THE STATISTICAL ANALYSIS OF POINT EVENTS ASSOCIATED WITH A FIXED POINT

Andrew B. Lawson

A Thesis Submitted for the Degree of PhD at the University of St Andrews

1991

Full metadata for this item is available in Research@StAndrews:FullText at:

http://research-repository.st-andrews.ac.uk/

Please use this identifier to cite or link to this item:

http://hdl.handle.net/10023/7294

This item is protected by original copyright
THE STATISTICAL ANALYSIS OF POINT EVENTS ASSOCIATED WITH A FIXED POINT.

Andrew B Lawson
Ph.D. degree
IMAGING SERVICES NORTH
Boston Spa, Wetherby
West Yorkshire, LS23 7BQ
www.bl.uk

TEXT CUT OFF IN THE ORIGINAL
DECLARATION

a) I, Andrew B Lawson, hereby certify that this thesis has been composed by myself, that it is a record of my own work, and that it has not been accepted in partial or complete fulfilment of any other degree or professional qualification.

Signed

Date 23/11/90

b) I was admitted to the Faculty of Science of the University of St Andrews, under Ordinance General No 12 on October 1986, and as a candidate for the degree of Ph.D. on October 1987.

Signed

Date 23/11/90

c) I hereby certify that the candidate has fulfilled the conditions of the Resolution and Regulations appropriate to the Degree of Ph.D.

Signature of Supervisor

Date 21 Nov 1990

23 Nov 1990
COPYRIGHT DECLARATION

A UNRESTRICTED

In submitting this thesis to the University of St Andrews I understand that I am giving permission for it to be made available for use in accordance with the regulations of the University Library for the time being in force, subject to any copyright vested in the work not being affected thereby. I also understand that the title and abstract will be published, and that a copy of the work may be made and supplied to any bona fide library or research worker.
ABSTRACT

This work concerns the analysis of point events which are distributed on a planar region and are thought to be related to a fixed point. Data examples are considered from Epidemiology, where morbidity events are thought to be related to a pollution source, and Ecology and Geology where events associated with a central point are to be modelled.

We have developed a variety of Heterogeneous Poisson Process (HEPP) models for the above examples. In particular, I have developed interaction and δ-dependence models for angular-linear correlation, with their ML estimation and associated score/Wald tests. In the Epidemiological case we have developed case-control models and tests.

The possibility of second-order effects being important has also led to the development of Bayesian Spatial Prior (BSP) models.

In addition, we have developed a new deviance residual for HEPP models and explored the use of GLIM for modelling purposes.

A variety of results were found in data analysis. In some cases HEPP models provide adequate descriptions of the process. In others, BSP models yield better fits. In general, the discrete case admits a simple spatial Poisson model for counts and does not require BSP model extensions.
ACKNOWLEDGEMENTS

First, I should like to thank my main supervisor, Professor R M Cormack for constant support and encouragement during the execution of this work. A number of lights have been turned on during our many discussions and I am deeply indebted to him for such illumination. I should also like to thank Dr P E Jupp who has acted as supervisor during Professor Cormack’s absence. As my research lies on the border between Spatial and Directional data analysis, the combination of supervisors has provided a balance between these areas.

Second, I should like to thank Jeremy Warnes for useful discussion concerning Empirical Bayes methods and Kriging, and Dr T Rolf Turner (University of New Brunswick) for use of his DELDIR program.

Third, I should like to thank Dr P Mason of the Institute of Terrestrial Ecology, Bush Estate for use of the Hebeloma Data Set, and General Register Office, Edinburgh for Mortality and population data for Buckhaven - Methil and Bonnybridge. I should also like to thank Dr Owen Lloyd and Dr Fiona Williams of the Environmental Epidemiology Unit, University of Dundee, for access to the Armadale data set.

Fourth, I would like to acknowledge the excellent typographical work of Shiela Wilson and Valerie Cobb, who have both provided speedy and accurate copy.

Finally, I should like to thank Pat for all her long-suffering patience and encouragement during the ups and downs.
TABLE OF CONTENTS

GLOSSARY

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

INTRODUCTION

1. Exploratory Data Analysis
   1.1 Kernel Density Estimation
   1.2 Special methods
      1.2.1 Kernel Interpolation
      1.2.2 Covariate Extraction
   1.3 Preliminary Testing of Mapped Patterns
      1.3.1 Radial Trend
      1.3.2 Angular anisotropy
      1.3.3 Radial-angular interaction
      1.3.4 Example

2. Point Process Modelling (Continuous)
   2.1 HEPP model definition
   2.2 Fixed-point HEPP models
      2.2.1 The polar model and sampling window
      2.2.2 Asymptotic Methods in Likelihood Analysis
   2.3 Likelihood Methods for HEPP models
      2.3.1 Single factor intensity models
      2.3.2 Composite intensity models
2.3.2.1 ML estimation
2.3.2.2 Hypothesis tests

3. Continuous Model: Extensions
3.1 Observed Heterogeneity
3.1.1 Case - Control Score Tests
3.1.1.1 Case a) Radial Trend
3.1.1.2 Case b) Angular Concentration
3.1.1.3 Case c) Peaked-Radial Effect
3.1.1.4 Case d) Interaction Effect
3.2 Unobserved Heterogeneity
3.2.1 Harmonic Intensity Terms
3.2.2 Spatial Prior Structure for Intensities
3.2.2.1 Cox Process Model
3.2.2.2 A Bayesian Spatial Prior (BSP) Model

4. Continuous Model: Goodness-of-Fit of Residual Analysis
4.1 Global Goodness-of-Fit
4.2 Residual Analysis
4.2.1 Binned Residuals
4.2.2 Individual Residuals
4.2.2.1 Deviance Residuals
4.2.2.2 Examples
4.2.2.3 Autocorrelation and Deviance Residuals
4.2.2.4 BSP Individual Residuals

5. Discrete Model: Introduction
5.1 The Human Morbidity Pattern
5.1.1 Functional forms for \( e \)
5.1.2 Likelihood Methods
5.1.2.1 Maximum Likelihood Estimation
5.1.2.1.1 Exponential Trend
5.1.2.1.2 Angular Concentration
5.1.2.1.3 Radial-Angular Interaction
5.1.2.2 Hypothesis Testing
5.1.2.2.1 Exponential Trend
5.1.2.2.2 Angular Concentration
5.1.2.2.3 Radial-Angular Interaction
5.1.2.3 General Testing with Observed Heterogeneity
5.1.2.4 Unobserved Heterogeneity
5.1.2.4.1 Harmonic Hazard Terms
5.1.2.4.2 Spatial prior structure for hazards
5.1.3 Discrete Model: Goodness-of-Fit and Residual Analysis
5.1.3.1 Global Goodness-of-Fit (GOF)
5.1.3.2 Residual Analysis

6. Variants of the Continuous/Discrete Models and Hazards
6.1 Model Variants
6.2 Hazard Variants

7. HEPP Models on GLIM
7.1 Normalising Constant Models
7.2 Numerical Approximation
7.3 Probabilistic Approximation
7.4 Integral Approximation Accuracy
7.5 One Dimensional Models
7.5.1 The von Mises Distribution
7.5.1.1 Numerical Comparison
7.5.2 A Test for 'von Misesness'
7.5.3 The Fisher Distribution
7.5.3.1 Data Example
7.5.4 Spatial HEPP Model

8. Model Simulation and Test Statistic Behaviour
8.1 Model Simulation
8.1.1 Continuous Models
8.1.1.1 HEPP Models
8.1.1.2 Bayesian Spatial Prior Models
8.1.2 Discrete Models
8.1.2.1 Multinomial Models
8.1.2.2 Bayesian Spatial Prior Models
8.2 Test Statistic Behaviour
8.2.1 Sampling Distributions
8.2.2 Numerical Power Studies
8.2.2.1 Continuous data sets
8.2.2.2 Discrete data sets

9. Analysis of Data Sets
9.1 Data Sets A, B and C
9.1.1 Respiratory Cancer Deaths (Data Set A)
9.1.2 Bronchitis and Pneumonia Deaths (Data Sets B, C)
9.1.1.2 Bonnybridge (Data Set B)
9.1.2.2 Buckhaven-Methil (Data Set C)
9.2 Data Sets D, E, F
9.2.1 Hebeloma Sporophores (Data Set D)
9.2.2 Oak Bark Beetles (Data Set E)
9.2.3 Lupinus Arboreas (Data Set F)
9.2.4 Volcanic Ejecta (Data Set G)
9.2.4.1 The 1935 example
9.2.4.2 The 1937 example
9.2.4.3 The 1938 example

DISCUSSION and CONCLUSIONS
REFERENCES
APPENDICES
**Glossary**

Listed below are the standard set of symbols used in this thesis. Any variation in this notation is defined where it is required.

- \( N(A) \) : number of events in a region A.
- \( \text{region} \) : a suitably defined area of the plane.
- \( A \) : region label.
- \( |A| \) : Lebesgue measure of A in \( \mathbb{R}^2 \) (area)
- \( A_i \) : Area of i-th Dirichlet tile.
- \( E(X) \) : expectation of a random variable \( X \).
- \( R, \Phi \) : random variables describing the radial and angular behaviour of points.
  
  Usually \( 0 < R < r_0 \)
  
  \( 0 < \Phi < 2\pi \).
- \( r_0 \) : radius of circular sampling window.
- \( r, \phi \) : realised values of random variables \( R \) and \( \Phi \)
- \( \mathbf{r} \) : coordinate pair \( (r, \phi) \)
- \( x, y \) : cartesian coordinates of point location
- \( \mathbf{x} \) : cartesian coordinate pair \( (x, y) \)
- \( b_\mathbf{r}(t) \) : disc with centre \( \mathbf{r} \) and radius \( t \)
- \( \mathbb{R} \) : the real line
- \( \mathbb{R}^2 \) : the entire plane
$S^1$ : the surface of a 1-d sphere (the circle)

$\lambda(\mathbf{r})$ : the intensity function at $\mathbf{r}$

$\rho$ : constant baseline intensity

$\Lambda(A)$ : $\int_A \lambda(\mathbf{r}) d\mathbf{r}$, the integral of $\lambda(\mathbf{r})$ over region $A$

$\lambda_i$ : $\lambda(.)$ evaluated at point $i$

$\sum_{i=1}^{n}$ : (unless otherwise stated)

$\xi_n$ : $(n \times 1)$ vector of mean surface values

$g_n$ : $(n \times 1)$ vector of linear predictors

$g(\mathbf{r})$ : linear predictor at $\mathbf{r}$

$m_k$ : number of events in region $k$

$p$ : number of regions

$e_k$ : hazard of any event in region $k$

$e_i$ : hazard associated with individual $i$ in region $k$

$n_k$ : total population of region $k$

$E_k$ : expected number of events in region $k$

$n_{ij}k$ : population of $i$-th age, $j$-th sex group in region $k$

$r_{ij}$ : national event rate for $i$-th age and $j$-th sex group

$n_a$ : number of age groups

$m_t$ : $\sum_{k=1}^{p} m_k$

$Zn/Fn$ : $(n \times p_a)$ matrix of covariate/trend
$\hat{\beta}$ : $(p_a \times 1)$ vector of parameters

$\Omega$ : $(p_a \times 1)$ vector of parameters

$p_a, p_b$ : number of parameters

$\kappa_n$ : $(n \times n)$ covariance matrix

d(i,j) : euclidean distance between points i, j

$\sigma^2$ : variance of a Gaussian Process

$R_a$ : covariance range parameter

\{ \} : set notation.
List of Figures

1) Data example maps:
   a) Data set (A): Armadale
   b) Data set (B): Bonnybridge
   c) Data set (C): Buckhaven-Methil
   d) Data set (D): Hebeloma sporophores 1975, 1978
   e) Data set (E): Oak Bark Beetles
   f) Data set (F): Lupinus arboreas
   g) Data set (G): volcanic ejecta 1935, 1937, 1938

Figures 2 to 8 consist of Gaussian kernel estimates of the a) radial, b) angular marginal distributions for the above data sets. Display c) is the surface 2-dimensional kernel estimate for the data set. All estimates were obtained using likelihood cross-validation. Figures 2) to 5) are point estimates and figures 5) to 8) are count estimates.

2) Armadale (A)
3) Hebeloma (D)
4) Lupinas (F)
5) Volcanic ejecta (G)
6) Bonnybridge (B)
7) Buckhaven-Methil (C)
8) Oak Bark Beetle (E)

9) Covariate extraction
   a) Armadale case-control i) population, ii) heart disease case control, iii) cancer, iv) extraction
   b) Bronchitis and pneumonia deaths with expected deaths extracted: Bonnybridge
   c) Bronchitis and pneumonia deaths with expected deaths extracted: Buckhaven-Methil

10) Simulations: $M(\mu_0, \kappa+\psi r)$

11) Simulations: $\delta$-dependence,

12) Marginal $g(\phi)$ of interaction $M(\mu_0, \kappa+\psi r)$

13) i) Point maps of deviance residual test data sets
   ii) Probability plots: CRICK, REDWOOD, PPSDRES, PPCNRES, Armadale marginal angular density (a-e)

14) Annular-sector method for regional integration

15) Point map: HEPP simulation on unit square ($n = 30$ points)

16) Simulations of a) Hebeloma 1975 data
    b) volcanic ejecta 1935 data

17) BSP simulation a) covariance range ($\alpha$) variation
    b) variance ($\sigma^2$) variation
    c) trend with range ($\alpha$) variation

18) Discrete simulation: a) trend b) peaked
19) Discrete simulation: a) angular concentration
   b) angular interaction
   c) concentration and interaction

20) Discrete BSP simulation: a) $\alpha = 1$, $\sigma^2 = 2$
    b) $\alpha = 0.1$, $\sigma^2 = 2$

21) Discrete BSP simulation: a) $\alpha = 1$, $\sigma^2 = 5$
    b) $\alpha = 1$, $\sigma^2 = 0.5$
    c) $\alpha = 10$, $\sigma^2 = 1$
    d) $\alpha = 0.1$, $\sigma^2 = 0.1$

22) continuous power studies: asymptotic critical regions

23) continuous power studies: monte carlo critical regions

24) discrete power studies: asymptotic results

25) discrete power studies: distance test comparison

26) Armadale: a) point map
    i) respiratory cancer
    ii) heart disease
    b) respiratory cancer expected death surface

27) Deviance Residual displays: Armadale
    a) fitted values, b) radial distance, c) angle, d) normal plot, e) surface view,
    f) surface contour map

28) Bonnybridge
a) enumeration distinct map (with inset location map)

b) SMR and death count surface (Bronchitis and Pneumonia)

29) Anscombe Residual displays: Bonnybridge
   a) fitted values, b) radial, c) angle, d) normal plot,
   e) surface view, f) surface contour

30) Buckhaven-Methil:
   a) enumeration district map
   b) SMR and death count surface (Bronchitis and Pneumonia)

31) Anscombe Residual displays: Buckhaven-Methil: reduced data set:
   a) surface view, b) surface contour, c) fitted values, d) normal plot

33) Deviance residual displays: Hebeloma 1975
   a) fitted values, b) radial distance, c) angle d) normal plot,
   e) surface view, f) surface contour

34) Deviance residual displays: Hebeloma 1978
   a) fitted values, b) radial distance, c) angle, d) normal plot,
   e) surface view, f) surface contour

   a) fitted values, b) normal plot, c) surface view, d) contour map

36) Oak log locations: Oak Bark Beetle data

37) Count map: Oak Bark Beetles

38) Anscombe Residual displays: Beetle Data: ltb offset
a) - f) as for fig (27)

39) Deviance residual displays: Lupinus Data
   a) - f) as for fig (27)

40) Standardised residual displays: BSP model: Lupinus data
   a) fitted value, b) normal plot, c) surface view
   d) surface contour

41) Deviance residual displays: 1935 volcanic bomb data
   a) - f) as for fig (27)

42) Standardised residual displays: BSP model: 1935 volcanic bomb data
   a) - f) as for fig (35)

43) Deviance residual displays: 1937 volcanic bomb data
   a) - f) as for figure (27)

44) Standardised residual displays: BSP model: 1937 volcanic bomb data
   a) - d) as for fig (35)

45) Deviance residual displays: 1938 volcanic bomb data
   a) - f) as for fig (27)

46) Standardised residual displays: BSP model: 1938 volcanic bomb data
   a) - d) as for fig (35)

47) Likelihood surface: \( \psi \)-\( \kappa \) optimisation (Appendix XII).
## List of Tables

1) Preliminary tests: data set A
2) Optimisation of $M(\mu_0, \kappa + \psi \tau)$ models on NAG (Appendix XII)
3) Numerical accuracy results 1-D weights
4) Numerical accuracy results 2-D weights
5) GLIM model results for the VOL 35 data example
6) ML results for Armadale (Data Set A)
7) Armadale: a) case-control and other model results  
   b) kernel score tests
8) GLIM model fits for Total Deaths: Bonnybridge
9) Score Test results: Bonnybridge
10) ML estimates of $\kappa, \psi, \mu_0$: Bonnybridge
11) Glim model fits for total Deaths: Buckhaven-Methil
12) Score test results: Buckhaven-Methil
13) ML estimates for $\kappa, \psi, \mu_0$: Buckhaven-Methil
14) Summary of Models for Bonnybridge and Buckhaven-Methil
15) Glim model fits for 1975, 1978 Hebeloma examples
16) Exploratory tests for Hebeloma Data
17) Glim model fits for Beetle Data: Polar Model
18) GLIM model results for Beetle Data: ltb offset
19) GLIM model results for Lupinus data
20) Exploratory tests for Lupinus data
21) Exploratory tests for Volcanic Bomb examples
List of Appendices

I  Computational Methods
II  Simulation Methods
III Research Papers
IV  Score Tests for Peakedness
V  Moments and other elements of the characterisation of the Interaction von Mises Distribution $(M(\mu_0, \kappa+\psi))$
VI  Derivation of Score and Wald Tests for Continuous and Discrete Models
VII GLIM macros
VIII Parameter and variance-covariance matrix approximations using the Delta Method
IX  Observed Information Matrix for the Case-Control Score Test for interaction
X  Asymptotic Sampling Distributions for test statistics
XI  Data sets
XII Numerical results for $\psi$-$\kappa$ optimisation
Introduction

The purpose of this reported research is the development of statistical techniques for the analysis of a point process associated with a fixed point. The motivation for this work lies in its application to the analysis of problems in Environmental Epidemiology and Ecology and Geology.

The detection of association between deaths recorded within an area and a pollution point source, (e.g. a factory chimney) is an example of such a problem found in environmental epidemiology. Published examples of such data are given in Lloyd (1982) and Lenihan (1985) and Carstairs and Low (1986).

In addition, there are a number of examples of data sets in Ecology where the spatial distribution of events is related to a fixed central point. Last et al (1984), Ford et al (1980) and Mason et al (1982) give examples of the spatial association of fruitbodies of sheathing mycorrhizal fungi around birch trees (Betula pendula). Yates (1983) has cited an example of the radial dispersion of oak bark beetle (Scolytus intricatus). The dispersal of seeds from a single plant would represent another example of such association. Palaniappan et al (1979) give an example of colonisation of a waste tip site by Lupinus Arboreus. The distribution of volcanic bombs/tephra after a volcanic eruption is an example of a geological event which involves dispersal by air. Minikami (1942) gives examples of a series of such events. Figure 1 displays examples of each data set.

Association with a single fixed point can be considered to be a special case of association with a realisation of another stochastic process of objects. Berman (1986) gives an example of modelling association between a point process (mineralisation sites) and a line process (faults) in a geological application. Conditional on the realisation of the line process, its components are considered to be fixed, and hence, the
case of a single fixed point is a specialisation of Berman's problem. Stoyan et al (1987) give other examples and some descriptive measures of association. The study of association between cancer cases and electromagnetic fields is a medical example.

The examples from Epidemiology and Ecology and Geology above, may show similar data structures but differ in that the hypotheses of interest are different and the level of data aggregation may vary considerably. In the epidemiology example, we are concerned with evidence for association between the fixed point and events to assess ultimately whether there is a causative link between these items. Hence, we are concerned with 'detection' of the imprint of a pollution source on a population of, possibly, variable susceptibility.

In the Ecological and Geological examples we are concerned with a movement out from a centre, either through dispersal in air or growth underground. In these examples, detection of the centre of dispersal is not important. However, appropriate modelling of this dispersal is important.

Different levels of data aggregation tend to lead to differences between the modelling approaches required by these two types of example. In the epidemiological example, a realisation of death locations may only be available as counts within census enumeration districts (eds), and hence arbitrary regionalisation of a point process must be considered.
Data Examples

The data sets considered in this work are as follows:

Epidemiological examples

A) Addresses of respiratory cancer death certificates for Armadale, West Lothian. The number of addresses (n) is 49. For this example the population structure of the area is available from 18 census eds. Deaths have been recorded for the period 1968-1974, and the 1971 census has been used as a population base. This data set was first analysed by Lloyd (1982) as part of a study of the perceived environmental hazard from a steel-making complex in central Armadale.

B) Counts of Bronchitis and Pneumonia deaths (1980-1982) in eds within a 5km radius circle of a chemical reprocessing plant (GR 835 815); Denny-Bonna Bridge area, Central Scotland. The number of eds studied is 200. Population information is available from the 1981 census.

This data set relates to the analysis made by the Lenihan Committee (Lenihan (1985)) on a perceived environmental hazard thought to relate to the Rechem Chemical reprocessing plant.

C) Counts of Bronchitis and Pneumonia deaths (1980-1982) in eds within a 3km radius circle of an industrial complex (GR 366 004); Buckhaven-Methil area, Eastern Fife. The number of eds studied is 62. Population and deprivation information are available from the 1981 census.

This data set is from an industrial area with a central industrial complex surrounded by domestic housing areas. No environmental hazard has been postulated hitherto.
The above data sets represent a spectrum of data aggregation level, from set A) which consists of exact locations and regional covariates to set B) where death counts in eds are available with population totals. Set C) can be regarded as a control data set and will be used in informal comparisons at a later stage.

Ecological Examples

D) Locations of Hebeloma Sporophores around a birch tree (1975, 1978) within a 5m square window. The number of point locations is 115 and 41 respectively. No covariate information is available.

This data set was analysed by Last et al (1984) and similar data sets were modelled by Byth (1980).

E) Counts of radio-labelled oak bark beetles found on experimental logs set at predetermined points around a central release point within a radius of 76m. The number of marked beetles is 115, the number of unmarked beetles is 366.

This data set was analysed by Yates (1983) and contains total populations at each location.

F) Locations of Lupinus arboreus plants found over a 6 year period around a central source plant on a waste tip site. The number of plants is 76.

This data set was analysed by Palaniappan et al (1979). No covariate information available for this example.

The above ecological data sets represent dispersal from a central point and consist of exact point locations ((D) and (F)) and counts (E). The data sets have only limited covariate information included.
Geological Examples

G) The locations of volcanic bombs after eruptions of Mt. Asama, Japan (1935, 1937, 1938). These data sets consist of mapped bomb locations found after the eruptions of April 20th 1935, April 10th 1937 and June 7th 1938. These data were presented by Minakami (1942). This data is characterised by dispersal under prevailing wind, and shows a simpler pattern than the biological examples considered above. No covariate information is available. Figure 1 a-g displays the data sets. In the case of count data a surface representation is given.

Spatial Models

The data examples given above have some common features. First, the basic unit of data is a point location or a count of points in a region. Second, the observations are measured relative to a fixed point. Third, the spatial pattern which is to be described is characterised by slowly varying point intensities i.e. we are concerned with first-order spatial properties or spatial trend.

Observations in the form of point locations or counts suggest a Point Process model could be appropriate. Location relative to a fixed point suggests that the Point Process should be modelled in polar coordinates (r,ϕ).

Finally, first-order properties can be modelled by an intensity function dependent on location, i.e. \( \lambda(x) \), and it is natural to consider a heterogeneous Poisson Process (HEPP) as a basic model. An HEPP model may be thought appropriate for fixed-centre data sets as the r and ϕ components can be related to the central point and relatively simple parametric forms for the (r,ϕ) distribution can be specified. Essentially, a bivariate distribution underlies this model assumption.

Second-order properties of point processes, i.e. their covariance structure, are characterised by the degree of clustering of points. A large literature has developed
concerning the modelling of such patterns (see, for example, Diggle (1983); Ripley (1981, p 155-168; Ripley (1988, p 49-67)). In the examples considered here, we do not consider second-order effects, except where underlying environmental heterogeneity will be modelled with a spatial prior density. This approach can be justified by the fact that we consider the spatial distribution of diseases with no known clustering aetiology (Bronchitis, Pneumonia, Respiratory Cancer), and of dispersal of plants, insects and volcanic ejecta from a central point. In either case, first-order effects must be modelled even though some environmental or subject heterogeneity may produce apparent clustering.

A number of methods are available for incorporation of such prior spatial structure, and we will review these alternatives in the following chapters.

First of all, we consider two basic types of model which could be applied to the above cases. In chapter 2 we outline a continuous model which could be applied in most data examples. Chapter 5 deals with a discrete model for human population structure which is related to the proportional hazards model in the time domain. This model was developed to allow the incorporation of covariates defined on individual susceptibles in the population considered.

The two basic models represent different ends of a spectrum from a homogeneous environment (continuous model) to a discretised/heterogeneous environment (discrete model). However, each model can be modified to move closer to the other formulation, and such modifications will be discussed in the following chapters.
CHAPTER 1

1. Exploratory Data Analysis

Exploratory methods for mapped point process data have developed traditionally via visual inspection of point maps to description of the intensity surface of the process by density estimation (see, for example, Byth (1980), Diggle (1981)). In most data examples, a stationary process is to be described and the isotropy of this process has led to the development of toroidally-corrected density estimates as a basic tool in graphical exploration. Alternative graphical techniques which depend on functions of point locations can also be utilised, and there appears to be scope for the application of multivariate EDA methods in this area.

Information about general spatial structure can be obtained from the Dirichlet tesselation or Delaunay triangulation of the points (see, e.g. Sibson (1980)). As the area of a Dirichlet tile (A) is related to the local point density, then 1/A is an intuitively appealing local intensity estimate. Hence a Dirichlet tile map yields an intensity 'picture'. The Delaunay triangulation yields a similar pictorial effect but also demonstrates the convex hull shape of the process. Convex hull peeling (Green, 1981) also yields information about anisotropic structure and shape.

The data examples considered here are not stationary and show anisotropy. In addition, as these examples are related to a fixed point, it is possible to consider the marginal structure of the radial and angular components of the processes.

Although the above general EDA methods are applicable in these examples, some adaptations of existing methods and new methods have been derived to account for these special cases.
Data Set (A): Armadale:
Respiratory Cancer death certificate locations
(1968-1974)
Data Set (B): Bonnybridge:
Bronchitis count surface
(h = 3) (1980-1982)

Pneumonia count surface
(h = 3) (1980-1982)
Data Set (C): Buckhaven Methil:
Bronchitis count surface
(h = 3) (1980-1982)

Pneumonia count surface
(h = 3) (1980-1982)
Data Set (D): Hebeloma Sporophores point map
1975, 1978
Data Set (E): Oak Bark Beetle count surface
(h = 7.426)

Data Set (F): Lupinus Arboreas point map
Data Set (G): point map of volcanic ejecta
1935, 1937, 1938
1.1 **Kernel Density Estimation**

Diggle (1981, p 55) has proposed a general toroidal edge-corrected kernel method for spatial point maps. The estimates thus produced can be viewed as contour plots or isometric views. Usually contours hold more detailed information, but isometric views are useful for general assessment.

When point data shows marked non-stationarity on visual inspection then it is inappropriate to use toroidal edge-correction in density estimation. The only method which accounts for edge effects is to use a guard area and, by implication, view a percentage of the map area at each edge as a border. Hence, standard two-dimensional kernel estimates can be used.

As the nature of the kernel is not critical, (compared to the choice of smoothing parameter), we have used a two-dimensional Gaussian kernel due to its computational simplicity. The choice of smoothing parameter (h) is however, important, as arbitrary over-smoothing can occur. Epanechnikov (1969) gives an IMSE criterion for obtaining an optimal h value which can be evaluated for specific target distributions. However, as we do not wish to make specific distributional assumptions, we have used likelihood cross-validation to estimate h opt. It is important to note that quite marked differences in h opt can be produced by these two criteria. Hence, as cross-validation is less parametric it is preferred.

The marginal radial or angular density of points can also be estimated. This form of density estimation is one-dimensional, but has to be adapted to the truncated positive real line for radial estimation, and to the circle for angular estimation. The first case may require reflection of data about the origin (Boneva et al, 1971). In addition, at the truncation point (r_0) a small section could be regarded as a guard area.
IMAGING SERVICES NORTH
Boston Spa, Wetherby
West Yorkshire, LS23 7BQ
www.bl.uk

PAGE MISSING IN ORIGINAL
where $K(\cdot)$ is a kernel function

$h$ is the smoothing parameter

$C$ is a normalising constant

$n$ is the number of points.

We can introduce a new variable $z(x_i)$, say measured at $x_i$, and re-write (1.1) as

$$\hat{z}(x) = C \sum_{i=1}^{n} z(x_i) K(x - x_i; h).$$

(1.2)

Here $K(\cdot)$ acts as a smoothing operator on $z$, and hence this operation can be used to interpolate $z$ to new locations. One advantage of this method is that it ensures that if $z$ is restricted to the positive real line then $\hat{z}(x)$ will be non-negative. This is not true of methods such as Universal Kriging, although log normal Kriging would protect against this effect. Hence, interpolation of counts can be directly achieved without the requirement of transformation.

The above method has been little documented (see, e.g. Ramlau Hansen (1983), Titterington (1985)) and the author knows of only one published application of this method where standardised mortality ratios (SMRS) are interpolated in time (Breslow and Day, 1987, pp 193-197). We have applied this method in the two-dimensional case to count data in regions. An example of this is the interpolation of region populations to the loci of a point process. This method also allows the exploratory examination of count data by contouring of an interpolated mesh. This method has been used for preliminary analysis of data sets B, C, E (Figures 6, 7, 8).
Figure 2 a)  
Armadale kernel estimate: a) radial ($h_{CV} = 3.373$)

Figure 2 b)  
Armadale kernel estimate: b) angular ($h_{CV} = 2.578$)
Figure 2 c)
Armadale kernel estimate: c) surface view ($h_{CV} = 14.669$)
Figure 3 a), b)
Hebeloma kernel estimate: a) radial \( h = 0.031 \);
b) angular \( h = 16.55 \).
Figure 3 a), b)
Hebeloma kernel estimate: b) radial $78: h = 0.0427$
   b) angular $78: h = 2.206$
Figure 3 c)
Hebeloma kernel estimate: c) surface view 75: \( h = 0.0497; 78: h = 0.123 \)
Figure 4
Lupinus Arboreas kernel estimate: a) radial (h = 8.2118)

Figure 4
Lupinus Arboreas kernel estimate: b) angular (h = 32.0)
Figure 4
Lupinus Arboreas kernel estimate: c) surface view (h = 16.141)
Figure 5
Volcanic ejecta kernel estimate: 1935 a) radial (h = 4.873)

Volcanic ejecta kernel estimate: 1935 b) angular (h = 21.839)
Figure 5
Volcanic ejecta kernel estimate: 1935 c) surface view (h = 9.697)
Figure 5
Volcanic ejecta kernel estimate: 1937 a) radial (h = 4.22)

Figure 5
Volcanic ejecta kernel estimate: 1937 b) angular (h = 16.476)
Figure 5
Volcanic ejecta kernel estimate: 1937 c) surface view (h = 8.397)
Figure 5
Volcanic ejecta kernel estimate: 1938 a) radial (h = 4.12)

Figure 5
Volcanic ejecta kernel estimate: 1938 b) angular (h = 1.351)
Figure 5
Volcanic ejecta kernel estimate: 1938 c) surface view (h = 8.646)
Figure 6
Bonnybridge kernel estimate: Bronchitis: a) radial ($h = 3.07$)

Figure 6
Bonnybridge kernel estimate: Bronchitis: b) angular ($h = 0.115$)
Figure 6
Bonnybridge kernel estimate: Pneumonia: a) radial \( h = 3.07 \)

\[ \text{Radial distance} \]

Figure 6
Bonnybridge kernel estimate: Pneumonia: b) angular \( h = 0.115 \)
Figure 6
Bonnybridge kernel estimate: Bronchitis: c) surface view (h = 7.773)

Figure 6
Bonnybridge kernel estimate: Pneumonia: c) surface view (h = 7.773)
Figure 7
Buckhaven-Methil kernel estimate: Bronchitis: a) radial ($h = 3.483$)

radial distance

Figure 7
Buckhaven-Methil kernel estimate: Bronchitis: b) angular ($h = 20.187$)

angle
Figure 7
Buckhaven-Methil kernel estimate: Pneumonia: a) radial (h = 3.483)

Figure 7
Buckhaven-Methil kernel estimate: Pneumonia: b) angular (h = 20.187)
Figure 7

Buckhaven-Methil kernel estimate: Bronchitis: c) surface view (h = 3.115)

Buckhaven-Methil kernel estimate: Pneumonia: c) surface view (h = 3.115)
Figure 8
Oak Bark Beetle kernel estimate: a) radial (h = 0.01)

Figure 8
Oak Bark Beetle kernel estimate: b) angular (h = 40.0)
Figure 8
Oak Bark Beetle kernel estimate: c) surface view (h = 7.426)
1.2.2 **Covariate Extraction**

The above methods are usually sufficient to describe the *spatial* structure of points or region counts in a mapped area. However, spatially-distributed covariate information is often associated with such data sets, and it can be important to examine the effect of such information on the spatial point process.

For example, the location of deaths in an area is related to area population, in that higher densities of populations should show higher numbers of deaths. This is often accounted for in regional count data by comparing a regional count to the expected number in the region based on its population structure (SMRS).

We have developed a new method, based on kernel estimation, for the extraction of baseline/covariate information.

Define the point process to be governed by an intensity

\[ \lambda^*(x) = H(x) \lambda(x) \]  

where \( H(x) \) is a baseline/covariate process and \( \lambda(x) \) is an underlying *spatial* intensity. \( H(x) \) could be a population process or even a case-control point process (see Diggle (1989)).

We now introduce a kernel estimator of the intensity

\[ \hat{\lambda}(x) = \frac{1}{C} \sum_{i=1}^{n} \hat{H}^{-1}(x_i) K(x-x_i; h) \]  

where

\[ \hat{H}(x_i) \] is the estimated covariate process evaluated at \( x_i \).

Note that \( \hat{H}(x_i) \) can be an intensity estimate of a case-control point process or the interpolation of a real valued covariate to \( x_i \). \( \hat{\lambda}(x) \) now may depend on the smoothing parameters of both \( \hat{H} \) and \( \hat{\lambda} \). The interpretation of a contoured surface of \( \hat{\lambda}(x) \), in the case-
Armadale case control i) population surface \( (h = 2.304) \)

Armadale case control ii) heart disease surface \( (h = 2.712) \)

Armadale case control iii) respiratory cancer surface \( (h = 14.669) \)
Figure 9 a)
Armadale case control iv) extraction surface (h = 5.186)
Bronchitis and Pneumonia deaths with expected deaths extracted: Bonnybridge (h = 3)
Figure 9 c) Bronchitis and Pneumonia deaths with expected deaths extracted: Buckhaven-Methil (h = 3)
control example, is based on the fact that any value > 1 is in excess of the baseline case-control. A similar interpretation is possible for other covariates if they are suitably normalised over the window area. Figure 9a) display the results of such extraction for data set A with a case-control point process of cardio-vascular disease. Figures 9 b), c) display the results of such extraction for data set B and C for Bronchitis deaths and baseline process of expected number of deaths.

1.3 Preliminary testing of Mapped Patterns

Visual inspection of point maps and contoured intensity surfaces demonstrate the general structure of data sets. In our examples, the dominant structure appears to be non-stationarity with marked anisotropy around the central point. There is only limited evidence of clustering.

As our aim is to assess whether there is evidence of spatial trend around the central point, we can employ simple statistical tests to assess trend components prior to statistical modelling.

1.3.1 Radial trend

Association with a central point may be displayed by a gradient or trend over radial distance from the centre. A number of tests are available for testing the null hypothesis of uniformity against trend alternatives for a 'radial-only' effect. Cox and Lewis (1966) derived the uniformity test, which in our case is given by

\[ U = \sum_{i=1}^{n} \frac{r_1 - \frac{1}{2} r_0}{r_0 \sqrt{\frac{1}{12n}}} \]  

(1.5)
where \( r_i \) = i-th radial distance
\[ r_0 = \text{under radius} \]
\[ n = \text{number of parts}. \]

Cliff and Ord (1981, p 107) also suggest a spacings test which can be adapted to annular areas i.e.

\[ S = 2n - 2 \sum_{j=1}^{n} jA(j) \]  \hspace{1cm} (1.6)

where \( A(j) = j\text{-th (ordered) annular area.} \)

Under the assumption of complete spatial randomness (CSR), we have \( E(S) = (n-1)/2 \) and \( \text{var}(S) = (n-1)/12 \) (Durbin, 1965). Values of \([S-E(S)]\) are useful, as large positive values imply regular spacing and negative values imply clustering or non-stationarity. The test given in (1.5) is the score statistic derived from the assumption of a heterogeneous poisson process with exponential trend i.e. \( \lambda(r) = \lambda_0 e^{\beta r} \). This test is UMP for any monotone alternatives (Gart and Tarone (1983)). Both \( U \) and \( S \) have an asymptotic standard normal distribution.

As the radial effect could be peaked as opposed to monotone decreasing we may also test for a Weibull shape parameter: \( H_0: \delta = 1 \) against \( H_1: \delta > 1 \), by use of a score statistic. The detailed derivation of this test is given in Appendix IV. This derivation is given as this author is unaware of such a test for a truncated Weibull in the literature.

In the case of count data it is also possible to apply a uniformity test like (1.5) by assuming that the hazard in each region \( (\ell) \) is \( E_{\ell} = E_{\ell} \exp(\beta_{\ell} r_{\ell}) \).

This leads to the score statistic
where \( \sum_{x=1}^{P} E_x \) is adjusted to equal \( n_T \) the total number of events in the study area

\[
n_T = \sum_{x=1}^{P} n_x
\]

\( n_x \) is the number of deaths in region \( x \), and \( r_x \) is the radial distance of the \( x \)-th region centre from the central point. \( R \) has an asymptotic \( \chi^2 \) distribution. The derivation of this test is similar to that given in Breslow et al (1983).

### 1.3.2 Angular Anisotropy

If angular uniformity is to be tested, then a variety of tests are available and are well documented (e.g. Mardia, 1972). Watson's \( U^2 \) test, Rao's spacing test or the Rayleigh test are well known examples which can be employed to detect deviations from angular uniformity.

For the case of counts in regions it is possible to use a new test for angular trend:

\[
T_h = \frac{\left[ \sum_{x} \cos (\phi_x - \hat{\mu}) (n_x - N_e E_x) \right]^2}{N_e \left[ \sum_{x} E_x \cos^2(\phi_x - \hat{\mu}) - \left( \sum_{x} E_x \cos(\phi_x - \hat{\mu}) \right)^2 / \left( \sum_{x} E_x \right) \right]}
\]

where \( N_e = n_T / \sum_x E_x \)

and \( \hat{\mu} = \tan^{-1}(NS/NC) \) and
where $\phi_k$ is the angle of the $k$-th region measured from the central point. $T_h$ has an asymptotic $\chi^2_1$ distribution. The detailed derivation of this new test is given in Chapter 5.

### 1.3.3 Radial-angular interaction

Separate tests of radial or angular uniformity ignore the possible effects of interaction or correlation between radial and angular components and hence could be misleading if carried out in isolation. A number of tests are available which measure general angular-linear correlation (e.g. Jupp and Mardia, 1980). However, for the case of a radial-angular interaction a new test has been developed by the author.

\[
W_s = \frac{\sum_{i=1}^{n} r_i \cos(\phi_i - \bar{x}_0) - \sum_{i=1}^{n} ri\hat{R}}{\sqrt{A'(K) \left( \sum_{i=1}^{n} r_i^2 - \frac{(\Sigma r_i)^2}{n} \right)}}
\]

(1.9)

where

\[\hat{R} = \frac{1}{n} \sqrt{S^2 + C^2}\]

\[S = \sum_{i=1}^{n} \sin \phi_i, \quad C = \sum_{i=1}^{n} \cos \phi_i\]

\[A(K) = \frac{I_1(K)}{I_0(K)}\]

and

\[A'(K) = 1 - \frac{A(K)}{K} - A(K)^2\]

where $I_1(.)$ and $I_0(.)$ are modified Bessel functions of the first kind of 1st and 0th order respectively. $W_s$ has an asymptotic standard normal distribution.
The detailed derivation of this test is given in Chapter 2.

For counts in regions a similar score statistic (eq 77) has been derived. Details of this test are given in Chapter 5.

1.3.4 Example

We now consider where the above preliminary testing can be employed prior to modelling of spatial pattern. (We consider all data sets more fully in Chapter 9) Data set A consists of death certificate addresses, and their spatial distribution may be related to the underlying population structure or to a case-control point process. In this example, we have utilised the 1971 census total population counts for 18 eds in the Armadale area, as a population covariate. In addition, we have selected cardio-vascular disease mortality as a disease which closely matches the age-specific-aetiology of respiratory cancer but should be little affected by an air pollution source (Lloyd, 1982). Hence, we utilised death certificate addresses for this disease as a case control point process.

Figures 9a show the effect of using likelihood cross-validated kernel estimates for population, case control, and respiratory cancer separately.

We have also employed 'covariate extraction' to this data set by extracting the case-control density estimate from the data. The result is given in Figure 9a). The individual density surfaces show a marked concentration of respiratory cancer to the south west of the foundry area, while the population structure and case-control show higher concentrations in the northern areas. The extraction of the case control produces a single 'dramatic' peak of respiratory cancer in the north west sector. This is not at all apparent from inspection of the individual surfaces. Lloyd (1982) has shown that the results of wind tunnel experiments, intended to simulate the distribution of pollutant outfall, supports a high concentration in a
north-westerly direction from the foundry. Table 1 displays the results of applying the preliminary tests listed above to the Armadale data set.

Table 1 Preliminary Tests: Data Set A.

The results of preliminary statistical tests for spatial trend effects in data set A, with n = 49 points.

- **L**: radial uniformity test (1.5)
- **S**: spacings test (1.6)
- **W**: Weibull Shape test
- **R**: Rayleigh test
- **U^2**: Watson's U^2 test
- **Ws**: Interaction score test (1.9)
- **M**: Mardia rank correlation test.

All tests have an asymptotic standard normal distribution except M which is \( \sim \chi^2_2 \), and R and U^2 which have a special distribution (Mardia, 1972). We have carried out Monte Carlo tests for each statistic, as the asymptotic distributions quoted above may not hold in this example.

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>S</th>
<th>W</th>
<th>R</th>
<th>U^2</th>
<th>Ws</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-4.231*</td>
<td>-3.394*</td>
<td>5.412*</td>
<td>0.544*</td>
<td>0.814*</td>
<td>2.39*</td>
<td>7.48*</td>
</tr>
</tbody>
</table>

* denotes significance in a one-tail Monte Carlo test at \( \alpha = 0.05 \).
The radial tests (L,S) both show significant non-uniformity and the negative sign suggests possible clustering or non-stationarity. The peakedness test (W) also shows a significant result with the radial parameter estimated under the null hypothesis. This suggests that a peaked effect occurs in the data. Watson's $U^2$ test shows high significance which suggests that preferred orientations exist in the data. Both the interaction score test (Ws) and the rank correlation coefficient are significant and this also suggests a significant correlation exists between radial and angular components. Hence, preliminary tests confirm the visual evidence that there is a preferred direction and strong radial trend in the original data. The case-control extraction, however, suggests a similar result although the direction of the effect is different from that suggested on visual inspection. A similar analysis to the above appears in Lawson (1988a).
CHAPTER 2

2. **Point Process Modelling (Continuous)**

We assume, initially that point locations are observed on a homogeneous planar region within a sampling window. The point locations could be realisations of death certificate addresses or locations of plants, fungi or ejecta. The objective of our analysis is to relate the point locations to a fixed point.

The first order structure of such examples can be modelled parametrically by a heterogeneous Poisson Process model (hereafter known as HEPP). The justification for such a formulation lies in the fact that these models allow for non-stationarity in the mean of the process, while the Poisson Process assumptions allow for independence of events. The models do not allow for clustering of points.

Further, it is possible to derive polar coordinate models which have simple parametric forms if related to fixed points. Patterns not so related often do not admit such simple models. A number of workers have examined HEPP models for general mapped processes (e.g. Rantschner (1973) and Kooijman (1979)). Diggle (1979) introduced a 5-parameter C-type Beta distribution for description of a general point intensity, although this did not provide an adequate fit to his Balsam Fir Seedling data set. However, it may be possible to use higher order intensity surfaces by introducing more power terms in (x,y) and interaction terms. This could lead to a better description of pattern but at the expense of parsimony.

Often recourse is made to density estimation (see, e.g. Diggle (1981, 1985)), for which parameterisation is limited to a smoothing constant and kernel function.
2.1 **HEPP model definition**

The first order properties of our point process model are governed by an intensity function:

\[ \lambda(\vec{r}) = \lim_{\|d\vec{r}\| \to 0} \left\{ \frac{E[N(d\vec{r})]}{\|d\vec{r}\|} \right\} \]

(1)

where \(d\vec{r}\) is an infinitesimal region which contains \(\vec{r}\).

The intensity function (1) forms the basis of the HEPP model. By suitable parameterisation of \(\lambda(\vec{r})\) we can derive a variety of models.

The basic assumptions of the HEPP model are (from Diggle (1983), p.52):

A1) \(N(A)\) has a Poisson distribution with mean \(\Lambda(A) = \int_A \lambda(\vec{r}) d\vec{r}\).

A2) Given \(N(A) = n\), the events in \(A\) form an independent random sample from the distribution on \(A\) with pdf equal to \(\lambda(\vec{r})/\Lambda(A)\).

Two consequences arise from definitions A1 and A2, which facilitate modelling of point patterns. First, counts in disjoint regions are independent. Second, conditional on the realisation of \(n\) events within a suitable window the locations of these events form a random sample from \(\lambda(\vec{r})/\Lambda(A)\). The first consequence allows simple parametric modelling of data which are only observed as events within regions. The second consequence allows the use of conventional likelihood methods when exact point locations are observed.
2.2 Fixed-point HEPP models

The data examples considered here can be described by an intensity of the general form

\[ \lambda(r) = \lambda(r, \phi) = \rho \exp \{ G(r, \phi; \mathbf{\theta}) \} \]  

(2)

where \( \mathbf{\theta} \) is an \((n \times 1)\) vector of parameters and \( \rho = \) constant baseline intensity.

We adopt an exponential form as our basic model. This ensures that \( \lambda(r) \) remains positive and hence avoids the use of specialised likelihood methods for parameter estimation (see, for example, Ogata (1983, 1988) and Berman and Rolf Turner (1988). The advantages of this form also lie in the link between the specification of \( \lambda(r)/\Lambda(A) \) and exponential family models with normalising constants. Oakes (1979) has exploited this link in survival analysis, and the construction of models for radial and angular components of variation is facilitated by such a form.

2.2.1 The polar model and sampling window

We define

\[ \lambda(r) = f(r) \cdot g(\phi, r)/r \]  

(3)

where \( f, g \) are 'suitable' functions which describe the radial and angular behaviour of points. Often \( f, g \) will be pdfs on \( \mathbb{R}^1 \) or \( S^1 \). We use the term 'suitable', as some latitude exists in the definitions, given that \( \lambda(r) \) will be normalised by \( \Lambda(A) \).

It is usual to observe point events within a sampling window, and hence we are usually concerned with an example of a completely mapped realisation. The effect of the window is two-fold. First, we do not have to consider a sampling procedure, as we will model the distribution of events conditional on the total number \( n \) in the window.
fixed $p$, the window size or shape will control $n$. Second, the boundary of a window will induce edge effects, as the nearest neighbours of a point can be censored at a boundary. Ripley (1988, Ch 3) discusses edge correction for nearest neighbour and interpoint distance methods. In the present context, as all measurements are made from a fixed point within the window there is no requirement to edge-correct parameter estimates.

The shape of the sampling window is important in the definition of suitable models. Rectangular windows are often used where processes have no a priori anisotropy. In the fixed-point case we wish to allow for the possibility of an anisotropic pattern around the fixed point. With the use of the polar coordinate system it is also natural to consider a circular window, in that each radial distance should be equally represented regardless of anisotropy. In addition, integration over $b_0(r_0)$ can often be performed analytically for simple forms of $\lambda(r)$.

### 2.2.2 Asymptotic Methods in Likelihood Analysis

The conventional asymptotic theory pertaining to hypothesis tests and maximum likelihood estimation should be applicable to the above HEPP models. First, the points of the process are IID, and hence no long range spatial correlation is entertained. Ripley (1988, pp 19-20), in discussion of Mardia and Marshall (1984) notes that asymptotic results derived by increasing the window size while keeping the intensity constant, will yield, even for low spatial correlation, the classical asymptotic results for IID variables.

Under a HEPP model, the points are IID and hence increasing the window should yield such results. The case where some short range correlation exists in the underlying data should not invalidate asymptotic results under this limiting operation. Examination of asymptotic behaviour by increase of intensity within a fixed window will not necessarily
yield the same results. In addition, asymptotic results concerning sampling distributions of estimators do not necessarily apply to small-sample situations. To allow for such correlations Berman (1986) and Berman and Diggle (1989) used a general stationary/isotropic model (ISI) and carried out monte carlo tests to detect first-order effects.

To accommodate possible inapplicability of asymptotic results we consider methods based on i) standard sampling distributions, and ii) monte carlo tests. We also consider, at a later stage, the effects of heterogeneity of environment on estimation and tests, as well as modelling such heterogeneity via spatial prior distributions.

### 2.2.3 Types of Intensity Model

#### 2.2.3.1 The Radial component

Often the spatial association between events and a fixed point takes the form of a distance decay or possibly a peak-then-decay. Hence, it may be appropriate to define f(r) as a Gamma (δ,λ) or Weibull (δ,λ) distribution on \( R^1 \). Both forms can describe peak-then-decay behaviour (δ > 1) and also purely exponential behaviour (δ = 1). To derive the full pdf of point events the normalising constant Λ(A) must be evaluated over the window. For ease of derivation of closed form results for Λ(A) we have used a Weibull (δ,λ) for f(r). The Gamma (δ,λ) has some advantages, in its membership of the exponential family and if numerical integration is easily available, then this model may provide a better description of some examples.

The Weibull model defined above can be justified on physical grounds. In the case of a point pollution source, it may be expected that diffusion of airborne material under a constant wind will produce a ground-level distribution of elliptical form, downwind of the
source. This is supported by the results of Diffusion theory which suggest that a Weibull
\( \delta = 2 \) can describe the radial distribution while the cross-wind spread is modelled by a
location-dependent variance (see, e.g. Pasquill (1974) Ch 5). Given such a distribution at
ground level, it may be expected that a homogeneous population with a uniform spatial
distribution should be affected in a similar way. However, non-constant wind speed and
direction can produce a more complex pattern of deposition and as dominant wind-direction
may be related to higher wind speeds it may not be responsible for 'local' deposition of
pollutants. In addition, populations are usually heterogeneous and not uniformly
distributed in space. These considerations do not invalidate the assumption of a simple
Weibull model for radial effects as we may expect that, given population in a location, the
conditional radial effect may be Weibull. More complex models to account for the
spectrum of wind directions and speeds are likely either to have many parameters or not to
be physically realistic (see, e.g. Jensen (1981)).

In ecological examples, the dispersal of plants and insects can be controlled by
similar wind effects, or a minimum effort/density-dependent control mechanism. The
pattern of Fungi distribution may be related to underlying tree root densities and inhibition
by competing species. Figures 2a) and 3a) depict kernel density estimates for respiratory
cancer deaths in Armadale (data set A), and Hebeloma sporophores in 1975 (data set D).
These estimates were obtained by the method of Byth (1982). These examples of radial
effects suggest that a peak-then-decay model could be reasonable as a description of the
first-order behaviour of these points.
2.2.3.2 **The Angular component**

Anisotropy of effects around a fixed point can be conveniently modelled by a standard directional pdf. The von Mises distribution \( M(\mu_0, \kappa) \) describes a unimodal density of points on \( S^1 \). \( \kappa \) defines the concentration of points around the mean angle \( \mu_0 \). This distribution is also related to the ground-level distribution of airborne pollutants. First, in combination with a Weibull radial distribution, this distribution can describe an elliptical concentration at a distance from the origin. Second, it is possible to show that the von Mises distribution is related to Brownian Motion (Kent, 1978).

This distribution can be extended to account for multimodal densities of points (Mardia (1972, p.120)), asymmetry and harmonic effects (Mardia (1988)). The justification for such a distributional form can be seen from Figs 2b) and 3b). These figures represent kernel estimates (Gaussian with data wrapped onto \(-2\pi\) and \(+4\pi\)) of angular observations for data set A) and data set D). Both data sets display a relatively smooth unimodal effect. In addition, the physical processes of wind-directed dispersion can produce a dominant plume effect and hence a concentrated effect around a mean angle.

2.2.3.3 **Radial-Angular Interaction**

Discussion in a) and b) above regarded the radial and angular components of intensity as acting independently to produce a model effect. It is also possible to conceive of correlation or interaction effects between \( R \) and \( \Phi \) which can be exhibited by natural point patterns. In the case of events related to an ambient dispersal process, the angular concentration of events found should decrease with increase of \( R \). The effect of a dominant wind may be to produce *increased* concentration at large \( R \) values and hence a
correlation/interaction effect should occur. Figure 10 and 11 depict the differences between a standard von Mises distribution and a von Mises which includes an interaction parameter.

The analysis of the above models vis-a-vis ML estimation and hypothesis test derivation is discussed in Section 2.3.

### 2.2.4 Evaluation of the Normalising Constant ($\Lambda(A)$)

For the above HEPP fixed point models it is convenient to analyse point events within a circular window: $b_0(r_0)$.

Hence, for likelihood analysis it is necessary to evaluate $\Lambda(b_0(r_0))$, and therefore necessary to integrate $\lambda(r)$ over the disc $b_0(r_0)$. i.e.

$$\Lambda(b_0(r_0)) = \int_{0}^{r_0} \int_{0}^{2\pi} \lambda(r, \phi) r \, dr \, d\phi. \quad (4)$$

Evaluation of (4) is simple if numerical integration is available. However, if $\lambda(r)$ has form (3) it is relatively simple to obtain a closed form for (4). The advantage of this will become apparent when we derive closed-form tests and ML estimators for the above model.

Chapter 3 describes a new method for the analysis of the above first-order point-process models. The von Mises distribution and other more complex directional models can be fitted using this method. In addition, the general method of Berman and Rolf Turner (1988) is described. This method allows the fitting of spatial HEPP models using Tessellation/Triangulation weights in GLIM. Modifications and extensions of their method have been used by this author and are discussed in that chapter.
Figure 10

M ($\mu_0, k+\psi r$) simulation for $f(r)$ as truncated Weibull on the unit disc

a) Uniform

b) $\delta = 5$, $\lambda = 1$, $k = 2$, $\psi = 1$, $\mu_\theta = 30^\circ$

c) $\delta = 5$, $\lambda = 1$, $k = 2$, $\psi = 50$, $\mu_\theta = 30^\circ$
\[ \delta^* = \delta_0 + \delta_1 \cos(\phi - \mu_0) \text{ and } M(\mu, K), \text{ on the unit disk} \]

- **a)** \( \delta_0 = 5, \lambda = 1, k = 2, \mu_0 = 30^\circ, \delta_1 = 0 \)
- **b)** \( \delta_0 = 5, \lambda = 1, k = 2, \mu_0 = 30^\circ, \delta_1 = 10.0 \)
- **c)** \( \delta_0 = 2, \lambda = 1, k = 100, \mu_0 = 0, \delta_1 = 0.1 \)
- **d)** \( \delta_0 = 2, \lambda = 1, k = 100, \psi = 50, \mu_0 = 0 \)
2.3 Likelihood Methods for HEPP Models

As the point events under the HEPP model are IID, standard likelihood methods can be used to derive parameter estimators and test statistics.

The log likelihood for a realisation of \( N(b(r_0)) = n \) events in a window is (without conditioning on the number of events):

\[
\mathcal{L}_u = \sum \sum_n \lambda(r_i, \phi_i) - \Lambda(b_0(r_0)).
\]  

(5)

Straightforward manipulation gives the likelihood, conditional on \( N(b_0(r_0)) = n \):

\[
\mathcal{L}_c = \sum \sum_n \lambda(r_i, \phi_i) - n \lambda \Lambda(b_0(r_0)).
\]  

(6)

For most of the following analysis we will be concerned with \( \mathcal{L}_c \) as our starting point, as we are concerned with the spatial structure within the window, without regard to the variation in \( N(b_0(r_0)) \) between realisations. The only material difference between \( \mathcal{L}_u \) and \( \mathcal{L}_c \) is that the baseline intensity \( \rho \) is absent from \( \mathcal{L}_c \) because of the conditioning. \( n \) is sufficient for \( \rho \) and hence we are usually not concerned with this intensity. The implication of the use of \( \mathcal{L}_c \) is that a constant term need not be specified in the definition of \( \lambda(r) \).

2.3.1 Single factor intensity models

The simplest models which can be used to characterise a spatial trend, in our context, are single factor intensities where only a radial or angular component are included. These can be defined, up to a constant, as:

\[
\begin{align*}
\text{case 1) radial} & : \lambda(r) = f(r)/r \\
\text{case 2) angular} & : \lambda(\phi) = g(\phi).
\end{align*}
\]  

(7)  

(8)
In case 1), \( f(r) \) could represent a single parameter exponential trend (\( \exp(\beta r) \)) or a two parameter peaked trend (Weibull/Gamma form). In the first case, maximum likelihood estimation is simple, as \( \Sigma r_i \) is sufficient for \( \beta \). In the second case, estimation proceeds as a truncated version of Weibull or Gamma distributions. The derivation of test statistics based on \( \xi_u \) or \( \xi_c \) are well documented for case (1). In particular, Cox and Lewis (1966, pp46-47) and Bain et al (1985) discuss the Score test for \( \beta \), which is \( U = (\bar{r} - \frac{1}{2} r_0)/(r_0/(1/12n)) \), the standard test for uniformity against an exponential trend alternative.

In the case of a two parameter model, score tests for \( \delta \) can be derived, as in Cox and Oakes (1984, p.44), where the exponential parameter \( \beta \) becomes a nuisance parameter and must be estimated under the null hypothesis.

For case 2), if \( g(\phi) \) is specified as a standard angular distribution such as \( M(\mu_0, \kappa) \), then ML estimation and derivation of test statistics are exactly as for the standard distribution. To see this, note that

\[
\xi_{vm} = \kappa \sum \cos(\phi_i - \mu_0) - n \ln 2\pi I_0(\kappa)
\]  

(9)

where

\[
I_0(\kappa) = \frac{1}{2\pi} \int_0^{2\pi} e^{\kappa \cos \phi} d\phi
\]

the modified Bessel function of the 1st kind, order zero and, for

\[
\lambda(r) = \exp(\kappa \cos(\phi - \mu_0))
\]

\[
\xi_{cvm} = \kappa \sum \cos(\phi_i - \mu_0) - n \ln \int_0^{2\pi} e^{\kappa \cos \phi} d\phi
\]  

(10)

(bar a constant).

Hence, the ML estimates for \( \kappa, \mu_0 \) are
\[ \hat{\mu}_0 = \bar{x}_0 = \tan^{-1} \left( \frac{S}{C} \right) \]

and

\[ \tilde{k} = A^{-1}(\tilde{R}) \]

where

\[ A(\kappa) = I_1(\kappa)/I_0(\kappa) \]

and

\[ I_1(\kappa) = I'_0(\kappa) \]

\[ \tilde{R} = \frac{1}{n} \sqrt{C^2 + S^2} \]

and

\[ S = \sum \sin \phi_i, \quad C = \sum \cos \phi_i. \]

In Chapter 7, we detail a method for fitting the above models which exploits the equivalence of (9) and (10), to allow the use of GLIM.

\[ 2.3.2 \quad \textbf{Composite intensity models} \]

Often we wish to characterise a spatial pattern by both radial and angular components. In the following analysis we will be concerned with intensities where both \( f(r) \) and \( g(\phi, r) \) are non-uniform. In the special case where \( \lambda(r) = f(r) \cdot g(\phi)/r \), with no dependence of \( g \) on \( r \) then the discussion of section a) is relevant and estimation and testing follow immediately from the separability of the likelihood.

Here, we consider \( g \) as a function of \( \phi \) and \( r \) and include a correlation/interaction parameter in our model.

A variety of types of correlation can be specified between \( R \) and \( \Phi \). In previous work on angular-linear correlation most attention has been given to dependence of the mean of a linear variable on an angular component (see, e.g. Mardia & Sutton (1978), Mardia (1975), Liddell & Ord (1978)). An alternative, but restrictive assumption is to allow the mean angle (\( \mu_0 \)) to depend on \( R \) (Gould, 1969).
In our case, we have a truncated linear variable on $\mathbb{R}^+$, and the type of dependence between $R$ and $\Phi$ depends on the examples considered. For a wind related effect, it may be appropriate to consider increased concentration of angle with radial distance from the source. This may occur as the result of periods of dominant strong winds. On the other hand, in some cases, notably the volcanic ejecta examples, there appears to be a dependence of the radial mean/mode of the density on angle.

These two situations are best described by two forms of dependency.

The first case will be termed *interaction* and can be described by the model

$$g(\phi, r) = \exp\left( (\kappa + \psi r) \cos (\phi - \mu_0) \right) / 2\pi \psi_0 (\kappa + \psi r)$$

(11)

$$0 < \phi < 2\pi$$

$$0 < r < \infty.$$  

This is an extension of the standard von Mises density by inclusion of an interaction parameter $\psi$. We denote this as $M(\mu_0, \kappa + \psi r)$.

For our formulation, we can directly derive the marginal distribution of $R$ but only the conditional distribution of $\Phi$, which is $M(\mu_0, \kappa + \psi r)$. The closed form marginal distribution of $\Phi$ is difficult to obtain for any non-trivial $f(r)$ function:

$$g(\phi) = \int_0^{r_0} f(r) \frac{\exp\left( (\kappa + \psi r) \cos (\phi - \mu_0) \right)}{2\pi \psi_0 (\kappa + \psi r)} \, dr$$

$$= \int_0^{r_0} f(r) \frac{\exp\left( (\kappa + \psi r) \cos (\phi - \mu_0) \right)}{2\pi \psi_0 (\kappa + \psi r)} \, dr.$$

We have evaluated $g(\phi)$ numerically. At 72 angular design points we have estimated $g(\phi)$ by numerical integration. Figure 12 a-d(i) display these densities for a range of $\kappa$ and $\psi$ parameters. We then fitted a $M(\mu_0, \kappa)$ distribution to the design points and estimated $\kappa$ by least squares. Figure 12 a-d(ii) displays these densities. The resulting distribution is
Figure 12
Marginal $g(\phi)$ of interaction $M(\mu_0, \kappa + \psi r)$

A) 1) $\delta = 2, \lambda = 2, \kappa = 2, \psi = 2, \mu_0 = 2$

B) 1) $\delta = 5, \lambda = 2, \kappa = 2, \psi = 2, \mu_0 = 2$

A) 2) $M(\mu_0, \hat{\kappa})$ where $\hat{\kappa}$ is fitted by least squares after numerical integration $\hat{\kappa} = 3.037$

B) 2) $\hat{\kappa} = 8.145$
Figures 12 c), d)

C) 1) $\delta = 2, \lambda = 8, \kappa = 0.5, \psi = 0.5, M_0 = 2$

D) 1) $\delta = 2, \lambda = 8, \kappa = 12, \psi = 2, M_0 = 2$

C) 2) $\hat{k} = 0.918$

D) 2) $\hat{k} = 13.667$
symmetric and closely follows \( M(\mu_0, \kappa) \), except for slight differences in tail probabilities. Hence, the marginal distribution of \( \Phi \) is approximately of standard von Mises form.

It appears that the form of density (11) has not been utilised before, although Jupp and Mardia (1989), in their review, give a general bivariate density (their 2.11) which can have (11) as a submodel. Johnson & Wehrly (1978) give an example of an exponential variable related to an angular variate via an interaction as above, but their model cannot be easily extended due to their requirements for marginal distributions of fixed type.

In our case, we assume a two parameter Weibull form

\[
f(r) = \lambda \delta r^{\delta-1} \exp\{-\lambda r^\delta\}
\]

for the radial component. This form allows for closed-form normalisation and simple derivation of analytical derivatives. This is not an important restriction in practice, however, as the form of test statistics derived from \( M(\mu_0, \kappa + \psi r) \) does not depend on \( f(r) \). We have derived the moments of the \( M(\mu_0, \kappa + \psi r) \) distribution and these are given in Appendix V. We have also derived ML estimators and test statistics for this model and these will be detailed after discussion of the second model for angular-linear dependence.

The second case will be termed *delta-dependence* and is a development from the dependence model of Mardia & Sutton (1978). Here, we do not have a linear variable which is normally distributed. Instead, we must consider a two parameter linear variable where the mean depends on both parameters. In the Weibull case, the mean depends on \( \delta \) and \( \lambda \). To simplify the dependency we have chosen to allow \( \delta \) to depend on angle. Hence, the effect of such a model should be to yield a different degree of 'peakedness' depending on direction. In detail, we assume that

\[
\delta = \delta_0 + \delta_1 \cos(\phi - \mu_0)
\]

with \( f(r) \) given by (12) and

\[
g(\phi) = \exp\{\kappa \cos(\phi - \mu_0)\}/2\pi I_0(\kappa).
\]
Hence, we have a five parameter model, equivalent to (11) and (12) but with \( \delta \) dependence. Note that for the interaction model \( \Phi \sim M(\mu_0, \kappa + \psi r) \) conditional on \( r \), while the marginal distribution of \( R \) is truncated Weibull. For the delta-dependence model, \( R \) has a truncated Weibull \((\delta, \lambda)\) conditional on \( \phi \), whereas \( \Phi \) has a marginal \( M(\mu_0, \kappa) \). We do not pursue the development of estimators or tests for the delta-dependence model as closed form results are not available in a simple form for ML estimation, or tests based on likelihood properties. We do, however, fit both the above models to available data sets using numerical estimation techniques. This is discussed in the data analysis sections. We present below new results for the interaction model which are also detailed in Lawson (1990a).

### 2.3.2.1 ML estimation

We use \( \mathcal{L}_c \) with \( f \) and \( g \) defined as in (11) and (12). Thus:

\[
\Lambda(b_0(r_0)) = 1 - \exp(-\lambda r_0^\delta). \tag{13}
\]

The log-likelihood can be written as the sum of two separate forms:

\[
\mathcal{L}_{cr} = n \ln(\lambda \delta) + (\delta - 2) \sum r_i \ln(r_i) - \lambda \sum r_i^\delta - n \ln\Lambda(b_0(r_0)) \tag{14}
\]

\[
\mathcal{L}_{c\phi} = \sum (\kappa + \psi r_i) \cos(\phi_i - \mu_0) - \sum \ln I_0(\kappa + \psi r_i) - \text{const.} \tag{15}
\]

\( \mathcal{L}_{cr} \) yields standard truncated Weibull parameter estimates for \( \lambda \) and \( \delta \). \( \mathcal{L}_{c\phi} \) yields non-standard estimates for \( \kappa \), \( \psi \) and \( \mu_0 \).

The ML estimates \( \hat{\kappa}, \hat{\psi} \) and \( \hat{\mu}_0 \) are the solutions of

\[
\sum \cos(\phi_i - \hat{\mu}_0) = \sum A(\hat{\kappa} + \hat{\psi} r_i) \tag{16}
\]

\[
\sum r_i \cos(\phi_i - \hat{\mu}_0) = \sum r_i A(\hat{\kappa} + \hat{\psi} r_i) \tag{17}
\]
\[ \hat{\mu}_0 = \tan^{-1} \left( \frac{\hat{\kappa} S + \hat{\psi} R_s}{\hat{\kappa} C + \hat{\psi} R_C} \right) \]  

\( \text{where} \quad R_s = \sum r_i \sin(\phi_i) \)

\( R_c = \sum r_i \cos(\phi_i). \)

It should be noted that for \( \hat{\psi} = 0 \), the normal equations reduce to those for the M(\( \mu_0, \kappa \)) distribution. Modified Newton numerical procedures can be used to find \( \hat{\kappa}, \hat{\psi}, \hat{\mu}_0 \). In the above model it can be shown that \( \psi \) and \( \kappa \) are orthogonal to \( \mu_0 \), in the sense of Cox and Reid (1987). However, \( \psi \) is not orthogonal to \( \kappa \) and this appears to affect the estimation of the parameters. Experience of numerical estimation using direct maximisation of \( \ell_c \phi \) shows that unbounded estimation of \( \kappa \) and \( \psi \) can lead to a trade-off between the estimated values.

Thus it is important to bound the estimation to yield \( \kappa, \psi > 0 \). Given that such trading-off may relate to the nature of the derivatives of the log-likelihood w.r.t. \( \kappa \) and \( \psi \), it is beneficial to utilise numerical routines which include analytical first derivatives. We have found E\( \Phi \)4KAF (NAG subroute library) to be reliable in the estimation of a variety of simulated point densities on the range \( 0 < \kappa < 20 \) and \( 0 < \psi < 20 \). Table 2 Appendix XII displays the results of these optimisation experiments.

It is possible to fit the above interaction model on GLIM. This is discussed in Chapter 7.

2.3.2.2 Hypothesis tests

It is possible to derive a likelihood ratio test (Lr) and a score test for the analysis of interaction in the above model. We consider that an interaction test is nested within a concentration hypothesis and so:
The $L_r$ test is defined by

$$L_r = 2\left[\xi_{H_1}(\hat{\kappa}, \hat{\mu}_0, \hat{\psi}) - \xi_{H_0}(\hat{\kappa}, \hat{\mu}_0)\right]$$

(19)

where

$$\hat{\kappa} = A^{-1}(\overline{R})$$

$$\hat{\mu}_0 = \tan^{-1}\left(\frac{S}{C}\right)$$

$$\overline{R} = \frac{1}{n} \sqrt{S^2 + C^2}$$

and $\hat{\kappa}$, $\hat{\mu}_0$, $\hat{\psi}$ are the ML estimates under the full interaction model. $L_r$ has the usual asymptotic $\chi^2$ distribution. A Taylor expansion of the $L_r$ statistic yields Wald's test (Cox and Hinkley, 1974, p.324; Rao, 1973, p.417-419). In the present case this is given by:

$$W_A = \hat{\psi}^2 \left[ \sum r_i^2 A' (\hat{\kappa} + \hat{\psi} r_i) - \frac{(\sum r_i A' (\hat{\kappa} + \hat{\psi} r_i))^2}{\sum A' (\hat{\kappa} + \hat{\psi} r_i)} \right].$$

(20)

This test requires the estimation of $\kappa$, and $\psi$ under the full model, but does provide a simpler test statistic, compared to the LR test. $W_A$ also has an asymptotic $\chi^2$ distribution.

It is often possible to avoid estimation of parameters by using a score test, which approximates the LR test by the gradient of $\xi_{H_1}$ with respect to $\psi$ evaluated at $\psi = 0$. The formal definition of this test is also given in Cox and Hinkley (op. cit.) and Rao (op. cit.). In the present case we can derive:

$$W_S = U_0^{-2} I_{\psi\psi}^{-1}$$

(21)

where $I_{\psi\psi}$ is the expected information under $H_0$, and
\[ r_{\psi,\psi}^{-1} = A'(\hat{\kappa}) (\sum r_i^2 - (\Sigma r)^2/n) \] (22)

\[ A'(\hat{\kappa}) = 1 - A(\hat{\kappa})/\hat{\kappa} - A(\hat{\kappa})^2 \] (23)

\[ U_0 = \Sigma r_i \cos(\phi_i - \bar{X}_0) - \Sigma r_i \bar{R} \] (24)

and

\[ A(\hat{\kappa}) = \bar{R} \text{ under } H_0. \]

Hence,

\[ W_A = \frac{\left( \Sigma r_i \cos(\phi_i - \bar{X}_0) - \Sigma r_i \bar{R} \right)^2}{A'(\hat{\kappa}) \left( \sum r_i^2 - \frac{(\Sigma r)^2}{n} \right)} \] (25)

This \( W_S \) test also has an asymptotic \( \chi^2 \) distribution. Each of the three tests \( (L_r, W_A, W_S) \) are often evaluated in the form \( W_i' \) as the signed square root of \( W_i \), to allow examination of any signed effects, and \( W_i' \) should have \( \approx N(0,1) \).

Detailed derivation of the Wald and Score tests is given in Appendix VI. Note that none of the above tests depend on the parametric form of \( f(r) \) and in that sense the tests are 'non-parametric'. This property is due to the fact that the log-likelihood can be written as the sum of two separate forms \( (\xi_c r^2 \text{ and } \xi_c \phi) \).

The statistic \( W_S \) is closely related to the general circular-linear correlation coefficient proposed by Mardia (1975), Jupp and Mardia (1980), and Liddell and Ord (1978) amongst others. It should be noted that the multiple correlation coefficient between \( r \) and \( (\cos \phi, \sin \phi) \) can be written

\[ \rho_{r\phi}^2 = \frac{[E_{rc} - E_r E_c]^2}{V_r V_c} \]

when \( \mu_0 = 0 \) and the sin moments disappear. Here

\[ E_{rc} = E(r \cos(\phi)) \]

\[ E_r = E(r) \]
$E_c = E(\cos(\phi))$

$V_r, V_c =$ variances of $r$ and $\cos \phi$ respectively.

Now, under $H_0: M(\mu_0, \kappa)$ the following moments are derived (see Appendix V for details).

$V_c = A'(\kappa); \quad V_r = \sigma_r^2; \quad E_c = \bar{R}$

$E_{rc} = \ln(rA(\kappa))/\Lambda(b_0(r_0))$

$E_r = \ln(r)/\Lambda(b_0(r_0))$

where $\ln(\cdot) = \int_0^{r_0} f(r) dr$.

When estimates of the above are substituted for these moments the statistic $W_S$ is derived.

Hence, $W_S$ is a measure of circular-linear correlation, and its form does not depend on the form of $f(r)$.

In a subsequent section, we examine numerically the power of the $L_r$ and $W_S$ statistics in comparison to a variety of related tests. In addition we apply the above model to a number of data examples.

Appendix I and II details the computational methods and programs used for ML estimation, test statistic calculation and numerical power studies.
CHAPTER 3

3. Continuous Model: Extensions

In the last Chapter a basic HEPP model for a point process was presented. Estimation methods and test construction were also discussed. In the form considered, this model can be criticised on a number of grounds. First, the model does not allow for possible underlying heterogeneity in the environment. Such heterogeneity could induce apparent clustering or sparseness within a point set. For example, in the ecological examples, variation in soil characteristics (data sets D, F) and/or local population density (data sets D, E, F) may cause such effects. In the epidemiological examples (data sets A, B, C) variation in the underlying population density and its age x sex structure will determine the propensity of locations to display point events. This latter case differs from the pure heterogeneity of the ecological examples in that existing individuals are 'marked' by a disease and a mapped realisation only consists of marked individuals.

3.1 Observed Heterogeneity

In some cases, it may be possible to observe heterogeneity by measurement of environmental covariates at specific locations. For the time domain, Cox (1972) has developed analysis of such 'modulated' point processes. This analysis can be carried over to the spatial case. We define:

\[ \lambda^*(r, \phi) = \lambda(r, \phi) h(\mathbf{z} \mathbf{\beta}) \]  

where

- \( \mathbf{\beta} = (1 \times p) \) parameter vector
- \( \mathbf{z} = (p \times 1) \) vector of covariates
- \( p = \) number of covariates.
Here \( \{z_i\} \) are observable at the point event locations. If they are not available they must be interpolated. This may only be sensible if \( z \) consists of spatially continuous variates such as population density or 'expected' mortality, suitably defined.

Note that in the analysis of multiplicative hazard models in Epidemiology, the baseline hazard rate and time-dependent standardised mortality ratio (SMR) can be smoothed by the kernel method to yield interpolant values at locations of mortality events (Breslow and Day (1987), 198-199).

Interpolation of irregularly spaced spatial data is the subject of a large literature (see, e.g. Ripley (1981), Ch 4). For observations made on a continuous variate the method of Kriging is often used. Universal Kriging requires the specification of a deterministic trend (trend surface) and a covariance for the spatial structure of the variate. Kriging is 'optimal' in the sense that it provides a linear estimator which is unbiased at the data points.

An alternative interpolation method which can be used for covariates which form point processes themselves, could be a kernel method in 2-d, such as given in Byth (1982) or Diggle (1981). We define:

\[
\lambda^*(r, \phi) = \tau(x, y) \lambda(r, \phi)
\]

and

\[
\lambda(x, y) = (2\pi N)^{-1} h^{-2} \sum_{i=1}^{N} \exp \left\{ -\frac{1}{2h^2} \left( (x-x_i)^2 + (y-y_i)^2 \right) \right\}
\]

where \( x = r \cos \phi; \quad y = r \sin \phi \)

\((x_i, y_i)\) are the \( N \)-set of observed point locations

\( h = \) smoothing constant.

This allows the possible use of case-control methods for spatial point processes (see, Diggle (1989)). The use of (27) in maximum likelihood estimation would require the numerical integration of \( \Lambda(h_0(r_0)) \).
An extension of the above ideas is to interpolate an SMR or population density spatially using the kernel method. Using a Gaussian kernel of width h, we can define:

\[ \hat{\theta}(x, y) = C \sum_{i=1}^{N} n_i \exp \left\{ -\frac{1}{2h^2} ((x-x_i)^2 + (y-y_i)^2) \right\} \]

where

\[ C = (2\pi N)^{-1} h^{-2} \]

\[ n_i^* = \begin{cases} O_i \text{ or } E_i & \text{for SMR in a region } i \\ n_i \text{ or } a_i & \text{for population density in a region } i \end{cases} \]

and

\[ \lambda^*(r, \phi) = \hat{\theta}(x, y) \lambda(r, \phi) \] (28)

ML estimation for the above interpolated intensity models (27) and (28), will be pursued further in data examples in a later chapter. Next we consider hypothesis testing for the above models. For the full covariate model (26) we further specify \( h(z_\beta) = \exp \{ z_\beta \} \). Hence, the resulting log-likelihood is:

\[ \ell_{cc} = \sum \lambda \ln \lambda(r_i, \phi_i) + \sum z_i \beta \cdot n \ln \Lambda(b_0(r_0)) \] (29)

and

\[ \Lambda(b_0(r_0)) = \int_{b_0(r_0)} \lambda(r, \phi) \exp \{ z_\beta \} \, dr. \] (30)

Define \( I_A(u) = \int_{b_0(r_0)} u \, dr \) and denote \( \alpha = (k \times 1) \) vector of intensity parameters and

\[ \lambda(r, \phi) = \exp \{ f \alpha \} \] (31)

where \( f = (1 \times k) \) vector of spatial variables. The parameterisation of \( f \) should consist of functions of \((r, \phi)\) which describe the relation with the fixed point, for example

\[ f^T = [r, \log r, \cos \phi, \sin \phi, r \cos \phi, r \sin \phi]. \]
Hence, the score vector for $\alpha_j$ is
\[
\frac{\partial \xi \epsilon}{\partial \alpha_j} = \sum f_j(r_i, \phi_i) - n A_p(f_j(r, \phi))
\] (32)
where
\[
A_p(f_j(r, \phi)) = E(f_j(r, \phi)) = \frac{I_A(f_j(r, \phi) \lambda^*(r, \phi))}{I_A(\lambda^*(r, \phi))}
\] (33)
and the observed information matrix has elements
\[
I_{\alpha_j \alpha_l} = n[A_p(f_j(r, \phi)) f_j(r, \phi) - A_p(f_j(r, \phi)) A_p(f_j(r, \phi))]
\] (34)
\[
= n[E(f_j f_j) - E(f_j) E(f_j)]
\]
where $f_j \equiv f_j(r, \phi)$.

Hence, under $H_0: \alpha = \beta = 0$, $A_p$ represents a simple areal average. If we are only interested in hypotheses concerning $\alpha$ then covariates $\zeta$ are 'nuisance' covariates and $\beta$ must be estimated under the null $H_0: \alpha = 0$. In this case, the information matrix will have additional elements:
\[
I_{\alpha_j \beta_\lambda} = n\{A_p(f_j(r, \phi) \zeta_\lambda(r, \phi)) - A_p(f_j(r, \phi)) A_p(\zeta_\lambda(r, \phi))\}
\] (35)
and I can be partitioned into
\[
I = \begin{bmatrix}
I_{\alpha \alpha} & I_{\alpha \beta} \\
I_{\beta \alpha}^T & I_{\beta \beta}
\end{bmatrix}
\]
From this score tests for the vector $\alpha$ or individual elements $\alpha_i$ can be constructed but closed form results do not arise easily.

A general test for $H_0: \alpha = 0$ against $H_1: \alpha \neq 0$ could be
\[
W_\alpha = U_{\alpha 0}^T \Gamma_{\alpha \alpha}^{-1}(0) U_{\alpha 0} \sim \chi^2_k
\] (36)
and
\[
U_{\alpha 0} = \frac{\partial \xi \epsilon}{\partial \alpha} |_{\alpha = 0}
\]
and 
\[ I_{\alpha\alpha}^{-1}(0) = (I_{\alpha\alpha}(0) - I_{\alpha\beta}^T(0) I_{\beta\beta}^{-1}(0) I_{\beta\beta}(0)) \]

where 
\[ I_{\alpha\alpha}^{-1}(0) = I_{\alpha\alpha}^{-1}_{|\alpha=0}. \]

Nuisance parameters cause problems with (29) or the test \( W_\alpha \). One method, which avoids direct estimation of \( \beta \), is profile likelihood (PL). This method consists of estimation of a subset of parameters (\( \beta \) say) by maximum likelihood, while keeping another set constant (\( \alpha \) say). The PL estimates \( \hat{\beta} \) are substituted into the full likelihood and thence likelihood methods can be applied to \( \alpha \). In terms of estimation, PL does not guarantee full ML estimates, it relies on the coincidence of a unimodal peak in both dimensions. However, PL can allow easier analytical treatment of otherwise difficult likelihood problems (see, for example, Wecker and Ansley (1983)).

Another method for treatment of nuisance parameters is to consider the parameter values to be realisations of a random variable with a predefined or assumed (prior) distribution. It is possible to simply 'integrate out' the parameters to leave the marginal distribution of the remaining parameters for use in estimation and testing (see, Cox and Hinkley (1974), p.401). We have not pursued these possibilities in connection with the log-likelihood \( l_{cc} \) or the score test \( W_\alpha \). However, these methods are considered in Section 3.2 where we discuss unobserved heterogeneity.

### 3.1.1 Case-Control Score Tests

The general score test for \( H_0 : \alpha = 0 \) in equation (36), can be used for testing components of trend when a case-control is present. Here we present a number of new score tests for radial and angular effects in the presence of a case-control.
For simplicity, we define \( \hat{\lambda}(x) \) to be the case-control intensity function. The full log-likelihood from (27) is

\[
\lambda_{ker} = \sum \lambda_n \hat{\lambda}(x_i) + \sum \lambda_n \lambda_i - n \ln \int \hat{\lambda}(x) \lambda(r) \, dr
\]

and we distinguish four intensity models:

a) \( \lambda(r) = \exp(-\beta r)/r \)

b) \( \lambda(\phi) = \exp(\kappa \cos(\phi - \mu_0))/r \)

c) \( \lambda(r) = \exp(\delta \ln r - \beta r)/r \)

d) \( \lambda(r,\phi) = \exp((\kappa + \psi r) \cos(\phi - \mu_0))/r \).

3.1.1.1 Case a) Radial Trend

For this intensity, we have

\[
\lambda_{\beta|H_0} = \sum t_i - n \int \hat{\lambda}(x) r \, dr / \int \hat{\lambda}(x) \, dr
\]

and

\[
\lambda_{\beta|H_0} = n \left[ \int \hat{\lambda}(x) r^2 \, dr - \left( \int \hat{\lambda}(x) r \, dr \right)^2 / \left( \int \hat{\lambda}(x) \, dr \right)^2 \right]
\]

and

\[
W_{\beta} = \frac{\bar{r} - E(r)}{\sqrt{\frac{1}{n}(E(r^2) - E(r)^2)}} \text{ as } N(0,1)
\]

where

\[
E(\cdot) = \int \hat{\lambda}(x) \, dr / \int \hat{\lambda}(x) \, dr.
\]
3.1.1.2 Case b) Angular Concentration

Under $H_0: \kappa = 0$, $\hat{\mu}_0$ is the solution of

$$\sum \sin(\phi_i - \hat{\mu}_0) = n \ E(\sin(\phi_i - \hat{\mu}_0))$$

and

$$\chi^2_{\kappa H_0} = \sum \cos(\phi_i - \hat{\mu}_0) - n \ E(\cos(\phi_i - \hat{\mu}_0))$$

and the Fisher information ($I_{\kappa\kappa}$) is

$$I_{\kappa\kappa} = n(E(\cos^2(\phi - \mu_0)) - E(\cos(\phi - \mu_0))^2).$$

Hence,

$$W_{\kappa} = \left[ \frac{1}{n} \sum \cos(\phi - \hat{\mu}_0) - E(\cos(\phi - \hat{\mu}_0)) \right] / p$$

where

$$p = \frac{1}{n} I_{\kappa\kappa}^{1/2}.$$

3.1.1.3 Case c) Peaked-Radial Effect

We consider a test for $\delta$ with $\beta$ as a nuisance parameter. The estimate of $\beta$ under $H_0: \delta = 0$ is the solution of

$$\frac{\int \hat{\tau}(x) r \ e^{-\beta r} \ dr}{\int \hat{\tau}(x) \ e^{-\beta r} \ dr}$$

and

$$\chi^2_{\delta H_0} = \sum \ln r_i - n \frac{\int \hat{\tau}(x) \ \ln r \ e^{-\beta r} \ dr}{\int \hat{\tau}(x) \ e^{-\beta r} \ dr}$$

and define $\lambda_0 = e^{-\beta r}$

$$\tau \equiv \hat{\tau}(x)$$

and

$$I(\cdot) = \int \cdot \ d\tau$$
\[ W_\delta = \frac{\sum \ln r_i/n - I(\tau \ln r \lambda_0)/I(\tau \lambda_0)}{\sqrt{\frac{1}{n} I^2 \delta}} \]

\[ I^{\delta\delta} = I^{\delta\delta} - \frac{1}{I^{\delta\delta}} \frac{I^{\beta\beta}}{I^{\beta\beta}} \]

and

\[ I^{\delta\beta} = n \left\{ \frac{I(\tau \ln r \lambda_0) I(\tau \lambda_0)}{I^2(\tau \lambda_0)} - \frac{I(\tau \ln r \lambda_0)}{I(\tau \lambda_0)} \right\} \]

\[ I^{\delta\delta} = n \left\{ \frac{I(\tau(\ln r)^2 \lambda_0)}{I(\tau \lambda_0)} - \frac{I^2(\tau \lambda_0)}{I^2(\tau \lambda_0)} \right\} \]

\[ I^{\beta\beta} = n \left\{ \frac{I(\tau^2 \lambda_0)}{I(\tau \lambda_0)} - \frac{I^2(\tau \lambda_0)}{I^2(\tau \lambda_0)} \right\} \]

where \( \lambda_0 = \lambda(\tau, \phi)|_{H_0} \).

3.1.1.4 Case d) Interaction Effect

In this case a relatively complex test statistic is derived. Here, \( \mu_0 \) and \( \kappa \) must be estimated under \( H_0 : \psi = 0 \), and \( \hat{\mu}_0 \) and \( \hat{\kappa} \) are the solutions of:

\[ \frac{1}{n} \sum \sin(\phi_i - \mu_0) = \frac{\int \tau \sin(\phi_i - \mu_0) \lambda_0 d\tau}{\int \tau \lambda_0 d\tau} \]

where \( \lambda_0 = \lambda(\tau, \phi)|_{H_0} \)

and

\[ \frac{1}{n} \sum \cos(\phi_i - \mu_0) = \frac{\int \tau \cos(\phi_i - \mu_0) \lambda_0 d\tau}{\int \tau \lambda_0 d\tau} \]
and the score test is

$$W_\psi = \frac{1}{n} \sum r_i \cos(\phi_i - \mu_0) - \frac{\int \tau \cos(\phi_1 - \mu_0) \lambda_0 d\tau}{\int \tau \lambda_0 d\tau} \sqrt{\frac{1}{n} I_{\psi} I_{\psi}}$$

where $I_{\psi}$ is the leading element of the inverse of the observed Fisher information matrix i.e.

$$I_{\psi} = [I_{\psi\psi} - I_{\psi\theta} I_{\theta\theta} I_{\theta\psi}]^{-1}$$

where $\theta$ represents the parameters $\kappa, \mu_0$

and

$$I_{\psi\psi} = n \left\{ \frac{\int \tau r^2 \cos^2(\phi_1 - \mu_0) \lambda_0 d\tau}{\int \tau \lambda_0 d\tau} - \left( \frac{\int \tau r \cos(\phi_1 - \mu_0) \lambda_0 d\tau}{\int \tau \lambda_0 d\tau} \right)^2 \right\}$$

and

$$I_{\psi\theta} = [I_{\psi\kappa}]$$

$$I_{\theta\psi} = [I_{\psi\mu_0}]$$

$$I_{\theta\theta} = \begin{bmatrix} I_{\kappa\kappa} & I_{\kappa\mu_0} \\ I_{\mu_0\kappa} & I_{\mu_0\mu_0} \end{bmatrix}$$

and, here

$$I_{\psi} = \left( I_{\psi\psi} - \frac{I_{\psi\kappa}^2 I_{\mu_0\mu_0} - 2 I_{\psi\kappa} I_{\psi\mu_0} I_{\mu_0\kappa} + I_{\kappa\kappa} I_{\psi\mu_0}^2}{I_{\kappa\kappa} I_{\mu_0\mu_0} - (I_{\mu_0\kappa})^2} \right)^{-1}$$

The expressions for the other information matrix elements are not given here due to their length. Appendix IX details all these elements.

### 3.2 Unobserved Heterogeneity
It is sometimes the case that data may not be available which relates to environmental heterogeneity. However, it may be considered that such effects are present a priori, and should be incorporated in a modelling approach. Two basic methods can be employed.

3.2.1 Harmonic Intensity Terms

It is possible to model general surface roughness by the inclusion of a number of harmonic terms in $\lambda(r,\phi)$. These terms should be functions of cartesian coordinates as general surface variation should not be related to a fixed point. Hence, we could define:

$$\lambda(r,\phi) = \exp\left\{ f + \sum_{k=1}^{m} c_k \cos k(\theta - \mu_x) + \sum_{k=1}^{m} d_k \sin k(\theta - \mu_x) \right\}$$

where

$$\theta = 2\pi x / x_{\text{max}}$$

$$\epsilon = 2\pi y / y_{\text{max}}$$

and

$$x = r \cos \phi, \quad y = r \sin \phi.$$ 

The fitting of such a model requires a) the introduction of a large number of parameters. b) The value $m$ is unspecified and hence must be assumed or estimated during fitting. c) $A(b_0(r_0))$ must be found numerically. The large number of parameters usually required to fit these harmonic terms yields a lack of parsimony.

We have not developed this approach in the present work, although it may be useful to develop tests for added harmonic terms as have been developed by Cox (1975) and Mardia et al. (1984) for the von Mises-Fisher distribution on the circle and sphere.
3.2.2 Spatial Prior Structure for Intensities

A more parsimonious approach to modelling unobserved heterogeneity is to consider some prior spatial structure exists in the process observed. For example, if a population is clustered in settlements spatial trend will only be observed between or within clusters where settlements occur.

3.2.2.1 Cox Process Model

One approach to this problem is to assume

a) $\Lambda(r)$ is a non-negative-valued Stochastic Process.

b) Conditional on $\{\Lambda(r) = \lambda(r)\}$, the events form a HEPP with intensity function $\lambda(r)$.

a) and b) above represent the definition of a Cox Process (Diggle, 1983, p.58). This formulation allows the specification of an underlying random structure. Diggle (op. cit.) notes that Poisson cluster processes can be subsumed within this class, and Cox and Isham (1980, p.73) note the possibility of the use of stationary Gaussian processes as models for $\Lambda(r)$.

We assume that $\Lambda(r)$ is a Cox Process with $E(\Lambda(r)) = \lambda(r)$ and, following Snyder (1975, p.301) we define the likelihood of a realisation as

$$L^* = E\left\{ \prod_{i=1}^{n} \Lambda(\tau_i, \phi_i) \exp \left[- \int_{b_0(r_0)}^{\Lambda(r)} \lambda(r)dr\right] \right\}$$  \hspace{1cm} (37)

The expectation in (37) is taken over the process $\Lambda(r)$, and in general is difficult to find. Snyder (1975, p.287) gives some examples in the time domain of random scaling models for which (37) can be evaluated exactly. However random scaling models do not lead to
any change in the *conditional* distributions of events given \( n \), as the random scaling is not spatially dependent. In our examples the prior structure is spatially dependent. One possible approach to evaluation of \( L^* \) is to approximate \( L^* \) by a Taylor series around the expectation \( \lambda(\tau) \):

\[
L^* = E\{L(\lambda(\tau)) + L'(\lambda(\tau))(\Lambda(\tau) - \lambda(\tau))
\]
\[
+ \frac{1}{2} L''(\lambda(\tau))(\Lambda(\tau) - \lambda(\tau))^2\}
\]
to second order
\[
= L(\lambda) + L''(\lambda) \frac{\sigma^2}{2}.
\]

Hence, the log likelihood is

\[
\ell^* = \ell_n \left[ L(\lambda) + L''(\lambda) \frac{\sigma^2}{2} \right]
\]

where

\[
\lambda \equiv \lambda(\tau).
\]

Now \( \ell(\lambda) = \sum \ell_n \lambda - \int_{b_0(t_0)} \lambda \, dt \). Hence, to second order, the log-likelihood \( \ell^* \) can be approximated by the log of the likelihood of a HEPP model (\( \ell(\lambda) \)) with additional terms representing the variance \( \left( \frac{\sigma^2}{2} \right) \) and covariances of the 'driving' process \( \Lambda(\tau) \). The difficulty of evaluating the term \( L''(\lambda(\tau)) \) for practical modelling, has led to an alternative method for specifying prior structure.

### 3.2.2.2 A Bayesian Spatial Prior (BSP) Model

Instead of considering a Cox Process, we consider \( \lambda(\tau) = \exp\{g(\tau)\} \) where \( g(\tau) \) is a linear function of \( f(x) \), and we call \( g(\tau) \) a 'linear predictor'. We consider \( \lambda(\tau) \) to be the intensity of a HEPP model. However, we now consider \( g(\tau) \) to have a prior spatial
structure. We do not restrict \( g(r) \) to non-negativity, as the log-link ensures positivity of \( \lambda(r) \).

As general environmental heterogeneity should not be associated with cluster or regular processes we confine attention to a Spatial Gaussian prior distribution for \( g(r) \).

Formally, we utilise \( \xi_u \) [eqn (5)] as a posterior log likelihood, and now re-define our variables in vector/matrix notation i.e.

\[
\begin{align*}
g_n &= (n \times 1) \text{ vector of linear predictor} \\
\xi_n &= (n \times 1) \text{ vector of spatial trend surface} = (f_n \xi) \\
t &= (n \times 1) \text{ unit vector.}
\end{align*}
\]

Hence, the prior distribution of \( g_n \) will be MVN \((f_n \xi, \kappa_n)\) and

\[
P_r(g_n; \varpi) \propto \exp \left\{ -\frac{1}{2} (g_n - \xi_n)^T \kappa_n^{-1} (g_n - \xi_n) \right\}
\]

where \( \kappa_n \) is the \((n \times n)\) covariance matrix of the process evaluated at the observed events and \( \kappa_{ij} = \sigma^2 \exp\left\{-d(i,j)/R_a\right\} \)

where \( d(i,j) = \text{euclidean distance between points } i \text{ and } j \).

In this formulation the spatial prior structure of \( g_n \) is controlled by the variance of the Gaussian Process \((\sigma^2)\) and by the covariance range parameter \((R_a)\).

Now the conditional log likelihood for data \( \{r_i, \phi_i\} \) given the realisation of \( g_n \) is given by

\[
\ell(g_n) = t'g_n - \int_{b_0(r_0)} e^{g(r)} \, dr.
\]

Hence, the posterior distribution of \( g_n \) for the data is

\[
P_0(g_n; \varpi) \propto \exp\{\ell(g_n)\} P_r(g_n; \varpi).
\]
Now the posterior is a function of an unknown vector $g_n$ and a vector of parameters $\theta$. We wish to estimate $\theta$ and we are not usually interested in $g_n$, although this vector can be estimated to yield a 'shrunken' estimator at each data point. Hence $g_n$ is usually regarded as a nuisance vector.

In our prior distribution there is also a variance $\sigma^2$ and a covariance range parameter $R_a$. We assume initially, that $\sigma^2$ and $R_a$ are known. A sensitivity analysis has suggested that exact specification of $R_a$ is not important for our covariance model (Warnes, 1986), and hence exact estimation may not be important. ML estimation of $R_a$ may be problematic as multiple maxima in the likelihood have been reported by Ripley, 1988, p.15-19 (see also Mardia and Marshall (1984), Mardia and Watkins (1989) and Warnes and Ripley (1987)). In some simulation experiments, the author has generated a series of thinned Gaussian-prior point processes. These experiments have suggested that perturbation of $\sigma^2$ has more effect on realisations than $R_a$ and they are relatively insensitive to $R_a$ variation. It is possible to estimate $\sigma^2$ by profile likelihood assuming $R_a$ known.

We have investigated two possible estimation schemes for the above model, with $\sigma^2$ and $R_a$ known.

The first scheme allows for estimation of $g_n$ with $\theta$ known, then to estimate $\theta$ after $g_n$ is estimated. We can use maximum a posteriori estimation (MAP) for this purpose. Warnes (1987) describes a similar estimation problem, where an intermediate parameter (our $g_n$) must be estimated before the parameter vector of ultimate interest. MAP estimation in our case amounts to maximising (40) wrt $g_n$. The log likelihood $\ell(g_n)$ contains an integral defined on the continuous function $g(r)$. As our maximisation is wrt a realisation $g_n$, we require a representation of $g(r)$ in terms of $g_n$. We can replace the integral in (39) by a numerical scheme which includes $g$ evaluated at the $n$ locations of the realisation. For example:
\[
\int_{b_0(r_0)} e^{g(x)} \, dx = \sum_{i=1}^{M} W_i^* e^{g_i}
\]

where the \(W_i^*\) are integration weights, and we include a set of \(m\) dummy points so that \(M = n + m\) \((m \geq 0)\). The right hand side of equation (41) can be rewritten as \(W_M e^{g_M}\) where \(W_M = (M \times M)\) diagonal matrix with

\[
W_{ii} = W_i^*.
\]

Here we have replaced an integral by a weighted function of \(g(x)\) evaluated at the data points.

For the simplest case, when \(M = n\) gives an adequate approximation to the integral in (41), we have that the posterior likelihood \((L_p)\) is given as:

\[
L_p \propto \exp \{t'g_n - W_n e^{g_n} - \frac{1}{2} (g_n - \xi_n)^T \kappa_n^{-1} (g_n - \xi_n)\}.
\]

Now for \(\xi_p = \log L_p\) we have

\[
\frac{\partial \xi_p}{\partial g_n} = t - W_n e^{g_n} - \kappa_n^{-1} (g_n - \xi_n) = \Omega
\]

\[
\Rightarrow g_n - \xi_n = \kappa_n(t - W_n e^{g_n}).
\]

Hence, for fixed \(\Omega\) the MAP estimate of \(g_n\) is the solution of

\[
\hat{g}_n = f_n \Omega + \kappa_n(t - W_n e^{\hat{g}_n}).
\]

We can obtain the profile likelihood of (42) with \(\hat{g}_n\) substituted and thence we obtain standard GLS estimates for \(\Omega\):

\[
\hat{\Omega} = (t_n^T \kappa_n^{-1} f_n)^{-1} t_n^T \kappa_n^{-1} \hat{g}_n
\]

with

\[
\text{cov}(\hat{\Omega}) = (t_n^T \kappa_n^{-1} f_n)^{-1}.
\]

Note that \(\hat{\Omega}\) can be found easily by OLS and Cholesky Decomposition of \(\kappa_n\) (see, e.g. Green (1984)). In addition, it should be noted that if \(A(b_0(r_0))\) were approximated by a
product rule method (see Berman and Rolf Turner (1988)) which involved function evaluations at a set of $m > 0$ dummy points (e.g. $g_m$, say), then (43) is replaced by a coupled set of equations for $\hat{g}_n$ and $\hat{g}_m$ respectively. The variance $\sigma^2$ can also be estimated from a profile likelihood once $\alpha$ has been estimated. Hence, it can be seen that

$$\sigma^2 = \frac{1}{n} (\hat{g}_n - f_n \hat{g})^T \kappa_n^{-1} (\hat{g}_n - f_n \hat{g})$$

(45)

where $\kappa_n = \frac{1}{\sigma^2} \kappa_n$.

The above solution for $\hat{g}_n$ in (43) is very close to that derived by Leonard (1978) for the time-domain case. That author obtained a general integral equation for a time-dependent Poisson process of the form

$$\hat{g}(t) = \mu(t) + \sum_{i=1}^{m} \kappa(x_i, t) - \int_{a}^{b} \kappa(s, t) \exp \{\hat{g}(s)\} ds$$

for $t \in [a, b]$. (46)

With the discretisation of the above integral the time case is exactly equivalent to (43) except that $\mu(t)$ is parameterised in our example and we consider a spatial process rather than a time-domain process.

Two important problems arise from the solution of the MAP estimation problem in (43), (44) and (45). First, the structure of (43) is that of $n$ coupled nonlinear equations, and hence solution is not straightforward. The structure of (43) is essentially that found in image restoration problems when a spatial prior is used. For large pixel arrays this optimisation problem is often solved by Stochastic relaxation or simulated annealing (Besag (1986); Geman and Geman (1984)). Geman and McLure (1987) have examined a continuous state space problem similar in structure to the present case where a posterior distribution consists of a Gibbs energy function and a Poisson likelihood term. Instead of utilising a two stage approach as above, they directly minimise $-\log L_p$ for their model,
using simulated annealing. This approach has not been adopted here, as it is possible to simplify the estimation problem under consideration by use of Taylor expansion methods.

In equation (42), $g_n$ is a nuisance vector and one alternative method of estimation is to 'integrate out' this vector over its range (see, for example, Warnes (1987, p.98)). The integration of (42) is difficult due to the normalisation. However, by the use of a Taylor expansion of the conditional likelihood (39), we derive a quadratic form, thus:

$$\mathcal{L}(g) = \mathcal{L}(\bar{g}) + (g - \bar{g})' \mathcal{L}'(\bar{g}) + \frac{1}{2} (g - \bar{g})' \mathcal{L}''(\bar{g})(g - \bar{g})$$

where $\bar{g} \equiv g_n$ (for short)

and $\mathcal{L}(\cdot) \equiv \mathcal{L}(\text{data } \cdot)$.

Hence, the log posterior likelihood of $g$ from (42) is

$$\mathcal{L}^*_g = \text{const} + \mathcal{L}(g) - \frac{1}{2} (g - \xi)^' \kappa_n^{-1}(g - \xi).$$

The posterior mode is

$$\bar{g}_{\text{mode}} = R^{-1}T$$

where

$$R = \kappa_n^{-1} \mathcal{L}''(\bar{g})$$

and

$$T = \kappa_n^{-1} \xi - \mathcal{L}''(\bar{g})\bar{g} + \mathcal{L}'(\bar{g}).$$

The variance-covariance matrix is $R^{-1}$. It is now possible to express (42) as

$$L_g^* = A \exp\{-\frac{1}{2} g' R g + g'T\}$$

where

$$A = \frac{|\kappa_n^{-1}|^{1/2}}{(2\pi)^{k/2}} \exp\{-\frac{1}{2} \xi' \kappa_n^{-1} \xi + f(\bar{g})\}.$$}

and

$$f(\bar{g}) = \mathcal{L}(\bar{g}) - \bar{g}' \mathcal{L}'(\bar{g})$$
It is possible, now, to 'integrate out' \( g \) from \( L^*_g \) as it is a function of a quadratic form in \( g \).

The result of

\[
\int \exp\left(-\frac{1}{2} g' R g + g' T \right) dg
\]

is well known. It is

\[
(2\pi)^k/2 |R|^{-1/2} \exp\left(\frac{1}{2} T' R^{-1} T \right).
\]

Hence, after integration, the posterior density is

\[
L^*_1 = |\kappa_n R|^{-1/2} \exp\left(-\frac{1}{2} \xi' \kappa_n^{-1} \xi + \frac{1}{2} T' R^{-1} T + \ell(\bar{g}) - \bar{g} \ell''(\bar{g}) \right).
\]

Define \( \tau = \ell'(\bar{g}) - \ell''(\bar{g}) \bar{g} \)

and

\[
T = \kappa_n^{-1} \xi + \tau \quad \text{and} \quad V_g = -\ell''(\bar{g}).
\]

Log \( L^*_1 \) is a quadratic form in \( \xi \). Differentiation wrt \( \xi \) of

\[
-\frac{1}{2} \xi' \kappa_n^{-1} \xi + \frac{1}{2} \xi' \kappa_n^{-1} R^{-1} \kappa_n^{-1} \xi + \xi' \kappa_n^{-1} R^{-1} \tau
\]

shows that the minimum occurs at

\[
\hat{\xi} = \left(\kappa_n + V_g^{-1} \right)^{-1} f_n'(\kappa_n + V_g^{-1})^{-1} \bar{g}
\]

with

\[
\text{cov}(\hat{\xi}) = \left( f_n'[\kappa_n + V_g^{-1}]^{-1} f_n \right)^{-1}.
\]

Clayton and Kaldor (1987) use a similar method for a conditional Poisson likelihood with a spatial Gaussian prior. However, they did not integrate out \( g \) and instead used the iterative EM algorithm to estimate \( g \) mode, which is a 'shrunken' estimator of \( g \). In their example, the estimator \( \bar{g} \) was the log (SMR) for an enumeration district. Our method yields direct one-step estimates of \( g \) and thence \( g \) mode based on \( \bar{g} \). For the case of a Poisson process unconditional likelihood we have
\[ \ell(g) = \sum g_i - \sum W_i e^{g_i} \]

and we require a saturated estimate of \( g_i \). This estimate is the solution of \( \ell_{g_i}(g) = 0 \)

\[ \ell_{g_i}(g) = 1 - W_i e^{g_i} \]

and hence \( g_i = -\ln W_i. \)

Often it is possible to replace \( W_i \) by \( A_i \), the Dirichlet tile area for the \( i \)th point. This estimate is intuitively plausible as \( \tilde{\lambda}_i \) would be \( \frac{1}{A_i} \), which is a local intensity estimate.

In addition, \( \ell'(\tilde{\lambda}) = 0 \) and \( \ell''(\tilde{\lambda}) = -I_n \) for this model so that

\[ \tilde{g}_{\text{mode}} = (\kappa_n^{-1} + I_n)^{-1} \cdot (\kappa_n^{-1} \tilde{\lambda} - \ln \Delta) \]

and \( \tau = -\ln \Delta \) and \( V_g = I_n. \)

Note that \( \tilde{g}_{\text{mode}} \) is the empirical Bayes estimate of \( g. \)

One disadvantage of either of the above estimation schemes, is the use of approximation (41) where the normalising constant is replaced by a weighted sum. Usually multiple integrals over areas are replaced by product rule schemes, where each dimension has a one-dimensional scheme associated with it and hence the integral is approximated by a sum of the form \( \sum_i \sum_j W_i W_j \lambda_{ij} \), where \( i,j \) denote the location in each dimension and \( \lambda_{ij} \) is the intensity evaluated at \((i,j)\) (see, for example, Davis and Rabinowitz (1984), Ch. 5). Berman and Rolf Turner (1988) suggested the use of a single \( n \)-set of weights associated with the \( n \) point locations of a point process. These weights could be based either on the areas of Dirichlet tesselation tiles or on average areas of Delaunay triangles (T-weights). This suggestion requires less data storage than the product rule method for their example. However, this method suffers from two related disadvantages. First, the method assumes a piecewise constant approximation over each area. Second, if \( n \)
is small large areas may have little support in the approximation. This latter effect is further compounded by a piecewise constant weighting system.

While the use of T-weights is advantageous from a data storage point of view, the above approximation methods could be improved. We discuss possible improvements in Chapter 7, where we evaluate the relative accuracy of the approximations. We do not pursue this problem at this point but we note that improved T-weights have been used in our analyses based on GLIM and Spatial Prior models.

The approximation of the normalising integral (41) can be improved by the addition of a set of m 'dummy' weight points, i.e.

\[ \int e^g(r) \, dr \approx \sum_{i=1}^{M} W_i * e^g_i \]

where \( M = n + m \).

This extension does not affect the BSP model significantly, as

\[ \mathcal{L}_{gigd}(\bar{g}) = 0 \]

\[ \mathcal{L}_{gdgd}(\bar{g}) = 0 \]

\[ \mathcal{L}_{gigi}(\bar{g}) = -I_n, \text{ where } d \text{ denotes the dummy weights} \]

and \( \mathcal{L}_{gigd}(\bar{g}) = 0, \text{ and } \mathcal{L}_{gd}(\bar{g}) = 0. \)

Hence, the enhanced data set of M points will yield a covariance matrix of \( \kappa_{n+m} \) but

\[ \mathcal{L}\mathcal{L}''(\bar{g}) = \begin{bmatrix} -I_n & 0 \\ 0 & 0 \end{bmatrix} \text{ and } (\kappa_n + I_n) \text{ remains positive definite.} \]
CHAPTER 4

4 Continuous Model: Goodness-of-Fit and Residual Analysis

4.1 Global Goodness-of-Fit

For models based on likelihoods, it is conventional to assess goodness-of-fit (GOF) by examination of the deviance, defined by

\[ D_{S1} = 2(\xi_{H_0} - \xi_{H_1}) \]

where \( \xi_{H_1} \) = maximum log-likelihood under hypothesis \( H_1 \)

and \( \xi_{H_0} \) = maximum log-likelihood under the saturated model: \( H_s \).

A saturated model is one in which there is a parameter per observation. It is possible to extend this idea to the comparison of nested models where one model depends on a subset of the parameter space of the other. We can define a relative deviance measure (\( \Delta \text{ dev} \)) as:

\[ D_{10} = 2(\xi_{H_1} - \xi_{H_0}) \]

and \( H_1 : \hat{\beta} = (\hat{\beta})_p \)

\( H_0 : \hat{\beta} = (\hat{\beta})_q \)

where \( q < p \leq N \).

If \( p = N \), then \( D_{S1} \) is derived. Standard likelihood theory suggests that, if the model describes the data well, then \( D_{S1} \sim \chi^2_{N-p} \). In the case of \( D_{10} \), if both models fit the data well then \( D_{10} \sim \chi^2_{p-q} \) and we would generally prefer the model corresponding to \( H_0 \) because it is more parsimonious. If the observed value of \( D_{10} \) is in the predefined critical region (for the \( \chi^2_{p-q} \) distribution) then we would reject \( H_0 \) in favour of \( H_1 \) on the grounds that \( \beta_1 \) provides a significantly better description of the data. Note that the \( \chi^2 \) distribution for \( D_{10} \) is an asymptotic result for most distributions, except for normal models for which it is an exact result.
In general, the interpretation of the deviance from a saturated model can be complicated by non-nested model comparisons. For comparison of nested models the change of deviance ($D_{10}$) as a relative measure of improvement is to be preferred and is also more reliably distributed as $\chi^2_{p-q}$ (McCullagh and Nelder (1989; 2.3)).

For non-nested models, it is possible to use the Akaike Information Criterion given by

$$AIC = -2 \log L(N_\beta) + 2(N_\beta + 1)$$

$$= 2[N_\beta + 1 - \log L(N_\beta)]$$

(47)

where $N_\beta = \text{number of parameters in model}$. Hence, the AIC is similar to Mallow's $C_p$ in the normal linear model. Essentially, AIC is minimised by parsimonious models with large likelihood.

For the point process models considered above both $\Delta \text{dev}$ and AIC can be applied directly. An alternative approximate measure of GOF can be derived by binning points into areal cells and comparing the counts for each cell ($n_i$) with the expected counts in each cell ($e_i$). Under the HEPP model we have:

$$e_i = n \int \lambda(r) \text{d}r / \Lambda(b_0(r_0)),$$

(48)

where $e_i$ is derived from a partition of the total number $n$ by the probability of being in cell $i$. Thus we can use

$$D = 2 \sum n_i \log(n_i/e_i)$$

(49)

or

$$\chi^2 = \sum (n_i - e_i)^2/e_i$$

(50)

as alternative measures of discrepancy. D is the LR statistic while $\chi^2$ is the score test derived for a Poisson contingency table model. These tests are asymptotically equivalent.
and $\sim \chi^2_{N_C-N_p}$, where $N_C$ = no of cells and $N_p$ = no of independent parameters estimated.

These tests can be affected by a number of problems. First, their sampling distributions are not $\chi^2$ if the model does not fit well. Second, sparse cells (i.e. $n_i = 0$) can have a significant effect upon these test statistics (see Agresti and Yang (1987); Zelterman (1987); Kramer and Woolson (1987); Koehler (1986)). It would appear that although sparseness modifications to tests can be made (Zelterman (1987)), the $\chi^2$ test appears to have best performance of all proposed tests for multinomial mixture alternatives with complete independence as $H_0$. Williams (1976) has suggested a modification to $D$ which allows for small values of $e_i$, applicable for closed-form ML estimates only. McCullagh (1986) has considered a different limiting distribution when cell counts need not be large. None of the above authors examine the sampling distributions of $D$ or $\chi^2$ under a log-linear model with non-closed form ML estimates.

An additional problem arises when there may be correlation in the data unaccounted for by the model. Although residuals such as $r_i^* = (n_i - e_i)$ will be mutually correlated due to the model fit, some additional autocorrelation may occur due to clustering of points. Note that a lack of fit of the model will also invalidate the large sample central $\chi^2$ distribution.

For present purposes we use deviance and AIC as our main guide to model relevance. The problem of loss of power by arbitrary regionalisation to produce the $n_i$ in combination with the above problems limits the use of $D$ and $\chi^2$ in our example.

4.2 Residual Analysis
The examination of residuals gives an indication of model GOF, as well as information on outlying or influential observations. Model assumptions can also be assessed.

4.2.1 Binned Residuals

Previous work on residuals for HEPP models have relied on binning of points in arbitrary cells or, in the special case of a time-domain process, of transformation of the time-axis (see, for example, Ogata (1988)). The first technique is applicable generally to spatial and time-domain processes.

We define the crude residual ($r_i$) as

$$r_i = n_i - e_i$$

where $n_i$ - count in region i

$$e_i = \text{expected number in region } i.$$ 

Here $e_i$ would be derived via (48). It is conventional to utilise a standardised version of the residual ($r_i^*$). The usual standardisation is the Pearson residual:

$$r_i^* = (n_i - e_i)/\sqrt{\text{var}(n_i)}. \quad (51)$$

For the Poisson distribution, $\text{var}(n_i) = e_i$. Alternatively, a standardised residual which has a closer normal distribution and hence can be more reliably examined via quantile plots is the Anscombe residual. This is defined as

$$r_{A_i}^* = \frac{3}{2} \left( \frac{n_i^{2/3} - e_i^{2/3}}{e_i^{1/6}} \right). \quad (52)$$
In general, individual standardised residuals can be considered as normal deviates and hence any values too far from zero can be regarded as unusual observations (outliers). The usual approximate cutoff for this purpose is $|r_i^*| > 2.5$. For spatial point processes, the contiguity of outliers is important, and hence it is appropriate to view their spatial structure as well as their overall distributional properties. In the data analysis sections reported later, we use normal Q-Q plots and $r_i^*$ versus fitted value plots as well as residual surface views to assess model validity.

Where a spatial model is found inadequate a variety of effects can occur in the residuals. If spatial clustering or regularity occurred in the data then an otherwise spatially random pattern of residuals would exhibit autocorrelation. Such effects can be detected by use of Moran’s I coefficient (Cliff and Ord (1981, 8.2). For small numbers of cells, it is possible to carry out a monte carlo test based on cell count residuals generated from the fitted model.

With points binned in cells, it is possible that a single cell may contain a cluster of points. Hence a single positive outlier can represent a spatial cluster here, and this can also be used as a diagnostic tool.

As in the non-spatial case any systematic error or model mispecification could lead to plateaux and basins in the residual field. This effect could be difficult to differentiate from autocorrelation if only an autocorrelation test were used. Hence, graphical display of residuals is very important.

Residual analysis for models with MVN spatial priors cannot rely on the distributional assumptions mentioned above. However, assuming the model is correct, the residuals should only be linearly dependent and should not have spatial correlation. Basic characteristics such as $E(r_i) = 0$ and $\text{var}(r_i) = \text{const}$, should be observable in graphical
plots. In addition, monte carlo tests can be performed to assess the significance of an autocorrelation test.

4.2.2 Individual Residuals

There are a number of disadvantages to the use of binned residuals. First, an arbitrary regionalisation must be employed. Hence, the resulting residual pattern is dependent on the regionalisation. Too fine a regionalisation can lead to cell sparseness, while a coarse mesh can obscure local effects. Inevitably, a loss of information must occur due to the binning process.

An alternative method is to base the residual on an individual point observation, as in the case of continuous data. This avoids arbitrary binning and allows the direct examination of each observation both in terms of its influence on, and closeness to, the model. We have developed a new residual analysis method suitable for HEPP models, based on individual observations.

4.2.2.1 Deviance Residuals

A residual suitable for likelihood models is the deviance residual (McCullagh and Nelder (1989, 2.4), Pregibon (1981), Pierce and Schafer (1986)). In the case of HEPP models, there is no natural or direct definition of an observation and its expectation and hence it is simpler to use the contribution of an observation to a model likelihood and thence compare that contribution at the maximum likelihood estimates to a saturated likelihood model.
Define the general deviance residual \( r_{di} \) as
\[
 r_{di} = \text{sign}(y_i - \mu_i) \sqrt{d_i}
\]  
where \( d_i \) = deviance contribution of the ith observation
\( y_i \) and \( \mu_i \) are the ith observation value and fitted value.

We can apply this definition in the case of a HEPP model by considering, first, the contribution of an individual observation (i) to the unconditional log likelihood for a HEPP model. We consider a new definition of this contribution by exploiting the fact that counts in disjoint sub-regions of \( \mathbb{R}^2 \) are independent and we consider an individual point and its surrounding area. The joint probability of \( n \) points in an area \( A \) is given by
\[
 \left( \prod_{i=1}^{n} \lambda_i \right) \exp \left( - \int_{A_i} \lambda(x) dx \right)
\]  
where \( \lambda_i \equiv \lambda(r_i, \phi_i) \).

If we consider a decomposition of the area into disjoint regions which each include only one point, then we can rewrite the joint probability
\[
 \prod_{i=1}^{n} \left( \lambda_i \exp \left( - \int_{A_i} \lambda(x) dx \right) \right).
\]  
Here, \( A_i \) is an arbitrary region around the ith point, but for later purposes it is natural and convenient to define \( A_i \) as the Dirichlet tile surrounding this point. The Dirichlet tile contains all points closer to the ith point than to any other data point. As this measure relates directly and consistently to the density of the point pattern, it is preferred over more arbitrary regionalisations.

We define the HEPP log-likelihood as
\[
 \ell = \sum \ln \lambda_i - \sum \int_{|A_i|} \lambda(x) dx.
\]
Here $\lambda_i$ is a function of the base intensity $\rho$ and $\lambda_i = \rho \lambda^{*}_i$ and

$$\mathcal{L} = n \ln \rho + \sum \ln \lambda^{*}_i - \rho \sum \int \lambda^{*}(r) \, dr.$$  

The deviance contribution $(d_i)$ requires a parameter per observation in the saturated log likelihood. In this case, $\rho$ is a nuisance parameter and we estimate it, and then use profile likelihood to estimate the model parameter vector $(\lambda^{*}_i)$. The ML estimate of $\rho$ is

$$\hat{\rho} = \frac{n}{\sum_{|A_i|} \int \lambda^{*}(r) \, dr}. \quad (57)$$

We now substitute $\hat{\rho}$ in $\mathcal{L}$ to give

$$\mathcal{L} = n \ln n - n \ln \sum_{|A_i|} \int \lambda^{*}(r) \, dr + \sum \ln \lambda^{*}_i - n. \quad (58)$$

Hence, bar a constant, this is the conditional log-likelihood of the HEPP model. We now require an estimate of $\lambda^{*}_i$ per observation. By ML estimation, we derive $\hat{\lambda}^{*}_i$ by assuming that

$$\int_{|A_i|} \lambda^{*}(r) \, dr = \lambda^{*}_i A_i, \quad (59)$$

a piecewise constant approximation across each tile.

Thus,

$$\sum_{|A_i|} A_i \frac{\hat{\lambda}^{*}_i - 1}{\sum \lambda^{*}_i A_i} = 0 \quad (60)$$

and

$$\hat{\lambda}^{*}_i = A_i^{-1}.$$  

Hence, the intuitive intensity estimator at point $i$ is the inverse of the density-dependent tile area $A_i$. The saturated model gives

$$\mathcal{L}^{\text{sat}}_i = - \ln A_i - 1.$$
while the fitted model contribution is
\[ \chi_i^\text{mod} = \chi n - \chi n \sum \hat{\lambda}_i A_i + \chi n \hat{\lambda}_i A_i - n \hat{\lambda}_i A_i / \sum \hat{\lambda}_i A_i \]

and the deviance contribution is
\[ d_i = \chi n \left( \frac{\sum \hat{\lambda}_i A_i}{n} - \frac{\chi n \hat{\lambda}_i A_i}{1 + n \frac{\sum \hat{\lambda}_i A_i}{\sum \hat{\lambda}_i A_i}} \right) \quad (61) \]

and
\[ r_{di} = \text{sign} \left( A_i^{-1} - \hat{\tau}_i \hat{\lambda}_i A_i \right) \sqrt{d_i} \quad (62) \]

where \( \tau = n / \sum \hat{\lambda}_i A_i \).

Note that under the saturated model \( \tau_i \) reduces to \( A_i^{-1} \).

The above crude residual (\( r_{di} \)) can be used in informal graphical checks for model goodness-of-fit. In addition, it is possible to 'standardise' \( r_{di} \) and to examine the diagonal elements of the 'hat' matrix (H) for influential observations. Define
\[ r_{di}^s = r_{di} / \sqrt{\hat{\phi}(1 - h_{ii})} \]

where \( \hat{\phi} = \text{estimated dispersion parameter (e.g. mean deviance) } \)

and
\[ h_{ii} = (W^{1/2}X)(V(\hat{\Theta}))(W^{1/2}X)^T \]

for the fitted model with estimated parameter vector \( \hat{\Theta} \). \( V(\hat{\Theta}) \) is the variance of these parameters, and W is the diagonal matrix of iterative weights used to define a generalised linear model i.e.
\[ W_{ii} = \nu_i^{-1} \left( \frac{d\lambda}{d\eta} \right)^2 \quad \text{and} \quad \eta = F_n \Theta. \]
We here note that (56) can be written as a weighted Poisson log-likelihood i.e.

\[ L = \sum A_i \left\{ \frac{n_i}{A_i} \ln \frac{\lambda_i}{\lambda} \right\} \]

where \( n_i \) is a 0/1 indicator of a point event \( A_i \), the Dirichlet tile area used as a prior weight. (The probabilistic justification for this is given in a later section dealing with model fitting on GLIM.) The elements \( h_{ii} \) of the hat matrix can be examined for influential observations in the usual way. Clusters of influential points can also be examined using surface plots of \( h_{ii} \) values.

The relevance of this new derivation of a deviance residual is not restricted to spatial HEPP models. As the Dirichlet tessellation of a line yields line segments with boundaries at the mid-points between data points, then this residual technique can be easily applied in the time domain. Not only does this allow for the residual analysis of point processes but it can also be applied to circular and spherical distributions and many other distributions whose density functions involve a normalising integral. The application of the weighted Poisson likelihood in estimating the parameters of a von Mises distribution has already been recorded (Lawson, 1988b, 1989a). The author has also used the method for estimation of the concentration parameter of a Fisher distribution (Lawson, 1990c).

### 4.2.2.2 Examples

Given below are some examples of the application of the deviance residual analysis detailed above. We consider the following examples:

a) Spatial pattern: the residuals should detect both clustering and/or regularity departures from the model, as well as isolated outliers;
Regularity: the centres of 42 biological cells (CRICKCELL)  
(Diggle, 1983, p.129)

Clustering: 62 Redwood seedlings (REDWOOD)  
(Diggle, 1983, p.129)

simulated HEPP model: points in unit square ($\lambda(x,y) = e^{-\lambda_1 x - \lambda_2 y}$)  
n = 50 $\lambda_1 = 1$ $\lambda_2 = 3$  (PPSDRES)

simulated Cox process: points in unit square  
$\lambda_x = 1$, $\lambda_y = 1$, $\alpha = 2$, $\sigma_x = 0.1$, $\sigma_y = 0.1$.

b) Angular Distribution: the residuals from a von Mises distribution  
fitted to the marginal angular distribution for the  
Armadale point pattern (n = 49) (PPARMRES).

The following figures 13 (a–e) show normal probability plots of deviance residuals ($r_{d_i}$).

The most dramatic effects detected by the probability plots appears to be the regularity of the Crickcell data. The flat central part of the plot suggests a narrow central section of the distribution i.e. regularity. Apart from some outliers that appear to be related to boundary area effects, the plots show quite close agreement to a fitted HEPP model. The Cox process has only mild clustering and hence does not show great departure from the estimated HEPP model. PPARMRES shows a short tailed distribution with some large spaces in the anti-modal area. This suggests that there are too few points in this area for a von Mises distribution. The $r_{d_i}$ fits plots show some isolated influential points. We
Figure 13) i)

Point maps (unit square) of the 4 test data sets used in assessment of deviance residuals (CRICK, REDWOOD, PPSDRES, PPCNRES)

a) CRICK  
b) REDWOOD  
c) PPSDRES  
d) PPCNRES
Figure 13) ii)

Probability plots
a) CRICK
b) REDWOOD
c) PPSDRES
d) PPCNRES
Figure 13) ii)

Probability plots  e) Armadale: angular distribution
  f) 2 simulations of standard normal samples:
      n = 25, n = 50
examine the above methods in greater detail when applied to the analysis of data examples in a later section.

4.2.2.3 Autocorrelation and Deviance Residuals

The detection of spatial autocorrelation amongst deviance residuals can be carried out as for binned residuals. We can use Moran's I coefficient. However, for this case it is computationally prohibitive to carry out monte carlo tests based on the fitted model, as this entails simulation and generation of tessellations at each iteration.

We have made recourse to conditioning on the observed residuals and testing I under randomisation of location labels (see, e.g. Cliff and Ord (1981, 2.3). This approach is data-dependent in that the residual realisation determines the significance testing method. This has parallels in the use of the Bootstrap for the determination of sampling distributions.

4.2.2.4 BSP Individual Residuals

For the BSP case it is also possible to derive individual residuals. We defined a crude residual \( \hat{e}_i \) in the terminology of 3.2.2.2.

Define,

\[ \hat{e}_i = \hat{g}_i - g \text{ mode } i \]

where \( \hat{g}_i \) is the saturated model estimate. In this definition, we use \( \hat{g}_i \) as a surrogate for 'data item' i and g mode i as the equivalent model value.
We would like to standardise this residual to allow the use of graphical assessment.

We define,

\[ r_i = \sqrt{\text{var}(\hat{e}_i)} \]

\[ \text{var}(\hat{e}_i) = \text{var}(g_i - g_{\text{mode}}) \]

and

\[ g_{\text{mode}} = R^{-1}(\kappa^{-1} \xi + \mathbf{1}) \]

where

\[ \xi = \ell'(g) - \ell''(g)g. \]

Hence,

\[ \text{var}(\hat{e}_i) = \text{var} \left( [I_n - R^{-1}(V_g + \kappa_n^{-1} F_n A^{-1} F_n(\kappa_n + V_g^{-1})^{-1})] g \right) \]

where

\[ A = F_n \left( \kappa_n + V_g^{-1} \right)^{-1} F_n \]

so that

\[ \text{var}(\hat{e}) = [I_n - P] \text{var}(g) [I_n - P]^\top \]

where

\[ P = R^{-1}(V_g + \kappa_n^{-1} F_n A^{-1} F_n(\kappa_n + V_g^{-1})^{-1}) \]

This standardisation has been used in the analysis of data examples in Chapter 9.

With regard to autocorrelation of residuals we have resorted to the method outlined for Deviance residuals in 4.2.2.3.
CHAPTER 5

5. Discrete Model: Introduction

In previous sections we have considered the development of HEPP models for point events with extensions to account for environmental heterogeneity or case-controls. These models may be appropriate for the analysis of ecological point patterns where events occur on a homogeneous plane, albeit subject to environmental gradients. The models can be applied to the medical case, but do not provide a natural method for analysis of events occurring in a heterogeneous population distributed in space. In this section, we consider a model based on the individual within a population where the motivation is the description of the occurrence of morbidity events within a demographically-heterogeneous population which has a variable spatial density.

5.1 The Human Morbidity Pattern

Human morbidity patterns are characterised by a host population whose characteristics vary over space, and morbidity events which occur only within this population. Hence, the spatial pattern of these events does not occur on a homogeneous surface, but depends on the local population characteristics. Unless an area has a homogeneous population mix, and a uniform spatial distribution, the use of simple tests for pure spatial structure can only be limited to exploratory data analysis. Two approaches could be adopted for the analysis of this data type. First, a continuous point process model could be used with population effects incorporated as covariates. For example, population density may be available for census enumeration districts (eds) and these values could be interpolated to the locations of morbidity events. More appropriately, it would also be possible to interpolate expected numbers of deaths for each ed to the morbidity event
locations. Alternatively, a mixed model where covariate information is available in EDS and point events are recorded, can be considered. These possibilities are considered in a later section.

In this section we consider a second option. A discrete model can be derived based on the population structure, where an individual contributes to the expected number of deaths in a region. Here the hazard for an individual at a location is related to location, and other environmental and individual covariates. Denote an individual's hazard at location \((t_i, \phi_i)\) as \(\varepsilon_i\). The total expected deaths for the \(k\)th region with \(n_k\) population will be \(\sum_{i=1}^{n_k} \varepsilon_i\). \(\varepsilon_i\) can be a function of the individual's medical history, individual locational covariables or regional covariables, such as age \(\times\) sex stratification or population density.

The derivation of likelihood models utilising these \(\varepsilon_i\), parallels that of Efron (1977) in time-domain proportional-hazards modelling.

First, we consider that given one event has occurred it happens in a region \(\ell\) with probability

\[
\Pr(\ell \mid \text{1 event}) = \frac{\sum_{i \in \ell} \varepsilon_i}{\sum_{\ell=1}^{P} \sum_{i \in \ell} \varepsilon_i}. \tag{63}
\]

(We do not utilise a double subscript to denote region.) Note that \(P\) equals the number of regions considered.

In addition, given an event in region \(\ell\) then the probability that it is individual \(k\) is given by

\[
\Pr(k \mid \text{1 event in } \ell) = \frac{\varepsilon_k}{\sum_{j \in \ell} \varepsilon_j}. \tag{64}
\]

Now, a variety of models can be formulated based on the above definitions. We assume that events in disjoint regions are independent. This assumption precludes any
consideration of spatial dependence or clustering of events in our first-stage analysis, although we do consider the incorporation of heterogeneity via suitable definition of the $e_i$.

At this point it should be noted that often in the discrete case all that can be observed is the aggregate effect of a pollution source on a population. Often only regional counts of events, rather than individual locations are available. In addition, the population characteristics are often only known up to a region average or total i.e. only $\sum_{i \in \ell} X_{ij}$ may be known but not $X_{1j}$ or $X_{2j}$. (Here $X_{ij}$ denotes the jth covariate, with $i$ representing an individual.) Two different likelihood models are derived when different resolution of observation is found. These are,

Case 1): full individual knowledge:

$$\ell_F = \sum_{k=1}^{n} \ell n \ e_k - n \ell n \sum_{i \in \ell} e_i$$  \hspace{1cm} (65a)

Case 2): regional counts available:

$$\ell_R = \sum_{\ell=1}^{p} m_{\ell} \ell n \sum_{i \in \ell} e_i - \left( \sum_{\ell=1}^{p} m_{\ell} \right) \ell n \sum_{\ell=1}^{p} \sum_{i \in \ell} e_i$$  \hspace{1cm} (65b)

where $m_{\ell} =$ no of events in region $\ell$.

Case 1) and 2) are two extremes in a spectrum of models available to describe different levels of data resolution. Some variants of these two cases will be discussed in Chapter 6.

In general, it is uncommon for there to exist enough information to assume the model in case (1). Only in controlled experiments can a risk set be completely specified. However, it is often the case that (65a) can be modified to admit exact information concerning events (e.g. age, sex, medical history of patients), but only region level information about the population. Hence we can sometimes use
\[ L_{FR} = \sum_{k=1}^{n} \ln n_k \varepsilon_k - n \ln \sum_{\ell=1}^{P} n_{\ell} \varepsilon_{\ell} \]  

(66a)

where \( n_k \) is the region population and \( \varepsilon_k \) is a regional hazard \( \varepsilon_k \) can be related to regional variables such as region centre coordinates, proportions of male/female, and age distributions.

Covariates such as % regional unemployment could also be included. We will consider (66a) in a later section. Current discussion will focus on the regional count likelihood (65b) as this relates to the most common level of data observation. Most records of morbidity or mortality are readily available for small census tracts (enumeration districts (eds)) or postal code sectors. Often, only counts of events are available, within such areas, in conjunction with regional (ed-based) covariate information. As such, the log likelihood \( L_{R} \) (65b) must also be modified to

\[ L_{DR} = \sum_{\ell=1}^{P} m_{\ell} \ln n_{\ell} \varepsilon_{\ell} - \left( \sum_{\ell=1}^{P} m_{\ell} \right) \ln \sum_{\ell=1}^{P} n_{\ell} \varepsilon_{\ell}. \]  

(66b)

Note that by consideration of case 2) the relevant likelihood becomes a standard multinomial form i.e.

\[ L = \prod_{\ell=1}^{P} \{ \Pr(\ell | 1 \text{ event}) \}^{m_{\ell}}. \]

In the next section we consider the specification of functional forms for \( \varepsilon_{\ell} = n_{\ell} \varepsilon_{\ell} \).

5.1.1 Functional forms for \( \varepsilon_{\ell} \)

The region-constant model (66b) assumes that each population member in region \( \ell \) has an equal risk defined by the hazard \( \varepsilon_{\ell} \). Hence \( \varepsilon_{\ell} \) specifies a piecewise constant hazard.
Usually $\varepsilon_\ell$ consists of population factors and environmental factors, as in the continuous model intensity $\lambda(\ell)$.

Specifically we consider the form

$$\varepsilon_\ell = h(z_\ell \beta) \cdot \lambda(\ell, \phi_\ell)$$

where $h(z_\ell \beta)$ is a function of a selection of demographic variables; while $\lambda(\ell, \phi_\ell)$ is a function of the pollution source location. It is usual to incorporate a summary measure of expected morbidity within $h(z_\ell \beta)$.

We can calculate the externally-standardised expected morbidity event rate within each region. This measure is defined as

$$E_\ell = \sum_i^{n_a} \sum_j^2 n_{ij\ell} \cdot r_{ij}$$

where

- $n_{ij\ell} = $ population of $i$th age, $j$th sex group in region $\ell$,
- $r_{ij} = $ national event rate for $i$th age and $j$th sex group
- $n_a = $ number of age groups.

Note that a descriptive measure of morbidity excess in a region is simply $(m_\ell/E_\ell) \times 100$, the standardised morbidity ratio (SMR).

Usually, if $E_\ell$ is specified in a model we define $\varepsilon_\ell$ as:

$$\varepsilon_\ell = E_\ell \cdot h^*(z_\ell \beta) \cdot \lambda(\ell, \phi_\ell)$$

and

$$\ell_{Re} = \sum_{\ell=1}^{P} m_\ell \ln E_\ell - \left( \sum_{\ell=1}^{P} m_\ell \right) \ln \sum_{\ell=1}^{P} E_\ell$$

where $n_\ell$ is now subsumed in $E_\ell$. The log-likelihood (69) is of the form used in Cohort Studies (Breslow et al., 1983), albeit in an unconditional (Poisson) form.
Often we consider \( h^*(.) \lambda(.) \) to be in the log-linear form so we can write (69) as

\[
\log \text{Re} = \sum \lambda m \ln E \lambda + \sum \lambda m f \lambda \lambda + \sum \lambda m z \lambda \beta
- \lambda m \ln \sum \lambda \varepsilon \lambda \lambda
\]

(70)

where

\[
\lambda^*(.) = \exp \{f \lambda \}
\]

\[
h^*(.) = \exp \{z \beta \}
\]

\[
m_t = \sum \lambda m \lambda.
\]

Note that the first term on the RHS of (70) is a constant. Note also that all covariables in \( f \) and \( z \) are considered to be 'regional' and in such a form can easily be analysed in GLIM.

Note that except for the background intensity rate, the multinomial and Poisson likelihoods are equivalent with respect to parameter estimation and hence any regional covariables can be fitted on GLIM by declaring a Poisson error with \( m \lambda \) as y-variate and \( \log E \lambda \) as offset.

Here, \( f \lambda \) represents the evaluation of spatial covariable terms at a suitably-defined region centre.

A more general spatial model can be formulated where a regional environmental hazard is represented by the average integrated intensity of the process i.e.

\[
\varepsilon \lambda = E \lambda \exp \{z \beta \} I_a \lambda / a \lambda
\]

(71)

where

\[
I_a \lambda = \int_a \lambda(r) d\lambda.
\]

Note that, in (71), if \( \lambda(r) \equiv \lambda \) then a piecewise constant rate follows.

The integral term in (71) requires evaluation over an arbitrary region \( a \lambda \). \( a \lambda \) may be a census tract or other administrative region and hence can have very irregular boundaries. As a consequence the evaluation of this integral is difficult even numerically. One method
of approximating $I_{a,k}$ is to enclose each region by two rays from the fixed centre, and two concentric arcs as shown in Figure 14a. Each ray is located at the rotational extreme of the region, while the arcs are placed at the nearest and furthest extreme of the region.

We define $r_k$ as the nearest point of region $k$ to the centre $c$. Also denote $\phi_k$ as the angle of the, clockwise, right enclosing ray. Then the area of the annular sector enclosing region $k$ is $A_k^*$, while $A_k$ is the area of region $k$. Now it is easy to integrate over $A_k^*$ and we can now make a linear approximation to the integral over $A_k$ and hence the region probability $(q_k)$, by

$$q_k = \frac{\int \lambda(L) \, dL}{\frac{A_k^*}{A_k} \Lambda(b_0(r_0))}$$

(72)

where

$$A_k^* = \frac{\Delta \phi_k}{2} \left[ 2r_k \Delta r_k + \Delta r_k^2 \right]$$
and
\[ I_{Ak} = \int_{Ak} \lambda(\ell)d\ell = \int_{\phi_k}^{\phi_k+\Delta\phi_k} \int_{r_k}^{r_k+\Delta r_k} \lambda(r,\phi)r \, dr \, d\phi. \] (73)

Note that this geometric approximation produces integrals over overlapping regions, albeit linearly shrunk, and hence some dependence between regions must ensue. The severity of this dependence will depend on the degree of irregularity in the regions.

One improvement in this technique is to relax the requirement that enclosing rays and arcs must be tangent to extremal points and allow them to move in so that the area of \( A_k^* \) more closely matches that of \( A_k \), to produce an average boundary in the plane of the arc or ray. Figure 14b depicts this modification.

Another alternative is to approximate \( I_{Ak} \) as
\[ I_{Ak} = \int_{\phi_k}^{\phi_k+\Delta\phi_k} \int_{r_1(\phi)}^{r_2(\phi)} \lambda(r,\phi)r \, dr \, d\phi \] (74)

where \( r_1(\phi) \) and \( r_2(\phi) \) are empirically estimated boundary curves, based on digitised boundary values. Kernel or spline methods could be used to evaluate \( r_1(\phi) \) and \( r_2(\phi) \). This option, while more accurate, will be computationally expensive if \( P \) is large and boundaries are very irregular. Note that for any numerical parameter estimation scheme, there would be \( 2P \) spline/kernel evaluations for each evaluation of \( \sum_{\ell=1}^{P} e_\ell \) in (71).

We have attempted to apply the above methods in the case of census enumeration districts. However, it was found that eds often have extremely irregular shapes, and hence the first method, (eq. 72,73) above would produce large inaccuracies and dependencies in the estimation process. In addition, the computational requirements of the second method (eq. 74) were deemed prohibitive and hence neither of these methods have been pursued in the analysis of the census ed data.
5.1.2 Likelihood Methods

5.1.2.1 Maximum Likelihood Estimation

Standard methods are available for ML estimation in model (66b or 69), the regional-count model. We assume that expected morbidity (E_\lambda) is included in e_\lambda, thus

\[ e_\lambda = E_\lambda h^*(z, \beta) \lambda(r_\lambda, \phi_\lambda) \]

and we also approximate I_{a\lambda}/a_\lambda by a region-constant rate \lambda_\lambda, evaluated at the region-centre (suitably defined). Hence e_\lambda can be written as \log e_\lambda = \log E_\lambda + z_\lambda \beta + f_\lambda \alpha and GLIM can be used directly for estimation purposes. If evaluation of I_{a\lambda} is required then a standard numerical maximisation would be required to evaluate (69). Some simple parametric forms of e_\lambda make the derivative of estimators relatively simple:

5.1.2.1.1 Exponential Trend: \( e_\lambda = E_\lambda e^{\beta r_\lambda} \)

We assume that there is a multiplicative effect of radial distance (r_\lambda) from the pollution source.

Hence:

\[ \lambda = \sum_{\lambda=1}^{P} \sum_{x=1}^{m_x} \ln E_\lambda + \beta \sum_{\lambda=1}^{P} \sum_{x=1}^{m_x} r_\lambda - (m_t) \ln \sum_{\lambda=1}^{P} E_\lambda e^{\beta r_\lambda} \]  

(75)

\[ \lambda' = \sum_{\lambda=1}^{P} m_x r_\lambda - m_t \sum_{\lambda=1}^{P} \frac{E_\lambda r_\lambda e^{\beta r_\lambda}}{E_\lambda e^{\beta r_\lambda}} \]  

(76)

and the ML estimate of \beta is the solution of
Here, we introduce a general expectation operation $E(.)$. This is defined to be the expectation of a random variable over the discrete distribution defined by the region probability $p_i r_i / m_i = E(r_i)$.

This general form $E(.)$ is used in a number of applications in later sections.

It is also possible to specify a peak-then-decline hazard which may include a shape parameter as equivalent to Gamma or Weibull extensions of the continuous case. Estimation is again straightforward. We do not pursue this here, although we evaluate such models in the data analysis section (Chapter 8).

5.1.2.1.2 **Angular Concentration**:

$\varepsilon_\phi = E_\phi \exp(\kappa \cos(\phi_k - \mu_0))$.

We again assume a multiplicative model for angular hazard and note that the standard form includes a concentration parameter $\kappa$ and a mean angle $\mu_0$. It is usual to consider such a parameterisation as the minimum required to describe a unimodal angular distribution. Note that we do not normalise $\varepsilon_\phi$, as normalisation is provided by $\sum_{i=1}^{P} \varepsilon_{\phi_i}$. 
The likelihood derived is
\[
\mathcal{L} = \sum_{\xi} m_{\xi} \ln E_{\xi} + \kappa \sum_{\xi} m_{\xi} \cos (\phi_{\xi} - \mu_0) \\
- m_t \ln \sum_{\xi} E_{\xi} \exp (\kappa \cos (\phi_{\xi} - \mu_0))
\] (77)

and
\[
\mathcal{L}'_\kappa = \sum_{\xi} m_{\xi} \cos (\phi_{\xi} - \mu_0) - m_t \frac{\sum \cos (\phi_{\xi} - \mu_0) \varepsilon_{\xi}}{\sum \varepsilon_{\xi}}
\] (78)

\[
\mathcal{L}'_{\mu_0} = \kappa \sum_{\xi} m_{\xi} \sin (\phi_{\xi} - \mu_0) - m_t \frac{\sum \sin (\phi_{\xi} - \mu_0) \varepsilon_{\xi}}{\sum \varepsilon_{\xi}}.
\] (79)

Hence, the ML estimates of \( \kappa, \mu_0 \) are solutions of
\[
\sum_{\xi} m_{\xi} \cos (\phi_{\xi} - \mu_0) = E(\cos(\phi_{\xi} - \mu_0))
\]
\[
\sum_{\xi} m_{\xi} \sin (\phi_{\xi} - \mu_0) = E(\sin(\phi_{\xi} - \mu_0)).
\]

In the case of a continuous distribution \( E(\sin(\phi_{\xi} - \mu_0)) = 0 \), but this is not necessarily the case here, due to irregular regionalisation. However, we can appeal to an asymptotic argument whereby the study window expands, while the intensity of events remains constant; then as the number of regions \( P \to \infty \), \( E(\sin(\phi_{\xi} - \mu_0)) \to 0 \). This argument relies on the hazard \( \varepsilon_{\xi} \) being symmetric about \( \mu_0 \).

An intuitive estimator of \( \mu_0 \) is the weighted sample mean angle \( \bar{X}_0 \) defined as
\[
\bar{X}_0 = \tan^{-1} \left( \frac{\text{NS}}{\text{NC}} \right)
\] (80)

where
\[
\text{NS} = \sum_{\xi} m_{\xi} \sin \phi_{\xi}
\]
\[
\text{NC} = \sum_{\xi} m_{\xi} \cos \phi_{\xi}.
\]
Now if \( E(\sin(\phi - \mu_0)) \to 0 \) then \( \bar{X}_0 \) is the ML estimator of \( \mu_0 \), so that it is only required to solve

\[
\sum \theta \cos(\phi - \bar{X}_0) = E(\cos(\phi - \bar{X}_0)) \quad \text{for } \hat{\theta}.
\]

We can also use the estimator \( \bar{X}_0 \) in the maximisation of the likelihood (77) to yield an initial estimate for \( \theta \), and thence to maximise jointly for \( \theta, \mu_0 \). Note that the above ML estimators can be obtained in GLIM via a parameter transformation, i.e. we use the Poisson error and log link and fit \( \cos \phi \) and \( \sin \phi \) with parameters \( \beta_1 \) and \( \beta_2 \) respectively and

\[
\hat{\theta} = \sqrt{\hat{\beta}_1^2 + \hat{\beta}_2^2} \quad \text{and} \quad \hat{\mu}_0 = \tan^{-1} \left( \frac{\hat{\beta}_2}{\hat{\beta}_1} \right).
\]

### 5.1.2.1.3 Radial-Angular Interaction:

\[
\varepsilon \mu = E \exp((\kappa + \psi r \mu) \cos(\phi - \mu_0))
\]

We consider that \( r \cdot \phi \) interaction or correlation is subsumed within a standard angular model as in 5.1.2.1.2. Hence, we replace \( \kappa \) by \( (\kappa + \psi r \mu) \).

The log-likelihood for this model is

\[
\mathcal{L} = \sum_{i} \theta \ln E_i + \kappa \sum_{i} \theta \cos(\phi_i - \mu_0) + \psi \sum_{i} r_i m_i \cos(\phi_i - \mu_0)
\]

\[
- \frac{\mu}{m_i} \sum_{i} \varepsilon_i
\]

and

\[
\mathcal{L}' = \sum_{i} m_i \cos(\phi_i - \mu_0) - m_i E(\cos(\phi_i - \mu_0))
\]

\[
\mathcal{L}' = \sum_{i} r_i m_i \cos(\phi_i - \mu_0) - m_i E(r_i \cos(\phi_i - \mu_0))
\]
\[ \mu_0 = \kappa \sum_{x} p_x \sin(\phi_x - \mu_0) + \psi \sum_{x} r_x m_x \sin(\phi_x - \mu_0) \]

\[- m_t E[(\kappa + \psi r_x) \sin (\phi - \mu_0)]. \quad (84)\]

Note that if we assume the previous conditions concerning asymptotic behaviour, then \( E[(\kappa + \psi r_x) \sin (\phi - \mu_0)] \to 0 \) as \( P \to \infty \).

Hence, define resultants

\[ R_1 = \sqrt{N C^2 + N S^2} \quad \text{and} \quad R_2 = \sqrt{N R S^2 + N R C^2} \]

where

\[ N R S = \sum_{x} p_x r_x \sin \phi_x \]

\[ N R C = \sum_{x} p_x r_x \cos \phi_x. \]

Hence the ML estimates are the solutions of

\[ \frac{R_2}{m_t} = E(r_x \cos(\phi_x - \hat{\mu}_0)) \]

\[ \frac{R_1}{m_t} = E(\cos(\phi_x - \hat{\mu}_0)) \]

and

\[ E[(\kappa + \psi r_x) \sin (\phi - \hat{\mu}_0)] = 0. \]

Note also that \( \hat{\kappa}, \hat{\psi} \) and \( \hat{\mu}_0 \) can be obtained on GLIM via a parameter transformation. If \( \cos \phi, \sin \phi \) and \( r \cos \phi, r \sin \phi \) are fitted with parameters \( \beta_1, \beta_2, \beta_3, \beta_4 \) respectively, then

\[ \hat{\kappa} = \sqrt{\hat{\beta}_1^2 + \hat{\beta}_2^2}; \quad \hat{\psi} = \sqrt{\hat{\beta}_3^2 + \hat{\beta}_4^2}; \quad \hat{\mu}_0 = \tan^{-1}\left(\frac{\hat{\beta}_2 + \hat{\beta}_4}{\hat{\beta}_1 + \hat{\beta}_3}\right). \]

This parameterisation is inefficient, as 4 GLIM parameters are used to estimate 3 derived parameters. One effect of this transformation could be to increase the difficulty in estimating the \( \kappa \) and \( \psi \) components of concentration as these parameters are not orthogonal. The estimate of \( \mu_0 \) should be little affected. The variances of these estimates can be
approximated by the delta method to $O(n^{-3/2})$. Appendix VIII gives a discussion of the continuous version of this transformation.

In general, all the above angular hazards $(b,c)$ can be extended to include a radial trend effect or other covariables. In such cases, further normal equations are required and it should be noted that the normalisation $\sum e_\xi$ depends on both functions of $\xi$ and $\Phi$ unlike the continuous case. This difference arises because, in the continuous case, the $\Phi$-variate can be integrated to give Bessel functions and this normalisation is included in $e_\xi$. This facility does not appear to be available in the discrete case.

5.1.2.2 Hypothesis Testing

LR, Wald and Score tests can all be derived for the variety of $e_\xi$ functional forms defined in the previous section. However, the complex form of LR and Wald tests which arise, for any but the simplest forms of $e_\xi$, prohibits their use. I have found that Score tests often have relatively simple closed forms and hence are more easily used or programmed. LR tests are available for all the above models on GLIM and this further suggests that the examination of Score tests may be preferred.

5.1.2.2.1 Exponential Trend

The score statistic for this hazard is well known as Rao's test (Breslow et al. (1983); Breslow and Day (1987), p.96) and is defined as

$$R = \sqrt{\frac{\sum_{\xi=1}^{P} r_\xi (m_\xi - E_\xi^*)}{\sqrt{\left(\sum_{\xi=1}^{P} r_\xi^2 E_\xi^* - \left(\sum_{\xi=1}^{P} r_\xi E_\xi^*\right)^2/m_t\right)}}}$$

Equation (85)
where $E_\mu$ has been adjusted so that $\sum_{\mu=1}^{p} E_\mu = m_t$, the population total, thus

$$E_\mu^* = \left( E_\mu / \sum_{\mu=1}^{p} E_\mu \right) m_t.$$

The test statistic $R$ has an asymptotic standard normal distribution, under $H_0 : \beta = 0$. It is usual to use 1df score tests in this form rather than $R^2 \sim \chi^2_1$, as $R$ allows interpretation of the sign of the numerator. Note that $R$ measures the distance weighted mean of the differences between observed counts $\{m_\mu\}$ and expected counts $\{E_\mu\}$.

Tarone (1982) and Gart and Tarone (1983) have pointed out that tests, such as the above, are uniformly most powerful and unbiased against monotone alternatives and, hence $R$ is optimal in this sense not only against an exponential hazard but also for any monotone effect. Hence, $R$ does not have strong parametric assumptions. Stone (1988) has proposed a test for detection of radial trend which, although nonparametric, does not have a simple sampling distribution. The form of the test statistic is more complex than $R$ and also depends on the ordering of regions with respect to radial distance. For the above reasons, $R$ should be preferred. Another advantage of such score tests is that they can be extended easily to accommodate non-monotone radial effects, such as a peak-then-decline function. Appendix IV gives details of such a test extension.

5.1.2.2.2 Angular Concentration

We can derive a score test based on the log likelihood (77), equations (78), (79) and second derivatives of (77). Appendix VI contains details of the elements of the information matrix for this model. As $\mu_0$ is a nuisance parameter, we must estimate it
under $H_0: \kappa = 0$. A consistent estimate of $\mu_0$ is $\bar{X}_0$ (given by (80)) and the derived score test statistic is

$$T_h = \frac{\sum_{\ell=1}^{P} \cos (\phi_{\ell} - \bar{X}_0)[m_{\ell} - E^*_{\ell}]}{\sqrt{\left[ \sum_{\ell} \cos^2(\phi_{\ell} - \bar{X}_0)E^*_{\ell} - \left( \sum_{\ell} E^*_{\ell} \cos (\phi_{\ell} - \mu_0) \right)^2 / m_1 \right]}}. \quad (86)$$

$T_h$ is also asymptotically $N(0,1)$.

This test also weights the difference in observed and expected morbidity by a measure of angular deviation from the mean angle. This test should be sensitive to time-averaged directional effects where concentration of events would be expected around a preferred direction related to the prevailing wind regime.

Note that if a radial hazard is also hypothesised, then the radial parameters would have to be estimated under $H_0$, or, if feasible, 'integrated' out.

5.1.2.2.3 Radial-Angular Interaction

We consider interaction nested within concentration and hence that $\kappa$ becomes $\kappa + \psi r_{\ell}$.

For this hazard a more complicated score test for $\psi$ is derived here. In this case the information matrix does not admit a reduction to simple form. The details of this matrix are given in Appendix VI. Suffice to note that

$$W_S = U.V^{1/2}_{\psi \psi} \sim N(0,1) \quad (87)$$

where

$$U = \left. \frac{\partial \xi}{\partial \psi} \right|_{\psi=0} \quad (88)$$
and the leading element of the inverse information matrix is

\[ V_{\psi\psi} = (I_{\psi\psi} - I_{\psi\lambda}^T I_{\lambda\lambda}^{-1} I_{\psi\lambda})^{-1} \]

(89)

and \( I_{\phi_1\phi_2} \) is the observed information for parameters \( \phi_1 \) and \( \phi_2 \) evaluated under \( H_0: \psi = 0 \).

The underline \( (\lambda) \) denotes a parameter vector. Note that \( \lambda^T = (\kappa, \mu_0) \) must be estimated under \( H_0 \) using maximum likelihood (Cox and Hinkley, 1974, p.324) or at least with a consistent estimator.

As noted for the continuous model case the \( \psi \) parameter is a measure of circular-linear correlation. Hence \( W_S \) can be used as a test for such correlation when angular data are grouped. Grouping can either occur in a spatial region, in which case \( E_\lambda \) will be a function of \( A_\lambda \), the region area, or, if data occurs grouped on \( S' \) the \( E_\lambda \) will be a function of the class interval on the circle.

5.1.2.3 General Testing with Observed Heterogeneity

The discrete model (66b) can be extended to include covariates relating to the population structure or environmental heterogeneity as in the continuous case. Hence, we can define

\[ \epsilon_\lambda = E_\lambda h^*(z_\lambda \beta) \lambda(\tau_\lambda, \phi_\lambda) \]

and now explicitly model the \( h^*(z_\lambda \beta) \) function. We will assume, as before, that \( h^* \) and \( \lambda \) are log-linear functions of observable covariates \( z_\lambda, f_\lambda \) so that

\[ \epsilon_\lambda = E_\lambda \exp\{z_\lambda \beta + f_\lambda \xi\} \]

(90)

and
\[ \mathcal{L}_f = \sum_{\xi} m_{\xi} \ln \mathcal{E}_{\xi} + \sum_{\xi} m_{\xi} [Z_{\xi}\beta + f_{\xi}\alpha] - \ln \mathcal{E}_{\xi} \sum_{\xi} \varepsilon_{\xi}. \]  

(91)

Now a range of tests can be derived, from this log likelihood, which are designed to test for the presence of particular components of spatial structure (i.e. \( f_{\xi} \)). Hence, we will be concerned with tests where the \( Z_{\xi} \) are nuisance variables and hence \( \beta \) is a nuisance parameter vector. From the log likelihood (91), we can derive a general score test statistic for testing for a subset of \( f_{\xi} \), in a similar manner to Cox (1972). Here,

\[ \mathcal{L}_{\alpha\eta} = \sum_{\xi} m_{\xi} f_{\xi\eta} - m_{\xi} \mathcal{E}(f_{\xi\eta}) \]  

(92)

where \( \mathcal{E}(f_{\xi\eta}) \) is the expectation over the normalised distribution based on \( \varepsilon_{\xi} \).

\[ - \mathcal{L}_{\alpha\eta\beta} = m_{\xi} [\mathcal{E}(f_{\xi\eta} Z_{\xi\beta}) - \mathcal{E}(f_{\xi\eta}) \mathcal{E}(Z_{\xi\beta})], \]  

(93)

(Note, that for a single covariable, say \( z_1 \), and single spatial variable \( f_1 = 1 \), then \( - \mathcal{L}_{\alpha\beta} = m_{\xi} \).

(sum of squares of \( z_1 \)).) Oakes (1979) shows a similar decomposition in the time case.

Denoting \( I_{\alpha\beta} = \{- \mathcal{L}_{\alpha\beta} \} \alpha_1 = \alpha_2 = \beta \) then we can define a score test for \( H_0: \alpha = \Omega \) against \( H_1: \alpha \neq \Omega \) as

\[ W_{\alpha} = U_{\alpha\alpha}^T I_{\alpha\alpha}^{-1}(0) U_{\alpha\alpha} \text{ as } \chi^2_k \]  

(94)

where \( \alpha \) is a \((k \times 1)\) vector and

\[ U_{\alpha0} = \mathcal{L}^T | \alpha = \bar{\Omega} \]

and \( I_{\alpha\alpha}^{-1}(0) \) is the leading element of the inverse information matrix evaluated under \( H_0 \).

The problem of nuisance parameters is also present in this test as \( \beta \) appears under \( H_0 \) and hence must be estimated or integrated out. One possibility in the above case is to estimate \( \beta \) using GLIM, and thence using these ML estimates as input to the above score test. Note, however, that the equivalent LR test is available on GLIM by fitting \( Z \) followed by \( f \).
5.1.2.4 Unobserved Heterogeneity

As in the continuous case, we can employ two basic methods to allow inclusion of unobserved heterogeneity in our discrete models:

5.1.2.4.1 Harmonic Hazard Terms

It is possible to extend the spatial hazard \( \lambda(x) = \exp\{ f(x) \} \), by including a number of high order terms in which describe general spatial heterogeneity. As in the continuous case, we could include harmonic terms such as \( \sum_{k=1}^{m} \tau_k \cos \{ k(\theta_k - \mu_x) \} \) and \( \sum_{k=1}^{m} \tau_k \sin \{ k(\varepsilon_k - \mu_y) \} \) where \( \theta_k = 2\pi x_k / X_{\text{max}} \) and \( \varepsilon = 2\pi y / Y_{\text{max}} \) and \( X = r \cos \phi \) and \( Y = r \sin \phi \). These terms could be included with the nuisance population covariates and would require the estimation of \( \tau_k \) for a range of \( k \) values, and \( \mu_x, \mu_y \). Hence this model approach is not parsimonious. Note that it is possible to fit such a model directly on GLIM. We do not pursue this approach due to its lack of parsimony.

5.1.2.4.2 Spatial prior structure for hazards

An alternative to the above approach is to assume that heterogeneity can be described by a Spatial prior distribution for a component or parameter in the model.

In the discrete case it is convenient to define

\[
\varepsilon_k = \exp\{ g(f_k, z_k) \}. \tag{95}
\]

We assume that \( g( ) \) is now a regional function and that \( g( ) \) is a realisation of a spatial Gaussian process and hence that \( g_p \) is MVN \((\Omega_p, K_p)\):
\[ \Pr(g_p; \mathcal{Q}) \propto \exp \left\{ -\frac{1}{2} (g_p - \xi_p)^T K_p^{-1}(g_p - \xi_p) \right\} \]  

where  
\[ g_p = (p \times 1) \text{ vector linear predictor} \]
\[ \mathcal{Q}_p = (n \times p) \text{ matrix of spatial and environmental variates} \]
\[ y = (p \times 1) \text{ vector of parameters} \]
\[ \xi_p = \mathcal{Q}_p y \]
\[ K_p = (p \times p) \text{ covariance matrix of the } P \text{ regions} \]

and  
\[ k_{ij} = \sigma^2 \exp \left\{ -\frac{d(i,j)}{R_a} \right\} \]

where  
\[ d(i,j) = \text{euclidean distance between region centres of } i \text{ and } j. \]

This formulation is similar to that used in the continuous case. Note that the conditional log-likelihood of the data in this case is multinomial, given \( g_p \). We use the equivalent Poisson log likelihood for the data  
\[ \mathcal{L}(\text{data} \mid g_p) = \sum_{\mathcal{I}} m_{\mathcal{I}} g_{\mathcal{I}} - \sum_{\mathcal{I}} e^{g_{\mathcal{I}}}. \]  

(97)

Note that this formulation is simpler than the conditional likelihood and includes the baseline hazard. Here the posterior distribution is given by  
\[ \Pr(g_p \mid \text{data}; \mathcal{Q}) = \exp \left\{ \sum_{\mathcal{I}} m_{\mathcal{I}} g_{\mathcal{I}} - \sum_{\mathcal{I}} e^{g_{\mathcal{I}} - \frac{1}{2} (g_p - \xi_p)^T K_p^{-1}(g_p - \xi_p)} \right\} \]  

(98)

(bar a constant). The MAP estimate of \( \hat{g}_p \) is easily seen to be the minimiser of  
\[ \hat{g}_p - \xi_p = K_p (m_p - \exp(\hat{g}_p)) \]  

(99)

for fixed \( y \). Note that this is similar to the continuous case equation (Leonard (1978), Whittle (1978)), except that \( t \) is replaced by the vector \( m_p \) and tessellation weights are included in the continuous formulation. Similar arguments to those of the continuous case, lead to profile likelihood estimates of \( y \) as
\[ \hat{\theta} = (Q_p^T K_p^{-1} Q_p)^{-1} Q_p^T K_p^{-1} \hat{\varphi}_p \] (100)

with \[ \text{cov}(\hat{\theta}) = (Q_p^T K_p^{-1} Q_p)^{-1}. \]

Note that \( \hat{\theta} \) can be obtained relatively quickly by OLS, as in the continuous case.

An iterative scheme can be employed, starting from the ML estimates of \( \hat{\varphi}_p \), using (99) and (100). We can also obtain \( \hat{\sigma}^2 \) after estimation of \( \hat{\varphi}_p \) and \( \hat{\theta} \) as

\[ \hat{\sigma}^2 = \frac{1}{n} (\hat{\varphi}_p - \hat{\varphi}_p^*)^T K_p^* (\hat{\varphi}_p - \hat{\varphi}_p^*) \] (101)

where \[ K_p^* = \frac{1}{\sigma^2} K_p. \]

Note that (97) to (99) were derived by Whittle (1978) for the time-domain, for constant mean (\( \xi = \text{constant} \)).

The numerical solution of the above iterative scheme is straightforward for the OLS stage (100) but (99) requires the solution of a system of \( p \) linked nonlinear equations, and this tends to be very inefficient in practice. As in the continuous case, we can avoid this problem by use of a Taylor series expansion of the data likelihood. This leads to the same one-step estimates of \( \varphi_p \), as in the continuous case, with

\[ \ell(\varphi) = \sum_{\xi} \sum_{g_{\xi}} m_{\xi} g_{\xi} - \sum_{\xi} e^{g_{\xi}} \]

and \[ \hat{g}_{\xi} = \ln m_{\xi} \]

\[ - \ell''(\varphi) = \text{diag} \{ m_{\xi} \} = D. \]

Hence,

\[ \tau = D \ln (m) \]

\[ R = K_p^{-1} + D \]
\[ T = K_p^{-1} \xi_p + D \ln(m) \]

and the empirical bayes estimate (\( \hat{\xi} \) mode) is
\[ \hat{\xi}_{\text{mode}} = (K_p^{-1} + D)^{-1} (K_p^{-1} \xi_p + D \ln(m)). \]

It should be noted that Clayton and Kaldor (1987) used the above Poisson data likelihood in their model, and the above analysis can be regarded as an improvement over their EM algorithm-based solution procedure.

5.1.3 Discrete Model: Goodness-of-Fit and Residual Analysis

5.1.3.1 Global Goodness-of-Fit (GOF)

In the discrete case we can also use the Deviance to measure model fit. As noted earlier the change-in-deviance (\( \Delta \) deviance) has a sampling distribution more reliably approximated by a \( \chi^2_{p-q} \) than the full Deviance and is used as a more reliable guide to relative improvement in nested models. GLIM provides these measures directly. Most discrete models considered here have Poisson likelihoods and hence GLIM can be used to provide estimation and GOF tests.

The problem of sparseness of the regional counts arises immediately with discrete data. Zero region counts are common in mortality data, particularly if a rare disease is considered. Hence the comments of Section 4.1 concerning LR and \( \chi^2 \) tests also apply here. Note that the points of the process are already binned in the discrete case. The bins are, however, irregularly bounded regions. In this case, we use the \( \Delta \) Deviance and AIC as general measures of GOF (the likelihood ratio statistic \( D \) is in fact the deviance for a Poisson likelihood). We also examine \( \chi^2 \) as a GOF measure.
5.1.3.2 Residual Analysis

As point process data are binned, the discrete case leads to consideration of Pearson or Anscombe residuals for Poisson data. Graphical methods also carry over from the continuous case.

Examples of testing regionalised medical data for autocorrelation can be found in Cliff and Haggett (1980), Lloyd et al. (1988), Whittemore et al. (1987). As in Section 4.2.1 we can use Moran's I coefficient. Monte Carlo evaluation of Moran's I is possible here as counts from the null model can be easily simulated and compared to observed counts to yield simulated residuals. Systematic model mis-specification can be verified by graphical methods.

Note that comments concerning Bayesian spatial prior models are also applicable here so that residual analysis can be carried out, using the above methods but less reliance should be placed on the sampling distribution of \( r_j \), and use made of graphical displays and autocorrelation tests.

In the next section we describe, briefly, a number of variants of the continuous-discrete model forms and the hazard or intensity functions, relevant to particular data sets.
6 Variants of the Continuous-Discrete Models and Hazards

6.1 Model Variants

A number of model variants can be considered which lie between the continuous and discrete models. One particular variant will be considered here, as it is relevant to the level of aggregation commonly found in census data.

It is often the case that point events are recorded exactly within a window, but other data in the form of covariates are only available at a regional level. For example, death certificate addresses may be known but the expected number of deaths or population structure may only be known within ed's. (Data set A is an example of this aggregation of covariate information.)

This tends to suggest a hybrid model:

\[
\Pr(\text{an event at } r_i, \phi_i) = \lambda^*(r_i, \phi_i) / \sum_{x=1}^P \lambda^*(r_{x}, \phi_{x}) \tag{102}
\]

and

\[
\lambda^*(r_i, \phi_i) = h(z_i \beta) \lambda(r_i, \phi_i; \omega) \tag{103}
\]

and

\[
\lambda^*(r_{x}, \phi_{x}) = h(z \xi \beta) \int_{a}^{x} \lambda(r, \phi, \omega) \, dr/a \xi \tag{104}
\]

or, if regional average spatial intensity is used then

\[
\lambda^*(r_x, \phi_x) = h(z \xi \beta) \lambda(r_x, \phi_x, \omega). \tag{105}
\]

Hence we can derive a likelihood of form (66a), given n events in the window

\[
x_{FA} = \sum_{i=1}^{n} \lambda \ln \lambda^*(r_i, \phi_i) - n \lambda \sum_{x=1}^P \lambda^*(r_x, \phi_x). \tag{106}
\]
For estimation, by maximum likelihood, of the spatial parameters, and assuming log-linear hazard $\lambda^*(\cdot)$, we need:

$$\mathbf{\ell}_{\alpha_\eta} = \sum_{i=1}^{n} f_{i\eta} - n \mathbb{E}_R(f_{\eta}) = 0 \quad (107)$$

and where $f_{i\eta} = (1 \times k)$ vector of spatial variables and $\mathbb{E}_R$ denotes expectation over the regional distribution of hazard. Hence, the ML estimate of $\alpha_\eta$ is the solution of

$$\sum_{i=1}^{n} f_{i\eta}/n = \mathbb{E}_R(f_{\eta}).$$

This estimate equates the regional average of $f_{\eta}$ over the model with the average over the data. In general, these estimates must be found numerically. Hypothesis tests for $H_0 : \alpha = 0$ against $H_1 : \alpha \neq 0$ can be based on LR or Score tests. For the LR test the two likelihoods are

$$\mathbf{\ell}_{H_0} = \sum_{i=1}^{n} z_i \hat{\beta} - n \sum_{i=1}^{p} \mathbb{E}_{\xi} \exp\{z_{i\xi}\} \quad (108)$$

$$\mathbf{\ell}_{H_1} = \sum_{i=1}^{n} z_i \hat{\beta} + \sum_{i=1}^{n} f_{i\eta} - n \sum_{i=1}^{p} \mathbb{E}_{\xi} \exp\{z_{i\xi} \hat{\beta} + f_{i\xi} \eta\}. \quad (109)$$

(We assume that a constant regional spatial hazard is modelled.)

Denoting ML estimates under $H_0$ and $H_1$ as $\hat{\alpha}$ and $\hat{\alpha}$ respectively, we have

$$LR = 2 \left\{ \left[ \hat{\alpha} - \hat{\alpha} \right] \sum_{i=1}^{n} z_i + \sum_{i=1}^{n} f_{i\eta} \hat{\alpha} + n \sum_{i=1}^{p} \frac{\mathbb{E}_{\xi} \exp\{z_{i\xi} \hat{\beta}\}}{\sum_{\xi=1}^{\tau} \mathbb{E}_{\xi} \exp\{z_{i\xi} \hat{\beta} + f_{i\xi} \eta\}} \right\}. \quad (110)$$

Note that $\hat{\beta}$ is the solution of

$$\sum_{i=1}^{n} z_{ei}/n = \mathbb{E}_R(z_{e}) \quad (\text{under } H_0)$$
and \( \hat{\beta} \) is the solution of

\[
\sum_{i=1}^{n} z_{E_i} / n = E_R(z_{E_i}) \quad \text{(under } H_1). \]

Note that all these estimates must be found numerically, due to the normalisation over all regions.

The score test formulation gives:

\[
\mathcal{X}^{\prime}_{\alpha \eta} = \sum_{i=1}^{n} f_{i \eta} - n E_R(f_{\eta}) \bigg|_{\alpha \eta = 0} = U_{\alpha \eta 0}
\]

and

\[
\mathcal{X}^{\prime \prime}_{\alpha \eta \beta e} = n \{ E_R(f_{\eta} z_e) - E_R(f_{\eta}) E(z_e) \} = I_{\alpha \eta \beta e}
\]

\( I_{\alpha \beta} = \{ I_{\alpha \eta \beta e} \} \) and \( \Gamma^{-1}_{\alpha \eta \beta e}(0) \) = leading element of the observed inverse information matrix, and

\[
W_S = U_{\alpha \eta 0}^T \cdot \Gamma^{-1}_{\alpha \eta}(0) \cdot U_{\alpha \eta 0} \cdot \chi^2_k
\]

This test can be restricted to a test for a specific vector element \( \alpha_{\eta} \), in which case \( W_S \sim \chi^2_1 \).

Note, that, in general, when a covariate vector \( z_{\xi} \) is present in the model neither the LR or score tests are of a simple form, even when testing for simple radial or angular effects.

However, when the population structure is incorporated by using \( E_{\xi} \) only (the expected events in region \( \xi \)), i.e. excluding the term \( \exp(z_{\xi \beta}) \), simple test forms arise.

For example, if we assume a simple exponential hazard then it is easy to show that the signed square root of \( W_S \) (\( W_A \) say) gives
This is just a comparison between the distance average of the point data and the population weighted region average. For the case of a simple angular hazard $W_A$ becomes:

$$W_A = \frac{\left( \sum_{i=1}^{n} r_i - n \frac{\sum_{i=1}^{n} E_i r_i}{E_T} \right) \left( \sum_{\xi=1}^{p} \frac{E_\xi r_\xi}{E_T} \right)}{\sqrt{n \left( \sum_{\xi=1}^{p} \frac{E_\xi r_\xi^2}{E_T} - \left( \sum_{\xi=1}^{p} \frac{E_\xi r_\xi}{E_T} \right)^2 \right)}} \sim N(0,1).$$

(114)

where

$$\hat{\mu}_0 = \tan^{-1} \left[ \frac{S}{\frac{n}{n \cdot E_T} - \frac{C}{C \cdot E_T}} \right]$$

where

$$S = \sum_{i=1}^{n} \sin \phi_i; \quad C = \sum_{i=1}^{n} \cos \phi_i$$

$$ES = \sum_{\xi=1}^{p} E_\xi \sin \phi_\xi; \quad EC = \sum_{\xi=1}^{p} E_\xi \cos \phi_\xi$$

$$E_T = \sum_{\xi=1}^{p} E_\xi.$$

Finally for the case of interaction, we have
\[
W_A = \sum_{i=1}^{n} r_i \cos (\phi_i - \mu_0) - n E_R(r_\xi \cos(\phi_\xi - \mu_0)) \quad \text{as } N(0,1)
\]

where \( E_R(\cdot) \) = expectation over the window of all regions under \( H_0 \)

i.e. \( \lambda(r_\xi, \phi_\xi) = \exp\{ \kappa \cos (\phi_\xi - \mu_0) \} \)

and

\( I_{\psi\psi}^{1/2}(0) \) is the leading element of the inverse observed information matrix under \( H_0 : \psi = 0. \)

The above hybrid model cannot be fitted directly by GLIM as the regional rates are different from those evaluated at the data points.

### 6.2 Hazard Variants

In all the above models we have assumed a multiplicative hazard, and have also used a log-linear model for derivation of a number of tests and ML estimators.

In medical examples, it may be argued that the spatial hazard around a point source is additive above a baseline effect. The rationale for this argument can be seen to lie in the belief that a pollution source would tend to increase above a baseline rate where it has particular influence. Hence, it may be more appropriate to examine an intensity/hazard such as

\[
\lambda^*(\psi) = P(\chi) h(z, \theta)(1 + \lambda(r, \phi))
\]

or

\[
E^*_\lambda = E_\xi h(z_\xi, \theta)(1 + \lambda(r_\xi, \phi_\xi))
\]
Muirhead and Darby (1987) have discussed a variety of such hazards for modelling of time-dependent radiation-induced cancer. Although, the parameter estimation in such a model may differ from the multiplicative case, it is still possible to fit any of these models in either the continuous or discrete case, using user-defined macros in GLIM, or purpose-written programs.

However, in the case of score tests, the effect of additive risk defined in (117) or (118), is to alter the variance estimate in the denominator of the test statistics by a constant amount. For example, the score test for multiplicative exponential trend given in (85), is

\[ W_M = \sum_{t=1}^{P} r_t (m_t - E_t^*) / S \]

where

\[ S = \sqrt{\left( \sum_{t=1}^{P} r_t^2 E_t^* - \left( \sum_{t=1}^{P} r_t E_t^* \right)^2 / m_t \right)} \]

Correspondingly, in the additive model we have

\[ \frac{\hat{\beta}}{\hat{\beta}_{H0}} = \sum \left( \frac{m_t r_t - e^{b r_t}}{(1 + e^{b r_t})} \right) - (\sum m_t) \frac{\sum E_t r_t e^{b r_t}}{\sum E_t (1 + e^{b r_t})} \]

and

\[ \frac{\hat{\beta}}{\hat{\beta}_{H0}} = \frac{1}{2} \sum m_t r_t - \frac{1}{2} (\sum m_t) \sum E_t r_t / \sum E_t . \]

In addition

\[ \frac{\hat{\beta}}{\hat{\beta}_{H0}} = (\sum m_t) \left[ \frac{\sum E_t r_t^2}{\sum E_t} - \frac{1}{4} \left( \frac{\sum E_t r_t^2}{\sum E_t} \right)^2 \right] - \frac{1}{4} \sum m_t r_t^2 . \]

Hence, the score statistic is
\[ W_A = \frac{\sum r_{\ell} (m_{\ell} - E_{\ell}^*)}{\sqrt{[\sum E_{\ell}^* r_{\ell}^2 - (\sum r_{\ell} E_{\ell}^*)^2/m_t] - \sum r_{\ell}^2 (m_{\ell} - E_{\ell}^*)}} \]

where \( m_t = \sum m_{\ell} \)

and \( E_{\ell}^* = \left( \frac{E_{\ell}}{\sum E_{\ell}} \right) m_t. \)

\( W_A \) differs from \( W_M \) only in its denominator and hence they may be expected to have different variances. Similar results hold for score tests for other parameters, and hence the score vector could be used to test under either model if Monte Carlo procedures are used.
CHAPTER 7

7 HEPP Models on GLIM

7.1 Normalising Constant Models

It can be shown that models which consist of a normalised intensity function, such as (1), can be fitted on GLIM by use of a numerical approximation to the log likelihood. A wide variety of Directional data models, as well as Spatial Point Process models are characterised by such a structure. Here we demonstrate the basic probabilistic and numerical approximations required by this technique. Berman and Rolf Turner (1988) (BRT) first suggested these approximations. We extend and amplify their ideas.

7.2 Numerical Approximation

We first consider the log likelihood (5):

\[ L_u = \sum \ln \lambda(\bar{x}_i) - \int_{A} \lambda(\bar{x}) d\bar{x} \]

= \sum \left[ \ln \lambda_i - \int_{A_i} \lambda(\bar{x}) d\bar{x} \right].

Here \( L_u \) is rewritten with the normalisation integral replaced by a sum of integrals over the dirichlet tiles of the process, as in (ch 4, eq 56). We now consider an approximation which will allow the use of a Poisson likelihood approximation to \( L_u \). Initially, we assume that \( \lambda(x) \) can be regarded piecewise constant within each Dirichlet tile and hence:

\[ L_{Ap} = \sum \left[ \ln \lambda_i - \lambda_i A_i \right] \]
$\xi_{Ap}$ is similar to a weighted Poisson log-likelihood if we introduce a variable $n_i/w_i$, where $n_i$ is a 0/1 indicator of a point event and $w_i$ is a weight ($= A_i$). The weighted Poisson log likelihood is then:

$$\xi_{wp} = \sum w_i \left( \frac{n_i}{w_i} \right) \ln \lambda - \frac{\lambda_i}{n_i}$$

This numerical approximation can be further justified by considering a probabilistic approximation.

### 7.3 Probabilistic Approximation

BRT have noted that a Bernoulli process on the line can be approximated by a Poisson model. We define a Spatial Bernoulli process with probability

$$p_\phi(x) = P_r(\text{an event in } \delta x) = \lambda_0(x) \delta x$$

$$N_i = \begin{cases} 
1 & \text{event in } \delta x \\
0 & \text{else.} 
\end{cases}$$

The planar window is divided into a rectangular mesh of small areas $\delta x$, chosen so that none of the cells contain more than one event. There are $M$ cells. The log likelihood for such a process is

$$\xi_{Ber} = \sum_{i=1}^{M} \{ N_i \log p_\phi i + (1 - N_i) \log (1 - p_\phi i) \},$$

as counts in disjoint regions are independent (Stoyan et al. 1987, 2.3). As the probability of an event in a specific cell $i$ will normally be small ($p_\phi i \ll 0.5$), we can make an approximation to $\log (1 - p_\phi i)$ from the first terms of a Taylor expansion. We replace $\log(1 - p_\phi i)$ by $-p_\phi i$ and this gives:
\[ \mathcal{L}_{\text{Ber}} = \sum_{i=1}^{M} \{ N_i \log P_{\phi_i} - (1 - N_i)P_{\phi_i} \} \]
\[ = \sum_{i=1}^{M} \{ N_i \log P_{\phi_i} - P_{\phi_i} + N_i P_{\phi_i} \}. \]

If we substitute \( \lambda_0(x)\delta_x \) for \( P_{\phi}(x) \) we have
\[ \mathcal{L}_{\text{Ber}} = \sum_{i=1}^{M} \{ N_i \log \lambda_{\phi}(x_i)\delta_x - \lambda_{\phi}(x_i)\delta_x + N_i \lambda_{\phi}(x_i)\delta_x \} \]
\[ = \sum_{i=1}^{M} \{ N_i \log \lambda_{\phi}(x_i) - \lambda_{\phi}(x_i)\delta_x + N_i \lambda_{\phi}(x_i)\delta_x + C \}. \]

Note that the first two terms are essentially those of \( \mathcal{L}_{wp} \), except that \( \delta_x \) is replaced by \( w_i \).

BRT exclude the third term and \( C \) is not dependent on the parameters.

### 7.4 Integral Approximation Accuracy

Given the possibility of using \( \mathcal{L}_{wp} \) within GLIM, it is important to consider the accuracy of the numerical approximation of \( \Lambda(A) \) by \( \sum \lambda_i a_i \). Usually a two-dimensional integral is approximated by a Product Rule numerical scheme (Davis and Rabinowitz, 1984, 5.6). This is defined as
\[
\Lambda(A) = \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} w_{1i} w_{2j} \lambda(i,j).
\]

Here the \( w_{1i}, w_{2j} \) are the weights of two separate one-dimensional approximation schemes, while \( \lambda(i,j) \) represents \( \lambda(.) \) evaluated at the point \( (i,j) \). This method requires the storage of \( M_1M_2 \) weights. BRT suggested the use of such a scheme by including the \( n \) data points in the scheme and using \( \mathcal{L}_{wp} \), not with tile area weights, but with an enlarged set of
M_1 M_2 \geq (n+2)^2$ mesh weights derived from trapezoid or Simpson's rule weights in each dimension.

The large number of mesh weights required by this scheme, prohibits its use on GLIM and an alternative extension to $\mathcal{W}_p$ was suggested. This involved a coarse grid of dummy points as well as data points, and the use of a single weight for each point. This gives a minimum-sized weight vector of $\geq n+4$. It was suggested that the Dirichlet tessellation or Delaunay triangulation of all the points could be used to provide weights. In the tessellation case, $w_i = A_i$, the area of the Dirichlet tile surrounding point $i$. If the Delaunay triangles are used then

$$w_i = \frac{\sum \text{area of all triangles sharing vertex } i}{3}.$$

The division by 3 is due to the fact that each triangle is shared by 3 points. One disadvantage of the above schemes, is that information contained in the tessellation of the data points alone is not available. This information could be useful when assessing residuals or fitting Bayesian models. However, if only the $n$ data points are used then there is a risk of numerical inaccuracy in evaluation of $\Lambda(A)$. We have developed an alternative scheme which uses only the $n$ data points in the tessellation but includes other weights available from the tessellation itself. Specifically, for data point $i$ we include the set $\{v_{ij} : j=1, n_{vi}\}$, where $v_{ij}$ are the tile vertex points for the $i$th point, and $n_{vi}$ is the number of vertices of the $i$th tile. We now calculate the $w_i$ for this extended set based on the triangles formed between the vertices and point $i$. We then use the Delaunay formula:

$$w_i = \frac{\sum \text{area of all vertex triangles for ith vertex}}{3}.$$

The data point, $i$, has a similar weight calculation. It is known (Miles, 1970) that for a homogeneous Poisson Process, the expected number of vertices on a Dirichlet tile is 6. Results for other processes have not been published, although it is possible to
demonstrate the result under general conditions. It is simple to show that for a tile with internal angles $\theta_1, \ldots, \theta_N$, we have $\sum_{i=1}^{N} (\pi - \theta_i) = 2\pi$. The expectation of the random sum on the LHS is $E(N)E(\pi - \theta_i)$. If each vertex is shared by 3 tiles then $E(\pi - \theta_i) = \pi/3$. Hence, the expected number of vertices is 6 for the standard Dirichlet tessellation tile. It follows that the expected number of integration weights will be $3N$. Hence the storage requirements on GLIM are dramatically reduced.

This method will always provide enough weights for any but the smallest $n$ values. Note, that the original $i$th tile area can be recovered by summation over all the weights associated with the $i$th data point or by multiplication by 3.

Applications of this tile vertex ($T_{\text{vert}}$) method will be considered in Section 7.5.4.

7.5 One Dimensional Models

When considering one-dimensional processes and their models, considerable simplifications are possible. First, the Dirichlet tessellation of a line consists of the length between successive mid-point bisections. Hence, storage considerations do not arise in this case. In fact, ordinary numerical integration schemes can be used, with data points added to the set of integration points. As it is simple to add or take out dummy points in the one-dimensional case, and storage is linearly increased then simple integration schemes are to be preferred over tessellation schemes as they are simpler to implement. For the one-dimensional case, $\xi_{wp}$ becomes:

$$\xi_{1D} = \sum w_i \left( \frac{n_i}{w_i} \right) \lambda n (S_j | F_{S_j}) - \lambda (S_j | F_{S_j})$$

where $F$ is the cumulative 'history' of the process.
We define a set of design points \( \{S_j\} \) so that \( 0 = S_1 < S_2 ... < S_M = S_T \), where \( S_T \) is the truncation point. This set contains as a subset the locations of point events \( X_i \in \{S_j\} \). We set \( N_j = 1 \) if \( S_j = X_j \) and \( N_j = 0 \) otherwise. Also \( \Delta_j = S_{j+1} - S_j, j = 1 ... M-1 \) denotes the length of the jth interval. A simple trapezoid rule type of approximation has been used here, so that:

\[
W_j = \begin{cases} 
\frac{1}{2} (S_2 - S_1), & j = 1 \\
\frac{1}{2} (S_{M+1} - S_M), & j = M \\
\frac{1}{2} (S_{j+1} - S_{j-1}), & j = 2,...,M-1 
\end{cases}
\]

where \( S_{M+1} = S_T \).

These weights give a linear approximation between each design point.

By suitable specification of \( \lambda(S_j | F_{S_j}) \) and \( S_T \) we can implement a variety of one-dimensional models. The following sections outline some possible applications.

### 7.5.1 The von Mises Distribution

We define \( S_T = 2\pi \) and then

\[
\lambda(\phi_i) = \exp(\beta_1 \cos \phi_i + \beta_2 \sin \phi_i)
\]

where

\[
\beta_1 = \kappa \cos \mu_0 \\
\beta_2 = \kappa \sin \mu_0.
\]

Now the log likelihood for a von Mises distribution is of the form

\[
\ell_{VM} = \sum \ln \lambda(\phi_i) - n \ln \int_0^{2\pi} \lambda(u)du.
\]

This is the conditional log likelihood (given n events), equivalent to \( \ell_{1D} \), except that in \( \ell_{1D} \) the baseline rate must be included. We can therefore use \( \ell_{1D} \) on GLIM and include the
GM term during a fit and this yields the required model fit. Note that GLIM will estimate \( \beta_1 \) and \( \beta_2 \) and a transformation is required to give

\[
\hat{\kappa} = \sqrt{\hat{\beta}_1^2 + \hat{\beta}_2^2} \quad \text{and} \quad \hat{\mu}_0 = \tan^{-1}\left(\frac{\hat{\beta}_2}{\hat{\beta}_1}\right).
\]

Delta method approximations to the variances and covariances of \( \hat{\kappa} \) and \( \hat{\mu}_0 \) which are \( o(n^{-3/2}) \) are available from:

\[
\text{var}(\hat{\kappa}) = (\hat{\beta}_1^2 \text{var}(\hat{\beta}_1) + \hat{\beta}_2^2 \text{var}(\hat{\beta}_2) + 2\hat{\beta}_1\hat{\beta}_2 \text{cov}(\hat{\beta}_1, \hat{\beta}_2))\kappa^2
\]

\[
\text{var}(\hat{\mu}_0) = (\hat{\beta}_2^2 \text{var}(\hat{\beta}_1) + \hat{\beta}_1^2 \text{var}(\hat{\beta}_2) + 2\hat{\beta}_1\hat{\beta}_2 \text{cov}(\hat{\beta}_1, \hat{\beta}_2))\kappa^4
\]

\[
\text{cov}(\hat{\kappa}, \hat{\mu}_0) = (\hat{\beta}_1 \hat{\beta}_2 \text{var}(\hat{\beta}_1) + \hat{\beta}_1 \hat{\beta}_2 \text{var}(\hat{\beta}_2) + \hat{\beta}_1^2 \text{cov}(\hat{\beta}_1, \hat{\beta}_2) - \hat{\beta}_2^2 \text{cov}(\hat{\beta}_1, \hat{\beta}_2))\kappa^3.
\]

\( \text{var}(\hat{\beta}_1), \text{var}(\hat{\beta}_2) \) and \( \text{cov}(\hat{\beta}_1, \hat{\beta}_2) \) are the variances and covariances of the estimated parameters on GLIM. These can easily be obtained via the $EXTRACT$ directive.

### 7.5.1.1 Numerical Comparison

In order to evaluate the numerical accuracy of the one-dimensional method in the case of the von Mises distribution we have carried out a simulation experiment. We simulated a \( M(2.094, \kappa) \) distribution (Best and Fisher (1979): IMSL routine : GGVMS) for \( \kappa = 0.5 \) and 5.0 respectively, and attempted to estimate \( \kappa \) and \( \mu_0 \) using conventional Newton methods (IMSL : ZXMWD, MMBS1\phi and MMBSII) and the above GLIM method with a variety of numbers of dummy weights (100, 300, 500, 1000). Table 3 displays the results of this study.
Table 3
Estimates of \( k \) and \( \mu_0 \) parameters using GLIM and IMSL routines ZXMWD and MMBSIO, MMBSII for Bessel Function evaluation. Samples of size 100 were generated from \( M(2.094, k) \) for \( k=0.5 \) and 5.0 respectively.

\[
\begin{array}{ccccccccc}
\text{GLIM: no. of dummy points} & \text{Newton methods} & \text{GLIM: no. of dummy points} & \text{Newton methods} \\
100 & 300 & 500 & 1000 & (IMSL) & 100 & 300 & 500 & 1000 & (IMSL)
\end{array}
\]

\[
\begin{array}{cccccccccccc}
\hline
k & 4.981 & 4.984 & 4.985 & 4.985 & 4.982 & 0.585 & 0.595 & 0.598 & 0.601 & 0.603 \\
\beta_0 & 2.084 & 2.084 & 2.084 & 2.084 & 2.174 & 2.233 & 2.247 & 2.253 & 2.257 & 2.242 \\
\text{var}(k) & 0.426 & 0.424 & 0.424 & 0.423 & 0.429 & 0.023 & 0.023 & 0.023 & 0.023 & 0.028 \\
\text{var}(\beta_0) & 0.002 & 0.002 & 0.002 & 0.002 & 0.002 & 0.061 & 0.058 & 0.058 & 0.058 & 0.057 \\
\text{cov}(k, \beta_0) & 0 & 0 & 0 & 0 & 2 \times 10^{-5} & 5 \times 10^{-4} & 2 \times 10^{-4} & 1 \times 10^{-4} & 1 \times 10^{-5} & 1 \times 10^{-5} \\
\text{var}(\beta_0^*) & 0.427 & 0.427 & 0.426 & 0.426 & 0.426 & 0.023 & 0.023 & 0.023 & 0.023 & 0.023 \\
\text{var}(\beta_0^*) & 0.002 & 0.002 & 0.002 & 0.002 & 0.002 & 0.061 & 0.059 & 0.058 & 0.058 & 0.057 \\
\hline
\end{array}
\]

* Exact asymptotic variances (Mardia, 1972, p. 125).

Standard errors are validly estimated in GLIM and \( O(n^{-3/2}) \) approximations to the variances of \( \hat{k} \) and \( \hat{\beta}_0 \) are also available from:

\[
\text{var}(\hat{k}) = (\hat{\beta}_1^2 \text{var}(\beta_1) + \hat{\beta}_2^2 \text{var}(\beta_2) + 2\hat{\beta}_1 \hat{\beta}_2 \text{cov}(\beta_1, \beta_2))/\hat{k}^4
\]

\[
\text{var}(\hat{\beta}_0) = (\hat{\beta}_1^2 \text{var}(\beta_1) + \hat{\beta}_2^2 \text{var}(\beta_2) - 2\hat{\beta}_1 \hat{\beta}_2 \text{cov}(\beta_1, \beta_2))/\hat{k}^4
\]

\[
\text{cov}(\hat{k}, \hat{\beta}_0) = (-\hat{\beta}_1 \hat{\beta}_2 \text{var}(\beta_1) + \hat{\beta}_1^2 \text{var}(\beta_2) + \hat{\beta}_2^2 \text{var}(\beta_1) - \hat{\beta}_1 \hat{\beta}_2 \text{cov}(\beta_1, \beta_2))/\hat{k}^3
\]

where \( \text{var}(\beta_1), \text{var}(\beta_2), \text{cov}(\beta_1, \beta_2) \) are the estimated (co)variances from GLIM for parameters \( \beta_1 \) and \( \beta_2 \). These are easily obtained by the use of the \texttt{SEXTRACT} directive.
The accuracy achieved on GLIM is comparable to Newton Methods. This is particularly true for the estimation of $\mu_0$ for large $\kappa$ for all GLIM dummy sets. In fact, in the case when $\kappa$ is low (0.5) and hence $\mu_0$ should be more difficult to estimate GLIM provides estimates closer to the true $\mu_0$, while having a similar variance to the Newton Method.


7.5.2 A Test for 'von Misesness'

The method of the previous section is not limited to the estimation of $\kappa$ or $\mu_0$ for a von Mises distribution. Cox (1975) suggested a Score test designed to detect the presence of higher order terms in the von Mises intensity function; that is

$$\lambda(\phi_i) = \exp(\alpha_1 \cos \phi_i + \beta_1 \sin \phi_i + \alpha_2 \cos 2\phi_i + \beta_2 \sin 2\phi_i).$$

We test $H_0, \alpha_2 = \beta_2 = 0$ against the alternative $\lambda(\phi_i)$ above. Mardia et al. (1984) have extended this to the spherical case.

It can be easily seen that a likelihood ratio test of the same hypothesis can be performed using the above GLIM method by examining the change in deviance when terms $\cos 2\phi$ and $\sin 2\phi$ are added to the model. This would be tested against $\chi^2_2$. This method now provides a hitherto inaccessible LR test procedure. Beyond this, we can fit a range of higher order terms to allow the assessment of smaller amplitude departures from 'von Misesness' e.g. $\cos \lambda \phi_i$, $\sin \lambda \phi_i$ where $\lambda \geq 2.$
We have analysed two examples using the above methods for inclusion of higher order terms. The first example is a simulation of 100 points from $M(\kappa, \mu_0)$ with $\kappa = 3, \mu_0 = 120^\circ$. This yields the following results, for a dummy mesh of 200 points:

<table>
<thead>
<tr>
<th>terms</th>
<th>deviance</th>
<th>df</th>
<th>$\Delta$dev (\Delta df)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1 + C + S$</td>
<td>220.2</td>
<td>297</td>
<td></td>
</tr>
<tr>
<td>$+ C2 + S2$</td>
<td>219.47</td>
<td>295</td>
<td>-0.73(2)</td>
</tr>
<tr>
<td>$+ C3 + S3$</td>
<td>215.03</td>
<td>293</td>
<td>-4.44(2)</td>
</tr>
<tr>
<td>$+ C4 + S4$</td>
<td>212.79</td>
<td>291</td>
<td>-2.24(2)</td>
</tr>
</tbody>
</table>

where $c_i, s_i$ represent $\cos i\phi, \sin i\phi$ terms.

Hence, none of the higher order terms add significantly to the model and $H_0$ is accepted in this case.

A data-based example was also analysed. We have fitted the above models to the marginal angular distribution of point events in the Armadale data set (set A). Fig 2b) displays the kernel estimate for the marginal angular distribution in Armadale. This demonstrates an increasing intensity of points from 0 to a peak around $240^\circ$ and then a marked decrease. There are few events in the south east quadrant. This pattern may suggest that lower amplitude terms may be significant. The following results were obtained:
It can be seen that the addition of the 3rd and 4th order terms yield a significant result for the order of entry specified here.

<table>
<thead>
<tr>
<th>terms</th>
<th>deviance</th>
<th>df</th>
<th>Δdev (Δdf)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 + C + S</td>
<td>222.57</td>
<td>246</td>
<td></td>
</tr>
<tr>
<td>+ C2 + S2</td>
<td>218.01</td>
<td>244</td>
<td>-4.57(2)</td>
</tr>
<tr>
<td>+ C3 + S3</td>
<td>205.80</td>
<td>242</td>
<td>-12.21(2)</td>
</tr>
<tr>
<td>+ C4 + S4</td>
<td>199.04</td>
<td>240</td>
<td>-6.76(2)</td>
</tr>
</tbody>
</table>

7.5.3 The Fisher Distribution

The method of the previous sections can be extended to the spherical case. While this is not related to the data examples used we include this here for record purposes. In general, Spherical distributions are characterised by normalising constants which are 2-dimensional integrals over the surface of sphere. Hence, if a suitable integration scheme is available there is no difficulty in applying $\mathcal{L}_M$ and fitting a variety of models in the spherical case with suitably defined $\lambda(\theta_i, \phi_i)$, where $\theta = \text{colatitude}$ and $\phi = \text{longitude}$ (polar coordinates). The tesselation method of Section 7.4 could be applied if a tesselation of the sphere surface were available. Currently, there appears to be no algorithm available for the Dirichlet Tesselation of the sphere, although Watson's Algorithm should be capable of adaptation (see, e.g. Sloan and Houlsby (1984)).
It is possible, however, to use one-dimensional methods for the special case of the Fisher Distribution.

Define
\[ g(\theta, \phi) = C_1 \exp \{ \kappa \{ \cos \mu_0 \cos \theta + \sin \mu_0 \sin \theta \cos(\phi - \nu_0) \} \} \sin \theta \]
where \( 0 < \theta < \pi \)
\( 0 < \phi < 2\pi \).

The pole for this distribution is \( \mu_0, \nu_0 \). If \( \mu_0 = \nu_0 = 0 \), then
\[ g_2(\theta, \phi) = C_2 \sin \theta \exp \{ \kappa \cos \theta \} \]
\( 0 < \theta < \pi \)
\( 0 < \phi < 2\pi \).

Hence \( \phi \) has a uniform distribution on \((0, 2\pi)\), and only \( \theta \) requires normalisation in \( C_2 \).

We proceed, as follows:

1) estimate \( \mu_0 \) and \( \nu_0 \) and make a transformation of \( \theta, \phi \) to
\( \theta', \phi' \) with \( \mu_0' = \nu_0' = 0 \)
\( \theta', \phi' \) now have density \( g_2(\theta', \phi') \).

2) On GLIM, fit \( \cos \theta \) with \( \log(\sin \theta) \) as a known parameter vector (\$OFFSET command).

The estimate of \( \kappa \) is the fitted parameter for \( \cos \theta \) in this model.
7.5.3.1 Data Example

We have analysed data set B5 of Fisher et al. (1987, p.284). This data set consists of 52 measurements of magnetic remanence from specimens of red beds from the Bowen Basin, Queensland, Australia. The coordinates are declination (Dec) and inclination (inc) of the specimens. Note that $\theta = \text{inc} + 90^\circ$ and $\phi = 360^\circ - \text{Dec}$. For this data example, we have calculated summary statistics, as follows:

$$\bar{R} = 0.1472$$

mean direction components:

0.7971

-0.0036

-0.6038

polar means ($\hat{\alpha}, \hat{\beta}$): 127.14°, 359.8°.

Using table A10, (Fisher et al. (1987), p.264) the estimate of $\kappa$ is 0.448. After rotation on GLIM using

$$A(\hat{\alpha}, \hat{\beta}, 0) = \begin{pmatrix}
\cos \hat{\alpha} \cos \hat{\beta} & \cos \hat{\alpha} \sin \hat{\beta} & -\sin \hat{\alpha} \\
-\sin \hat{\beta} & \cos \hat{\beta} & 0 \\
\sin \hat{\alpha} \cos \hat{\beta} & \sin \hat{\alpha} \sin \hat{\beta} & -\cos \hat{\alpha}
\end{pmatrix}$$

and

$$\begin{pmatrix}
\sin \theta' \cos \phi' \\
\sin \theta' \sin \phi' \\
\cos \theta'
\end{pmatrix} = A(\hat{\alpha}, \hat{\beta}, 0) \begin{pmatrix}
\sin \theta \cos \phi \\
\sin \theta \sin \phi \\
\cos \theta
\end{pmatrix}$$

we fitted $\cos \theta'$ with $\log(\sin \theta')$ as offset and we obtained the following results:

<table>
<thead>
<tr>
<th>terms</th>
<th>deviance</th>
<th>df</th>
</tr>
</thead>
</table>


1 + cos $\theta'$ 217.09 250

(with 200 dummy points)

<table>
<thead>
<tr>
<th>term</th>
<th>parameter estimate</th>
<th>S.E.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.015</td>
<td>0.1432</td>
</tr>
<tr>
<td>2</td>
<td>0.4438</td>
<td>0.2449</td>
</tr>
</tbody>
</table>

The GLIM estimate of $\kappa$ is 0.4438. This is very close to the table estimate. Note that this does not imply that the Fisher Distribution is a 'good' model for this data set. Hence, GLIM appears to provide a flexible means of estimating parameters for such a model, but has the major advantage that the model can be modified or extended easily by inclusion of higher order terms of covariates.

### 7.5.4 Spatial HEPP Model

We here give two examples of the application of the tesselation weight method for spatial HEPP models. The first example is a simulation of a HEPP model on the unit square. The model has IF $\lambda(x) = \exp(-\lambda_x x_1 - \lambda_y x_2)$ where $\lambda_x = 1.0$ and $\lambda_y = 2.0$. We have simulated this model, using the conditioning method, for $n = 30$ points. Figure 15 displays the distribution of points generated. We have estimated $\lambda_1$ and $\lambda_2$ by direct maximisation of the log likelihood (IMSL: ZXMWD) to give ML estimates of $\hat{\lambda}_x = 1.463$ and $\hat{\lambda}_y = 1.858$. On GLIM we have used a variety of dummy schemes to allow estimation of the parameters. The following results (Table 4) have been obtained, and provide an assessment of the accuracy of the numerical approximation. It can be seen that even using
Figure 15
Point map: 30 points simulated on the unit square
the data points alone, the parameter estimates are close to the original estimates from direct maximisation. The addition of a dummy *mesh* appears to yield estimates slightly closer to the original estimates, but not to the true parameter values. The vertex weights appear to reduce the value of estimates, except in the case of a 7×7 dummy mesh which produces the

<table>
<thead>
<tr>
<th>no. of points</th>
<th>terms</th>
<th>dev (df)</th>
<th>Null Dev (df)</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>5.762</td>
<td>10.61(27)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.359</td>
<td>20.5(29)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.4075)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.685)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.763)</td>
<td></td>
</tr>
<tr>
<td>30 + (5×5) dummies</td>
<td></td>
<td>5.852</td>
<td>43.8(52)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.377</td>
<td>55.99(54)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.390)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.645)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.671)</td>
<td></td>
</tr>
<tr>
<td>30 + 152 vertices</td>
<td></td>
<td>4.641</td>
<td>66.89(174)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.02</td>
<td>74.26(176)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.397)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.702)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.793)</td>
<td></td>
</tr>
<tr>
<td>30 + (5×5) dummies +290 vertices</td>
<td></td>
<td>4.519</td>
<td>111.24(334)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.029</td>
<td>120.1(336)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.3917)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.647)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.661)</td>
<td></td>
</tr>
<tr>
<td>30 + (7×7) dummies</td>
<td></td>
<td>5.884</td>
<td>66.0(76)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.421</td>
<td>78.6(78)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.3917)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.654)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.674)</td>
<td></td>
</tr>
<tr>
<td>30 + (7×7) dummies +420 vertices</td>
<td></td>
<td>4.764</td>
<td>132.0(492)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.245</td>
<td>144.0(494)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.397)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.659)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.704)</td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Numerical accuracy results for 2-D HEPP parameter estimation on GLIM.
closest approximation to the true parameter values (i.e. 1.245, 1.918). Note that the profile likelihood estimate of the baseline intensity (term 1) is 4.6994 (i.e. \( \ln (n/\Lambda) \)). The vertex weights appear to yield a closer estimate of this parameter. For examples with larger numbers of datapoints, smaller dummy arrays would be required.

The second example of the use of tessellation weights is in the analysis of data set G, the distribution of volcanic ejecta after an eruption of Mt Asama. We will illustrate the method with the data for 1935 (vol 35). This is displayed in Figure 1g. We will discuss more fully modelling of this data set in Chapter 9.

Vol 35 consists of point locations of 46 ejecta distributed around an eruption centre. The surface and marginal density estimates (Figures 5a, b, c), suggest a peaked radial effect with a strong angular concentration in the east. Either an interaction model, with terms in \( r \cos \phi \) and \( r \sin \phi \), or a \( \delta \)-dependence model with terms in \( \ln(r) \cos \phi \) and \( \ln(r) \sin \phi \) may be appropriate. We have fitted both models using Delaunay triangulation weights with a mesh of dummy points which provide a convex hull approximation to a circular window. The Delaunay triangles always form a convex hull and, hence, a fine mesh of dummy points on the circular boundary should yield a convex hull approximation to a circle.

On GLIM we fit the terms \( r, \ln(r), \cos \phi, \sin \phi, r \cos \phi, r \sin \phi, \ln(r) \cos \phi, \ln(r) \sin \phi \), in different combinations.

Note that by a suitable transformation we can obtain \( \kappa, \psi, \) and \( \mu_0 \) for the interaction model:

\[
\hat{\kappa} = \sqrt{\hat{\beta}_1^2 + \hat{\beta}_2^2}, \quad \hat{\psi} = \sqrt{\hat{\beta}_3^2 + \hat{\beta}_4^2}
\]

\[
\hat{\mu}_0 = \tan^{-1}\left(\frac{\hat{\beta}_2 + \hat{\beta}_4}{\hat{\beta}_1 + \hat{\beta}_3}\right)
\]
where $\beta_1$ and $\beta_2$ are the GLIM estimates of the parameters estimated for $\cos \phi$ and $\sin \phi$ terms. $\beta_3$ and $\beta_4$ are the GLIM estimates for the $r \cos \phi$, $r \sin \phi$ terms.

Appendix VII contains details of the GLIM macros used, while Appendix VIII contains details of the parameter approximations and variance-covariance matrix approximations used.

The following results (Table 5) were obtained for the vol 35 data set:

<table>
<thead>
<tr>
<th>terms</th>
<th>estimate</th>
<th>SE*</th>
<th>dev</th>
<th>df</th>
<th>$\phi_{max}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-5.506</td>
<td>0.147</td>
<td>137.6</td>
<td>75</td>
<td>-299.3</td>
</tr>
<tr>
<td>1</td>
<td>-132.27</td>
<td>16.172</td>
<td>36.6</td>
<td>73</td>
<td>-248.8</td>
</tr>
<tr>
<td>R</td>
<td>-0.863</td>
<td>0.102</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LR</td>
<td>43.839</td>
<td>5.432</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$\chi_{sat} = -230.5$

* adjusted for mean deviance

terms: $R = r$; $LR = \chi_n r$

In this case, the 'best' polar coordinate model is simply a function of radial distance (R) and its log (LR). There is no additional angular component of significance, after the radial terms are fitted. Hence a pattern of radial decline with a large peak is predicted. We will discuss this example in more detail in Chapter 9.
CHAPTER 8

8 Model Simulation and Test Statistic Behaviour

In this Chapter we present simulated realisations of both continuous and discrete models discussed in previous chapters, and the results of numerical power studies of test statistics and evaluation of Monte Carlo statistical testing.

8.1 Model Simulation

8.1.1 Continuous Models

8.1.1.1 HEPP Models

A general rejection method of simulation from HEPP models has been suggested by Lewis and Shedler (1979), (see also Ogata (1981)). However, the models proposed here have bivariate pdfs, which are amenable to a simple conditioning method. All simulations are also conditional on $n$ events.

The model $\lambda(t) = f(r) \cdot g(\phi, r)/r$ given in (11), (12) can be simulated by

a) generation from the marginal of $R$ and conditional on $R = r$, generate $\Phi$ from $M(\mu_0, \kappa+\psi r)$. For $M(\mu_0, \kappa)$ models the $R, \Phi$ components are independent and hence the simulation is straightforward. In the case of the delta dependence model, we

a) generate from the marginal $M(\mu_0, \kappa)$ of $\Phi$ and conditional on $\Phi = \phi$, generate $R$ from a Weibull $(\lambda, \delta)$, where $\delta = \delta_0 + \delta_1 \cos(\phi - \mu_0)$.

A range of HEPP realisations in a circular window are displayed in Figs 10a, b, c, 11 a, b, c, d. Figure 10 demonstrates the model for

$$f(r) = \lambda \delta r^{\delta-1} \exp (-\lambda r^{\delta})/r$$
\[ g(r, \phi) = \exp \left( (\kappa + \psi r) \cos (\phi - \mu_0) \right) / 2\pi I_0(\kappa + \psi r) \]

while Fig 11 displays the \( \delta \)-dependence as defined above.

In each case we display a realisation of \( n = 100 \) points. Fig 10a demonstrates a homogeneous Poisson Process. Fig 10b demonstrates the effect of high concentration (\( \kappa = 2 \)) with exponential distance decline and low \( \psi \) (\( \lambda = 1; \delta = 5; \psi = 1 \)), while figure 10c displays a similar distribution (\( \lambda = 1; \delta = 5; \kappa = 2 \)) with high interaction (\( \psi = 50 \)). Figure 11 displays the effects of the inclusion of \( \delta_1 \) in the model. Figure 11a depicts no \( \delta \)-dependence (\( \delta_1 = 0; \kappa = 2 \)) with peaked radial trend (\( \lambda = 1; \delta_0 = 5 \)). Figure 11b depicts high \( \delta \)-dependence and a distance-peak effect (\( \delta_1 = 10; \kappa = 2; \lambda = 1; \delta_0 = 5 \)). Figures 11 c, d display high concentration and interaction. It is apparent that interaction acts as a concentration parameter except that at greater radial distances it produces greater concentration relative to the equivalent setting of the concentration parameter. In section 2.3.2, we demonstrated the closeness of the marginal distribution of \( \Phi \) to a von Mises form. The \( \delta \)-dependence parameter (\( \delta_1 \)) tends to allow an anisotropic distance peak.

It is possible to demonstrate the applicability of such models to data examples by generating simulations of the point patterns based on the model parameters estimated from the data. We consider the full analysis of data examples in Chapter 9. However, we present here two examples of simulated realisations which suggest 'visually' the general feasibility of the HEPP models. We have simulated Data set d) and Data set g) and Figures 16a,b represent these data sets. These figures should be compared with Figures 1)d) and 1)g).
Figure 16 a), b)

Simulation of 1975 Hebeloma data using ML estimates from a 5 parameter HEPP model: $\hat{s} = 3.85$, $\lambda = 12.907$, $R = 0.00$, $\phi = 2.24$, $\hat{\kappa}_0 = 2.625$

Simulation of 1935 volcanic ejecta data using ML estimates from a 5 parameter HEPP model: $\hat{s} = 7.322$, $\lambda = 0.0158$, $R = 0.0$, $\phi = 0.698$, $\hat{\kappa}_0 = 5.224$
8.1.1.2 **Bayesian Spatial Prior Models (BSP)**

For these models it is possible to consider a two stage method. First, generate on a fine mesh of points a vector \( g \) distributed as MVN \((g_n, \kappa_n)\). Second we thin the intensity \( \lambda(t) = e^{g(t)} \) using the Lewis and Shedler Method. The details are given in Appendix II. This is essentially the method suggested by Diggle (1983, p.60) for simulation of Cox processes.

We assume that \( g_n = F_n \beta \) and \( \sigma^2 \), the prior variance and \( \alpha \), the covariance range parameter, are fully specified prior to simulation. We also condition on \( n \) events, and hence in this method must generate a large number of mesh points to be certain of achieving \( n \). We use 10,000 points in the unit square, approximately 0.785 of this number will lie within the unit disc. The generation of large numbers of MVN variates requires special techniques (see Appendix II).

We have generated a selection of four realisations of this log-normal intensity process in Figure 17. Figures 17 a) represent the variation in covariance range parameter \( (\alpha) \), 17 b) the variance \( (\sigma^2) \) of \( g \). Figure 17 c) represents the variation in range with trend parameters \( (\lambda, \delta, \kappa, \phi, \mu_0) \). An overall comparison between the HEPP models and the BSP models suggests that variation in \( \sigma^2 \) can produce spatial variation of intensity of point events (Fig 17 b). Also, the range parameter \( (\alpha) \) appears to have little effect (over the conventional range) on the simulations (Figure 17 a, c). This effect may be due to the log-normal model structure, in that any generated \( g \) value less than 0 will yield very low acceptance probabilities. This would become important if \( g_n \) is also close to 0. Hence, increasing \( \sigma^2 \) in this case will yield greater proportions of \( g \)'s < 0. Whereas, for fixed \( \sigma^2 \) alteration of \( \alpha \) should not yield a large increase in numbers of negative \( g \)'s.
Figure 17 a), b)

BSP simulation on the unit disk

a) variation in range parameter ($\alpha$) (no trend)

b) variation in process variance ($\sigma^2$)
Figure 17 c)

BSP simulation

on the unit disk  c) variation in range parameter ($\kappa$) (trend: $\kappa = 1$)
8.1.2 Discrete Models

8.1.2.1 Multinomial Models

Simulation of multinomial models can be easily achieved by direct generation from (63) using the alias method (Kronmal and Peterson (1979); Kemp and Kemp (1987)). This involves the evaluation of (63) for each of P regions, and is easily achieved using a standard library routine (e.g. IMSL : GGDA). The regional system used for such a simulation must be designed for a specific purpose. In the present case, our ultimate goal is to evaluate the power of test procedures and carry out Monte Carlo testing. Hence, it is appropriate to design a regular region mesh, which can be used for such purposes. The use of such a mesh does not allow the simulation of the arbitrary regionalisation of eds. Any data example which included eds would require a digitised map of, at least, the region centres if not region boundaries.

Here, we confine our simulation examples to a regular mesh of annular segments around a central point. The number of radial or angular cells in the mesh can be varied as required.

The region probabilities given in (53) can have a variety of hazard functions and it is easy to incorporate covariates or $E_5$s (expected events) in this simulation method. (In the continuous case, resort must be made to the thinning method, if covariates are to be included in HEPP models.) Figures 18 and 19 show a range of simulations produced for a variety of spatial trend parameters and a variety of expected death patterns in the regions. Figure 18 depicts a) with trend, and b) peaked trend. Figure 19 shows the effect on variation in angular parameters.
Figure 19 depicts the effects of concentration and interaction. Counts are high around the mode and sparse close to the anti-mode.

A regional simulation could be performed by generation of a HEPP model and binning the points into regional cells. However, these models are not exactly equivalent as the discrete simulation is population-based and not an approximation to an area-based HEPP model. This said, it might be possible to 'adjust' such a HEPP simulation to reproduce the discrete case. We do not pursue this here.

\section{8.1.2.2 Bayesian Spatial Prior Models (BSP)}

The BSP models in the discrete case are relatively simple to simulate. Computational costs restrict us to a small number (P) of regions. In our simulations, and usually in real examples, P << 1000. In our data examples P < 200, and for multinominal simulations we do not need to consider meshes with more than 200 cells. Hence the spatial MVN (\(\xi_n, \kappa_n\)) prior density for \(g_\xi\) can easily be simulated using Cholesky decomposition. In the author's experience, decomposition is relatively efficient up to n-vector sizes of 400-500.

The number of events simulated in each cell can be derived by the alias method applied to the cell probabilities: 
\[
\exp(g_\xi) / \sum_{\xi=1}^P \exp(g_\xi).
\]

Figures 20 and 21 depict simulations of the discrete model with log-normal hazard, with and without trend. Figure 20, 21 a), b), c), d) represent different values of \(\alpha\) and \(\sigma^2\) in the prior density. An effect similar to that of the continuous case (8.1.1.2) is found here, i.e. variation in \(\alpha\) does not yield as large an effect as variation in \(\sigma^2\).
Figures 18 and 19

Figure 18
Discrete simulation (5 x 10 radial grid)
(a) exponential radial trend ($\lambda = 5$)
(b) peaked ($\delta = 1.5, \lambda = 3$)

\[ n = 200 \]

\[ \delta = 0.5 \quad \lambda = 5 \quad k = 1 \quad \psi = 0 \quad \mu = 1 \]

\[ \delta = 1.5 \quad \lambda = 3 \quad k = 1 \quad \psi = 0 \quad \mu = 1 \]

Figure 19
Discrete simulation (5 x 10 radial grid) $n = 400$
(a) angular concentration ($k = 5$)
(b) angular interaction ($\psi = 5$)
(c) concentration and interaction ($k = 5, \psi = 5$)
Discrete BSP simulation (5 x 10 radial grid)  
$n = 400, \lambda = 5, K = 5, \delta = 1, \mu_0 = 1$

a) $\alpha = 1, \sigma^2 = 2$

b) $\alpha = 0.1, \sigma^2 = 2$. 

Figure 20
Figure 21

Discrete BSP simulation (5 x 10 radial grid) \( n = 400 \) \( \lambda = 5, \delta = 1, \mu_0 = 1 \)

a) \( \lambda = 1, \sigma^2 = 5 \)
b) \( \lambda = 1, \sigma^2 = 0.5 \)
c) \( \lambda = 10, \sigma^2 = 1 \)
d) \( \lambda = 0.1, \sigma^2 = 0.1 \)
8.2 Test Statistic Behaviour

Simulation allows a visual assessment of model behaviour but our ultimate goal is to examine the sampling behaviour of test statistics and the adequacy of asymptotic approximations and to assess the relative power of these tests.

8.2.1 Sampling Distributions

In previous sections a number of models were presented and associated test statistics were derived. These test statistics were based on likelihood methods and all had approximate asymptotic chi-squared or standard normal sampling distributions. This 'approximation' can be invalid for small samples. It is also useful to be able to assess tests over a wide range of sample sizes. Uncertainty concerning test critical regions can be overcome by the use of Monte Carlo testing, which involves simulation of $H_0$ $m$ times, using estimated parameter values as if known to produce $t_i^*$ ($i = 1, m$) and calculation of the joint ranking of the test statistic $t^*$ among the $m+1$ results. $H_0$ is rejected in a one-sided test of size $k/(m+1)$ if $t^*$ ranks $k$th or larger (Diggle (1983, p.7)). Ripley (1987, p.173-174) notes that power loss relative to an exact test is, for exact concave power function $\beta(\alpha)$

$$\beta_m(\alpha)/\beta(\alpha) \geq 1 - \left[ \frac{(1 - \alpha)}{2\pi m\alpha} \right]^{1/2}.$$
Hence for a 5% level the lower bound of the power of a M.C. test relative to an exact test is 0.825 \((m = 99)\) and 0.945 \((m = 999)\). Ripley (op. cit.) further suggests that 99 is a reasonable value for \(m\) in most significance testing contexts.

In the following sections, we use both asymptotic and Monte Carlo critical regions for evaluation of test statistics. In cases where the null hypothesis has nuisance parameters, we replace them by their ML estimates under \(H_0\), and simulate the profile likelihood for Monte Carlo testing.

8.2.2 Numerical Power Studies

In previous sections a number of test statistics have been developed but have not been compared with other tests which are aimed at detection of particular departures from \(H_0\). The criterion often used for assessment of a test statistic is its power function. This function measures the probability of rejecting \(H_0\) under both \(H_0\) and \(H_1\). In essence, the power function describes how good a test is at detecting a particular alternative. Test statistics designed to detect a general type of alternative can have markedly different power functions for a particular \(H_1\).

In this study, we use simulation to assess the power of a range of test statistics using asymptotic and Monte Carlo critical regions. The stages of the method used are 1) select values for all nuisance parameters, 2) simulate \(H_0\) \(m\) times, 3) calculate the \(j\)th test statistics \(t_{ij}\) for the \(i\)th simulation \((i=1, m)\), 4) for each \(j\) compare all \(t_{ij}\) against \% points of sample distribution, 5) power = no. of rejections/\(m\). Usually, we can define either a one- or two-tailed test, appropriately, and use a number of \% point levels. In the numerical results reported here, we give only the 5\% level for one-tailed tests. Other recorded levels
(1%, 0.1%) behaved in a similar manner. Results for two-tailed tests are similar to those for one-tailed tests. We can define two cases: tests for continuous and discrete data types.

8.2.2.1 Continuous data tests

In the continuous case, we have developed a likelihood ratio (LR) and Score (WS) test for interaction based on the $M(\mu_0, \kappa+\psi r)$ distribution. This test is designed to detect a particular form of angular-linear correlation and hence it is appropriate to compare such a test with other tests for such effect. The tests here considered are the angular-linear correlation coefficient ($R^2$) (Mardia & Sutton (1978); Jupp and Mardia (1980); Liddell and Ord (1978); Mardia (1976)), Moore's test (M) for directionality (Moore, 1980), Mardia's rank correlation coefficient (RM), Mardia (1975) and Fisher and Lee's rank test (FL) (Fisher and Lee, 1981).

The test $R^2$ is a general parametric angular-linear correlation coefficient, while test $M$ is a non-parametric test of concentration and interaction. The test RM is a rank-based non-parametric equivalent to $R^2$, and FL is a non-parametric angular-linear correlation test based on Kendall's Tau procedures. In our case, FL is taken as the incomplete statistic, which is an average of 200 subsets of 4 cases sampled with replacement from the realisation.

We have included a range of both parametric and non-parametric tests to provide an adequate range of comparison to LR and WS. All the statistics have known asymptotic sampling distributions. Details of these are given in Appendix X. A variety of values of $\lambda$, $\delta$ and $\kappa$ were assumed for the distance distributions and angular concentration to allow for possible dependence of power on these effects. Figure 22 a)-f) depict the results of simulations for the 6 tests described above using asymptotic critical regions. Three different sample sizes were used ($n = 30, 100, 500$) and one-sided tests at 5% level only
Figure 22 a), b), c) Numerical Power Studies: Continuous case: Asymptotic critical regions $n = 30, 100, 500$

downwards

a) $\delta = 1, \lambda = 0.001, \kappa = 0.001$

b) $\delta = 1, \lambda = 1.0, \kappa = 0.001$

c) $\delta = 2, \lambda = 1.0, \kappa = 0.001$
Figure 22 d), e), f)

d) \( \delta = 1, \lambda = 0.001, K = 1.0 \)

e) \( \delta = 1, \lambda = 1.0, K = 1.0 \)

f) \( \delta = 2, \lambda = 1.0, K = 1.0 \)

y axis: \( \Pi(\psi) \); x axis: \( \psi \) symbols

- Moore's test (M)
- Angular-Linear correlation (R2)
- Score test (WS)
- Likelihood ratio test (LR)
- Mardia's Rank test (RM)
- Fisher-Lee test (FL)
are given. Two parameter levels were used for each nuisance parameter. These were: \( \delta = 1 \) or \( 2 \), \( \lambda = 0.001 \) or \( 1.0 \); \( \kappa = 0.001 \) or \( 1.0 \).

Overall, the results show satisfactory behaviour for WS over a range of parameter values with power close to the nominal level under \( H_0 \) and a steep increase in power in the range \( 1 < \psi < 5 \). This behaviour is more marked for large sample sizes. Both LR and R2 show more gradual power increases with increasing \( \psi \) value. They attain 100\% rejection only for higher levels of \( \psi \) than WS. In addition R2 always yields a very high rejection under \( H_0 \) which suggests that the size for this test is incorrect.

In smaller samples (\( n = 30 \)), neither test achieves 50\% rejection at \( \psi = 10.0 \). With a peaked radial distribution (\( \delta = 2 \)) both LR and R2 are less powerful than under a monotonic distribution. The non-parametric tests RM and FL show differing results. In general, RM is most powerful, and has similar behaviour to WS for large \( n \) (\( n = 500 \)). FL has very poor power characteristics and is little enhanced by sample size increase or other variations in parameters. This is similar to, though more dramatic than, the findings of Bain et al (1985) who examined parametric and non-parametric tests for simple trend. A peaked radial distribution (\( \delta = 2 \)) reduces the power of both RM and FL, although at \( n = 500 \), RM still behaves as WS.

The effect of negligible concentration (\( \kappa = 0.001 \)) on the tests is marked. First, M shows a profile like WS, but rejects at lower \( \psi \) values. As M depends on concentration it yields 100\% rejection for \( \kappa = 1.0 \) for all \( \psi \) values. Hence, M is powerful only where interaction alone occurs. LR, WS, RM and R2 all have increased power and steeper profiles than when \( \kappa > 0.001 \). Note, however, that R2 does not show a better response at the larger sample size (\( n = 500 \)).

Overall, it appears that WS has the most favourable characteristics across a wide variety of parameter settings and sample sizes. R2 and LR show similar though more gradual profiles. The best non-parametric test is RM which shows little power loss in large
Numerical Power Studies: Continuous case: Monte Carlo critical regions \( n = 50 \) only

a) \( \delta = 1, \lambda = 0.001, \kappa = 0.001 \)
b) \( \delta = 1, \lambda = 1.0, \kappa = 0.001 \)
c) \( \delta = 2, \lambda = 1.0, \kappa = 0.001 \)
d) \( \delta = 1, \lambda = 0.001, \kappa = 1.0 \)
e) \( \delta = 1, \lambda = 1.0, \kappa = 1.0 \)
f) \( \delta = 2, \lambda = 1.0, \kappa = 1.0 \)
samples (n = 500) compared with its parametric competitors. Moore's test (M) can be recommended only when there is negligible concentration, when it outperforms RM and WS.

Given that the above results use only asymptotic distributions it may be considered that comparisons are not strictly valid for n = 30, particularly in the case of RM and FL. However, the power differences are obvious from other sample sizes and both non-parametric tests perform relatively poorly even at n = 100. In addition, the author has examined the form of the Monte Carlo null distributions of test statistics, and kernel estimates of these distributions suggest close agreement with a $N(0,1)$, although in some cases a slight positive bias may make the power calculation anti-conservative.

Figure 23 a)-f) display the results of a small monte carlo power study of the above test statistics. We have examined the sample size n = 50 only, as this represents the smallest size encountered in the current study. The overall results are very similar to those of Figure 22 with n = 30 or 100. The notable differences are in the position of Moore's test. Even for n = 50 the test shows 100% rejection under $\kappa = 1.0$. The relatively low power of the Fisher & Lee test is further supported by the very low profile when $\lambda > 0.001$. All tests appear to have lower power profiles when radial parameters are increased, than in the asymptotic case. There is also some evidence of some slight changes in relative position of tests.

Overall, the score test (WS) demonstrates favourable characteristics across a wide variety of parameter settings.

Note that the power study above is purely against an interaction alternative i.e. ($\psi > 0$). Other power studies of angular-linear correlation, for example, as reported in Fisher & Lee (1981), examine different correlation models under the alternative. Hence, this author has reported that FL has relatively bad power characteristics, whereas under other alternatives the power of FL is comparable to RM. This author has examined some of
these alternatives and it appears that FL does perform relatively well against RM. Hence, the evidence suggests that only under our model does FL perform badly.

8.2.2.2 Discrete data tests

In the discrete case we have developed a score test for interaction (WS) given in (77)-(79). The equivalent LR statistic (LR) is available in GLIM and hence was not discussed in an earlier section. However, it would be advantageous to compare such a test with WS and other equivalent angular-linear correlation tests available for the discrete case. As far as this author is aware, there are no tests of either parametric or non-parametric form which have been developed for this case. One obvious way to derive a general correlation test would be to consider a weighted angular-linear correlation coefficient. An intuitive modification of (R2) would be to allow weighting of the sample correlations: \( r_{12} = \text{corr}(x, \cos \theta) \), \( r_{13} = \text{corr}(x, \sin \phi) \), \( r_{23} = \text{corr}(\cos \theta, \sin \theta) \) in the general correlation coefficient (R2):

\[
R_{x\theta}^2 = (r_{12}^2 + r_{13}^2 - 2r_{12}r_{13}r_{23})/(1 - r_{23}^2).
\]

The correlation \( \text{corr}(x,y) \) can then be defined as

\[
\text{corr}(x,y) = \frac{\sum_{\ell=1}^{P} w_x x_{\ell} y_{\ell} - \bar{x} \left( \sum_{\ell} w_y y_{\ell} \right)}{\sqrt{\left( \sum_{\ell} w_x x_{\ell}^2 - \left( \sum_{\ell} w_x x_{\ell} \right)^2 \right) \left( \sum_{\ell} w_y y_{\ell}^2 - \left( \sum_{\ell} w_y y_{\ell} \right)^2 \right)}}
\]

where

\( w_x \) = weight for \( \ell \)th region

\( x_{\ell}, y_{\ell} \) are the \( \ell \)th region variables

\( \bar{x}, \bar{y} \) are weighted means of \( x \) and \( y \) over all regions.
Note that \( w \) could represent the region event count \( m \), the expected events in the region \( E \), or even the SMR \( = m / E \). The main difficulty in the use of (109) is that this weighted form of \( R^2 \) will not necessarily have the same asymptotic distribution as the original \( R^2 \). It is possible to simulate the sampling distribution of the weighted \( R^2 \) under \( H_0 \) and thence use a kernel estimate or spline to evaluate particular % points and hence to evaluate the power function.

For the purposes of general testing using this statistic it is possible to carry out Monte Carlo tests as described previously, or, alternatively it is possible to use a Bootstrap method. The Bootstrap has been shown to yield an asymptotic distribution equal to the true distribution in the finite sample case (see Efron (1982, p.34) or Ripley (1987, p.175)). Hence it is possible to estimate both the bias \( E_b(R^2) \) and variance \( \text{var}_b(R^2) \) via the bootstrap and for large use \( z_B = \frac{R^2 - E_b(R^2)}{\text{var}_b(R^2)} \) as a standard normal variate. The bootstrap can also be used in numerical power studies, as \( z_B \) can be calculated for any sample from \( H_0 \) and hence could be compared to the % points of the \( N(0,1) \) distribution.

In the numerical study we have examined similar parameter settings to the continuous case. We have used the bootstrap distribution of statistic \( R^2 \) to allow comparison with WS and LR defined above. We used 1000 simulations of \( H_0 \) and examined three% points as in the continuous case. We only present results for one-tail tests and 5% level as other test levels show similar results. Figures 24 a)-f) depict the results of these power studies.

In general, these displays show very steep power profiles. For example, most statistics show 100% rejection by \( \psi = 3.0 \). In the continuous case, only at \( n = 500 \) do most statistics show such behaviour.
Figure 24: Discrete power studies: asymptotic results \( n_M = 400, n_F = 400 \)

Power curves for a One-sided test at \( \alpha = 0.05 \) level for 3 angular-linear tests.

Each column represents a unique combination of parameters \((\delta, \lambda, \kappa)\). Each curve represents a different test.

Column: 
- a) \( \delta = 0.001; \lambda = 0.001; \kappa = 0.001 \)
- b) \( \delta = 0.001; \lambda = 1.0; \kappa = 0.001 \)
- c) \( \delta = 2.0; \lambda = 1.0; \kappa = 0.001 \)
- d) \( \delta = 0.001; \lambda = 0.001; \kappa = 1.0 \)
- e) \( \delta = 0.001; \lambda = 1.0; \kappa = 1.0 \)
- f) \( \delta = 1.0; \lambda = 1.0; \kappa = 1.0 \)

Symbol: 
- \( \Delta \) discrete correlation \( R_{\text{discrete}} \)
- \( \triangledown \) LR test
- + Score test

y axis: \( \Pi(\psi) \); x axis: \( \psi \)
Figure 25

Discrete power studies: \( n_m = 400, n_f = 400 \)

Power curves for a one-sided test of \( H_0: \lambda = 0 \) against \( H_1: \lambda > 0 \), at \( \alpha = 0.05 \) level for 2 distance trend tests.

\( \Delta \) : distance score test
\( \nabla \) : stone's test

a) \( \delta = 0.001, \ k = 0.001, \ \gamma = 0.001, \ \mu = 1 \)
b) \( \delta = 0.1, \ k = 0.001, \ \gamma = 0.001, \ \mu = 1 \)
c) \( \delta = 0.5, \ k = 0.001, \ \gamma = 0.001, \ \mu = 1 \)

\( y \) axis: \( \Pi(\lambda) \); \( x \) axis: \( \lambda \)
The effect of increased $\lambda$ is to lower the power of most tests, as in the continuous case, while high $\kappa$ values lead to 100% rejection at the lowest $\psi$ values. Large values of $\delta$ appear to lead to high rejection rates even under $H_0$. The $R_{x\theta}^2$ test shows very high rejection rates in most cases, whereas the LR and score tests show similar profiles. The only exception to this is the parameter setting $(\delta=1; \lambda=1; \kappa=1)$, where the $R_{x\theta}^2$ and score test yield 100% rejection under $H_0$ while the LR test is close to its nominal $H_0$ level. However, both the score and LR tests achieve a level close to nominal, in most other cases.

A small comparison of the distance score test (eq 85) and Stone's (1988) test was made and the results are shown in Figure 25. We examined the effect of variation in peakedness (non-monotonicity) on either test. Stone's test yields a high rejection rate under $H_0$ when $\delta = 0.0001$. As $\delta$ is increased the score test yields higher rates until 100% rejection occurs at $\delta = 0.5$. As the $\delta$ parameter is not estimated under $H_0$, such an effect could be quite significant in the analysis of radial trend.
CHAPTER 9

9 Analysis of Data Sets

In this section, we analyse data sets A-G using the relevant models developed in previous sections. For the range of continuous and discrete models fitted on GLIM, we have used the following covariables: R (radial distance); LR (log(R)); C (cos(φ); S(sin(φ)); RC (R*cos(φ)); RS (R*sin(φ)); LRC (LR*cos(φ)); LRS (LR*sin(φ)). R and LR represent exponential trend and peakedness respectively. C and S yield the angular concentration and mean while RC and RS signify angular-linear interaction. It can be shown that, to fit the continuous δ-dependence model with a log link on GLIM then LRC and LRS terms must be used. Hence the inclusion of significant LRC and LRS terms implies that peakedness has angular dependence in the data. We will first of all consider the Epidemiological Examples in data sets A, B and C.

9.1 Data Sets A, B and C

We will consider data set A separately from B and C as A represents a set of point locations while B and C are regionalised count data sets.

9.1.1 Respiratory Cancer Deaths (Data set A)

This example consists of respiratory cancer death certificate address locations in a small town in West Central Scotland (Armadale). The data consist of all deaths from ICD code 162 in the period 1968-1974. A number of workers have studied the unusually high levels of respiratory cancer found during the period and subsequently* (Lloyd, 1982; Lloyd et al. (1985)). A hypothesised cause of the increased incidence of such cancer was the presence of a steel foundry in the town and there is some visual evidence of a radial and angular concentration of cases (see Figure 1 a)) (Figure 1 a) depicts the foundry location as the central point.)

* SMR of ≈ 150 for each year between 1968-1974
An initial examination of this data by Lloyd (1982) examined the marginal radial distribution of the events only. While this shows a marked distance-decay, it does not account for angular variation or radial-angular interaction. Hence the radial effect could be a spiral effect which may not be consistent with the model. We have also produced Gaussian kernel estimates of the marginal radial and the marginal angular distribution (Figure 2 a), 2 b). This does suggest a strong radial effect with a marked distance peak. In addition, the marked angular peak at 250-260° mirrors the visual conclusion.

At this point, we will not consider the effect of population on the probability model. Instead, as an exploratory tool, we fit a HEPP model and examine ML estimates and tests for radial, angular and interaction effects. In cases where no information on population structure is available, this would be an appropriate form of model. We examine a composite intensity model using a truncated Weibull distance effect (eq (11)) and $M(\mu_0, \kappa+\psi r)$ model for the angular effect (eq (12)). We have compared this model with a variety of alternative forms. We have fitted a simple $M(\mu_0, \kappa)$ with the Weibull model, a full additive model (eq 117; 72) with 2 Weibull distance and 3 von Mises parameters, and a 5 parameter $\delta$-dependence model. As these models have been fitted by direct log-likelihood maximisation it has not been possible to evaluate all combinations of parameters for 5 parameter models. Table 6 gives details of the above model fits. However, it appears that amongst the multiplicative models of low dimensionality (1,2,3), the interaction von-Mises (3) yields the minimal AIC and highest log-likelihood. Comparison of 5 parameter models suggests that the Weibull-von Mises (4) is the best model, although model (3) yields a much lower $\xi_{\text{max}}$. The A deviance between model (2) and (3) is 185.88 on 1 degree of freedom. It is possible that different parameterisations of models (5) or (6) may yield higher $\xi_{\text{max}}$ values. We have not investigated this further. It is unlikely that the inclusion of $\delta$, in the 2 parameter model ($\lambda$, $\delta$ model (1)) would alter the $\xi_{\text{max}}$ value from -129.80 to above +18.72 (model (3)). The addition of the $\delta_1$ parameter has only increased the log
likelihood by 73.8 between model (1) + (2) and model (6). The additive model shows the lowest log likelihood for any 5 parameter model, and we have not considered this model further.

| Table 6 |

ML Results for Armadale (Set A)

(Standard errors are given in brackets)

<table>
<thead>
<tr>
<th>(1) W(λ, δ)</th>
<th>λ</th>
<th>δ</th>
<th>κ</th>
<th>ψ</th>
<th>μ₀</th>
<th>λ_max</th>
<th>AIC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.392</td>
<td>1.983</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-129.80</td>
<td>265.7</td>
</tr>
<tr>
<td>(0.008)</td>
<td>(0.024)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(2) M(µ₀, κ)</td>
<td>-</td>
<td>-</td>
<td>1.307</td>
<td>-</td>
<td>3.171</td>
<td>-74.22</td>
<td>146.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(0.071)</td>
<td></td>
<td>(0.029)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(3) M(µ₀, κ+ψr)</td>
<td>-</td>
<td>-</td>
<td>0.002</td>
<td>1.306</td>
<td>3.024</td>
<td>+18.72</td>
<td>34.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(0.58)</td>
<td>(0.686)</td>
<td>(0.159)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(4) W(λ, δ)</td>
<td>0.578</td>
<td>1.636</td>
<td>0.576</td>
<td>0.613</td>
<td>3.101</td>
<td>-111.14</td>
<td>234.3</td>
</tr>
<tr>
<td>M(µ₀, κ+ψr)</td>
<td></td>
<td></td>
<td>(0.083)</td>
<td>(0.580)</td>
<td>(1.070)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.214)</td>
<td></td>
<td></td>
<td>(0.220)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(5) full 5 parm : additive</td>
<td>1.011</td>
<td>1.118</td>
<td>0.001</td>
<td>0.182</td>
<td>1.348</td>
<td>-144.4</td>
<td>300.8</td>
</tr>
<tr>
<td></td>
<td>(0.440)</td>
<td>(0.360)</td>
<td>(1.860)</td>
<td>(3.830)</td>
<td>(0.050)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(6) full 5 parm : δ dependence</td>
<td>0.435</td>
<td>0.930</td>
<td>0.728</td>
<td>0.225</td>
<td>1.684</td>
<td>-130.3</td>
<td>272.6</td>
</tr>
<tr>
<td></td>
<td>(0.065)</td>
<td>(0.456)</td>
<td>(0.565)</td>
<td>(0.061)</td>
<td>(0.176)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

These results tend to suggest a pattern characterised by slight angular-radial interaction rather than concentration. This may be explained by the fact that point events appear to cluster in the south west of the foundry in close proximity, but at greater distances from the 'centre' the modal angle changes and fewer events are found in this south-west direction. This feature is observed on the display of the angular kernel density.
estimate (Fig 2b)) and on the point map (Fig 1a)). Table 1 (Sect. 1.3.4) displays the results of individual tests of radial and angular uniformity. Both tests for radial distance and peakedness are significant as well as conventional tests for angular uniformity and interaction.

However, based on the likelihood analysis above, it appears that the ('pure' spatial) pattern is best described by a 3 parameter interaction model.

The analysis can be taken beyond the exploratory stage by consideration of the additional information available in this example. First, the age × sex structure of the population is known at the level of enumeration districts. Hence regional covariates are available (and would usually be available in such studies). Population density or other demographic variables should directly affect the probability of an event in a particular area. At one extreme, no population means no possible event, while with a large susceptible population, a large number of events should occur. Hence any spatial effect should be observed, above that expected in the local population. In fact, it is standard practice in Epidemiological analysis of regional counts to compare counts with expected counts, often by use of SMRS or other standard measures.

In the present case, two levels of data aggregation are readily available, and except for the regionalisation of point events, which loses spatial information, there are two basic methods for incorporation of regional covariate information. First, the regional covariates can be interpolated to the point event locations and thence a standard HEPP model can be applied albeit with covariates. A model such as eq (26)

\[ \lambda^*(r,\phi) = \lambda(r,\phi) \, h(z) \]

i.e.

\[ \lambda^*(r,\phi) = \lambda(r,\phi) \, h(z) \]

could be employed with, for example, the population, population density or SMR as interpolated covariate. If the SMR is used, then it could be regarded as an offset in a GLIM model fit. We have used log (expected deaths) interpolated to data points in GLIM fits.
using T-weights in the Berman-Rolf-Turner method. The T-weights were T-del weights as we used the convex hull of the circular window for construction (see Chapter 7).

The interpolation of these covariates could be performed as in eq (27) or eq (28) using kernel or Spline methods. For continuous data it is also possible to employ Universal Kriging for interpolation. Kriging is formally equivalent to MAP estimation for Gaussian Processes (Warnes, 1987, App III) and is also related to Spline smoothing and penalised likelihood methods (see Watson (1984), and Lancaster and Salkaukas (1986, 11.6)). The equivalence of kernel methods, splines and moving average smoothers has also been demonstrated, in the time-domain, by Diggle (1990a, 2.2).

Region population counts or population densities are not sampled point events but types of average. Hence, interpolation to a point within a region is not the same process as interpolation between sample points: We have assumed, given the aggregate level of such data, that a simple kernel smoother such as (1.2) would be adequate. This allows the automatic assessment of smoothing by cross validation. We have used likelihood cross validation (Silverman, 1986, 87-88).

Second, it is possible to consider a model which includes a background point process as case control for respiratory cancer. In the original study of the Armadale 'epidemic', Lloyd (1982) examined the distribution of a 'control' disease thought to affect a similar population structure but which is unrelated to the environmental cause in question. In that case, Myocardial disease (ICD 410-414) was used for the period 1968-1974. Death certificate addresses for such heart disease were mapped for the 74 cases found in that period. Figure 26 a) and b) depicts the distribution of the lung cancer and heart disease cases. The justification for the use of heart disease as a 'control' for respiratory cancer was that both tend to affect older age groups and males predominantly. It was also suggested that such a control disease should be unrelated to a smelter location. This raises the problem of whether a spatial effect could be apparent in lung cancer data because people
Armadale

a) point map:

i) respiratory cancer

ii) heart disease
Armada b) respiratory cancer expected death surface (h = 11.261)
who work in a smelter also live near it. In addition, heart disease could be related to occupational conditions which might reflect worker’s place of residence rather than a general population effect.

For the Armadale case, it appears that no more suitable control disease could be found, (Lloyd, personal communication, 1988). We have therefore used Myocardial disease as our control point process.

Visual inspection of Figure 26 suggests that most Myocardial cases are found in the western and north-western areas of Armadale - a distribution quite unlike that of lung cancer which has a south-western angular peak. The radial distribution of cases were also noted to differ, in that a marked decrease in lung cancer cases occurs with distance from the foundry. In Lloyd’s original analysis, the population structure and/or a control disease were not incorporated directly into the analysis of lung cancer. Diggle (1989, 1990b) has suggested interpolation of such case-control point events, using density estimation, to the point event locations of the disease under study. He also suggests that by such covariate interpolation, there will be no need to consider the demographic structure of the population (Diggle (1989), personal communication).

Here, we have considered both case-control interpolation of heart disease, and log (ED) interpolation as this does not ignore any possible extra information important only to the disease considered. In addition, we have also directly modelled the different aggregation levels of the data via the log-likelihood (eq 106)

\[ \phi_{FA} = \sum_{i=1}^{n} \ln \lambda^*(r_i, \phi_i) - n \ln \sum_{\phi = 1}^{P} \lambda^*(r, \phi) \]

In its exact form this likelihood cannot be handled by GLIM, although approximations could be implemented. For example, the continuous model (eq 14,15) could be used but a regional ‘factor’ with P levels could be fitted. Alternatively, the regional covariate could be assumed to be piecewise constant and hence assigned directly to the point events.
However, this latter option is essentially interpolation and is not different from the first method.

The following results have been derived from the three methods mentioned above.

**Table 7a**

<table>
<thead>
<tr>
<th>Armadale: Case-control and other model results</th>
<th>* Case Control</th>
<th>* expected deaths (ED)</th>
<th>Mixed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deviance (df)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>null : 92 (78)</td>
<td>null : 77 (78)</td>
<td>null : 77 (78)</td>
<td>-295.5 $\chi_{\text{max}}^2$</td>
</tr>
<tr>
<td>model : 73 (76)</td>
<td>model : 66.5 (75)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AIC</td>
<td>583.6</td>
<td>582.2</td>
<td>601.0</td>
</tr>
<tr>
<td>l</td>
<td>2.78 (0.207)</td>
<td>3.064 (0.405)</td>
<td>0.659 (0.412) $\hat{\lambda}$</td>
</tr>
<tr>
<td>R</td>
<td>-</td>
<td>0.034 (0.018)</td>
<td>2.503 (1.910) $\hat{\delta}$</td>
</tr>
<tr>
<td>C</td>
<td>-0.935 (0.275)</td>
<td>-</td>
<td>0.446 (0.312) $\hat{\kappa}$</td>
</tr>
<tr>
<td>S</td>
<td>-0.331 (0.217)</td>
<td>-</td>
<td>0.770 (0.370) $\hat{\psi}$</td>
</tr>
<tr>
<td>RC</td>
<td>-</td>
<td>-0.001 (0.014)</td>
<td>3.33 (0.509) $\hat{\mu}_0$</td>
</tr>
<tr>
<td>RS</td>
<td>-</td>
<td>-0.02 (0.008)</td>
<td></td>
</tr>
</tbody>
</table>

* estimated on GLIM using 30 convex hull Delaunay dummies

Table 7a displays the 'optimal' model found using GLIM for case control and ED models. The mixed model is reported as a direct maximisation for a 5 parameter model. Initially we used direct maximisation for the case control and ED models and this showed higher $\chi_{\text{max}}^2$ values for these 5 parameter models than the mixed models.
Table 7b

Kernel Score Tests for Armadale

Tests given here are for radial (R), angular (TH) and interaction (I) effects, as described in Section 3.1.1.

The background intensity process is a) ischaemic heart disease, and b) expected lung cancer deaths

<table>
<thead>
<tr>
<th>test</th>
<th>result</th>
<th>smoothing</th>
</tr>
</thead>
<tbody>
<tr>
<td>a) R</td>
<td>-20.617 *</td>
<td>( h_{cvopt} = 2.712 )</td>
</tr>
<tr>
<td>TH</td>
<td>0.721</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>-0.389</td>
<td>( \hat{k} = 0.992 ) ( \hat{\mu}_0 = 200' )</td>
</tr>
<tr>
<td>b) R</td>
<td>-23.106 *</td>
<td>( h_{cvopt} = 11.26 )</td>
</tr>
<tr>
<td>TH</td>
<td>28.669 *</td>
<td>( \hat{k} = 0.418 ) ( \hat{\mu}_0 = 146.6' )</td>
</tr>
<tr>
<td>I</td>
<td>19.325 *</td>
<td></td>
</tr>
</tbody>
</table>

* denotes significance at 5% in a one-tail monte carlo test

Hence we confine our comments to a comparison of the Case-control and ED models on GLIM. Each model was fitted using the likelihood cross-validation h estimate. We have found that results can vary considerably depending on the choice of h, as noted by Diggle (1990a).
In general, the model which used the respiratory cancer expected deaths interpolated to the point locations yielded the lowest deviance under the null model i.e. 77 on 78 degrees of freedom.

We have fitted a variety of variables in different combinations using both models. The optimal combination using the ED model has terms $1 + R + RC + RS$ i.e. a grand mean, radial distance ($R$), radial-angular correlation ($r \cos \phi : RC$; $r \sin \phi : RS$). This model produced the lowest deviance for models where each component yield a 'significant' change in deviance ($\Delta \text{dev}$). We have examined models without the grand mean but these yield increases in deviances.

It is interesting to note that under this model the mean angle is estimated as 1.537 (88.06°) which appears to reflect the effect shown by the covariate extraction method i.e. the residual peak of cases occurs to the north of the foundry. This also supports the results of wind tunnel experiments by Lloyd (1982), which demonstrated a dominant deposition of particulate material in the north of the study area. We have examined the deviance residuals from this fit. Figure 27 displays the results. Plots against fitted values and against $r$ and angle (Figs 27 a,b,c) show a reasonable spread around 0. There are four points which have high positive residuals. This suggests areas smaller than expected and could signal clustering of points. It would be expected that there will be correlation between adjacent residuals, hence if a 'cluster' of high positive residuals occurs then a number of high negative residuals may also be expected. The residual surface (Fig 27 e,f) shows a simple peak of low positive value in the south-west of the foundry. This area has low expected deaths.

The autocorrelation coefficient ($z_1 = 6.389$ under randomisation) suggests clustering of residuals but the surface shows a peak at the large positive residuals only. Hence, it appears that the fitted model accounts for the variation quite well and has allowed the isolation of an anomalous cluster of points.
Figure 27

a) Expected deaths

<table>
<thead>
<tr>
<th>Standard Deviation</th>
<th>Expected Deaths</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.300</td>
<td>2.34</td>
</tr>
<tr>
<td>1.750</td>
<td>1.87</td>
</tr>
<tr>
<td>1.250</td>
<td>1.45</td>
</tr>
<tr>
<td>1.000</td>
<td>1.03</td>
</tr>
<tr>
<td>0.750</td>
<td>0.61</td>
</tr>
<tr>
<td>0.350</td>
<td>-0.25</td>
</tr>
<tr>
<td>-0.500</td>
<td>-0.67</td>
</tr>
<tr>
<td>-1.000</td>
<td>-1.09</td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.52</td>
</tr>
<tr>
<td>-2.000</td>
<td>-1.94</td>
</tr>
</tbody>
</table>

b) Heart disease

<table>
<thead>
<tr>
<th>Standard Deviation</th>
<th>Heart Disease</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.300</td>
<td>2.30</td>
</tr>
<tr>
<td>1.750</td>
<td>1.91</td>
</tr>
<tr>
<td>1.250</td>
<td>1.48</td>
</tr>
<tr>
<td>1.000</td>
<td>1.04</td>
</tr>
<tr>
<td>0.750</td>
<td>0.61</td>
</tr>
<tr>
<td>0.350</td>
<td>0.18</td>
</tr>
<tr>
<td>-0.500</td>
<td>-0.25</td>
</tr>
<tr>
<td>-1.000</td>
<td>-0.67</td>
</tr>
<tr>
<td>-1.500</td>
<td>-1.09</td>
</tr>
<tr>
<td>-2.000</td>
<td>-1.52</td>
</tr>
</tbody>
</table>

a) fitted value
b) radial distance
c) angle
d) normal quantile plot
e) surface view of standardised residuals (z axis) in X-Y coordinates (h = 5.186)

Figure 27 Expected Deaths

f) contour map of e)
e) surface view of standardised residuals (z axis) in X-Y coordinates (h = 5.186)

f) contour map of e)
9.1.2 **Bronchitis and Pneumonia Deaths (Data sets B, C)**

These two data sets consist of counts of events within enumeration districts. Both data sets concern Bronchitis (ICD 490-493) and Pneumonia (ICD 480-486) mortality in the areas of Bonnybridge, Central Scotland (set B) and Methil, East Fife (set C).

These data sets were chosen with a view to examination of the spatial relation between a putative source of environmental pollution and diseases which have a short latency period and can be related to an environmental cause. The main diseases which show short latency could be skin diseases, eye disease and respiratory disorders.

Due to the problems related to the use of morbidity records (e.g. Hospital admissions) we have used mortality records. Mortality is often used as a surrogate morbidity measure. However, the effects of environmental pollution may be to increase morbidity at a younger-age, in the short term. Extreme pollution could increase the mortality of low risk groups as well as high risk groups. Hence, small effects may appear only in high risk groups. Gross pollution effects are likely to produce increases also in low risk groups. In the case of Bronchitis and Pneumonia, the high risk groups are very young children and those over 65. Only under severe conditions would the adult working population be expected to show mortality effects.

Other possible causes of raised incidences of such diseases are housing/deprivation status e.g. damp housing conditions, or mode of employment. For example, residents who also work in or at an establishment which is a putative pollution source may have a higher risk due to occupational status.

We have used mortality for respiratory diseases as these are likely to have higher *morbidity* numbers in each ed than for skin or eye disorders, and hence any raised incidence should be reflected in mortality.
A high standardised mortality ratio (SMR) may or may not occur for the whole window, but there could still be an identifiable excess of cases associated with the putative source. In our analysis we condition on the number of events in the window, and hence a raised incidence does not necessarily affect the small scale spatial pattern within the window. Nonetheless, the fact that there is a high SMR suggests adverse health conditions within this area.

9.1.1.2 Bonnybridge (Data Set B)

This data set consists of 182 eds within a 5 km radius circle centred on a putative pollution source. This area was chosen as concern has been expressed as to an environmental health risk related to a chemical reprocessing plant run by Rechem International Ltd. A Government enquiry (Lenihan, 1985) has reviewed the general morbidity in this area. The basis of analysis for their enquiry was a 5 km circle not centred on the reprocessing plant. In addition, the 200 eds within their circular window were not examined for spatial distribution, but were amalgamated and the whole area compared to other areas of 'similar' population structure. Hence, only a gross exceedence of morbidity in the area could be detected. Lenihan (op. cit.) found a 'significant' SMR for general respiratory disease in the Bonnybridge area, although he did not examine the distribution of each disease within their area.

We have chosen a circular window as any radial effects are given equal weight regardless of direction. We have chosen a 5 km radius as this allows a relatively large number of eds to be included while not including effects which are beyond the dispersion range for an air pollution effect. In addition, a number of large industrial areas would also be included if a larger radius were used and hence competing risks may be included.
We have included all eds which were intersected by the window boundary. Data were obtained on 182 eds for the period 1980-1982. The data consisted of deaths from Bronchitis and Pneumonia for each of 2 sex classes and 9 age classes. Demographic structure was also made available in these same age × sex classes. Fig 28 depicts the ed map of the area. Due to the large variation in size and large number of eds we have not labelled the districts. Appendix XI contains a listing of the data used.

As the data form counts within arbitrary regions, with only regional covariate information, we have employed likelihood equation (69). We have included expected numbers of deaths (ED) calculated from national (Scotland) and regional (window) rates for both Bronchitis and Pneumonia. Due to the small numbers found in each age × sex class in each ed we have only considered total deaths in each sex class or total deaths regardless of sex.

We have assumed a constant regional rate in (69) in our initial analysis. Hence, we use GLIM with Poisson error and log (Eₐ) as offset. Table 8 displays the relevant results for both diseases.
Bonnybridge (Denny): a) enumeration district map
(inset: location map)
Figure 28

&)

dead count surface

(h = 7.773) (Bronchitis)

(h = 7.773) (Pneumonia)
Figure 28

\[(h = 3) \quad \text{SMR (Bronchitis)}\]

\[(h = 3) \quad \text{SMR (Pneumonia)}\]
Table 8

GLIM Model Fits for Total Deaths

*Bronchitis and Pneumonia: Bonnybridge*

a) **Ordinary fit**

<table>
<thead>
<tr>
<th>model</th>
<th>1</th>
<th>1+R+LR+C+S+RC+RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bronchitis : Dev</td>
<td>129.3</td>
<td>118.6</td>
</tr>
<tr>
<td>df</td>
<td>181</td>
<td>175</td>
</tr>
<tr>
<td>Pneumonia : Dev</td>
<td>223.0</td>
<td>181.0</td>
</tr>
<tr>
<td>df</td>
<td>181</td>
<td>175</td>
</tr>
</tbody>
</table>

b) **log (total population) offset**

<table>
<thead>
<tr>
<th>model</th>
<th>1</th>
<th>1+R+LR+C+S+RC+RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bronchitis : Dev</td>
<td>131.5</td>
<td>118.4</td>
</tr>
<tr>
<td>df</td>
<td>181</td>
<td>175</td>
</tr>
<tr>
<td>Pneumonia : Dev</td>
<td>216.0</td>
<td>181.3</td>
</tr>
<tr>
<td>df</td>
<td>181</td>
<td>175</td>
</tr>
</tbody>
</table>

c) **log (total expected deaths) offset**

<table>
<thead>
<tr>
<th>model</th>
<th>1</th>
<th>1+R+LR+C+S+RC+RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bronchitis : Dev</td>
<td>117.0</td>
<td>104.0</td>
</tr>
<tr>
<td>df</td>
<td>181</td>
<td>175</td>
</tr>
<tr>
<td>Pneumonia : Dev</td>
<td>179.0</td>
<td>147.5</td>
</tr>
<tr>
<td>df</td>
<td>181</td>
<td>175</td>
</tr>
</tbody>
</table>
Table 8 displays distinct differences between Bronchitis and Pneumonia for this area. We have examined both models with and without offsets, and with and without a grand mean. The expected deaths were calculated based on age × sex stratification in each ed. In general similar results were found for sex-specific model fits so we confine our discussion here to models for total deaths. First, it is apparent that the lowest deviance whether under the null model or 7 parameter model is found with the log (expected deaths) offset. The main difference between Bronchitis and Pneumonia appears to be that the Poisson model yields a closer fit to Bronchitis under the null or with covariates. That is the deviance for Bronchitis is lower than that for Pneumonia. Overall, the intercept term appears to reduce the deviance to close to the degrees of freedom. If the model were correct then the deviance should approximately equal the df, at least for large df. This assumes negligible sparseness in the data, but can still be used as a general guide to model fit.

We now only consider models for total deaths under the log (total expected deaths) offset. Below we present the 'best' models for Bronchitis and Pneumonia from among the seven r-φ covariates available, i.e. we only include terms which yield a significant reduction in deviance during variable selection.

<table>
<thead>
<tr>
<th>terms</th>
<th>est</th>
<th>SE*</th>
<th>dev</th>
<th>terms</th>
<th>est</th>
<th>SE*</th>
<th>dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>-0.01</td>
<td>0.169</td>
<td>(179)</td>
<td>R</td>
<td>-0.136</td>
<td>0.034</td>
<td>(179)</td>
</tr>
<tr>
<td>S</td>
<td>-0.95</td>
<td>0.275</td>
<td>AIC:</td>
<td>LR</td>
<td>4.764</td>
<td>1.211</td>
<td>AIC:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-213.6</td>
<td></td>
<td></td>
<td></td>
<td>-259.4</td>
</tr>
</tbody>
</table>

* standard errors adjusted for mean deviance.
The above results suggest that Bronchitis has a predominantly angular relation with the putative pollution source. On the other hand, Pneumonia yields a predominantly radial effect, as radial distance (R) and the log of radial distance (LR) are significant. However, the model yields a higher deviance for Pneumonia than Bronchitis.

Table 9 displays the score test results for radial/angular interaction discussed earlier.

<table>
<thead>
<tr>
<th></th>
<th>Bronchitis</th>
<th>Pneumonia</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>1.763</td>
<td>12.321 *</td>
</tr>
<tr>
<td>TH</td>
<td>0.084</td>
<td>16.154 *</td>
</tr>
<tr>
<td>WS</td>
<td>1.288</td>
<td>1.577</td>
</tr>
</tbody>
</table>

All results are for one-tail tests; all tests are ~ N(0,1), although the above results are based on Monte Carlo testing. * denotes significance at \( \alpha = 0.05 \) level (one-tail).

Note that Bronchitis shows little spatial association, except that the sin \( \phi(S) \) term is significant which suggests an N-S axis of concentration. Pneumonia shows high significance for radial and angular concentration tests. Neither disease displays any significant interaction effects. Pneumonia shows a peaked-distance effect as well as exponential decline.
Table 10

ML estimates of $\kappa, \psi, \mu_0$: Bonnybridge

<table>
<thead>
<tr>
<th></th>
<th>Bronchitis</th>
<th>Pneumonia</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\kappa$</td>
<td>0.1448 (0.024)</td>
<td>0.799 (0.157)</td>
</tr>
<tr>
<td>$\psi$</td>
<td>$2.86 \times 10^{-5}$ (1.6 $\times 10^{-5}$)</td>
<td>$1.125 \times 10^{-7}$ (1.2 $\times 10^{-6}$)</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>4.8665 (0.324)</td>
<td>1.098 (0.105)</td>
</tr>
</tbody>
</table>

Standard errors given in brackets

We have also derived ML estimates for $\kappa, \phi, \mu_0$ for the interaction model for Bronchitis and Pneumonia. Table 10 displays these results. Note that the association of Bronchitis with a sin effect ($s$) is related to the $\mu_0$ estimate given in Table 10. The mean angle for Bronchitis is approximately SSE.

We have also examined the model fits obtained when $n_\xi$ instead of $E_\xi$ is used as an offset. In general, the results follow the pattern found in the GLIM results, and similar patterns of 'significance' occur in parameter estimates.

Residual analysis tends to show differences between Bronchitis and Pneumonia. The plot of Anscombe residuals fitted values for Pneumonia tend to show an even spread around 0, although a few large positive outliers occur. The patterned nature of the fitted value plots reflects the fact that fitted values are fitted to collections of integer counts. In addition sparseness in count data can lead to an excess of small negative residuals occurring. The normal plots show approximate straight lines. Bronchitis however shows more patterned residual plots, while the normal plot shows a concentration of negative
Anscome residual displays: Bonnybridge

- a) fitted value
- b) radial distance (r)
- c) angle (θ)
- d) normal quantile plot
Figure 29

Bronchitis

e) surface view of standardised residuals (z axis) in X-Y coordinates (h = 7.772)

f) contour map of e)
200
Figure 29  Pneumonia

e) surface view of standardised residuals (z axis) in X-Y coordinates (h = 7.772)

f) contour map of e)
residuals between -0.25 and -1.0. Hence, there appears to be few observations close to the model fitted values. Figures 29 a-b) display typical plots of both diseases.

The residual surfaces (Fig 29 e,f) show isolated peaks/troughs of residuals. In the Bronchitis case, the areas immediately west and east of the putative source show clusters of low negative residuals. These levels of residual are also related to influential points. The high leverages are mainly concentrated at values of residual between 0 and -1.0. However, the autocorrelation is -0.0132, which using an MC test has probability 0.19 and is not significant. The pneumonia surface shows a single trough of negative residuals to the east of the putative source. The autocorrelation is -0.0146 and has MC probability of 0.34 and is not significant.

9.1.2.2 Buckhaven-Methil (Data Set C)

This area was examined as it is a small industrial community, similar in population size to the Bonnybridge community. It has a 'central' industrial area with a number of possible environmental pollution sources. There is no current environmental health concern related to this industrial activity, as far as the author is aware. The area is different from Bonnybridge in that it has an estuarine location. As a consequence of the above, this site can be considered as a type of 'control' area for the Bonnybridge data. Note that due to the smaller physical size of this community than the general Denny-Bonnybridge area we have examined a 3 km circular window, centred on a steel foundry site (ATO1) (see Figure 30a)). We have included all 62 eds which are wholly within or intersect the circle boundary. The eds included lie in all directional sections around the central point, although a general NE-SW coastal orientation is apparent for the settlement as a whole. The effect of such an orientation could be to produce preferred directional results if population (n_E) or expected deaths (E_E) were not included. We have analysed radial and angular effects using E_E in all cases.
Figure 30 a)

Buckhaven-Methil a) enumeration district map
Figure 30 b) BUCKHAVEN-METHIL DEATH COUNT surface

SMR surface
(h = 3)
bronchitis

BUCKHAVEN-METHIL DEATH COUNT surface

Bronchitis (h = 3.115)

(h = 3)
Pneumonia

Pneumonia (h = 3.115)
For the data set some additional covariate information is available, in the form of employment status of ed populations. As in the Bonnybridge example we have examined Bronchitis and Pneumonia mortality for years 1980-1982 for the 62 eds in the region. Unlike the Bonnybridge case there is not a high SMR for these diseases for the period in question.

We have applied a model to this data which is similar to the Bonnybridge example (likelihood (69)). The results of GLIM model fitting are given in Table 11.
Table 11
GLIM Model Fits for Total Deaths

Bronchitis and Pneumonia: Methil

a) Ordinary fit

<table>
<thead>
<tr>
<th>model</th>
<th>1</th>
<th>1+R+LR+C+S+RC+RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bronchitis : Dev</td>
<td>67.1</td>
<td>54.5</td>
</tr>
<tr>
<td>df</td>
<td>61</td>
<td>55</td>
</tr>
<tr>
<td>Pneumonia : Dev</td>
<td>119.6</td>
<td>98.3</td>
</tr>
<tr>
<td>df</td>
<td>61</td>
<td>55</td>
</tr>
</tbody>
</table>

b) log (total population) offset

<table>
<thead>
<tr>
<th>model</th>
<th>1</th>
<th>1+R+LR+C+S+RC+RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bronchitis : Dev</td>
<td>67.1</td>
<td>56.26</td>
</tr>
<tr>
<td>df</td>
<td>61</td>
<td>55</td>
</tr>
<tr>
<td>Pneumonia : Dev</td>
<td>122.9</td>
<td>95.44</td>
</tr>
<tr>
<td>df</td>
<td>61</td>
<td>55</td>
</tr>
</tbody>
</table>

c) log (total expected deaths) offset

<table>
<thead>
<tr>
<th>model</th>
<th>1</th>
<th>1+R+LR+C+S+RC+RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bronchitis : Dev</td>
<td>63.4</td>
<td>50.14</td>
</tr>
<tr>
<td>df</td>
<td>61</td>
<td>55</td>
</tr>
<tr>
<td>Pneumonia : Dev</td>
<td>105.08</td>
<td>92.62</td>
</tr>
<tr>
<td>df</td>
<td>61</td>
<td>55</td>
</tr>
</tbody>
</table>
In general, the grand mean null fit appears to be a reasonable model for Bronchitis, and this is true with and without offsets. For Bronchitis, the addition of spatial variables shows a significant effect for all models, although the final deviance remains close to the df value in all cases. Pneumonia, on the other hand, does not admit the null model as a feasible model: in most cases the deviance is almost double the degrees of freedom. Spatial variables also produce reductions in deviance but not close to the df value. In all cases the log (total expected deaths) offset yields the lowest deviance under null or seven parameter models. We will only consider this model in the following discussion.

The following results are for the model with variables which yield significant reductions in deviance during variable selection.

<table>
<thead>
<tr>
<th>Bronchitis</th>
<th>Pneumonia</th>
</tr>
</thead>
<tbody>
<tr>
<td>dev : 52.99</td>
<td>dev : 93.53</td>
</tr>
<tr>
<td>df : 59</td>
<td>def : 59</td>
</tr>
<tr>
<td>aic : -99.97</td>
<td>aic : -181.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>term</th>
<th>est</th>
<th>se*</th>
<th>term</th>
<th>est</th>
<th>se*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.565</td>
<td>0.165</td>
<td>1</td>
<td>-0.764</td>
<td>0.332</td>
</tr>
<tr>
<td>RC</td>
<td>-0.052</td>
<td>0.021</td>
<td>RC</td>
<td>0.0344</td>
<td>0.022</td>
</tr>
<tr>
<td>RS</td>
<td>-0.017</td>
<td>0.026</td>
<td>RS</td>
<td>0.0494</td>
<td>0.039</td>
</tr>
</tbody>
</table>

* S.e.s are adjusted for mean deviance

In general, both disease types show a significant angular-radial interaction effect only, no other r-th variates were found to add significantly. The pattern for Bronchitis shows a better fit (dev : 52.99, df : 59) than Pneumonia (dev : 93.53, df : 59). The overall orientation of the town of Methil, determined by its seaside location would support findings
of an angular concentration confined to a linear (radial) zone along the coast. There appears to be little pure radial or angular effects in either case. Hence, it may be suggested that the pure spatial model is 'feasible' for Bronchitis but other variables must be associated with the Pneumonia pattern. The similarity between the Bronchitis/Pneumonia cases for Methil and Bonnybridge is quite striking in that in both areas the spatial model appears to yield a much better model for Bronchitis than for Pneumonia. However, the lack of fit for Pneumonia is more marked in the Methil case. As the Methil example represents an area which has no 'perceived' environmental risk, it may be important that r-th variables do not explain the variation, whereas they appear to provide a better model, even in the Pneumonia example for Bonnybridge.

As RC and RS have opposite signs for Bronchitis and Pneumonia in Methil, it is to be supposed that the areas of angular concentration are in opposite directions. This is supported by the SMR surfaces (Fig 30b) for the area.

We have also analysed this data set using the score tests as in section 9.1.2.1. Table 12 displays the results for the R, TH, WS tests.

<table>
<thead>
<tr>
<th></th>
<th>Bronchitis</th>
<th>Pneumonia</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>2.754 *</td>
<td>3.161 *</td>
</tr>
<tr>
<td>TH</td>
<td>1.077</td>
<td>2.199 *</td>
</tr>
<tr>
<td>WS</td>
<td>2.419 *</td>
<td>3.822 *</td>
</tr>
</tbody>
</table>

All results are for one-tail tests; all tests are as $\sim N(0,1)$, although test significances are given for monte carlo tests at 5% level.
The test results show similar patterns for both diseases in that radial and interaction effects are significant. However, Bronchitis shows lesser effect than Pneumonia, while there is no significant angular effect. These results are supported by the ML estimation results in Table 13.

Table 13
ML estimates for $\kappa, \phi, \mu_0$: Buckhaven-Methil

<table>
<thead>
<tr>
<th></th>
<th>Bronchitis</th>
<th>Pneumonia</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\kappa$</td>
<td>$0.40 \times 10^{-5}$ ($1.0 \times 10^{-5}$)</td>
<td>$0.39 \times 10^{-5}$ ($1.2 \times 10^{-6}$)</td>
</tr>
<tr>
<td>$\psi$</td>
<td>0.666 (0.159)</td>
<td>0.944 (0.205)</td>
</tr>
<tr>
<td>$\mu_0$</td>
<td>3.452 (0.569)</td>
<td>1.300 (0.102)</td>
</tr>
</tbody>
</table>

Standard errors in brackets

In both cases, concentration ($\hat{\kappa}$) is negligible and there is a small interaction term ($\hat{\psi}$). The lower TH test values reflect this effect. Note that the mean angle for Bronchitis is approximately WSW, while that of Pneumonia is NNE approximately.

As in the Bonnybridge case we have examined the model fits obtained when $n_\varphi$ is offset instead of $E_\varphi$ and for separate sex classes. In general, these reflect the results presented here, and while parameter estimates vary, the general conclusions remain the same.

Residual analysis for this area (Fig 31 a-b) differs for Bronchitis and Pneumonia. Most Bronchitis residuals lie between ±1.5, with 2 higher residuals (2.34, 1.78). The patterned nature of the fitted value plot again reflects discrete counts with continuous fitted values. The normal plot shows an approximate straight line with the exception of a high concentration in the region of -1.0. This value also contains a high leverage point (> 0.3).
Anscombe Residual displays: Buckhaven-Methil

a) fitted value
b) radial distance
c) angle
d) normal quantile plot
Figure 31 e) Bronchitis

(e) surface view of standardised residuals (z axis) in X-Y coordinates (h = 3.023)

f) contour map of e)
surface view of standardised residuals (z axis) in X-Y coordinates (h = 11)

f) contour map of e)
In general, the spread of residuals around 0 appears consistent with model assumptions although the excess of negative residuals (~ -1.0) may suggest sparseness. The residual surface (Fig 31 e,f) shows a peak north-east of the centre (0,0) and 3 troughs aligned on a NE-SW axis.

The autocorrelation coefficient ($I = -0.0291$) achieves a probability of 0.93 in an MC test and is not significant.

Pneumonia displays a similar plot of residuals versus fitted values as Bronchitis, except there are more large +ve and -ve residuals: (2.158, 6.419, -2.262). The large +ve residual of 6.419 corresponds to an ed with 12 deaths. I have checked the validity of this observation and it was found to be correct. The normal plot shows a concentration between 0 and -1.0, and the large +ve residual is also evident (Fig 31 d), otherwise an approximate straight line is found. There is a single high leverage point (0.38) which corresponds with the cluster of points around -1.0. The large +ve residual (6.419) does not appear to be influential.

We have examined the fit of a variety of models with this large observation given zero weight (item 6).

The lowest deviance is still achieved by log (ED) as offset and the following table shows the 'best' model achieved:

<table>
<thead>
<tr>
<th></th>
<th>null fit</th>
<th>model : 1+C+S+RC+RS</th>
</tr>
</thead>
<tbody>
<tr>
<td>deviance</td>
<td>51.85</td>
<td>43.68</td>
</tr>
<tr>
<td>df</td>
<td>(60)</td>
<td>(56)</td>
</tr>
</tbody>
</table>

Hence, the exclusion of item 6 yields a deviance reduction from 92.62 (55) to 43.68 (56). The C and S terms are now also significant. The parameter estimates produced by this fit are:
Figure 32

Anscombe Residual displays: Buckhaven-Methil reduced data set (Pneumonia only)
e) surface view (h = 3.023)

f) surface contour map
The C and RS terms are now negative, while S and RC are positive.

We have examined residual surfaces (Fig 32 a,b) with and without item (6) used for model fitting. In either case the surfaces are smoothly varying except for the large positive residual at item 6. In fact the autocorrelation coefficient is reduced by exclusion of (6) from the fit, but in either case is not significant ((6) included: I = -0.0297; exact MC prob = 0.72; (6) excluded : I = -0.0161; exact MC prob = 0.30). The plots of anscombe residuals versus fitted values and normal quantiles (Figs 32 c,d) shows that the exclusion of item (6) reduces the variability of residuals (generally -1.6 < r < 2.0) but increases the residual for item (6) to 8.63. Hence the model appears to fit more closely the bulk of the data with item (6) excluded.

<table>
<thead>
<tr>
<th>term</th>
<th>estimate</th>
<th>se*</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>-0.98</td>
<td>0.256</td>
</tr>
<tr>
<td>C</td>
<td>-1.07</td>
<td>0.641</td>
</tr>
<tr>
<td>S</td>
<td>1.93</td>
<td>0.744</td>
</tr>
<tr>
<td>RC</td>
<td>0.14</td>
<td>0.051</td>
</tr>
<tr>
<td>RS</td>
<td>-0.18</td>
<td>0.076</td>
</tr>
</tbody>
</table>

* adjusted for mean deviance
Table 14

Summary of Models for Bonnybridge and Buckhaven-Methil

<table>
<thead>
<tr>
<th></th>
<th>Bronchitis</th>
<th></th>
<th>Pneumonia</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>dev df</td>
<td></td>
<td>dev df</td>
</tr>
<tr>
<td>null</td>
<td>117.0 181</td>
<td>null</td>
<td>179.0 181</td>
</tr>
<tr>
<td>Denny</td>
<td>optimal</td>
<td>109.8 179</td>
<td>optimal</td>
</tr>
<tr>
<td>l+C+S</td>
<td></td>
<td>1+R+LR</td>
<td></td>
</tr>
<tr>
<td>null</td>
<td>63.4 61</td>
<td>null</td>
<td>105.0 61</td>
</tr>
<tr>
<td>optimal</td>
<td>52.99 59</td>
<td>optimal</td>
<td>93.5 59</td>
</tr>
<tr>
<td>l+RC+RS</td>
<td></td>
<td>1+RC+RS</td>
<td></td>
</tr>
<tr>
<td>Methil</td>
<td>item 6 omitted from fit</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>optimal</td>
<td>43.68 56</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1+C+S+RC+RS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

all models have log (expected deaths) offset.

In general it is possible to compare the results of data set B and C to assess whether there is any spatial evidence of a link between the putative source and Bronchitis and Pneumonia mortality. Table 14 gives a summary of these results. For Bonnybridge and Buckhaven-Methil, Bronchitis is well described by the population structure, although some angular and interaction effects occur. The interaction effects may be due to the orientation of the settlement on the estuarine site in the Methil case. In both areas Pneumonia is less well characterised by the constant rate (grand mean) model (although the omission of item (6) produces the lowest null deviance). The radial effects are significant in both, while radial-
angular interaction is characteristic of Methil and pure angular effects are also significant there. This is reflected in the parameter estimates. Of these results, the significant radial effects for Pneumonia at Bonnybridge are of greatest import.

9.2 Data Sets D, E, F

These data sets represent ecological examples of point processes which may be modelled by particular classes discussed in previous sections. The intention in the present case is to fit an appropriate model, rather than test a hypothesis about spatial association with a fixed point.

9.2.1 Hebeloma Sporophores (Data Set D)

This example consists of locations of sheathing mycorrhizal fungi (Hebeloma Species) around a birch tree on an experimental 5m square plot. The realisations of Hebeloma distribution are for 1975 and 1978. The development of these patterns have been discussed by Last et al. (1984), Ford et al. (1980) and Mason et al. (1982). In addition, a previous statistical analysis has been carried out by Byth (1980). This previous analysis developed kernel density estimation methods for the polar coordinate case (Byth, 1982) and also modelled the point events as $\phi$-stationary processes (Byth, 1980). This approach consists of assuming that within certain fixed angular sectors the process is stationary in $\phi$ and the assessment of second-order properties related to these sectors (see, Byth (1980, 1981). It is possible to consider certain of these Hebeloma patterns as 'first-order' in that they display smooth variation over space (see, Fig 1 d)). Hence, rather than
use a complex second-order method we could employ a HEPP model which allows variation in $r$ and $\phi$.

We have considered a HEPP model (eq (6)) with composite intensity (eq (11), (12)). Hence, as in the exploratory phase of modelling the Armadale data example, we assume a homogeneous environment. In this case such homogeneity should be more valid, as there is no underlying population structure, although some unobserved environmental effect may exist.

We have used the T-del weight method to fit a variety of models on GLIM for 1975 and 1978: we have used 30 convex hull dummies. In this case we have specified a circular window of radius equal to half the plot size. No data points are found outside this window. Table 15 gives details of the GLIM model fits for these data.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>1975</th>
<th>1978</th>
</tr>
</thead>
<tbody>
<tr>
<td>dev</td>
<td>765</td>
<td>143.6</td>
</tr>
<tr>
<td>df</td>
<td>144</td>
<td>71</td>
</tr>
<tr>
<td>null</td>
<td>245</td>
<td>31.37</td>
</tr>
<tr>
<td>'best' model</td>
<td>136</td>
<td>67</td>
</tr>
<tr>
<td>1+R+LR+C+S</td>
<td>245</td>
<td>31.37</td>
</tr>
<tr>
<td>+RC+RS+</td>
<td>245</td>
<td>31.37</td>
</tr>
<tr>
<td>LRC+LRS</td>
<td>245</td>
<td>31.37</td>
</tr>
<tr>
<td>$\xi_{\text{sat}}$ : -531.2</td>
<td>$\xi_{\text{sat}}$ : 81.67</td>
<td></td>
</tr>
<tr>
<td>$\xi_{\text{max}}$ : -653.7</td>
<td>$\xi_{\text{max}}$ : 65.99</td>
<td></td>
</tr>
</tbody>
</table>

* SES adjusted for mean deviance
It is apparent from Table 15, that the null fit for 1978 is significantly better than for 1975. After assessment of a variety of polar coordinate models, the 1975 data yielded a reduction in deviance to a level almost twice the degrees of freedom (i.e. 245 on 136 degrees of freedom). It appears that the polar variables account for about 70% of the variation in 1975. The residual plots (Figs 33 a-f) show a concentration of high positive residuals at low fitted values, but there are 3 large negative residuals, the largest being -2.928. The residual surface (Fig 33 e,f) shows concentric peaks and troughs. The autocorrelation is very high ($z_I = 20.80$ under randomisation).

| Table 16 |

| Exploratory Tests for Hebeloma Data |

<table>
<thead>
<tr>
<th></th>
<th>1975</th>
<th>1978</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>-16.31 *</td>
<td>-9.26 *</td>
</tr>
<tr>
<td>S</td>
<td>-16.19 *</td>
<td>-10.08 *</td>
</tr>
<tr>
<td>W</td>
<td>13.08 *</td>
<td>9.861 *</td>
</tr>
<tr>
<td>R</td>
<td>0.409 *</td>
<td>0.397 *</td>
</tr>
<tr>
<td>U²</td>
<td>1.089 *</td>
<td>0.388 *</td>
</tr>
<tr>
<td>Wₛ</td>
<td>3.386 *</td>
<td>0.546</td>
</tr>
<tr>
<td>M</td>
<td>19.41 *</td>
<td>8.474 *</td>
</tr>
</tbody>
</table>

L : radial uniformity test (1.5)
S : spacing test (1.6)
W : Weibull shape test
R : Rayleigh test
U² : Watson's $U^2$ test
Wₛ : Interaction score test (1.9)
M : Mardia rank correlation test

* denotes significant at $\alpha = 0.05$ in a one-tail Monte Carlo test
Figure 33
Deviance residual displays: Hebeloma: 1975

a) fitted value

b) radial distance (r)

c) angle (th)

d) normal quantile plot
Figure 33

Deviance residual displays: Hebeloma 1975

Surface view of standardised residuals (z axis) in X-Y coordinates (h = 4.965)

f) contour map of e)
Figure 34
Deviance residual displays: Hebeloma 1978

<table>
<thead>
<tr>
<th>stand</th>
<th>res</th>
<th>v</th>
<th>fit</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.300</td>
<td>3.000</td>
<td>2.700</td>
<td>2.400</td>
</tr>
<tr>
<td>-0.300</td>
<td>-0.600</td>
<td>-0.900</td>
<td>-1.200</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>stand</th>
<th>res</th>
<th>v</th>
<th>r</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.300</td>
<td>3.000</td>
<td>2.700</td>
<td>2.400</td>
</tr>
<tr>
<td>-0.300</td>
<td>-0.600</td>
<td>-0.900</td>
<td>-1.200</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>stand</th>
<th>res</th>
<th>v</th>
<th>th</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.300</td>
<td>3.000</td>
<td>2.700</td>
<td>2.400</td>
</tr>
<tr>
<td>-0.300</td>
<td>-0.600</td>
<td>-0.900</td>
<td>-1.200</td>
</tr>
</tbody>
</table>

- a) fitted value
- b) radial distance (r)
- c) angle (th)
- d) normal quantile plot
Deviance residual displays:
Hébéoma 1978

Figure 34

e) surface view of standardised residuals (z axis) in X-Y coordinates (h = 0.122)

f) contour map of e)
Figure 35

1975

1978

a) fitted value
b) normal quantile plot
Standardised residual displays:
BSP model fit:
Hebeloma: 1975

c) surface view: z axis: standardised residuals

d) contour plot (h = 4.975)
Figure 35

Standardised residual displays:
BSP model fit:
Hebeloma: 1978

c) surface view: z axis: standardised residuals

d) contour plot h = 0.122
In contrast, the 1978 data appears to be better modelled by a constant rate (dev = 143.67 on 71 df) and the 'best' combination of polar variables yields dev = 31.36 on 67 degrees of freedom. In this case, the deviance is much lower than the degrees of freedom. A simpler radial-angular model is found, compared to 1975, and this is supported by the pattern of test significance in Table 16.

The residual plots (Figs 34 a-f) show some clustering of high positive and negative residuals at 2/5 of window radius. This distance is where the highest point density lies. The residuals vary widely in this case: the highest positive residuals are 3.127, 3.027, 2.553, 2.349 and the lowest negative residual is -2.344. However, the normal plot shows a closer linear tendency than 1975, albeit with a large gap between 0 and +1.0. The residual surface is less peaked than 1975, but a large single peak in the SE predominates. Although the residuals are autocorrelated this is much reduced (zI = 8.195 under randomisation).

Overall, it appears that the simple polar models do not account for enough of the spatial variation to produce residuals with appropriate properties i.e. negligible autocorrelation, linearity in normal plots etc. The extent of positive autocorrelation suggests that some form of clustering or heterogeneity may be present.

We have considered the application of a BSP model for both data sets to allow for heterogeneity or spatial correlation. In both cases, the parameter combination of $\alpha = 0.1$ and $\sigma^2 = 0.1$ was found to yield the 'best' fit with the full polar model (const, r, ln(r), cos $\phi$, sin $\phi$, r cos $\phi$, r sin $\phi$).

This fit gave the following parameter estimates.
<table>
<thead>
<tr>
<th>parameter</th>
<th>estimate</th>
<th>se</th>
<th>estimate</th>
<th>se</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-9.1610</td>
<td>0.1455</td>
<td>14.6197</td>
<td>0.1265</td>
</tr>
<tr>
<td>R</td>
<td>9.1621</td>
<td>0.1735</td>
<td>-14.6208</td>
<td>0.0205</td>
</tr>
<tr>
<td>ξₙR</td>
<td>-2.0050</td>
<td>0.7850</td>
<td>6.4159</td>
<td>0.5614</td>
</tr>
<tr>
<td>cosφ</td>
<td>-1.037</td>
<td>0.2368</td>
<td>0.0089</td>
<td>0.2805</td>
</tr>
<tr>
<td>sinφ</td>
<td>0.5422</td>
<td>0.5422</td>
<td>0.0748</td>
<td>0.3931</td>
</tr>
<tr>
<td>rcosφ</td>
<td>1.0377</td>
<td>0.5687</td>
<td>-0.0042</td>
<td>0.4347</td>
</tr>
<tr>
<td>rsinφ</td>
<td>-0.5424</td>
<td>0.1783</td>
<td>-0.0756</td>
<td>0.3255</td>
</tr>
<tr>
<td>ξ_max</td>
<td>-453.17</td>
<td></td>
<td>-887.15</td>
<td></td>
</tr>
<tr>
<td>ZI</td>
<td>0.371</td>
<td></td>
<td>1.185</td>
<td></td>
</tr>
</tbody>
</table>

The above results demonstrate that the BSP model appears to fit the 1975 data more closely than the 1978 example. 1975 has an ξ_max (-453.17) higher than the HEPP model, whereas 1978 has a much lower value. In addition the 1975 normal plot is quite close to linear and 1978 is highly curved. The autocorrelation for 1975 is also negligible (0.371); although that for 1978 is not high (1.185). The residual surfaces (fig 35 e,f) reflect this pattern as a single large peak appears in 1975 whereas a number of peaks occur in 1978.

In general, it appears that the 1975 data set can be modelled solely by polar coordinate functions with a BSP model extension. Although, there is a 70% reduction in deviance, the 'best' polar model HEPP deviance remains higher than the degrees of freedom. The 1978 data, however, appear to admit a simple radial-angular HEPP model.
and, apart from the residual pattern, appears to give a reasonable fit. The BSP model yields a decreased likelihood but significantly reduces the autocorrelation in the residuals.

9.2.2 Oak Bark Beetles (Data Set E)

This data set consists of counts of radio-labelled Oak Bark Beetles (Scolytus Intricatus), found on concentric sets of Oak logs. These data form part of an experiment concerning the dispersal behaviour of newly-emergent Beetles, in connection with the Beetle's potential as a vector for oak wilt disease (Yates (1983)). Figure 36 depicts the experimental layout used in the experiment. Radio-labelled beetles were released from the central point and later the sample logs were assessed for numbers of Radio-labelled Beetles \( n_b \) and non-labelled Beetles \( n_p \). These latter Beetles are thought to represent relatively local emergence compared to dispersal of the former type. Hence, the data set consists of polar coordinates of predetermined sample points, and numbers of \( n_b \) and \( n_p \) respectively.

This example differs from the count examples from Epidemiology, as the counts are located at points and not regionalised. However, the discrete constant rate model (eq 69) is particularly appropriate in this case, as no assumption of a piecewise constant rate is required. We have fitted such a model to the \( n_b \) counts. We examined a variety of models related to radial and angular effects, and to population structure e.g. the total number of beetles \( T_b = n_b + n_p \).

Table 17 below displays the results of fitting polar functions and \( T_b \).
Figure 36

The layout of host logs (●) and the position of the source of 11P marked oak bark beetles (indicated by *). Numbers above each log are the total numbers of marked beetles caught, those below are numbers of unmarked beetles.

Oak log locations: Oak Bark Beetles experiment
(ITE; Monks Wood)
Count map: Oak Bark Beetles
Table 17

GLIM Model for Beetle Data: Polar Model

<table>
<thead>
<tr>
<th>terms</th>
<th>dev</th>
<th>df</th>
<th>Δdev</th>
<th>δdf</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>299.9</td>
<td>68</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+LR</td>
<td>152.4</td>
<td>67</td>
<td>-147.5</td>
<td>1</td>
</tr>
<tr>
<td>+RC+RS</td>
<td>147.4</td>
<td>65</td>
<td>-5.05</td>
<td>2</td>
</tr>
<tr>
<td>+LRC+LRS</td>
<td>126.7</td>
<td>63</td>
<td>-20.74</td>
<td>2</td>
</tr>
<tr>
<td>+C+S</td>
<td>121.7</td>
<td>61</td>
<td>-5.01</td>
<td>2</td>
</tr>
<tr>
<td>+tb</td>
<td>61.83</td>
<td>60</td>
<td>-59.87</td>
<td>1</td>
</tr>
</tbody>
</table>

final model estimates:

<table>
<thead>
<tr>
<th>terms</th>
<th>estimates</th>
<th>Ses*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.350</td>
<td>0.4476</td>
</tr>
<tr>
<td>LR</td>
<td>-1.221</td>
<td>0.1376</td>
</tr>
<tr>
<td>RC</td>
<td>0.0454</td>
<td>0.0440</td>
</tr>
<tr>
<td>RS</td>
<td>-0.002</td>
<td>0.0452</td>
</tr>
<tr>
<td>LRC</td>
<td>-0.851</td>
<td>0.6110</td>
</tr>
<tr>
<td>LRS</td>
<td>-0.577</td>
<td>0.6261</td>
</tr>
<tr>
<td>C</td>
<td>1.234</td>
<td>0.8135</td>
</tr>
<tr>
<td>S</td>
<td>1.152</td>
<td>0.8720</td>
</tr>
<tr>
<td>tb</td>
<td>0.121</td>
<td>0.0179</td>
</tr>
</tbody>
</table>

*Ses adjusted for mean deviance

The null model yields a poor fit to this data (dev : 299.9, df = 68). The 'optimal' model for r-φ variables is given by 1+LR+RC+RS+LRC+LRS+C+S. However, only when tb is added does the deviance fall to the appropriate degrees of freedom. By inclusion of the population term only the ln<sub>nr</sub> term becomes significant and yields a significant t ratio (t = -
8.872). Hence for the model order used a radial peaked form and population structure are important. The sparseness of counts of marked beetles must also be considered here. This may reduce the deviance and hence direct comparison of deviance and df may be difficult.

The residuals from this fit lie between -1.2 and 1.6, with the exception of one point which has a value of 2.515. The plot against fitted values shows a reasonable spread. The high influence values are all between -1.0 and 0.0, and the high positive residual is not influential. The normal plot shows a high concentration of points between -0.5 and -1.5 while other residuals fall closer to the equality line. The residual surface is undulating and smooth with a peak at the high positive residual. the autocorrelation coefficient is 0.01508, which has MC prob = 0.19, which is not significant. Hence, although the residual range and autocorrelation is acceptable, the excess of low negative residuals suggests an inadequate model.

We have also considered the inclusion of T_b and log(T_b) (~tb) before the polar functions. The largest reduction in deviance is given by ~tb, as shown below. The inclusion of ~tb allows the population structure to be admitted prior to inclusion of polar functions and this reduces the significant covariates to R only (dev = 43.05 on 66 df). Hence population space/territory appears to explain a large amount of the variation, while a

<table>
<thead>
<tr>
<th>terms</th>
<th>Dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>1+tb</td>
<td>209.14</td>
<td>67</td>
</tr>
<tr>
<td>1+~tb</td>
<td>145.08</td>
<td>67</td>
</tr>
<tr>
<td>1 + ~tb + R</td>
<td>43.05</td>
<td>66</td>
</tr>
</tbody>
</table>
simple exponential radial decline of marked beetles also explains a large amount of the variation.

It is further possible to gain a degree of freedom, by using $tb$ as an offset. In this case, we have:

<table>
<thead>
<tr>
<th>terms</th>
<th>Dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>145.65</td>
<td>68</td>
</tr>
<tr>
<td>1+R</td>
<td>45.7</td>
<td>67</td>
</tr>
</tbody>
</table>

In addition, no other polar functions are significant.

Our following discussion is based on the $tb$ offset model with 1+R terms.

Table 18

GLIM model results for Beetle data : $tb$ offset

<table>
<thead>
<tr>
<th>term</th>
<th>estimate</th>
<th>ses*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.054</td>
<td>0.1167</td>
</tr>
<tr>
<td>R</td>
<td>-0.074</td>
<td>0.0068</td>
</tr>
</tbody>
</table>

* ses adjusted for mean deviance

Table 18 gives details of the parameter estimates and standard errors.
Figure 38
Anscombe Residual displays: Beetle Data: ltb offset

a) fitted value

b) radial distance

c) angle

d) normal quantile plot
Anscombe Residual displays: Beetle Data: ltb effect

Figure 38

surface view of standardised residuals (z axis) in X-Y coordinates (h = 11.035)
The residual plots (Figs 38 a-f) show a reasonable spread of values: all residuals are between -1.5 and 1.75 except for one high positive value (2.618) and one high negative value (-1.877). The residual versus fit plot shows a lot of residuals at low fitted values. In this plot some 'garland' patterns occur, again due to discrete count data with associated continuous fitted values. There is no suggestion of a radial or angular relation with residuals. The high leverage values are related to high fitted values. The normal plot shows an approximate linear relation although there are local concentrations of residuals between 0 and -1.0 which translate the line. The residual surface (Fig 38 e,f) is relatively smooth with peaks in the NE and SW directions. The autocorrelation of the residuals is negligible ($I = -0.0029$, with MC prob = 0.25 not significant).

Overall, it may be concluded that the model using log (total beetles) as offset is an adequate representation of the spatial structure of the Beetle dispersal. With radial distance included the deviance is reduced considerably. The parameter estimate (-0.074 with $Se = 0.0068$) has $t$ value = -10.88 and is significant. The residual patterns appear reasonable, although some clusters do occur on the normal plot. These clusters of points are not spatially correlated, however.

\textbf{9.2.3 Lupinus Arboreus (Data Set F)}

This example concerns the dispersal and colonisation of Lupinus arboreus on china clay wastes. The data are derived from a study of such colonisation over a 6 year period in relation to soil environmental gradients (Palaniappan \textit{et al.} (1979)).

The data are recorded as point locations around a central primary plant. Figure 1 f) depicts the data structure. As can be seen in this figure, the points form a concentric concentration, although some regularity may also be present. We have fitted a basic HEPP model to the data with radial and angular components. We have not attempted to model the
component of regularity in the data. Table 19 displays the results of estimating using GLIM with Delaunay weights.

---

Table 19

GLIM model results for LUPINUS data

<table>
<thead>
<tr>
<th>terms</th>
<th>est</th>
<th>se*</th>
<th>dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-6.084</td>
<td>0.1147</td>
<td>74</td>
<td>105</td>
</tr>
<tr>
<td>1</td>
<td>-15.32</td>
<td>2.057</td>
<td>36</td>
<td>103</td>
</tr>
<tr>
<td>+R</td>
<td>-0.095</td>
<td>0.014</td>
<td></td>
<td></td>
</tr>
<tr>
<td>+LR</td>
<td>3.733</td>
<td>0.710</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* adjusted for mean deviance

---

The GLIM results demonstrate that the lupinus data is essentially isotropic but there is a marked radial decrease and a peaked effect. This is reflected in tests for radial and angular effects (Table 20). Here, the radial tests are significant (uniformity test (L), spacing test (S), Weibull shape test (W)) whereas the angular and interaction tests show no effects.

---

Table 20

Exploratory Tests for LUPINUS data

<table>
<thead>
<tr>
<th>L</th>
<th>S</th>
<th>W</th>
<th>R</th>
<th>U²</th>
<th>WS</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>-13.608</td>
<td>-13.38</td>
<td>8.87</td>
<td>0.142</td>
<td>0.098</td>
<td>0.745</td>
<td>1.473</td>
</tr>
</tbody>
</table>

* denotes significance at α = 0.05 level in a one-tail monte carlo test
Hence there appears to be a radial dispersal with little preferred angular effect. Examination of the residual plots (Fig 39 a,f) show a reasonable spread around 0, although there is some concentration of residuals just below 0. There is one marked negative residual (item 63; -3.38) and another at -2.02. There are three high positive residuals. The residual surface (Fig 39 e,f) shows a large dip and peak close to each other. The autocorrelation is not significant (I = -0.729 under randomisation), although it is negative, which suggests slight regularity or inhibition in the pattern. Overall, the low deviance and small standard errors suggest a reasonable model, although the presence of a number of large residuals and an excess of negative residuals between -1.5 and 0.0 suggests that an improved model could be achieved.

We have also considered the inclusion of general spatial variables in the model i.e. functions of x and y, rather than polar coordinates. After a variety of combinations were considered the 'best' model achieved was, as follows:

<table>
<thead>
<tr>
<th>terms</th>
<th>dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>1+X2+Y2+R</td>
<td>31</td>
<td>102</td>
</tr>
<tr>
<td>Xl ⇒ X²</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>²max = -516.7</td>
<td></td>
</tr>
</tbody>
</table>

For one degree of freedom we have reduced the model deviance by 5. This leaves radial distance (R) as the only significant polar variable. The parameter estimates are

<table>
<thead>
<tr>
<th>terms</th>
<th>estimates</th>
<th>se*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-6.953</td>
<td>0.418</td>
</tr>
<tr>
<td>X2</td>
<td>-0.0007</td>
<td>0.0001</td>
</tr>
<tr>
<td>Y2</td>
<td>-0.0009</td>
<td>0.0001</td>
</tr>
<tr>
<td>R</td>
<td>0.0702</td>
<td>0.015</td>
</tr>
</tbody>
</table>

* adjusted for mean deviance
Figure 39
Deviance Residual displays: Lupinus Arboreus

a) fitted value

b) radial distance

c) angle

d) normal quantile plot
Deviance Residual displays: Lupinus Arboreus

Figure 39

surface view of standardised residuals (z axis) in X-Y coordinates (h = 18.026)

f) contour map of e)
Figure 40
Standard residual displays: BSP model: Lupinus Arboreus

a) fitted value

b) normal quantile plot
Standardised residual displays:
BSP model: Lupinus Arboreus

c) surface view: z axis: standardised residuals

d) contour plot: \( h = 0.1744 \)
However, the residual plots and surfaces show little differences from the original model, and the large residuals are not reduced significantly.

We have also fitted a BSP model to this data. The optimal values of $\alpha, \sigma^2$ were $\alpha = 0.5, \sigma^2 = 0.5$, and this yielded the following parameter estimates.

<table>
<thead>
<tr>
<th>term</th>
<th>estimate</th>
<th>se</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.836</td>
<td>0.6942</td>
</tr>
<tr>
<td>R</td>
<td>0.996</td>
<td>0.6700</td>
</tr>
<tr>
<td>log R</td>
<td>3.841</td>
<td>0.1579</td>
</tr>
<tr>
<td>cos $\phi$</td>
<td>-3.534</td>
<td>0.1872</td>
</tr>
<tr>
<td>sin $\phi$</td>
<td>0.779</td>
<td>0.5282</td>
</tr>
<tr>
<td>r cos $\phi$</td>
<td>2.669</td>
<td>0.6048</td>
</tr>
<tr>
<td>r sin $\phi$</td>
<td>-0.814</td>
<td>0.1379</td>
</tr>
</tbody>
</table>

$L_{\text{max}} = -463.0$

The residuals from this fit yield a smooth undulating surface (Fig 40 c,d) and non-significant autocorrelation ($z_l = 1.374$, under randomisation). The normal plot shows a linear trend, but is offset from the equality line. The residual range is -2.26 to +1.23.

Overall, the HEPP model appears to fit the lupinus data reasonably well, although some high residual values have led to examination of a BSP model. This model has provided a continuous normal plot (unlike the HEPP model) but shows an asymmetric residual distribution.

The main spatial effects seem to be radial in that exploratory testing and model fitting tends to yield significance for radial and/or log (radial) variables. In this case it may be concluded that a model for non-stationarity with a radial covariate is appropriate but that the regular component in the data should also be modelled. A non-stationary Markov model may be appropriate for, at least, some part of the data set.
9.2.4 Volcanic Ejecta (Data Set G)

The last data examples concern the location of volcanic bombs (ejecta) after the eruption of a volcano. The data consists of a complete mapping of bomb locations after eruptions of Mt Asama, Japan in 1935, 1937 and 1938. In each of these examples there is evidence of a radial peak of deposition as well as a pronounced angular effect. The 1935 example shows a ring structure around the central point, while 1937 is similar with a high intensity of points and a marked peak in the radial distribution. The 1938 example shows a marked angular concentration due east of the centre, with an outlying cluster of points south-east of the centre. Figure 1g displays the point maps of the eruptions and Figures 5 a,b) display the marginal angular and radial cross-validated kernel estimates and Figures 5c) display the cross-validated kernel surface isometric views and contour plots.

These displays suggest a smoothly varying intensity structure with radial and angular components. The 1937 example may be supposed to display \( \delta \)-dependence in that it has a marked radial mode which depends on angle, whereas 1938 has a pattern more characteristic of an interaction model. The 1935 example has a marked radial mode but has less angular dependence than 1937.

Preliminary tests of the patterns (Table 21) shows significant results for radial (\( \lambda \)), peakedness (\( \delta \)) and angular concentration (Watson's U\(^2\)), while angular-radial interaction tests are all significant for each year, under monte carlo assumptions.
Table 21

Exploratory tests for Volcanic Bomb examples

<table>
<thead>
<tr>
<th></th>
<th>1935</th>
<th>1937</th>
<th>1938</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>-10.06*</td>
<td>-17.71*</td>
<td>-20.05*</td>
</tr>
<tr>
<td>S</td>
<td>-10.70*</td>
<td>-18.99*</td>
<td>-21.57*</td>
</tr>
<tr>
<td>W</td>
<td>12.56*</td>
<td>14.96*</td>
<td>18.12*</td>
</tr>
<tr>
<td>R</td>
<td>0.447*</td>
<td>0.401*</td>
<td>0.930*</td>
</tr>
<tr>
<td>U²</td>
<td>0.502*</td>
<td>1.182*</td>
<td>9.328*</td>
</tr>
<tr>
<td>WS</td>
<td>9.458*</td>
<td>7.847*</td>
<td>39.51*</td>
</tr>
<tr>
<td>M</td>
<td>27.15*</td>
<td>111.96*</td>
<td>11.31*</td>
</tr>
</tbody>
</table>

* denotes significance at α = 0.05 level in a one-tail monte carlo test

9.2.4.1 The 1935 example

As noted in Chapter 7, we have used GLIM with Delaunay weights to fit the relevant models. Table 5 (Sect 7.5.4) demonstrates the results of the fitting process. Essentially no radial effect is apparent and angular components provide a large change in deviance ($\Delta\text{dev} = 18.2, \Delta\text{df} = 2$). The $\delta$-dependence terms (LRC, LRS) also reduce the deviance by an amount ($\Delta\text{dev} = 4.9; \Delta\text{df} = 2$). This does not reach the $\chi^2_{0.05,2}$ value, but may be included given that it is above the 10% level which is often used in forward selection.

While polar variables alone reduce the deviance significantly, the value is high compared to the degrees of freedom. It is possible to consider the inclusion of x-y variates in the model as well. These can be used to allow for unobserved heterogeneity. In this example, we have used two strategies. First, we have fitted the 'best' polar model and then
added 'significant' x-y variates up to 4th order. Second, we fitted x-y variates and then polar variates. The best models found are:

<table>
<thead>
<tr>
<th>model</th>
<th>dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>1+R+LR+</td>
<td>39.0</td>
<td>60</td>
</tr>
<tr>
<td>C+S+LRC+LRS +X+Y+X2+Y2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1+X+Y+X2+Y2 +X4+Y4+X2Y+Y2X +X2Y2+LRC+LRS</td>
<td>37.4</td>
<td>59</td>
</tr>
</tbody>
</table>

here Xi ⇒ X^i etc.

Hence the lowest deviance model included polynomial terms x and y and δ-dependence polar terms (LRC : log(r).cos φ; LRS : log(r).sin φ). We give below the final estimates for the last model:

<table>
<thead>
<tr>
<th>terms</th>
<th>estimate</th>
<th>se*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-12.41</td>
<td>1.5689</td>
</tr>
<tr>
<td>X</td>
<td>0.1276</td>
<td>0.0508</td>
</tr>
<tr>
<td>Y</td>
<td>-0.2139</td>
<td>0.0975</td>
</tr>
<tr>
<td>X2</td>
<td>0.0078</td>
<td>0.0016</td>
</tr>
<tr>
<td>Y2</td>
<td>0.0034</td>
<td>0.0011</td>
</tr>
<tr>
<td>X4</td>
<td>-1.9×10^-6</td>
<td>4.2×10^-7</td>
</tr>
<tr>
<td>Y4</td>
<td>-5.7×10^-7</td>
<td>1.9×10^-7</td>
</tr>
<tr>
<td>X2Y</td>
<td>-5.4×10^-6</td>
<td>1.4×10^-5</td>
</tr>
<tr>
<td>Y2X</td>
<td>-1.1×10^-5</td>
<td>5.7×10^-6</td>
</tr>
<tr>
<td>X2Y2</td>
<td>-2.4×10^-6</td>
<td>5.5×10^-7</td>
</tr>
<tr>
<td>LRC</td>
<td>-1.34</td>
<td>0.5916</td>
</tr>
<tr>
<td>LRS</td>
<td>2.43</td>
<td>1.083</td>
</tr>
</tbody>
</table>

\[ \chi_{max} = -277.0 \]

* adjusted for mean deviance
The standardised residuals from this fit lie in the range -1.75 to +1.25, with one large negative residual (-2.145) and 3 large positive residuals (2.112, 2.275, 1.89). There is a gap with no values between 0.0 and 1.0 and a large concentration of negative residuals between 0.0 and -1.0. The probability plot reflects this structure (Fig 41 d). The residual surface (Fig 41 e,f) shows a linear peak to the west of the centre and a trough in the east. In fact, the autocorrelation coefficient ($Z_l$) is 16.71 (under randomisation) which represents high positive autocorrelation.

Overall the model yields a low deviance but an unacceptable residual pattern. We have also fitted a Bayesian Spatial Prior (BSP) model as an alternative to allow for the possibility of unobserved heterogeneity or clustering in the data. We assumed a variety of $\sigma^2$ and $\alpha$ values for the prior, with a 5 parameter polar model. By trial and error, we found that the $\sigma^2 = 0.5$, $\alpha = 0.5$, gave the lowest MSE. The following results were obtained.

<table>
<thead>
<tr>
<th>term</th>
<th>estimate</th>
<th>se</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-17.841</td>
<td>0.5776</td>
</tr>
<tr>
<td>$r$</td>
<td>17.225</td>
<td>0.6084</td>
</tr>
<tr>
<td>$\ln(r)$</td>
<td>-8.769</td>
<td>0.4591</td>
</tr>
<tr>
<td>$\cos \phi$</td>
<td>-0.418</td>
<td>0.1091</td>
</tr>
<tr>
<td>$\sin \phi$</td>
<td>-0.560</td>
<td>0.5888</td>
</tr>
<tr>
<td>$r \cos \phi$</td>
<td>0.111</td>
<td>0.7761</td>
</tr>
<tr>
<td>$r \sin \phi$</td>
<td>1.593</td>
<td>0.1791</td>
</tr>
</tbody>
</table>

$\xi_{\text{max}} = -265.0$

The residuals from this fit show a good approximation to a straight line on a normal plot (Fig 42 d). The residuals range from -2.4 to +2.0. The residual surface (Fig 42 c,d) shows a smooth form with a number of peaks and troughs.
Figure 41
Deviance Residual displays: 1935 volcanic ejecta

a) fitted value

b) radial distance

c) angle

d) normal quantile plot
Deviance
Residual displays:
1935 volcanic ejecta

(e) surface view of standardised residuals (z axis) in X-Y coordinates (h = 5.923)

(f) contour map of (e)
Figure 42
Standardised Residual displays: BSP model: 1935 volcanic ejecta

a) fitted value

b) normal quantile plot
Figure 42

c) surface view: z axis: standard residuals

d) contour plot (h = 5.914)
There is some autocorrelation ($z_1 = 2.178$ under randomisation). Hence, in the case of the 1935 data a BSP model with polar variables appears to provide a good model for the point distribution, although the HEPP model also yields a large reduction in deviance and hence a feasible model.

### 9.2.4.2 The 1937 example

We have used GLIM with Delaunay weights to fit the relevant models. As in the 1935 example, we have fitted polar-cartesian variates in different orders:

<table>
<thead>
<tr>
<th>term</th>
<th>dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1$</td>
<td>387.0</td>
<td>168</td>
</tr>
<tr>
<td>$1 + R + LR + C + S + LRC + LRS$</td>
<td>239.1</td>
<td>162</td>
</tr>
</tbody>
</table>

The best polar model reflects the $\delta$-dependence suggested by the kernel surfaces in that LRC and LRS terms are included. However, it appears that polar variables alone do not provide a feasible model for this data set. We have also fitted cartesian (x-y) and polar variables as follows:

<table>
<thead>
<tr>
<th>term</th>
<th>dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1 + X + Y + X^2 + Y^2 + X_4 + Y_4 + X_2 Y + Y_2 X + X_2 Y_2 + R + LR$</td>
<td>207.0</td>
<td>157</td>
</tr>
</tbody>
</table>
Hence, in this case even an x-y-r-\( \phi \) model does not reduce the deviance to an acceptable level. The parameter estimates for this model are, as follows:

<table>
<thead>
<tr>
<th>terms</th>
<th>estimate</th>
<th>se*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-117.3</td>
<td>33.32</td>
</tr>
<tr>
<td>X</td>
<td>0.0182</td>
<td>0.014</td>
</tr>
<tr>
<td>Y</td>
<td>0.00092</td>
<td>0.006</td>
</tr>
<tr>
<td>X2</td>
<td>0.049</td>
<td>0.012</td>
</tr>
<tr>
<td>Y2</td>
<td>0.051</td>
<td>0.012</td>
</tr>
<tr>
<td>X4</td>
<td>-2.1x10^{-6}</td>
<td>5.0x10^{-7}</td>
</tr>
<tr>
<td>Y4</td>
<td>-2.6x10^{-6}</td>
<td>5.9x10^{-7}</td>
</tr>
<tr>
<td>X2Y</td>
<td>1.66x10^{-6}</td>
<td>4.61x10^{-6}</td>
</tr>
<tr>
<td>Y2X</td>
<td>1.07x10^{-6}</td>
<td>1.16x10^{-5}</td>
</tr>
<tr>
<td>X2Y2</td>
<td>-4.72x10^{-6}</td>
<td>1.02x10^{-6}</td>
</tr>
<tr>
<td>R</td>
<td>-4.991</td>
<td>1.305</td>
</tr>
<tr>
<td>LR</td>
<td>64.51</td>
<td>18.20</td>
</tr>
</tbody>
</table>

\( \hat{\phi}_{\text{max}} = -732.7 \)

* adjusted for mean deviance

Examination of the residuals for the model shows a number of values of \( |r_i| > 2.0 \), albeit mainly negative (2.01, 2.1, -2.25, -2.49). The residual plots (Figs 43 a-f) show a reasonable spread of results, although a gap between 0 and +1 is apparent here. The normal plot is linear in the lower section, below the gap and thence is flatter than expected above +1.0. The residual surface (Fig 43 e,f) is characterised by a ring of peaks and troughs. The residuals are significantly autocorrelated (\( z_1 = 26.08 \) under randomisation).

The above results suggest that a BSP model may also be appropriate for this data set.

By trial and error a model with \( \sigma^2 = 0.1 \), and \( \alpha = 0.5 \) was fitted: The following parameter estimates were obtained:
Figure 43
Deviance Residual displays: 1937 volcanic ejecta

a) fitted value

b) radial distance

c) angle

d) normal quantile plot

d) normal quantile plot

---

stand res vs fit
0.000 0.0400 0.0800 0.1200 0.1600 0.2000 0.2400 0.2800 0.3200 0.3600
stand res vs th
0.00 1.60 3.20 4.80 6.40 8.00 9.60 12.0 24.0 36.0 48.0 60.0 72.0 84.0
surface view of standardised residuals (z axis) in X-Y coordinates (h = 3.9008)

f) contour map of e)
Figure 44
Standardised Residual displays: BSP model: 1937 volcanic ejecta

a) fitted value

b) normal quantile plot
c) surface view: z axis: standardised residuals

d) contour plot (h = 0.050)
The residuals show a reasonable range, in that the majority of points lie between ±1.9. There is one large negative residual (-2.5) and 2 high, positive values (+2.048, +2.11). On the whole, the normal plot (Fig 44 b) follows a straight line remarkably well.

The residual surface (Fig 44 c,d) shows a concentric ridge of peaks and troughs with a single large peak in the south east. The autocorrelation is not significant (zI = 1.401, under randomisation). Hence, the residual evidence suggests a good fit by the BSP model in this case. It appears that after underlying heterogeneity has been accounted for, there is a slight radial effect and a more marked radial interaction effect.

### 9.2.4.3 The 1938 example

We have also used GLIM with Delaunay weights to fit the relevant models in this case. In this case a model confined to polar functions yielded the following results:

<table>
<thead>
<tr>
<th>term</th>
<th>dev</th>
<th>df</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1016</td>
<td>198</td>
</tr>
<tr>
<td>1+C+S</td>
<td></td>
<td>191</td>
</tr>
<tr>
<td>+RC+RS</td>
<td>333</td>
<td></td>
</tr>
<tr>
<td>+LR+LRC+LRS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The addition of x-y covariates yielded non-convergence and subsequent failure to cycle. The inclusion of such terms before polar functions in general yielded higher deviances than the above model. Hence, we confine discussion to the above polar model. It appears that the null model does not provide a reasonable fit, but the polar model reduces the deviance considerably and achieves approximately 66% explanation. However, the deviance remains large compared to the degrees of freedom. This suggests that other factors may be important.

The parameter estimates for this model are:

<table>
<thead>
<tr>
<th>terms</th>
<th>estimate</th>
<th>se*</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>211.22</td>
<td>61.648</td>
</tr>
<tr>
<td>C</td>
<td>-337.09</td>
<td>63.891</td>
</tr>
<tr>
<td>S</td>
<td>424.55</td>
<td>117.697</td>
</tr>
<tr>
<td>RC</td>
<td>-0.540</td>
<td>0.107</td>
</tr>
<tr>
<td>RS</td>
<td>1.638</td>
<td>0.428</td>
</tr>
<tr>
<td>LR</td>
<td>-53.092</td>
<td>14.530</td>
</tr>
<tr>
<td>LRC</td>
<td>91.052</td>
<td>15.758</td>
</tr>
<tr>
<td>LRS</td>
<td>-127.276</td>
<td>34.703</td>
</tr>
</tbody>
</table>

\[ \chi_{\text{max}} = -790.3 \]

* ses adjusted for mean deviance

As can be seen, all parameters yield 't' ratios of > 3 and are well estimated. Hence, this example admits both peaked radial and radial-angular interaction and \( \delta \)-dependence effects.

The residual fitted values plot (Fig 45a) shows a general cloud of residuals related to fitted values with 5 large negative residuals below -2.7 and one large positive residual at +2.4. Angular residuals are clustered around the 0 (rad) point which is the approximate centre of the angular distribution (\( X_0 = 6.127 \) (rad)). There is no radial clustering (Figs 45 b,c). The normal plot (Fig 45d) shows linearity but a substantial gap between 0 and +1.0. The large residuals lie off the line. The residual surface (Fig 45 e,f) is quite undulating...
Figure 45
Deviance residual displays: 1938 volcanic ejecta

a) fitted value

b) radial distance

c) angle

d) normal quantile plot
surface view of standardised residuals (z axis) in X-Y coordinates (h = 2.776)

f) contour plot of e)
Figure 46
Standardised residual displays: BSP model: 1938 volcanic ejecta

a) fitted value

b) normal quantile plot
c) surface view: z axis: standardised residuals

f) contour plot (h = 0.021)
with peaks in the south and troughs in the north. The autocorrelation is very high ($z_I = 24.318$, under randomisation). The large peak in the south-west of the plot corresponds to the cluster of points evident in Fig 1g, separated from the main data 'cloud'.

We have also considered a BSP model for this data set. By trial and error we have found that $\sigma^2 = 0.1$ and $\alpha = 10.0$ yield the 'best' model. For this case, the parameter estimates are as follows:

<table>
<thead>
<tr>
<th>term</th>
<th>estimate</th>
<th>se</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.290</td>
<td>0.846</td>
</tr>
<tr>
<td>$R$</td>
<td>1.229</td>
<td>0.402</td>
</tr>
<tr>
<td>log $R$</td>
<td>3.194</td>
<td>0.096</td>
</tr>
<tr>
<td>$\cos \phi$</td>
<td>-1.579</td>
<td>0.419</td>
</tr>
<tr>
<td>$\sin \phi$</td>
<td>-0.982</td>
<td>0.802</td>
</tr>
<tr>
<td>$r \cos \phi$</td>
<td>1.498</td>
<td>0.771</td>
</tr>
<tr>
<td>$r \sin \phi$</td>
<td>0.986</td>
<td>0.088</td>
</tr>
</tbody>
</table>

$\chi_{\text{max}}^2 = -504.51$

The residuals from this model show a reasonable normal plot, although there is some 'snaking' apparent (Fig 46b). No gap is apparent between 0 and +1.0. The residual surface (Fig 46c,d) has changed considerably from the HEPP model. The only peaks on the surface are now in the centre of the area, and an outlying cluster in the south of the area. The autocorrelation is reduced but still significant ($z_I = 15.218$, under randomisation). This suggests that high positive residuals are associated with the isolated clusters.

Overall it appears that the BSP model does improve upon the polar HEPP model fit, and the residual evidence supports this contention.

However, the model does not produce negligible residual autocorrelation and hence neither model is entirely satisfactory.
Discussion and Conclusions

We have considered a range of point and count data sets related to a fixed point. We have developed HEPP models to describe the non-stationarity of the data sets and developed new tests associated with these models. In some cases, we have developed special models such as the case-control/SMR model in epidemiology and BSP models applied to the Hebeloma and volcanic bomb data sets. We have achieved varying degrees of success. In general, simple HEPP models based on polar functions have not been successful in providing a complete description of the data sets. The obvious exceptions to this are the Lupinus data (set F), the 1935 ejecta data and 1978 Hebeloma data which both share a very low mean deviance (albeit undesirable residual patterns). At least in the Lupinus example, regularity as an alternative may be important and hence for this case a simple HEPP model may require a Markov process extension. This has not been explored here. On the other hand, the BSP models, applied where appropriate, have yielded better model fits and usually acceptable residual patterns. The exception to this is the 1938 ejecta data, where the autocorrelation remained high. The discrete case has shown that simple Poisson models appear adequate to describe spatial disease variation in eds and in the dispersal of Beetles. In all these cases low mean deviances and acceptable residual patterns were achieved, with simple polar functions with or without suitable offsets. As a consequence of this, we have not needed to apply BSP models in the discrete case. A major problem which arises when BSP models are used, is the possibility of near-confounding of trend and covariance structure. We have approached this problem by applying a grid search for those covariance parameters which maximise the posterior probability for fixed trend components. Allowing the trend components to vary could lead to trade-offs occurring between trend and covariance parameters, as it is always possible to 'model' second order structure by using harmonic trend terms (see Sect 3.2.1 and 5.1.2.4.1). One possible improvement may be to iterate the fitting process by re-estimation
of covariance parameters from residuals, and hence re-fitting. The use of REML may also be appropriate. One common problem in our datasets which may require further analysis, is the gap in low positive residual values found in normal plots. This appears in the discrete case and in many examples of point patterns. It does not occur, however, in the deviance residuals for a simulated HEPP process, nor is it commonly found in BSP model residuals. Although we have found HEPP models applicable only in some data sets, the score tests which have been developed from these models have wide applicability, as they would continue to measure departure from the null hypothesis even when a BSP or other model for second order effects were appropriate. Given the need to use monte carlo critical regions with these tests, the objection to the use of asymptotic distributions for statistic evaluation is removed. An underlying BSP model may alter the power of such tests as well, and this is another area for further research. With regards to further research and topics not explored in this work, a number of areas could be examined. First, the possibility of estimating the centre of pollution may be possible in some cases, although it is likely that a flat likelihood surface may arise in this context and standard errors may be very large.

Second, there is scope for examination of higher order asymptotics of the score tests derived here. Peers (1971) has derived the appropriate non-centrality parameter for score test corrections. However, given the application of these tests for spatially correlated data, the validity of asymptotic results may be in doubt and hence Monte Carlo testing avoids the need for such corrections.

Third, it is possible to derive a score test for autocorrelation under the BSP model. This is being pursued at present.

Finally, the generality of the results for BSP models should allow such models to be applied in non-polar coordinate systems and to a wide variety of data likelihoods, where environmental or unobserved heterogeneity is thought to be important.
REFERENCES


Lawson, A B (1990a) Statistical Analysis of a Direction-Dependent Planar Point Process, Ch. 16, Polymodel XII, Ellis Horwood.


Lawson, A B (1990c) Score Tests for Detection of Spatial Trend in Morbidity Data, Accepted by "Statistics in Medicine".


Lewis, P A W (1965) Some Results on Tests for Poisson Processes, Biometrika, 52, 1+2, 67-77.


Snyder, D L (1975) Random Point Processes, Wiley.


Williams, D A (1976) Improved likelihood ratio tests for complete contingency tables, Biometrika, 63, 1, 33-37.


Appendix I  Computational Methods

In this Appendix, we detail the computational methods used. Where programs have been written by the author, these are enclosed. We do not detail numerical methods used within the GLIM package, as these are well documented elsewhere. Simulation methodology is detailed in Appendix II.

A) HEPP Models

Maximum likelihood estimation for these models is performed by a modified Newton algorithm (NAG : E4KAF) for direct maximisation of the likelihood. Note that in the case of \( \lambda(r) = f(r)g(r,\phi)/r \), the log-likelihood can be maximised separately for the f and g components. A number of numerical convergence problems were encountered, initially, during optimisation for M(\( \mu_0, \kappa+\phi \)) models. These problems appeared to be related to the possible trade-off between \( \kappa \) and \( \psi \) during maximisation. The problems were overcome when a routine requiring specification of first derivatives of the likelihood was used. The difference appears to be related to the numerical, as opposed to analytical, estimation of derivatives. Figure 47 displays a typical likelihood surface for \( \psi-\kappa \) maximisation.

Programs:

All programs used are written in Fortran 77 running under the VAX/VMS operating system.

name: JJW

Fits all 5 parameter intensity models via maximum likelihood. The program provides residual analysis, and graphical displays of original data.
name: **VMLIK**

Fits standard \( M(\mu_0, \kappa) \) models only.

name: **POWER**

This large program is designed for numerical power studies of test statistics. It can provide estimation for LR interaction tests as well as a range of angular-linear correlation, uniformity and concentration test statistics. It provides a simulation base for numerical comparison of test statistics. It can also be used to derive simple summary statistics and tests for a single data set.

Numerical procedures used are from IMSL and NAG: MMBSI\( \phi \), MMBSI1, (Bessel functions), ZXGSN (one-dim search for \( \hat{\kappa} \)), E\( \phi \)4KAЕ (LR testing with \( M(\mu_0, \kappa+\psi r) \) and \( M(\mu_0, \kappa) \)), GGUBFS, GGVMS (simulation).

name: **KERNTEST**

This program calculates test statistics for the kernel-case-control model as described in section 3.1.1. It is possible to obtain Monte Carlo one-tail probabilities for specific rest statistics using the program as well.

B) **Bayesian Spatial Prior Models (continuous)**

These models require the solution of a system of linear equations similar to those of ordinary GLS estimation. The only difference arises in the nature of the covariance structure and that an estimate of \( \lambda \) at each data point is required. The methods employed
are based on Cholesky decomposition followed by OLS, or direct Generalised Inversion and other matrix operations.

**Programs**

name: NEWKP2

This program provides estimates of $\hat{\theta}$ and $\text{var}(\hat{\theta})$ for a fixed covariance structure (variance and covariance range parameter specified), and uses initial estimates: $\hat{\lambda}_i = \frac{1}{A_i}$ where $A_i$ is the Dirichlet tile area for the i-th point.

C) **Discrete Multinomial Models**

The solution of maximisation problems for discrete models is only required for test statistics evaluation, as GLIM provides ML estimates via the Poisson error for the constant rate models applied here. GLIM also provides LR tests for spatial effects. To evaluate other test statistics, it is necessary to carry out ML estimation for nuisance parameters. In addition assessment of numerical power of test statistics requires maximisation of likelihoods.

In the discrete case, the likelihoods consist of sums of exponential terms (e.g. eq 69, 70) and can produce non-uniform convergence problems from different points, even when analytical derivatives are provided. We have resorted to a global optimisation system which uses a mesh of start coordinates and finds local maxima. It then selects a subset for iterative improvement and thence maximum selection. This appears to provide consistent solutions for 3 parameter angular models.

**Programs:**
name: SIMREG

This is a large general purpose program which provides a simulation base for test statistics evaluation, and evaluation of summary and test statistics for data sets. In addition, graphical facilities are available.

The maximisation procedure used is IMSL:ZXMWD which provides a Global method of optimisation. The simulation base uses IMSL routines: GGDA (alias method) and GGNSM (MVN prior generation).

D) Bayesian Spatial Prior Models (Discrete)

The discrete case models are close in structure to the continuous case and the same basic estimation method is used. Initial estimates of $\hat{\theta}_i = \ln(n_i)$ are employed.

Program:

name: NEWKRIGE2

This has a similar structure to NEWKP2.

E) Smoothing and Interpolation Programs

A number of smoothing and interpolation programs have been developed.

Programs:

name: KRIGE
This program performs 'Universal Kriging' as in Ripley (1981, p 49-50). A covariance structure is assumed and predictions at interpolation points are available.

The $\mathbf{\hat{A}}$ parameter vector is found by GLS which is carried out by Cholesky Decomposition and inversion (IMSL : LUDECP, LINV1P) and thence OLS by matrix operations. Surfaces up to cubics can be fitted.

name: FISH

This program carries out one-dimensional nonparametric density estimation using a Gaussian kernel. For single data input it uses IMSL : NDKER to estimate the density. For polar coordinate data, special own-written routines are used for $r$ (including reflection about 0) and $\phi$ (including toroidal copies). Boneva, et al (1971) and Byth (1982) discuss these implementation methods. Likelihood cross-validation is used to estimate $h_{opt}$ (Silverman, 1986).

name: SURFISH

This program carries out two-dimensional kernel estimation using a Gaussian kernel. This estimation routine is own written. Graphical output of contour and perspective plots is provided. The GINOSURF library is used. Likelihood cross-validation is used to estimate $h_{opt}$.

F) Simulation Programs

A number of programs have been developed to provide simulations of processes discussed here. The most important programs concern HEPP model simulation and BSP simulation.

Programs:

286
name: **PPGEN**

This uses similar routines to POWER but is intended to provide individual realisations of HEPP models. It uses IMSL: GGUBFS and GGVMS and provides graphical output via the GINO library. All displays of HEPP models have been generated by PPGEN.

name: **GAUSS**

This program generates individual realisations of BSP models, via simulation of an MVN mesh and thinning. (Details of this simulation method are given in Appendix II.) Meshes used are normally (100 × 100). These may be thinned from 10,000 to $n = 100-500$ depending on the sample size required.

MVN generation is by Cholesky Decomposition (IMSL: GGNSM) for small meshes (up to $20 \times 20$). For large meshes we use the Harmonic average method with modification to avoid trigonometric evaluations. Graphical output is provided based on GINO library routines. All figure displays of BSP models have been produced by GAUSS. A subroutine version of GAUSS is used with POWER.
Appendix II Simulation Methods

A number of different models have been simulated in this study and a range of simulation methods have been used.

A) HEPP models

In general, HEPP models can be simulated by thinning (Lewis and Shedler (1979), Ogata (1981)). However, for the polar coordinate intensity model with $g(\phi,r)$ defined as $M(\mu_0, \kappa+\psi r)$ and $f(r)$ as truncated Weibull (eq 11 and 12 respectively), we can use a simple conditioning argument for a bivariate distribution (see, e.g. Rubinstein (1981, p 59). If a marginal distribution for one variable is available then the conditional distribution of the second variable can be used, and a pair of variate values can be generated. In the case of eq 11 and 12 the marginal distribution of $R$ is truncated Weibull, and the conditional distribution of $\Phi$ is $M(\mu_0, \kappa+\psi r)$.

Hence, our algorithm is

1) generate $u_r = -\frac{1}{c_2} (\log (1 - ab))^{1/c_1}$

where $ab = u_1(1 - \exp[-c_0^{c_1}])$

and $c_2 = \lambda = \text{scale parameter}$

$c_1 = \delta = \text{shape parameter}$

$u_1 \sim U(0,1)$

2) set $\varepsilon = \kappa + \psi u_r$
3) generate $uq$ from $M(\phi,\varepsilon)$ using Best and Fisher's envelope rejection method (Best and Fisher (1979))

4) repeat 1)-3) for $n$ sample points.

This algorithm encompasses a range of possible models and suffices for the class of HEPP models utilised here.

B) BSP models (Continuous)

Bayesian spatial prior models can, in principle, be simulated in two stages. First, simulation of a realisation of the prior. Second, conditional on this realisation, simulation of the point process realisation. This is essentially the method advocated by Diggle (1983, p 58) for Cox processes, where the intensity measure $\Lambda$, say, is a spatial stochastic process (non-negative) and conditional on the realisation $\Lambda^*$, say, we generate the point events as a HEPP model. For this case thinning is usually used. In our case, we have a Gaussian spatial prior distribution for the log intensity. Hence, this is equivalent to a Gaussian process realised at the point event sites. Our algorithm is as follows:

1) Generate a fine mesh ($axb$) where $a, b >> 50$ and which covers the sampling window.

2) Generate a realisation ($g_n$) of MVN ($\mu_n, K_n$) at the $n$ ($=axb$) mesh points for given $\mu_n, K_n$.

3) Set $\lambda^*_1(r) = \exp(g_1)$

4) Accept ($\{\lambda^*_1(r) | r_i < r_0\} = \{\lambda^c_i\}_{i=1,n_c}$

5) Find $\lambda^c_M = \max_{\lambda_i} \{\lambda^c_i\}$
6) Accept \((\lambda_i^c \mid U_1 \geq \lambda_i^c / \lambda_M^c)\)

where \(U_1 \sim U(0,1)\).

To generate a fixed number of events \(m\) (\(\leq n_c\)) we select \(\lambda_i^c\) randomly from the set of labels \(1 \rightarrow n_c\) and execute until \(m\) is reached. For small meshes this must be done by sampling without replacement. However, for the large meshes needed to provide continuity of the prior surface \((a, b = 100, \text{ at least})\), it is usually adequate to use a discrete uniform distribution for the labels.

Although the algorithm is conceptually simple, the generation of an MVN mesh, of large size is not straightforward. The technique usually used for small vector sizes \((n)\) is Cholesky Decomposition (Ripley (1987), p 99), but this can become very inefficient for meshes with \(n \geq 1000\). In the author's experience the method becomes inefficient above \(a = b = 20\).

The usual alternative to this method is the 'Turning Band' method (Journel and Huijbregts (1978, Ch VII), Ripley (1987, p 108-110), Brooker (1985)). This method uses a one dimensional simulation at fixed points on \(\mathbb{R}^1\), the simulated values being assigned as constants to the area surrounding the fixed point (the band). A number \((n^*)\) of uniform rotations around the origin and thence projection onto the generation line yields a value from the band in which the point falls. The method requires the specification of a one-dimensional covariance which is structured so that when the \(n^*\) rotations are averaged the resulting realisation has the required two-dimensional covariance. Simulation of Turning Bands can be performed via fast fourier transforms. Davis et al (1981) give an algorithm for generation of a one dimensional process on \(\mathbb{R}^1\).

An alternative to the 'Turning Band' method has been derived by Mejia and Rodriguez-Iturbe (1974) for two dimensional simulations. The method is used extensively in hydrological and pedological sciences (see Russo (1984), Freeze (1980)). The method
consists of simulating a number of samples of a process on a mesh. The samples are averaged to give an MVN realisation. The theoretical justification for the algorithm is given in Mejia and Rodriguez-Iturbe (1974). Essentially we simulate the covariance structure of the mesh, via the spectral distribution function for a given covariance model. The trend values ($\mu_{ij}$) are added at a later stage. We have modified the original algorithm to avoid excessive trigonometric calculation.

The algorithm is as follows:

**Algorithm : Harmonic average Method (HA)**

1) Set $\alpha_A$ (covariance range parameter)

2) set $W_m = \alpha_A \left[ \left( \frac{1}{1 - U_1} \right)^2 - 1 \right]^{1/2}$ where $U_1 \sim U(0,1)$

3) $i = 1, j = 1$

4) set $\phi_m = 2\pi U_2$

5) set $V_1 = 2U_3 - 1$

$V_2 = 2U_4 - 1$

$h_5 = V_2^2 + V_1^2$

if distance to $(V_1, V_2) < 1$ then

$e_{ij} = \left( \frac{2}{N} \right)^{1/2} \sum_{i=1}^{N} \cos (W_m(x_i \sin \gamma_m + y_j \cos \gamma_m) + \phi_m)$

where

$\sin \gamma_m = \frac{V_2}{\sqrt{V_2^2 + V_1^2}}$

$\cos \gamma_m = \frac{V_1}{\sqrt{V_2^2 + V_1^2}}$
\( x_i, y_j \) are mesh point coordinates

and \( \phi_m \sim U(0, 2\pi) \)

ELSE

restart 5)

6) form \( g_{ij} = \mu_{ij} + \sigma e_{ij} \)

where \( \mu_{ij} \) = trend values at point \((i, j)\)

\( \sigma^2 \) = process variance.

Note: \( U_1, U_2, U_3, U_4 \) are \( \sim U(0, 1) \).

Usually we rely on a large value of \( N \) to justify the approximation to a MVN(0, \( K_n \)). \( N \geq 50 \) will usually suffice. Note that this requires 200 random numbers per mesh point. In comparison the turning band method only requires the generation of one line followed by \( n^* \) rotations and projections. Ripley (1981) suggests that a small value of \( n^* \) suffices. However, others use much larger values (e.g. Brooker, (1985), \( n^* = 100 \)).

We have used the HA method in our simulations. We are aware that the turning band method may be more efficient. The HA method is easily programmed, however, and is relatively fast compared to Cholesky Decomposition.

C) Discrete Models

In general, it is simple to simulate from a discrete model if the cell probabilities are known, in which case, a general pmf generator based on the alias or table-look up method can be used (Kemp and Kemp (1987)). Our models use likelihoods based on a cell
probability $\varepsilon_\xi / \sum_{\xi=1}^P \varepsilon_\xi$. Hence after $\varepsilon_\xi$ is defined the normalization is a sum and thence the pmf is formed.

**Algorithm**

1) set up $\Phi$, $m$ (number in sample)

2) for all cells: set $\varepsilon_\xi(\Phi_i)$

3) form normalisation: $C = \sum_{\xi=1}^P \varepsilon_\xi(\Phi_i)$

4) generate by alias method using $P_\xi = \varepsilon_\xi(\Phi_i)/C$ as cell probabilities.

The alias method is available in the IMSL library (GGDA).

D) **BSP Models** (discrete)

The general principle used for continuous BSP models can be applied in the discrete case. However, the generation is usually simpler. We define

$$\varepsilon_\xi = \exp(g_\xi)$$

$g_\xi$ has a Spatial Gaussian prior and hence in a region system a realisation of $g_\xi$ will be MVN ($\mu_\xi; K_\xi$). Conditional on this realisation then we generate cell counts from $P_\xi = e^{g_\xi} / \sum_{\xi=1}^P e^{g_\xi}$.
Algorithm

1) Set up $\mathbf{B}; \alpha; \sigma; \mu_n = F_n \mathbf{B}$.

2) Generate $g_n \sim MVN(\mu_n; K_n)$

3) form $\varepsilon_\mathbf{x} = e g_\mathbf{x}$ for all $\mathbf{x}$

4) form $C = \sum_{\mathbf{x}=1}^{P} e g_\mathbf{x}$

5) generate, via alias method,

$$m \text{ samples from } P_\mathbf{x} = \varepsilon_\mathbf{x}/C.$$ 

Note: an unconditional sample could be generated by replacing 4) and 5) by a Poisson generator with $\lambda_\mathbf{x} = e g_\mathbf{x}$ for each region (e.g. IMSL: GGPON).

The above algorithm is not as complex as the continuous case, as the number of regions is usually $<< 1000$ and hence the Cholesky method can be used (IMSL: GGNSM).
Appendix III  Research Papers

Listed below are published and submitted research papers currently extant. The papers all relate to the current research topic:


*JAS, 15, 2, 225-234.*

Lawson, A (1988) Fitting the von Mises distribution on GLIM.

*JAS, 15, 2, 255-260.*

Lawson, A (1989a) The von Mises Distribution on GLIM.

*GLIM NEWSLETTER, 19, December 1989.*

Lawson, A (1989b) Contribution to "Cancer Near Nuclear Installations".

*JRSS, A, 152, 3, 374-375.*


Conditionally accepted by *Comm. in Stats., Theory & Methods.*


Conditionally accepted by *Stats in Med.*


Submitted to Biometrika.

Lawson, A (1990) GLIM and Normalising Constant Models in Spatial and Directional Data Analysis.

Accepted by *Computational Statistics and Data Analysis.*

Appendix IV  Score Tests for Peakedness

1) Weibull case (Continuous)

The continuous model likelihood with \( f(r) \) defined as a truncated Weibull (eq 12) yields log likelihoods \( \mathcal{L}_{cr}, \mathcal{L}_{c\phi} \) (eq 14, 15). These log likelihoods can be maximised separately for estimation purposes, as \( \Lambda(b_0(r_0)) \) is purely a function of \( \lambda \) and \( \delta \). A test for simple radial trend was mentioned in an earlier section, but tests for a peaked shape in the radial form have not been considered. The reliability literature contains examples of Weibull tests and Cox and Oakes (1984, p.44) derive a test for an ordinary Weibull shape parameter. Here we derive the score test for testing

\[ H_0 : \delta = 1 \quad \text{against} \quad H_1 : \delta > 1 \]

for the log likelihood

\[
\mathcal{L}_{cr} = n \sum r_i \lambda \delta + (\delta - 2) \lambda \sum r_i \delta - n \sum \frac{\lambda}{e^{\lambda r_0}} \delta
\]

hence, under \( H_0 \)

\[
U_0 = n + \sum r_i \delta - \lambda \sum r_i \delta - n \left[ \frac{\lambda}{e^{\lambda r_0}} \delta \right]
\]

where \( \lambda \) is the ML estimate under \( H_0 \)

i.e. the solution of

\[
\frac{1}{\lambda} - \frac{r_0}{e^{\lambda r_0} - 1} = \hat{r}
\]

where \( \hat{r} = \sum r_i/n \).
The observed information elements are, under $H_0$:

$$I_{\delta \delta} = n + \hat{\lambda} \sum r_i (\ln r_i)^2 + n \left[ \frac{\hat{\lambda}_0 (\ln r_0)^2}{e^{\hat{\lambda}_0} - 1} - \frac{(\hat{\lambda}_0 (\ln r_0)^2 e^{\hat{\lambda}_0}}{(e^{\hat{\lambda}_0} - 1)^2} \right]$$

$$I_{\delta \lambda} = \sum r_i \ln r_i + n \left[ \frac{r_0 \ln r_0}{e^{\hat{\lambda}_0} - 1} - \frac{\hat{\lambda}_0^2 \ln r_0 e^{\hat{\lambda}_0}}{(e^{\hat{\lambda}_0} - 1)^2} \right]$$

$$I_{\lambda \lambda} = n \hat{\lambda}^2 - n \left[ \frac{r_0^2 e^{\hat{\lambda}_0}}{(e^{\hat{\lambda}_0} - 1)^2} \right]$$

and the score test statistic is

$$W_\delta = U_0(\nu_{\delta \delta})^{1/2} \sim N(0,1)$$

where

$$\nu_{\delta \delta} = (I_{\delta \delta} - I_{\delta \lambda}^2/I_{\lambda \lambda})^{-1}.$$

An LR test can also be defined for this model but has a more complex form and is not given here.

2) **Power case (Discrete)**

We consider the discrete model likelihood (59), with

$$e_2 = E_2 \exp(\beta r_2 + \zeta \ln r_2).$$
This parameterisation has the advantage of providing a log-linear model, in \( r_x \) and \( \ln r_x \).

For testing, 

\[
H_0 : \zeta = 0 \quad \text{against} \quad H_1 : \zeta \neq 0
\]

we derive the score statistic \((W_\zeta)\), as follows:

\[
\mathcal{L} = \sum m_x \ln E_x + \beta \sum m_x r_x + \zeta \sum m_x \ln r_x
\]

\[
- (\sum m_x) \ln \sum E_x \exp(\beta r_x + \zeta \ln r_x)
\]

\[
\mathcal{L}_{r,H_0} = \sum m_x \ln r_x - m_t \frac{\sum \ln r_x E_x e^{\hat{\beta} r_x}}{\sum E_x e^{\hat{\beta} r_x}}
\]

where 

\[
m_t = \sum m_x
\]

\[
\hat{\beta} = \text{ML estimate of } \beta \text{ under } H_0
\]

(i.e. solution of \( \sum m_x r_x/m_t = E_{H_0}(r_x) \))

\[
\mathcal{L}_{\beta} = \sum m_x r_x - (\sum m_x) \frac{\sum r_x \hat{E}_x}{\sum \hat{E}_x}
\]

where \( \hat{E}_x = E_x e^{\hat{\beta} r_x} \).

Denoting \( \chi^2_{\phi_1\phi_2} = - 2 \ln \phi_1\phi_2 \), we have

\[
I_{\zeta|H_0} = m_t \left[ \frac{\sum r_x \ln r_x \hat{E}_x}{\sum \hat{E}_x} - \frac{(\sum \ln r_x \hat{E}_x)(\sum r_x \hat{E}_x)}{(\sum \hat{E}_x)^2} \right]
\]

\[
I_{\zeta\beta|H_0} = m_t \left[ \frac{\sum (\ln r_x)^2 \hat{E}_x}{\sum \hat{E}_x} - \left( \frac{\sum \ln r_x \hat{E}_x}{\sum \hat{E}_x} \right)^2 \right]
\]
The leading element of the observed inverse information matrix is

\[ I_{\beta\beta} = m_t \left[ \frac{\sum r_k^2 \hat{e}_k}{\sum \hat{e}_k} - \left( \frac{\sum r_k \hat{e}_k}{\sum \hat{e}_k} \right)^2 \right]. \]

From standard theory, \( W_\zeta \) has an asymptotic standard normal distribution.
Appendix V  Moments and other elements of the characterisation of the Interaction von Mises Distribution \( (\mathcal{M}(\mu_0, \kappa + \psi r)) \)

Define the joint pdf of \( R, \Phi \) as \( h(r, \phi) \) and

\[
\lambda(r) = \lambda(r, \phi) = f(r)g(\phi, r)/r \quad 0 < r < r_0
\]

\[
0 < \phi < 2\pi
\]

where \( f(r) \) and \( g(\phi, r) \) are suitable radial and radial-angular model functions and \( h(r, \phi) = \lambda(r, \phi)/\Lambda(b_0(r_0)) = \lambda/\Lambda \) (for short).

Define

\[
E(g) = \int_{b_0(r_0)}^{r_0} g\lambda(r)dr/\Lambda
\]

and

\[
I_{r_0}(g) = \int_0^{r_0} g f(r)dr
\]

where \( g \) is an arbitrary function. Hence, for \( g \), with the above von Mises distributional form the following moments are easily derived:

\[
\alpha_1 = E(\cos \phi) = \cos \mu_0 I_{r_0}(A(\kappa + \psi r))/\Lambda
\]

\[
\alpha_2 = E(\cos \phi) = \cos \mu_0 I_{r_0}(A(\kappa + \psi r))/\Lambda
\]

\[
\beta_1 = E(\sin \phi) = \sin \mu_0 I_{r_0}(A(\kappa + \psi r))/\Lambda
\]

\[
\beta_2 = E(\sin \phi) = \sin \mu_0 I_{r_0}(A(\kappa + \psi r))/\Lambda
\]

\[
\alpha_2 = E(\cos 2\phi) = 2\mu_0 I_{r_0}(1 - 2A(\kappa + \psi r)/(\kappa + \psi r))
\]

\[
\beta_2 = E(\sin 2\phi) = 2\mu_0 I_{r_0}(1 - 2A(\kappa + \psi r)/(\kappa + \psi r))
\]

\[
\alpha_{12} = E(\cos^2 \phi) = \frac{1}{2} (1 + \alpha_2)
\]
\[ \beta_{12} = \mathbb{E}(\sin^2 \phi) = \frac{1}{2} (1 - \alpha_2) \]

\[ \text{var} \left( \cos \phi \right) = \frac{1}{2} (1 + \alpha_2) - \cos^2 \mu_0 \mathbb{I}_0^2 (A(\kappa + \psi r))/\Lambda^2 \]

\[ \text{var} \left( \sin \phi \right) = \frac{1}{2} (1 - \alpha_2) - \sin^2 \mu_0 \mathbb{I}_0^2 (A(\kappa + \psi r))/\Lambda^2. \]

Define the following measures based on a random sample of size \( n \) from \( h(r, \phi) \):

\[
\begin{align*}
C &= \sum \cos \phi_i; \quad S = \sum \sin \phi_i; \quad RC = \sum r_i \cos \phi_i; \\
RS &= \sum r_i \sin \phi_i; \quad R_1 = \sqrt{C^2 + S^2}; \quad R_2 = \sqrt{RC^2 + RS^2}
\end{align*}
\]

\[
\begin{align*}
C^2 &= \sum \cos^2 \phi_i + 2 \sum_{i < j} \cos \phi_i \cos \phi_j \\
S^2 &= \sum \sin^2 \phi_i + 2 \sum_{i < j} \sin \phi_i \sin \phi_j
\end{align*}
\]

\[
\begin{align*}
E(C^2) &= \frac{n}{2} (1 + \alpha_2) + n(n-1)\alpha_1^2 \\
E(S^2) &= \frac{n}{2} (1-\alpha_2) + n(n-1)\beta_1^2
\end{align*}
\]

\[
\begin{align*}
E(C) &= n \cos \mu_0 \mathbb{I}_0 (A(\kappa + \psi r))/\Lambda \\
E(S) &= n \sin \mu_0 \mathbb{I}_0 (A(\kappa + \psi r))/\Lambda \\
E(RC) &= n \cos \mu_0 \mathbb{I}_0 (rA(\kappa + \psi r))/\Lambda \\
E(RS) &= n \sin \mu_0 \mathbb{I}_0 (rA(\kappa + \psi r))/\Lambda
\end{align*}
\]

also note that:

\[
\alpha_{R12} = E(r^2 \cos^2 \phi) = \mathbb{I}_0 \left( \frac{r^2}{2} (1 + \delta_1 (\kappa + \psi r)) \right)/\Lambda
\]
\[ \beta_{R12} = E(r^2 \sin^2 \phi) = \text{Ir}_0 \left(\frac{r^2}{2} (1 - \delta_1(\kappa + \psi \tau)) \right)/\Lambda \]

where \( \delta_1(\kappa + \psi \tau) = \cos 2\mu_0(1 - 2A(\kappa + \psi \tau)/(\kappa + \psi \tau)) \)

\[ E(R_1^2) = n + n(n-1) \text{Ir}_0^2 (A(\kappa + \psi \tau))/\Lambda^2 \]
\[ E(R_2^2) = n \text{Ir}_0(r^2)/\Lambda + n(n-1)\text{Ir}_0^2[rA(\kappa + \psi \tau)]/\Lambda^2 \]
\[ \text{cov} (\cos \phi, \sin \phi) = \frac{1}{2} \beta_2 - \alpha_1 \beta_1 \]
\[ \text{cov} (C,S) = n \left( \frac{1}{2} \beta_2 - \alpha_1 \beta_1 \right) \]
\[ \text{var}(RC) = n(\alpha_{R12} - \alpha_{R1}^2) \]
\[ \text{var}(RS) = n(\beta_{R12} - \beta_{R1}^2) \]
\[ \text{cov}(RC,RS) = n \left\{ \text{Ir}_0 \left[ \frac{r^2}{2} \Phi_1 (\kappa + \psi \tau) \right] - \alpha_{R1} \beta_{R1} \right\} \]

where

\[ \Phi_1(\kappa + \psi \tau) = \sin 2\mu_0(1 - 2A(\kappa + \psi \tau)/(\kappa + \psi \tau)) \]
Appendix VI  Derivation of Score and Wald Tests for Continuous and Discrete Models

a) Hepp Models

The score test for \( r \cdot \phi \) interaction is derived as follows:

The log likelihood from (15) is

\[
\ell = \sum (\kappa + \psi r_i) \cos (\phi_i - \mu_0) - \sum n_i I_0 (\kappa + \psi r_i) - \text{const}
\]

and

\[
\ell_{\kappa} = \sum \cos(\phi_i - \mu_0) - \sum A(\kappa + \psi r_i)
\]

\[
\ell_{\psi} = \sum r_i \cos(\phi_i - \mu_0) - \sum r_i A(\kappa + \psi r_i)
\]

\[
\ell_{\mu_0} = \psi \sum r_i \sin(\phi_i - \mu_0) + \kappa \sum \sin(\phi_i - \mu_0)
\]

where \( A(x) = I_1(x)/I_0(x) \)

and

\( I_n(x) \) is the modified Bessel function of nth order.

Hence

\[
U_{0|H_0} = \sum r_i \cos (\phi_i - \hat{\mu}_0) - \sum r_i A(\kappa).
\]

The observed information elements under \( H_0 \) are

\[
I_{\psi \psi} = \sum r_i^2 (1 - A^2(\kappa)) - A(\kappa)/\kappa = \sum r_i^2 A'(\kappa)
\]

\[
I_{\psi \mu_0} = - \sum r_i \sin (\phi_i - \mu_0)
\]

\[
I_{\psi \kappa} = \sum r_i A'(\kappa)
\]

\[
I_{\kappa \mu_0} = - \sum \sin (\phi_i - \mu_0)
\]

\[
I_{\kappa \kappa} = n A'(\kappa)
\]
\[ I_{\mu_0 \mu_0} = \kappa \Sigma \cos (\phi_i - \mu_0) \]

\[
I = \begin{bmatrix}
\Sigma r_i^2 A'(\kappa) & \Sigma r_i A'(\kappa) & -\Sigma r_i \sin (\phi_i - \mu_0) \\
\Sigma r_i A'(\kappa) & nA'(\kappa) & -\Sigma \sin (\phi_i - \mu_0) \\
-\Sigma r_i \sin (\phi_i - \mu_0) & -\Sigma \sin (\phi_i - \mu_0) & \kappa \Sigma \cos (\phi_i - \mu_0)
\end{bmatrix}.
\]

Note that under \( H_0, \hat{\mu}_0 = \bar{X}_0 \) and hence the expected information is estimated by

\[
I = \begin{bmatrix}
A'(\kappa) \Sigma r_i^2 & A'(\kappa) \Sigma r_i & 0 \\
A'(\kappa) \Sigma r_i & nA'(\kappa) & 0 \\
0 & 0 & knA(\kappa)
\end{bmatrix}
\]

and the leading element of the inverse matrix \((I^\psi \psi)\) is

\[
I^\psi \psi = I_{\psi \psi} - I_{\psi \Delta}^T I^{-1}_{\Delta \Delta} I_{\psi \Delta}
\]

where

\[
I_{\psi \Delta} = \begin{bmatrix}
A'(\kappa) \Sigma r_i \\
0
\end{bmatrix}, \quad I^{-1}_{\Delta \Delta} = \begin{bmatrix}
1/nA'(\kappa) & 0 \\
0 & 1/knA(\kappa)
\end{bmatrix}.
\]

Hence, the score test (eq 25) is equivalent to

\[
W^\psi = \frac{\Sigma r_i \cos (\phi_i - \bar{X}_0) - A'(\kappa) \Sigma r_i}{\sqrt{A'(\kappa) \left( \Sigma r_i^2 - \frac{(\Sigma r_i)^2}{n} \right)}} \sim N(0,1)
\]

i.e.

\[
W^\psi = \sqrt{WS}.
\]

Note that the Wald test for \( \psi \)-interaction can be derived as
where $\hat{\phi}$ denotes ML estimation under the full model. Now,

$$
\text{I}\text{V}\text{I} \mathcal{V} \mathcal{I} (\hat{\phi}, \hat{\psi}) = \left[ \sum_i r_i^2 A'(\hat{\kappa} + \hat{\psi} r_i) - \frac{(\sum r_i A'(\hat{\kappa} + \hat{\psi} r_i))^2}{\sum A'(\hat{\kappa} + \hat{\psi} r_i)} \right].
$$

b) **Discrete Models**

i) **Angular Concentration**

The score test for $H_0 : \kappa = 0$ against $H_1 : \kappa > 0$ in the discrete model was given in equation (76), and is based on the log likelihood (67)

$$
\ell = \text{const} + \kappa \sum_{\ell} m_{\ell} \cos (\phi_{\ell} - \mu_0)
$$

$$
- m_t \ln \sum_{\ell} E(\ell) \exp(\kappa \cos (\phi_{\ell} - \mu_0))
$$

and

$$
\ell' = \sum_{\ell} m_{\ell} \cos (\phi_{\ell} - \mu_0) - m_t \frac{\sum \cos (\phi_{\ell} - \mu_0) \epsilon_{\ell}}{\sum \epsilon_{\ell}}
$$

$$
\ell'_{\mu_0} = \kappa \sum m_{\ell} \sin (\phi_{\ell} - \mu_0) - m_t \frac{\kappa \sum \sin (\phi_{\ell} - \mu_0) \epsilon_{\ell}}{\sum \epsilon_{\ell}}
$$

where $\epsilon_{\ell} = E(\ell) \exp(\kappa \cos (\phi_{\ell} - \mu_0))$.

Now

$$
U_0 = \ell'_{\mu_0} = \sum m_{\ell} \cos (\phi_{\ell} - \mu_0) - m_t \frac{\sum \cos (\phi_{\ell} - \mu_0) E(\ell)}{\sum E(\ell)}
$$
and $\hat{\mu}_0$ is a consistent estimate under $H_0$

$$I_{kk} = I_{kk} - I_{k\mu_0}^2/\mu_0\mu_0$$

where

$$I_{k\mu_0|H_0} = m_t \left\{ \frac{\Sigma \cos^2(\phi_k - \hat{\mu}_0)E_k}{\Sigma E_k} - \left( \frac{\Sigma \cos(\phi_k - \hat{\mu}_0)E_k}{\Sigma E_k} \right)^2 \right\}$$

$$I_{\mu_0\mu_0|H_0} = -\Sigma m_g \sin(\phi_k - \hat{\mu}_0) + m_t \frac{\Sigma \sin(\phi_k - \hat{\mu}_0)E_k}{\Sigma E_k}$$

$$I_{\mu_0\mu_0|H_0} = 0.$$ 

Under expectation all the sin moments disappear and so $I_{k\mu_0|H_0} = 0$ also.

Hence

$$I_{kk} = m_t \left\{ \frac{\Sigma \cos^2(\phi_k - \hat{\mu}_0)E_k}{\Sigma E_k} - \left( \frac{\Sigma \cos(\phi_k - \hat{\mu}_0)E_k}{\Sigma E_k} \right)^2 \right\}$$

and if we adjust the $E_k$, so that $m_t = \Sigma E_k$ then

$$T_h = \frac{\Sigma \cos(\phi_k - \hat{\mu}_0)[m_k - E_k]}{\sqrt{\left( \Sigma \cos^2(\phi_k - \hat{\mu}_0)E_k - (\Sigma \cos(\phi_k - \hat{\mu}_0)E_k)^2/m_t \right)}}$$

A consistent estimate of $\mu_0$ under $H_0$ could be

$$\hat{\mu}_0 = \tan^{-1}\left( \frac{NS}{NC} \right)$$
where \( N_S = \sum_{g} m_g \sin \phi_g \), \( N_C = \sum_{g} m_g \cos \phi_g \). Note that it is also possible to extend this model to incorporate a radial hazard (Lawson, 1990c). Gart and Tarone (1983) note that in the case of nuisance parameters, conditioning of sufficient statistics leads to score tests which are UMPU asymptotically.

ii) Angular-Radial Interaction

The score test for \( H_0 : \psi = 0 \) against \( H_1 : \psi > 0 \) in the discrete model was given in equations (77-79). We assume that

\[
E_r = E_r \exp\left[\left(\kappa + \psi r_x\right) \cos (\phi_x - \mu_0)\right]
\]

and use

\[
x = \text{const} + \kappa \sum_{g} m_g \cos(\phi_g - \mu_0) + \psi \sum_{g} r_g m_g \cos(\phi_g - \mu_0)
\]

\[- m_t \sum_{g} E_r \mu_0.
\]

Hence,

\[
x'_{\psi} = \sum_{g} m_g r_g \cos(\phi_g - \mu_0) - m_t \frac{\sum_{g} r_g \cos(\phi_g - \mu_0)E_r}{\sum E_r}
\]

\[
x'_{\kappa} = \sum_{g} m_g \cos(\phi_g - \mu_0) - m_t \frac{\sum \cos(\phi_g - \mu_0)E_r}{\sum E_r}
\]

\[
x'_{\mu_0} = \kappa \sum_{g} m_g \sin(\phi_g - \mu_0) + \psi \sum_{g} m_g r_g \sin(\phi_g - \mu_0)
\]
Denote 
\[ e_{0\xi} = E_\xi \exp\left(\hat{\kappa} \cos(\phi_\xi - \bar{X}_0)\right) \]

where \( \bar{X}_0 \) is defined previously

and
\[ R = m_y / \sum e_{0\xi} \]

The score test is
\[ WS = U_0.1^\psi \psi^{-1/2} \]

where
\[ I^\psi = I^{\psi \psi} - I^{\psi \lambda} \lambda^{-1} I^{\psi \lambda} \]

and
\[ I^{\psi \lambda} = \begin{bmatrix} I_{\psi \kappa} \\ I_{\psi \mu_0} \end{bmatrix} \quad \text{and} \quad I^{\lambda \lambda} = \begin{bmatrix} I_{\kappa \kappa} & I_{\mu_0 \kappa} \\ I_{\kappa \mu_0} & I_{\mu_0 \mu_0} \end{bmatrix} \]

and
\[ I^{\psi \kappa} = R \left\{ \sum r_\xi c_\xi^2 e_{0\xi} - \frac{1}{\Sigma \varepsilon_{0\xi}} (\Sigma c_\xi e_{0\xi})(\Sigma r_\xi c_\xi e_{0\xi}) \right\} \]

\[ I_{\kappa \kappa} = R \left\{ \sum c_\xi^2 e_{0\xi} - \frac{1}{\Sigma \varepsilon_{0\xi}} (\Sigma c_\xi e_{0\xi})^2 \right\} \]

\[ I_{\psi \mu_0} = - \sum m_\xi r_\xi s_\xi \]
\[ + R \left\{ \sum r_\xi s_\xi e_{0\xi} + \hat{\kappa} (\Sigma r_\xi c_\xi s_\xi e_{0\xi}) - \frac{1}{\Sigma \varepsilon_{0\xi}} (\Sigma s_\xi e_{0\xi})(\Sigma r_\xi c_\xi e_{0\xi}) \right\} \]
\[ I_{\kappa \mu_0} = -\Sigma m_\lambda s_\lambda \]
\[ + R\left\{ \Sigma s_\lambda e_{0\lambda} + \hat{k}\left( \Sigma c_\lambda s_\lambda e_{0\lambda} - \frac{1}{\Sigma e_{0\lambda}}(\Sigma s_\lambda e_{0\lambda})(\Sigma c_\lambda e_{0\lambda}) \right) \right\} \]
\[ I_{\mu_0 \mu_0} = \hat{k} \Sigma m_\lambda c_\lambda \]
\[ + R\left\{ \hat{k}^2\left( \Sigma s_\lambda^2 e_{0\lambda} - \frac{1}{\Sigma e_{0\lambda}}(\Sigma s_\lambda e_{0\lambda})^2 \right) - \hat{k} \Sigma c_\lambda e_{0\lambda} \right\} \]
\[ I_{\psi \psi} = R\left\{ \Sigma r_\lambda^2 c_\lambda^2 e_{0\lambda} - \frac{1}{\Sigma e_{0\lambda}}(\Sigma r_\lambda c_\lambda e_{0\lambda})^2 \right\} \]

and

\[ c_\lambda = \cos(\phi_\lambda - \bar{X}_0) \]
\[ s_\lambda = \sin(\phi_\lambda - \bar{X}_0). \]
Appendix VII  GLIM macros for Special Models

Listed below are the GLIM macros used to fit normalising constant models and calculate parameter transformations.

1) Macro INT

This macro is used to set up integration weights for one dimensional HEPP models. The macro has a single input argument: vector %1 : the vector of data points. In addition, scalar %z must be set to the number of dummy points. The macro returns the weights in vector %w and the sorted data and dummy points in vector %ths. The trapezoid rule is used, and the macro is currently set for end point %m = 360.

```
$macro int
$del w s k the nt x j h ths ns nw a$
$var %z x$
$cal %m=360$
$cal x=%cu(360/(%z+1))$
$cal %q=%z+%nu$
$var %q the nt a b j k w ns nw$
$cal nt=1 : nt=%cu(nt)$
$cal j=%le(nt,%nu) : nt=j$
$cal k=%ne(j,1) : k=k*%cu(k)$
$cal j-j*%cu(j)$
$cal the=%l(j)$
$cal the=the*%ne(j,0)+x(k)$
$sort ns nt the$
$sort ths the the$
$cal a(1)=0.5*(ths(2)-ths(1))$
$cal a(%q)=0.5*(%m-ths(%q))$
$cal %e=%q-2 $var %e s$
$cal s=1 : s(1)=2 : s=%cu(s)$
$cal a(s)=0.5*(ths(s+1)-ths(s-1))$
$del s k the nt x j b$endmacro
```

2) Macro GRUN

This macro calculates summary statistics and the $\kappa$ estimate for a Fisher distribution. The data must be $\phi$: vector ph, $\theta$: vector th. Macro INT is called, after rotation to the mean axis. In addition, macros for calculation of inverse cos and tan functions are called (TANMI, ARCOS). These must be user supplied.
$\text{mac grun}$
$\text{cal } cth=\text{th}+\pi/2.$
$\text{cal } pth=\text{ph}+\pi/2.$
$\text{cal } ct=\sin(cth)$
$\text{cal } cp=\sin(pth)$
$\text{cal } st=\sin(th)$
$\text{cal } sp=\sin(ph)$
$\text{cal } l=st*cp$
$\text{cal } m=st*sp : n=ct$
$\text{cal } a=acu(l) : b=acu(m) : c=acu(n)$
$\text{cal } r=\sqrt{a^2+b^2+c^2}$
$\text{pri } \text{resultant(R)}: \ r$
$\text{cal } p=r/\nu$\text{print } 'rbar:' $ p$
$\text{pri } R_x,R_y,R_z : ' a $ b $ c$
$\text{print } 'l,m,n means'$
$\text{cal } p=a/r$ $\text{print } a$
$\text{cal } p=b/r$ $\text{print } b$
$\text{cal } p=c/r$ $\text{print } c$
$\text{cal } d=a/c/r$
$\text{use tanml } b \ a \ t \ v \ q \ u$
$\text{use arcos } d \ m \ h \ i \ j$
$\text{pri } \text{polar means(xbar,ybar)} : ' m $ v$
$\text{cal } x=\sin(m+\pi/2.)$
$\text{cal } y=\sin(m)$
$\text{cal } z=\sin(v+\pi/2.)$
$\text{cal } w=\sin(v)$
$\text{cal } l=2x*z*l+2x*w*m-2y*n$
$\text{cal } m=2w*l+2z*m$
$\text{cal } n=2y*z*l+2y*w*m+2x*n$
$\text{use arcos } n \ m \ h \ teml$
$\text{cal } thpr=thpr*57.295795$
$\text{print } 'thpr' /thpr$
$\text{cal } z=200$
$\text{cal } m=180$
$\text{use int } thpr$
$\text{units } q$
$\text{cal } w=a$
$\text{cal } nw=ns/w$
$\text{wei } w$
$\text{yvar } nw$ err p
$\text{cal } c=0.01745$
$\text{cal } ths=ths*c$
$\text{cal } lo=log(\sin(ths)/(2*\pi))$
$\text{pri } 'lo'$\text{print } lo$
$\text{off } lo$
$\text{cal } c=\sin(ths+\pi/2)$
$\text{fit } c$
$\text{endm}$

3) Macros VM2P, VM3P

These macros provide parameter transformations and their covariance matrices, as described in appendix VIII.

a) VM2P
The arguments to this macro are the position of the \( \cos \phi \) parameter (%1) and \( \sin \phi \) parameter (%2) in model formula. The macro calculates \( \kappa \) and \( \mu_0 \) estimates and their covariance matrix.

\[
\begin{align*}
&\text{The arguments to this macro are the position of the } \cos \phi \text{ parameter } (%1) \text{ and } \sin \phi \text{ parameter } (%2) \text{ in model formula. The macro calculates } \kappa \text{ and } \mu_0 \text{ estimates and their covariance matrix.}
\end{align*}
\]

b) **VM3P**

The arguments to this macro are the positions of the \( \cos \phi \) (%1), \( \sin \phi \) (%2), \( \rho \cos \phi \) (%3), and \( \rho \sin \phi \) (%4) parameters in the model formula. The macro calculates \( \kappa, \psi, \mu_0 \) estimates and their covariance matrix. An external macro COVLOC is called to extract the appropriate covariance from \%VC.

\[
\begin{align*}
&\text{The arguments to this macro are the positions of the } \cos \phi \text{ (%1), } \sin \phi \text{ (%2), } \rho \cos \phi \text{ (%3), and } \rho \sin \phi \text{ (%4) parameters in the model formula. The macro calculates } \kappa, \psi, \mu_0 \text{ estimates and their covariance matrix. An external macro COVLOC is called to extract the appropriate covariance from } \%VC.}
\end{align*}
\]
Seal
\[ x = \frac{2 * a * b * e + b * e * f}{k * k} \]

$\text{pri} ' \text{var(k)} :' \ x$

Seal
\[ x = \frac{2 * c * d * e + c * e * g + d * d * h}{p * p} \]

$\text{pri} ' \text{var(psi)} :' \ x$

Seal
\[ x = \frac{-t * e - t * g + f + h}{r} \]

$\text{pri} ' \text{var(mu)} :' \ x$

Seal
\[ x = \frac{-a * e * f + e * f + a * e + a * e}{k * r} \]

$\text{pri} ' \text{cov(k,m)} :' \ x$

Seal
\[ x = \frac{a * c * b + a * d * m + b * e * n + b * d * o}{k * p} \]

$\text{pri} ' \text{cov(k,psi)} :' \ x$

Seal
\[ x = \frac{-c * t * j + d * g + d * t * m + d * o}{p * r} \]

$\text{pri} ' \text{cov(psi,pmu)} :' \ x$

$\text{ends} !$
Appendix VIII  GLIM parameter estimates and variance-covariance matrix elements for the $K, \theta, \mu_0$ Interaction model

On GLIM, we can use a tessellation weight method to fit the $M(K+\psi r, \mu_0)$ model. We proceed by fitting terms: $\cos \phi, \sin \phi, r \cdot \cos \phi, r \cdot \sin \phi$ with parameter estimates $\hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3, \hat{\beta}_4$ respectively.

By a straightforward parameter transformation we can derive the ML estimates of $K, \psi, \mu_0$ by invariance. Thus,

$$\hat{K} = \sqrt{\hat{\beta}_1^2 + \hat{\beta}_2^2}, \quad \hat{\psi} = \sqrt{\hat{\beta}_3^2 + \hat{\beta}_4^2}$$

$$\hat{\mu}_0 = \tan^{-1}\left\{\frac{\hat{\beta}_2 + \hat{\beta}_4}{\hat{\beta}_1 + \hat{\beta}_3}\right\}.$$

For simplicity of notation we now replace $\hat{\beta}$ by $\beta$. Hence any reference to a $\beta$ parameter implies the GLIM parameter estimate. In addition, $\text{var}(\beta_i)$ and $\text{cov}(\beta_i, \beta_j)$ implies the GLIM parameter variances and covariances.

We employ the delta method to give $O(n^{-3/2})$ approximations to the variances and covariances of $\hat{K}, \hat{\psi}$ and $\hat{\mu}_0$. Define,

$$u = \frac{\beta_2 + \beta_4}{\beta_1 + \beta_3}$$

$$\text{var}(\hat{K}) = \left(2\beta_1\beta_2 \text{cov}(\beta_1, \beta_2) + \beta_1^2 \text{var}(\beta_1) + \beta_2^2 \text{var}(\beta_2)\right)/\hat{K}^2$$

$$\text{var}(\hat{\psi}) = \left(2\beta_3\beta_4 \text{cov}(\beta_3, \beta_4) + \beta_3^2 \text{var}(\beta_3) + \beta_4^2 \text{var}(\beta_4)\right)/\hat{\psi}^2$$

$$\text{var}(\hat{\mu}_0) = a\{- u \text{var}(\beta_1) - u \text{var}(\beta_3) + \text{var}(\beta_2) + \text{var}(\beta_4)\}$$

$$+ b\{- u \text{cov}(\beta_1, \beta_2) + u^2 \text{cov}(\beta_1, \beta_3) - \text{cov}(\beta_1, \beta_4)\}$$
where
\[ a = \left(1 + u^2 \right) \left(\beta_1 + \beta_3\right)^{-1} \]

and
\[ b = 2 \left(1 + u^2 \right) \left(\beta_1 + \beta_3\right)^{-2} \]

\[
\text{cov}(\hat{\kappa}, \hat{\mu}_0) = a \cdot \hat{\kappa}^{-1} \left\{ - \beta_1 u \text{var} (\beta_1) + \beta_2 \text{var} (\beta_2) + \beta_1 \text{cov} (\beta_1, \beta_2) - \beta_1 u \text{cov}(\beta_1, \beta_3) - \beta_2 u \text{cov}(\beta_2, \beta_1) + \beta_1 \text{cov}(\beta_1, \beta_4) - \beta_2 u \text{cov}(\beta_2, \beta_3) + \beta_2 \text{cov}(\beta_2, \beta_4) \right\}
\]

\[
\text{cov}(\hat{\kappa}, \hat{\psi}) = (\hat{\kappa} \cdot \hat{\psi})^{-1} \left\{ \beta_1 \beta_3 \text{cov}(\beta_1, \beta_3) + \beta_1 \beta_4 \text{cov}(\beta_1, \beta_4) + \beta_2 \beta_3 \text{cov}(\beta_2, \beta_3) + \beta_2 \beta_4 \text{cov}(\beta_2, \beta_4) \right\}
\]

\[
\text{cov}(\hat{\psi}, \hat{\mu}_0) = a \cdot \hat{\psi}^{-1} \left\{ - \beta_3 u \text{var} (\beta_3) + \beta_4 \text{var} (\beta_4) - \beta_4 u \text{cov}(\beta_4, \beta_1) + \beta_4 \text{cov}(\beta_4, \beta_2) - \beta_4 u \text{cov}(\beta_4, \beta_3) + \beta_3 \text{cov}(\beta_3, \beta_2) + \beta_3 u \text{cov}(\beta_3, \beta_4) - \beta_3 \text{cov}(\beta_3, \beta_1) \right\}
\]

GLIM macros for the calculation of the above estimates are given in Appendix VII.
Appendix IX  
**Observed Information Matrix for the Case Control Score**

**Test for Interaction**

We, here, give the elements of the observed information matrix for the angular-linear interaction model in the presence of a case-control process, under \( H_0 : \psi = 0 \).

We define

\[
\tau = \hat{\tau}(\lambda), \quad T = \int t\lambda_0 \, d\lambda
\]

\[
\lambda_0 = \lambda(r, \phi)_{H_0}
\]

\[
C = \cos(\phi - \hat{\mu}_0)
\]

\[
S = \sin(\phi - \hat{\mu}_0)
\]

\[
I_{\psi\psi} = n \left\{ \frac{\int \tau r^2 C^2 \lambda_0 d\lambda}{T} - \left( \frac{\int \tau C \lambda_0 d\lambda}{T^2} \right)^2 \right\}
\]

\[
I_{\psi\mu_0} = - \Sigma r \sin(\phi_i - \hat{\mu}_0)
\]

\[
+ n \left\{ \frac{1}{T} \int \tau r S \lambda_0 d\lambda + \frac{\hat{\kappa}}{T} \int \tau r C S \lambda_0 d\lambda - \frac{\hat{\kappa}}{T^2} \left( \int \tau r C \lambda_0 d\lambda \right) \left( \int \tau r S \lambda_0 d\lambda \right) \right\}
\]

\[
I_{\psi\kappa} = n \left\{ \frac{1}{T} \int \tau r C^2 \lambda_0 d\lambda - \frac{1}{T^2} \left( \int \tau r C \lambda_0 d\lambda \right) \left( \int \tau r C \lambda_0 d\lambda \right) \right\}
\]

\[
I_{\kappa\kappa} = n \left\{ \frac{1}{T} \int \tau C^2 \lambda_0 d\lambda - \frac{1}{T^2} \left( \int \tau C \lambda_0 d\lambda \right)^2 \right\}
\]
\[ I_{\mu_0} = - \Sigma \sin (\phi_i - \hat{\mu}_0) \]
\[ + n \left\{ \frac{1}{T} \left( \int \tau S \lambda_0 d\xi \right) + \frac{\hat{\kappa}}{T} \int \tau C \lambda_0 d\xi - \frac{\kappa}{T^2} \left( \int \tau C \lambda_0 d\xi \right) \left( \int \tau S \lambda_0 d\xi \right) \right\} \]

\[ I_{\mu_0 \mu_0} = \kappa \Sigma \cos (\phi_i - \hat{\mu}_0) \]
\[ + n \hat{\kappa} \left\{ - \frac{1}{T} \left( \int \tau C \lambda_0 d\xi \right) + \frac{\kappa}{T} \left( \int \tau S^2 \lambda_0 d\xi \right) - \frac{\kappa}{T^2} \left( \int \tau S \lambda_0 d\xi \right)^2 \right\}. \]
Appendix X  Asymptotic Sampling Distributions for test Statistics

We list below the asymptotic sampling distributions for the main test statistics used in this study.

1) Score, LR and Wald tests

These test specified for a(1 x k) parameter vector \( \mathbf{a} \), say, with \( H_0: \mathbf{a} = 0 \) against \( H_1: \alpha \neq 0 \) are as \( \chi^2_k \).

2) Other Tests

Rayleigh Test (R) : \( 2nR^2 \sim \chi^2_2 \)

Moore's test (M) : given in Moore (1980)
Watson's \( U^2 \) test (U2) : see Mardia (1972, p182)
Mardia rank correlation test (RM) : \( \chi^2_2 \)
Angular-linear correlation (\( r^2 \)) : \( -n \ln(1 - r^2) \sim \chi^2_2 \)
Fisher-Lee test (FL) : incomplete statistic in standardised form : \( N(0,1) \)
Radial Uniformity test (L) : \( N(0,1) \)
Spacings test (S) : \( N(0,1) \)
Discrete angular-linear correlation (\( R^2_\theta \)) (Bootstrapped) : \( N(0,1) \)
Appendix XI  Data Sets

The data sets used in this study are listed below. For point data, the data is usually listed as (r,\(\phi\)) coordinates with \(\phi\) in degrees, unless otherwise stated. For count data, the grid reference (E,N) of each ed. followed by the disease count and age \(\times\) sex population numbers are given.

A)  Armadale

i)  respiratory cancer deaths (n = 49)

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>17.5</td>
<td>7</td>
<td>27.0</td>
<td>186</td>
</tr>
<tr>
<td>13.5</td>
<td>18</td>
<td>30.5</td>
<td>187</td>
</tr>
<tr>
<td>16.5</td>
<td>18</td>
<td>33.0</td>
<td>187</td>
</tr>
<tr>
<td>15.0</td>
<td>28</td>
<td>15.5</td>
<td>192</td>
</tr>
<tr>
<td>25.0</td>
<td>61</td>
<td>18.0</td>
<td>193</td>
</tr>
<tr>
<td>15.5</td>
<td>99</td>
<td>20.0</td>
<td>198</td>
</tr>
<tr>
<td>30.0</td>
<td>100</td>
<td>18.0</td>
<td>202</td>
</tr>
<tr>
<td>50.5</td>
<td>105</td>
<td>20.0</td>
<td>212</td>
</tr>
<tr>
<td>32.5</td>
<td>109</td>
<td>23.0</td>
<td>219</td>
</tr>
<tr>
<td>55.5</td>
<td>110</td>
<td>21.5</td>
<td>219</td>
</tr>
<tr>
<td>53.5</td>
<td>115</td>
<td>20.5</td>
<td>222</td>
</tr>
<tr>
<td>16.0</td>
<td>118</td>
<td>27.5</td>
<td>228</td>
</tr>
<tr>
<td>49.5</td>
<td>130</td>
<td>34.5</td>
<td>228</td>
</tr>
<tr>
<td>44.5</td>
<td>136</td>
<td>20.0</td>
<td>229</td>
</tr>
<tr>
<td>32.0</td>
<td>138</td>
<td>45.0</td>
<td>229</td>
</tr>
<tr>
<td>12.5</td>
<td>143</td>
<td>25.0</td>
<td>230</td>
</tr>
<tr>
<td>18.0</td>
<td>143</td>
<td>18.0</td>
<td>230</td>
</tr>
<tr>
<td>44.0</td>
<td>148</td>
<td>15.5</td>
<td>230</td>
</tr>
<tr>
<td>22.0</td>
<td>162</td>
<td>23.5</td>
<td>233</td>
</tr>
<tr>
<td>53.0</td>
<td>166</td>
<td>47.5</td>
<td>234</td>
</tr>
<tr>
<td>49.5</td>
<td>167</td>
<td>11.0</td>
<td>240</td>
</tr>
<tr>
<td>56.0</td>
<td>170</td>
<td>39.0</td>
<td>240</td>
</tr>
<tr>
<td>57.0</td>
<td>172</td>
<td>21.0</td>
<td>242</td>
</tr>
<tr>
<td>59.0</td>
<td>178</td>
<td>12.5</td>
<td>260</td>
</tr>
<tr>
<td>32.0</td>
<td>182</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cardiovascular deaths (n = 153)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.39120</td>
<td>12.00000</td>
<td>52.33008</td>
<td>135.5000</td>
</tr>
<tr>
<td>18.46944</td>
<td>12.00000</td>
<td>39.50408</td>
<td>135.5000</td>
</tr>
<tr>
<td>16.41728</td>
<td>18.00000</td>
<td>51.30400</td>
<td>137.0000</td>
</tr>
<tr>
<td>10.77384</td>
<td>42.50000</td>
<td>44.63448</td>
<td>137.0000</td>
</tr>
<tr>
<td>12.31296</td>
<td>44.00000</td>
<td>38.99104</td>
<td>137.0000</td>
</tr>
<tr>
<td>11.56430</td>
<td>49.50000</td>
<td>26.67808</td>
<td>138.0000</td>
</tr>
<tr>
<td>10.26080</td>
<td>61.00000</td>
<td>51.30400</td>
<td>138.5000</td>
</tr>
<tr>
<td>14.36512</td>
<td>61.00000</td>
<td>27.19112</td>
<td>138.5000</td>
</tr>
<tr>
<td>11.28688</td>
<td>65.00000</td>
<td>23.08680</td>
<td>139.0000</td>
</tr>
<tr>
<td>12.82600</td>
<td>66.50000</td>
<td>35.91280</td>
<td>140.0000</td>
</tr>
<tr>
<td>14.87816</td>
<td>70.50000</td>
<td>21.54768</td>
<td>142.0000</td>
</tr>
<tr>
<td>20.00856</td>
<td>72.00000</td>
<td>31.80848</td>
<td>142.5000</td>
</tr>
<tr>
<td>20.00856</td>
<td>81.00000</td>
<td>41.55624</td>
<td>142.5000</td>
</tr>
<tr>
<td>13.33904</td>
<td>82.00000</td>
<td>25.65200</td>
<td>151.0000</td>
</tr>
<tr>
<td>15.39120</td>
<td>90.00000</td>
<td>45.66056</td>
<td>153.5000</td>
</tr>
<tr>
<td>27.70416</td>
<td>90.00000</td>
<td>43.60840</td>
<td>156.5000</td>
</tr>
<tr>
<td>19.49552</td>
<td>100.000</td>
<td>38.99104</td>
<td>157.5000</td>
</tr>
<tr>
<td>23.08680</td>
<td>104.000</td>
<td>31.80848</td>
<td>157.5000</td>
</tr>
<tr>
<td>22.06072</td>
<td>105.500</td>
<td>29.24328</td>
<td>157.5000</td>
</tr>
<tr>
<td>69.26040</td>
<td>106.000</td>
<td>25.65200</td>
<td>158.0000</td>
</tr>
<tr>
<td>51.30400</td>
<td>106.000</td>
<td>40.01712</td>
<td>158.5000</td>
</tr>
<tr>
<td>24.62592</td>
<td>106.000</td>
<td>30.78240</td>
<td>158.5000</td>
</tr>
<tr>
<td>66.69590</td>
<td>107.500</td>
<td>50.79096</td>
<td>158.5000</td>
</tr>
<tr>
<td>65.69612</td>
<td>108.000</td>
<td>51.30400</td>
<td>159.0000</td>
</tr>
<tr>
<td>61.56480</td>
<td>110.000</td>
<td>49.76488</td>
<td>159.0000</td>
</tr>
<tr>
<td>46.17360</td>
<td>110.500</td>
<td>42.06928</td>
<td>159.0000</td>
</tr>
<tr>
<td>52.84312</td>
<td>111.000</td>
<td>50.27792</td>
<td>160.5000</td>
</tr>
<tr>
<td>43.09536</td>
<td>111.000</td>
<td>52.33008</td>
<td>161.0000</td>
</tr>
<tr>
<td>34.88672</td>
<td>111.000</td>
<td>43.09536</td>
<td>162.5000</td>
</tr>
<tr>
<td>17.95640</td>
<td>111.000</td>
<td>31.80848</td>
<td>162.5000</td>
</tr>
<tr>
<td>44.12144</td>
<td>111.500</td>
<td>47.71272</td>
<td>163.0000</td>
</tr>
<tr>
<td>29.24328</td>
<td>111.500</td>
<td>30.78240</td>
<td>163.0000</td>
</tr>
<tr>
<td>44.63468</td>
<td>112.000</td>
<td>50.27792</td>
<td>164.0000</td>
</tr>
<tr>
<td>33.36760</td>
<td>112.000</td>
<td>25.65200</td>
<td>164.0000</td>
</tr>
<tr>
<td>44.12144</td>
<td>112.200</td>
<td>53.86920</td>
<td>167.0000</td>
</tr>
<tr>
<td>34.88672</td>
<td>112.200</td>
<td>17.95640</td>
<td>167.0000</td>
</tr>
<tr>
<td>48.22576</td>
<td>112.500</td>
<td>47.19968</td>
<td>168.0000</td>
</tr>
<tr>
<td>44.63468</td>
<td>113.000</td>
<td>40.01712</td>
<td>168.0000</td>
</tr>
<tr>
<td>33.86064</td>
<td>113.000</td>
<td>28.73024</td>
<td>168.0000</td>
</tr>
<tr>
<td>47.19968</td>
<td>113.500</td>
<td>39.50408</td>
<td>168.5000</td>
</tr>
<tr>
<td>34.37368</td>
<td>113.500</td>
<td>27.70416</td>
<td>168.5000</td>
</tr>
<tr>
<td>47.71272</td>
<td>114.000</td>
<td>45.14752</td>
<td>170.0000</td>
</tr>
<tr>
<td>28.21720</td>
<td>114.000</td>
<td>25.13896</td>
<td>170.5000</td>
</tr>
<tr>
<td>20.52160</td>
<td>114.000</td>
<td>61.05176</td>
<td>171.0000</td>
</tr>
<tr>
<td>21.54768</td>
<td>115.000</td>
<td>51.30400</td>
<td>171.0000</td>
</tr>
<tr>
<td>52.84312</td>
<td>117.000</td>
<td>56.94744</td>
<td>172.0000</td>
</tr>
<tr>
<td>48.73880</td>
<td>119.000</td>
<td>34.37368</td>
<td>173.0000</td>
</tr>
<tr>
<td>48.22576</td>
<td>120.000</td>
<td>47.45620</td>
<td>173.0000</td>
</tr>
<tr>
<td>19.49552</td>
<td>122.000</td>
<td>57.97352</td>
<td>175.0000</td>
</tr>
<tr>
<td>48.73880</td>
<td>122.000</td>
<td>61.56480</td>
<td>175.5000</td>
</tr>
<tr>
<td>47.71272</td>
<td>122.000</td>
<td>55.92136</td>
<td>176.5000</td>
</tr>
<tr>
<td>47.19968</td>
<td>124.000</td>
<td>17.44336</td>
<td>176.5000</td>
</tr>
<tr>
<td>23.08680</td>
<td>124.000</td>
<td>16.95640</td>
<td>178.0000</td>
</tr>
<tr>
<td>22.06072</td>
<td>124.000</td>
<td>21.54768</td>
<td>181.0000</td>
</tr>
<tr>
<td>39.50408</td>
<td>124.500</td>
<td>20.52160</td>
<td>184.0000</td>
</tr>
<tr>
<td>23.08680</td>
<td>125.500</td>
<td>23.59984</td>
<td>192.0000</td>
</tr>
</tbody>
</table>
iii) **expected respiratory cancer deaths** (r,th,expd) (18 eds)

<p>| | | | | | | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>49.01</td>
<td>1.823</td>
<td>2.992</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41.39</td>
<td>1.997</td>
<td>3.986</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43.64</td>
<td>2.834</td>
<td>2.903</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35.40</td>
<td>2.434</td>
<td>4.316</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24.92</td>
<td>2.328</td>
<td>3.453</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24.14</td>
<td>1.880</td>
<td>2.713</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.44</td>
<td>1.388</td>
<td>3.825</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.11</td>
<td>2.078</td>
<td>5.444</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19.04</td>
<td>2.689</td>
<td>3.792</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37.63</td>
<td>2.919</td>
<td>3.348</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26.95</td>
<td>3.196</td>
<td>5.010</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24.73</td>
<td>3.614</td>
<td>3.800</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.62</td>
<td>3.885</td>
<td>5.499</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24.34</td>
<td>4.186</td>
<td>3.336</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.52</td>
<td>4.498</td>
<td>4.676</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42.41</td>
<td>4.258</td>
<td>3.228</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43.31</td>
<td>12.41</td>
<td>1.187</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.79</td>
<td>0.5972</td>
<td>0.8194</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

iv) **population** (18 eds)

| (E) | (N) | M1 | F1 | M2 | F2 | M3 | F3 | M4 | F4 | M5 | F5 | M6 | F6 | M7 | F7 | M8 | F8 | M9 | F9 |
|-----|-----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| A01 | 2938 | 6692 | 13 | 12 | 47 | 29 | 52 | 45 | 14 | 20 | 32 | 35 | 51 | 56 | 32 | 23 | 01 | 08 | 00 | 04 |
| A02 | 2937 | 6690 | 30 | 18 | 39 | 34 | 37 | 38 | 35 | 31 | 15 | 23 | 48 | 54 | 35 | 27 | 09 | 10 | 02 | 00 |
| A03 | 2932 | 6685 | 41 | 40 | 45 | 36 | 63 | 81 | 52 | 46 | 27 | 29 | 09 | 14 | 14 | 19 | 09 | 05 | 07 | 05 |
| A04 | 2935 | 6687 | 14 | 08 | 47 | 61 | 45 | 32 | 24 | 22 | 34 | 45 | 33 | 36 | 37 | 31 | 14 | 06 | 01 | 07 |
| A05 | 2937 | 6686 | 08 | 06 | 34 | 18 | 24 | 28 | 08 | 09 | 14 | 15 | 24 | 27 | 29 | 26 | 07 | 14 | 04 | 07 |
| A06 | 2939 | 6687 | 25 | 28 | 57 | 76 | 19 | 11 | 32 | 49 | 46 | 39 | 18 | 23 | 11 | 06 | 10 | 20 | 03 | 08 |
| A07 | 2941 | 6685 | 12 | 11 | 35 | 22 | 16 | 18 | 29 | 33 | 28 | 26 | 22 | 20 | 19 | 18 | 10 | 24 | 07 | 16 |
| A08 | 2939 | 6685 | 09 | 06 | 28 | 17 | 25 | 17 | 09 | 14 | 11 | 17 | 30 | 27 | 30 | 30 | 19 | 29 | 08 | 12 |
| A09 | 2937 | 6684 | 14 | 09 | 22 | 21 | 29 | 32 | 18 | 22 | 15 | 18 | 25 | 17 | 12 | 26 | 12 | 13 | 10 | 06 |
| A10 | 2933 | 6684 | 21 | 10 | 29 | 28 | 23 | 28 | 13 | 14 | 20 | 23 | 18 | 21 | 17 | 15 | 08 | 11 | 08 | 12 |
| A11 | 2935 | 6682 | 17 | 09 | 46 | 35 | 32 | 40 | 22 | 20 | 23 | 32 | 29 | 28 | 21 | 26 | 20 | 23 | 08 | 09 |
| A12 | 2936 | 6680 | 07 | 12 | 12 | 16 | 14 | 20 | 13 | 17 | 10 | 14 | 17 | 20 | 21 | 28 | 14 | 17 | 05 | 09 |
| A13 | 2938 | 6680 | 10 | 05 | 25 | 18 | 15 | 13 | 18 | 18 | 18 | 22 | 20 | 18 | 31 | 37 | 27 | 35 | 03 | 06 |
| A14 | 2938 | 6678 | 19 | 14 | 17 | 30 | 25 | 25 | 13 | 13 | 21 | 17 | 10 | 08 | 16 | 19 | 14 | 13 | 06 | 03 |
| A15 | 2940 | 6680 | 17 | 06 | 34 | 18 | 33 | 28 | 23 | 19 | 16 | 19 | 23 | 27 | 26 | 20 | 22 | 20 | 04 | 05 |
| A28 | 2935 | 6689 | 39 | 22 | 28 | 35 | 49 | 65 | 54 | 50 | 23 | 24 | 24 | 24 | 20 | 26 | 08 | 14 | 05 | 04 |
| A29 | 2933 | 6687 | 60 | 45 | 42 | 45 | 45 | 65 | 63 | 58 | 35 | 39 | 20 | 15 | 08 | 07 | 02 | 03 | 00 | 03 |
| A30 | 2943 | 6684 | 12 | 10 | 33 | 36 | 07 | 09 | 21 | 19 | 10 | 15 | 09 | 05 | 02 | 07 | 03 | 04 | 01 | 02 |
IMAGING SERVICES NORTH
Boston Spa, Wetherby
West Yorkshire, LS23 7BQ
www.bl.uk

PAGE MISSING IN ORIGINAL
325
B) Bonnybridge
ii)

Pneumonia deaths (182 eds)
r

87.6641
85.6154
89.0505
88.6454
94.9210
114.7563
90.3549
77.0260
49.6488
114.8434
26.4008
32.2025
34.1321
34.4819
31. 3050
30.4138
27.7308
25.5539
22.8254
23.4094
27.5136
26.0000
29.1548
31. 0161
32.6956
33.2415
31.6228
35.4401
34.5398
35.4683
38.2753
39.5601
37.1080
, 40.0250
41.4849
34.6699
39.2046
37.1214
39.3954
38.4838
42.9418
41.1825
44.2945
48.3011
45.6180
44.7772
44.9444
44.6430
50.6952
49.3964
49.2443
53.8516
52.3259
51.1077
46.3249
54.0833
52.6973
61. 8466
61.3270
60.3738

th
4.9076
4.9602
4.9274
4.9054
4.9139
4.8876
4.8011
4.7384
5.3116
5.0039
5.6321
5.6521
5.7277
5.7677
5.8196
5.8048
5.8357
5.7495
5.7804
5.9344
5.9500
5.8884
5.8963
5.9207
5.8743
5.9139
5.9614
5.9971
5.8973
5.9079
5.9367
6.0014
6.0382
6.0564
6.0148
5.7032
5.6244
5.8075
5.8650
5.8544
5.7987
5.7761
5.7893
5.8103
5.8294
5.8695
5.9192
6.0111
6.0239
5.9102
5.8650
5.9027
5.8281
5.7495
5.7132
5.6952
5.7005
3.5401
3.5615
3.6052

Population (18 age/sex groups)

nd
0
0
0
0
0
0
0
0
0
0
0
1
0
0
0
0
2
0
0
0
0
1
1
0
1
1
0
0
0
2
0
1
1
0
1
0
1
2
0
1
0
1
1
6
0
3
0
0
2
1
0
0
1
1
1
1
0
0
0
0

8
6
15
14
5
7
1
3
2
5
17
26
24
8
19
27
12
17
19
8
8
8
12
8
7
2
5
3
0
6
5
4
10
8
9
5
3
1
6
3
9
8
4
3
3
4
6
1
2
3
9
11
7
11
8
17
10
7
16
32

5
5
7
11
8
7
1
0
3
10
24
16
25
5
29
13
20
16
18
8
7
12
13
9
11
3
0
2
5
1
2
2
6
3
10
2
1
1
2
13
8
10
3
5
4
5
5
3
4
4
5
4
6
8
8
19
13
7
19
29

26
11
58
48
20
3
1
3
11
20
49
70
76
16
39
62
57
41
49
34
35
34
16
20
9
26
1
9
10
5
3
16
17
35
9
8
16
3
24
25
21
30
35
13
20
21
37
7
11
11
30
24
17
27
38
41
28
19
43
35

20
10
42
38
14
8
3
7
7
22
40
57
62
28
54
64
57
39
43
77
32
39
10
23
7
24
6
8
6
9
5
19
22
39
8
12
12

o

16
22
28
39
42
12
23
16
31
5
11
24
25
35
26
21
31
32
21
24
38
38

27
13
41
49
16
10
1
0
10
11
32
42
36
24
62
56
41
62
47
34
39
46
20
24
15
33
1
30
18
13
5
29
28
42
21
20
12
11
15
27
33
27
36
15
29
43
32
6
24
14
25
34
34
24
55
26
29
26
45
18

33
19
48
47
12
13
5
1
7
16
42
40
34
15
58
47
39
34
39
20
35
27
27
21
19
24
6
19
20
12
8
20
18
33
24
18
9
7
18
23
28
29
43
17
20
40
38
5
17
17
25
24
31
19
40
27
39
14
46
23

20
15
31
28
11
6
3
3
5
18
46
46
45
23
35
37
30
25
34
22
19
28
17
16
13
12
5
11
14
12
6
9
14
14
21
5
6
4
10
21
18
22
20
9
12
15
18
5
16
11
16
16
17
14
14
54
26
11
72
69

21
10
27
28
12
10
2
4
6
13
49
54
54
15
34
33
28
27
35
22
17
22
12
14
12
11
5
9
12
7
5
17
21
14
21
6
8
2
15
21
21
30
24
7
10
12
20
2
12
10
13
17
14
19
13
55
27
14
67
72

25
13
31
20
9
8
3
5
8
20
21
45
43
16
31
41
31
27
29
19
30
23
9
20
12
24
4
14
7
12
3
16
11
36
8
11
10
9
20
23
22
17
24
9
13
8
38
8
13
12
22
21
25
19
35
34
20
22
19
32

16
16
37
26
10
7
3
2
7
22
28
46
40
21
32
42
38
30
32
21
31
31
7
18
7
28
2
13
9
8
6
14
15
35
7
12
7
10
17
18
30
19
34
10
19
15
43
9
12
17
24
22
30
25
46
29
25
18
20
25

15
16
21
33
7
8
3
4
10
14
17
18
20
19
22
30
26
31
27
19
13
24
18
21
17
22
0
24
5
8
2
25
23
29
16
6
11
15
18
26
25
28
33
23
23
36
32
4
11
14
23
19
23
30
46
15
24
16
8
6

21
10
20
35
10
8
3
0
7
12
9
19
13
19
32
25
22
31
28
27
23
23
16
19
20
20
2
16
8
21
3
26
28
32
14
5
9
19
18
29
25
29
33
30
25
49
35
5
20
26
28
21
30
31
51
11
17
16
11
5

16
14
28
14
10
8
2
0
7
11
14
8
6
8
14
17
18
32
17
28
28
19
17
17
21
26
3
20
12
28
7
17
23
14
8
11
15
29
7
25
33
31
22
20
27
48
28
9
12
24
31
13
16
23
24
4
10
4
11
4

22
12
26
20
7
10
0
3
8
9
15
9
6
16
8
19
19
32
19
38
33
23
17
24
25
21
6
24
8
32
4
26
26
19
11
7
26
39
7
30
39
32
19
21
32
41
21
10
13
20
32
14
21
22
20
3

16
9
7
5

18 25
8 10
411
8 12
3 7
4 8
3 2
1 0
8 3
9 8
2 2
14 5
4 4
5 10
9 9
6 19
16 20
10 6
5 6
18 20
19 17
17 10
23 24
19 26
13 29
13 24
8 7
9 14
11 19
15 25
9 13
30 34
13 14
11 16
8 10
1 1
18 17
34 49
2 3
26 25
18 18
18 18
10 15
22 34
24 26
15 19
7 4
10 15
16 21
6 24
16 22
10 15
1311
16 19
10 12
2 3
8 11
9 9
10 17
3 3

7
4
6
7
1
4
2
0
1
3
2
4
0
2
5
3
3
4
0
4
8
5
17
11
9
4
2
8
4
8
4
12
7
4
1
0
5
13
1
3
3
2
16
12
8
4
2
6
10
9
6
8
1
13
1
0
4
7
0
1

10
6
5
6
4
2
3
0
1
4
1
7
3
13
4
5
5
7
1
8
9
16
21
21
17
6
6
12
16
19
7
31
11
15
8
2
15
23
3
8
8
3
13
16
14
14
3
7
32
23
14
16
11
10
2
1
5
15
1
1


<table>
<thead>
<tr>
<th>Value</th>
<th>Decimal</th>
<th>Integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>43.8634</td>
<td>0.4229</td>
<td>2</td>
</tr>
<tr>
<td>47.8539</td>
<td>0.4563</td>
<td>23</td>
</tr>
<tr>
<td>50.0000</td>
<td>0.4900</td>
<td>16</td>
</tr>
<tr>
<td>52.5730</td>
<td>0.4536</td>
<td>21</td>
</tr>
<tr>
<td>52.9179</td>
<td>0.1773</td>
<td>37</td>
</tr>
<tr>
<td>52.4591</td>
<td>0.3247</td>
<td>25</td>
</tr>
<tr>
<td>52.3411</td>
<td>0.3067</td>
<td>129</td>
</tr>
<tr>
<td>59.0000</td>
<td>0.1416</td>
<td>14</td>
</tr>
<tr>
<td>59.0172</td>
<td>0.1071</td>
<td>16</td>
</tr>
<tr>
<td>60.0666</td>
<td>0.0750</td>
<td>127</td>
</tr>
<tr>
<td>60.2173</td>
<td>0.0837</td>
<td>12</td>
</tr>
<tr>
<td>60.2080</td>
<td>0.0585</td>
<td>7</td>
</tr>
<tr>
<td>60.1973</td>
<td>0.4634</td>
<td>0</td>
</tr>
<tr>
<td>60.6757</td>
<td>2.8095</td>
<td>26</td>
</tr>
<tr>
<td>60.0832</td>
<td>0.7673</td>
<td>10</td>
</tr>
<tr>
<td>63.5498</td>
<td>2.7557</td>
<td>9</td>
</tr>
<tr>
<td>63.9257</td>
<td>2.7321</td>
<td>1</td>
</tr>
<tr>
<td>63.0989</td>
<td>0.2690</td>
<td>2</td>
</tr>
<tr>
<td>65.5359</td>
<td>0.2679</td>
<td>0</td>
</tr>
<tr>
<td>65.1548</td>
<td>0.2610</td>
<td>2</td>
</tr>
<tr>
<td>65.5287</td>
<td>2.5899</td>
<td>21</td>
</tr>
<tr>
<td>65.4819</td>
<td>2.6261</td>
<td>11</td>
</tr>
<tr>
<td>65.3879</td>
<td>2.6192</td>
<td>13</td>
</tr>
<tr>
<td>65.9857</td>
<td>2.6012</td>
<td>1</td>
</tr>
<tr>
<td>65.1205</td>
<td>2.5105</td>
<td>15</td>
</tr>
<tr>
<td>65.0135</td>
<td>2.4711</td>
<td>1</td>
</tr>
<tr>
<td>65.8782</td>
<td>2.4330</td>
<td>0</td>
</tr>
<tr>
<td>65.8904</td>
<td>0.2429</td>
<td>0</td>
</tr>
<tr>
<td>66.8541</td>
<td>2.9229</td>
<td>0</td>
</tr>
<tr>
<td>66.6118</td>
<td>1.7235</td>
<td>0</td>
</tr>
<tr>
<td>66.1332</td>
<td>1.6374</td>
<td>0</td>
</tr>
<tr>
<td>66.0086</td>
<td>1.5880</td>
<td>0</td>
</tr>
<tr>
<td>66.0000</td>
<td>1.5708</td>
<td>0</td>
</tr>
<tr>
<td>66.0351</td>
<td>1.5357</td>
<td>0</td>
</tr>
<tr>
<td>66.0000</td>
<td>1.5078</td>
<td>0</td>
</tr>
<tr>
<td>66.5478</td>
<td>1.9513</td>
<td>0</td>
</tr>
<tr>
<td>68.8443</td>
<td>1.8664</td>
<td>10</td>
</tr>
<tr>
<td>69.2140</td>
<td>1.8371</td>
<td>21</td>
</tr>
<tr>
<td>69.6788</td>
<td>1.8131</td>
<td>8</td>
</tr>
<tr>
<td>88.5268</td>
<td>1.8103</td>
<td>9</td>
</tr>
<tr>
<td>88.0568</td>
<td>1.8351</td>
<td>10</td>
</tr>
<tr>
<td>86.4002</td>
<td>1.8523</td>
<td>12</td>
</tr>
<tr>
<td>86.4781</td>
<td>1.4364</td>
<td>0</td>
</tr>
<tr>
<td>86.6023</td>
<td>2.1911</td>
<td>3</td>
</tr>
<tr>
<td>86.0058</td>
<td>1.9772</td>
<td>3</td>
</tr>
</tbody>
</table>

The table above contains values with their corresponding decimals and integers.
C) Buckhaven-Methil

### Bronchitis and Pneumonia Deaths (62 eds)

<table>
<thead>
<tr>
<th>ref, E, N, NB, Np</th>
<th>population (18 age × sex groups)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 372 016 00 00</td>
<td>02 373 018 00 00</td>
</tr>
<tr>
<td>02 373 018 00 00</td>
<td>03 375 019 00 00</td>
</tr>
</tbody>
</table>
### Hebeloma Sporophores

#### 1975 (n = 115)

<p>| Value 1 | Value 2 | Value 3 | Value 4 | Value 5 | Value 6 | Value 7 | Value 8 | Value 9 | Value 10 | Value 11 | Value 12 | Value 13 | Value 14 | Value 15 | Value 16 | Value 17 | Value 18 | Value 19 | Value 20 | Value 21 | Value 22 | Value 23 | Value 24 | Value 25 | Value 26 | Value 27 | Value 28 | Value 29 | Value 30 | Value 31 | Value 32 | Value 33 | Value 34 | Value 35 | Value 36 | Value 37 | Value 38 | Value 39 | Value 40 | Value 41 | Value 42 | Value 43 | Value 44 | Value 45 | Value 46 | Value 47 | Value 48 | Value 49 | Value 50 | Value 51 | Value 52 | Value 53 | Value 54 | Value 55 | Value 56 | Value 57 | Value 58 | Value 59 | Value 60 | Value 61 | Value 62 | Value 63 | Value 64 | Value 65 | Value 66 | Value 67 | Value 68 | Value 69 | Value 70 | Value 71 | Value 72 | Value 73 | Value 74 | Value 75 | Value 76 | Value 77 | Value 78 | Value 79 | Value 80 | Value 81 | Value 82 | Value 83 | Value 84 | Value 85 | Value 86 | Value 87 | Value 88 | Value 89 | Value 90 | Value 91 | Value 92 | Value 93 | Value 94 | Value 95 | Value 96 | Value 97 | Value 98 | Value 99 | Value 100 | Value 101 | Value 102 | Value 103 | Value 104 | Value 105 | Value 106 | Value 107 | Value 108 | Value 109 | Value 110 | Value 111 | Value 112 | Value 113 | Value 114 | Value 115 |</p>
<table>
<thead>
<tr>
<th></th>
<th>1978 (n = 42)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1.180957</td>
</tr>
<tr>
<td></td>
<td>1.162206</td>
</tr>
<tr>
<td></td>
<td>1.129842</td>
</tr>
<tr>
<td></td>
<td>1.108625</td>
</tr>
<tr>
<td></td>
<td>1.106040</td>
</tr>
<tr>
<td></td>
<td>1.165136</td>
</tr>
<tr>
<td></td>
<td>1.104410</td>
</tr>
<tr>
<td></td>
<td>1.065909</td>
</tr>
<tr>
<td></td>
<td>1.10350</td>
</tr>
<tr>
<td></td>
<td>1.060536</td>
</tr>
<tr>
<td></td>
<td>1.051713</td>
</tr>
<tr>
<td></td>
<td>0.4037078</td>
</tr>
<tr>
<td></td>
<td>0.3118108</td>
</tr>
<tr>
<td></td>
<td>0.2889793</td>
</tr>
<tr>
<td></td>
<td>1.023946</td>
</tr>
<tr>
<td></td>
<td>1.125224</td>
</tr>
<tr>
<td></td>
<td>1.1069187</td>
</tr>
<tr>
<td></td>
<td>1.066511</td>
</tr>
<tr>
<td></td>
<td>1.164818</td>
</tr>
<tr>
<td></td>
<td>1.133187</td>
</tr>
<tr>
<td></td>
<td>1.098397</td>
</tr>
<tr>
<td></td>
<td>0.3439840</td>
</tr>
<tr>
<td></td>
<td>1.102017</td>
</tr>
<tr>
<td></td>
<td>0.2856606</td>
</tr>
<tr>
<td></td>
<td>1.150784</td>
</tr>
<tr>
<td></td>
<td>1.162754</td>
</tr>
<tr>
<td></td>
<td>1.095858</td>
</tr>
<tr>
<td></td>
<td>0.9509206</td>
</tr>
<tr>
<td></td>
<td>1.120179</td>
</tr>
<tr>
<td></td>
<td>1.206763</td>
</tr>
<tr>
<td></td>
<td>1.106960</td>
</tr>
<tr>
<td></td>
<td>1.062836</td>
</tr>
<tr>
<td></td>
<td>0.9867117</td>
</tr>
<tr>
<td></td>
<td>0.8618475</td>
</tr>
<tr>
<td></td>
<td>0.8743501</td>
</tr>
<tr>
<td></td>
<td>0.8289029</td>
</tr>
<tr>
<td></td>
<td>0.7822052</td>
</tr>
<tr>
<td></td>
<td>0.9068494</td>
</tr>
<tr>
<td></td>
<td>0.8673108</td>
</tr>
<tr>
<td></td>
<td>0.8721703</td>
</tr>
<tr>
<td></td>
<td>0.8124235</td>
</tr>
<tr>
<td></td>
<td>0.3527322</td>
</tr>
</tbody>
</table>
E) Oak Bark Beetle

\((r, \phi, \text{nm, nu})\)

\[
\begin{array}{c}
34 & 1 & 0 & 0 \\
23 & 4.5 & 0 & 4 \\
33.5 & 13 & 1 & 15 \\
14.0 & 21.5 & 0 & 1 \\
24.0 & 22.0 & 0 & 2 \\
8 & 24.5 & 0 & 1 \\
34 & 26.5 & 0 & 1 \\
37 & 33 & 0 & 9 \\
23 & 41.5 & 0 & 3 \\
39 & 45 & 0 & 7 \\
14 & 45 & 2 & 1. \\
5 & 53.5 & 11 & 5 \\
23 & 65.0 & 1 & 2 \\
37 & 71.5 & 2 & 30 \\
38 & 78 & 0 & 3 \\
22 & 84 & 0 & 3 \\
37 & 90.5 & 4 & 33 \\
13 & 90.5 & 5 & 7 \\
7 & 98 & 1 & 1 \\
37 & 102 & 0 & 3 \\
22 & 106 & 1 & 4 \\
37 & 115 & 2 & 28 \\
22 & 125 & 1 & 9 \\
12 & 125 & 0 & 4 \\
37 & 127.5 & 0 & 7 \\
37 & 140 & 1 & 28 \\
21 & 145.5 & 1 & 0 \\
7 & 145.5 & 4 & 4 \\
37 & 152 & 1 & 26 \\
36 & 164 & 0 & 1 \\
22 & 164 & 0 & 3 \\
12.5 & 170 & 0 & 3 \\
32 & 176.5 & 0 & 3 \\
3 & 179.5 & 9 & 5 \\
31 & 188 & 1 & 9 \\
22 & 192 & 1 & 4 \\
8 & 192 & 3 & 8 \\
32 & 199 & 1 & 6 \\
12 & 202 & 0 & 2 \\
32 & 207 & 1 & 9 \\
22 & 217 & 0 & 1 \\
35 & 219 & 3 & 12 \\
33 & 231 & 2 & 3 \\
12 & 234 & 0 & 0 \\
22 & 242 & 1 & 3 \\
38 & 244.5 & 0 & 3 \\
8 & 254 & 14 & 7 \\
38 & 255 & 0 & 5 \\
22 & 255 & 0 & 1 \\
16 & 263 & 8 & 0 \\
39 & 266.5 & 0 & 0 \\
4 & 274 & 16 & 5 \\
22 & 274 & 1 & 6 \\
38 & 278 & 0 & 1 \\
10 & 287.5 & 0 & 1 \\
24 & 325 & 0 & 4 \\
39 & 337 & 0 & 0 \\
9 & 338 & 1 & 3 \\
6 & 342 & 13 & 7 \\
23 & 347 & 0 & 1 \\
15 & 352 & 1 & 0 \\
38 & 289 & 0 & 0 \\
15 & 290 & 0 & 0 \\
24 & 290 & 1 & 4 \\
40 & 292 & 0 & 1 \\
24 & 301 & 0 & 2 \\
38 & 308.5 & 0 & 1 \\
15 & 313 & 0 & 0 \\
39 & 319 & 0 & 1 \\
\end{array}
\]

nm: number of marked beetles

nu: number of unmarked beetles.
F) *Lupinus Arboreas* \( (n = 76) \)

<table>
<thead>
<tr>
<th>69.9</th>
<th>3.5</th>
<th>1</th>
<th>73.5</th>
<th>227.0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>29.0</td>
<td>2.5</td>
<td>3</td>
<td>43.0</td>
<td>229.0</td>
<td>4</td>
</tr>
<tr>
<td>45.0</td>
<td>9.5</td>
<td>4</td>
<td>54.0</td>
<td>236.0</td>
<td>2</td>
</tr>
<tr>
<td>58.0</td>
<td>13.5</td>
<td>3</td>
<td>64.0</td>
<td>238.0</td>
<td>1</td>
</tr>
<tr>
<td>50.0</td>
<td>16.5</td>
<td>1</td>
<td>34.0</td>
<td>256.5</td>
<td>3</td>
</tr>
<tr>
<td>66.5</td>
<td>21.0</td>
<td>4</td>
<td>57.0</td>
<td>275.0</td>
<td>1</td>
</tr>
<tr>
<td>75.0</td>
<td>21.5</td>
<td>1</td>
<td>24.0</td>
<td>279.0</td>
<td>5</td>
</tr>
<tr>
<td>91.0</td>
<td>22.5</td>
<td>1</td>
<td>37.0</td>
<td>283.0</td>
<td>3</td>
</tr>
<tr>
<td>54.0</td>
<td>28.0</td>
<td>5</td>
<td>45.0</td>
<td>289.0</td>
<td>1</td>
</tr>
<tr>
<td>35.5</td>
<td>30.0</td>
<td>5</td>
<td>65.0</td>
<td>305.0</td>
<td>1</td>
</tr>
<tr>
<td>72.0</td>
<td>39.5</td>
<td>3</td>
<td>42.0</td>
<td>309.5</td>
<td>3</td>
</tr>
<tr>
<td>59.0</td>
<td>44.5</td>
<td>4</td>
<td>92.0</td>
<td>322.5</td>
<td>1</td>
</tr>
<tr>
<td>86.0</td>
<td>46.5</td>
<td>1</td>
<td>54.0</td>
<td>323.0</td>
<td>2</td>
</tr>
<tr>
<td>75.5</td>
<td>47.0</td>
<td>1</td>
<td>43.5</td>
<td>327.0</td>
<td>4</td>
</tr>
<tr>
<td>64.0</td>
<td>61.5</td>
<td>5</td>
<td>94.0</td>
<td>338.0</td>
<td>2</td>
</tr>
<tr>
<td>35.0</td>
<td>66.5</td>
<td>5</td>
<td>99.0</td>
<td>336.0</td>
<td>1</td>
</tr>
<tr>
<td>65.0</td>
<td>70.0</td>
<td>2</td>
<td>55.0</td>
<td>344.5</td>
<td>2</td>
</tr>
<tr>
<td>50.0</td>
<td>72.0</td>
<td>5</td>
<td>83.0</td>
<td>352.0</td>
<td>1</td>
</tr>
<tr>
<td>61.0</td>
<td>77.5</td>
<td>1</td>
<td>53.0</td>
<td>355.0</td>
<td>4</td>
</tr>
<tr>
<td>62.0</td>
<td>85.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50.0</td>
<td>94.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30.5</td>
<td>95.5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36.5</td>
<td>95.0</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>61.0</td>
<td>95.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22.5</td>
<td>100.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>52.5</td>
<td>103.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.0</td>
<td>105.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42.5</td>
<td>109.0</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>56.0</td>
<td>113.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26.0</td>
<td>116.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>49.0</td>
<td>122.5</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60.0</td>
<td>125.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>69.5</td>
<td>131.5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37.0</td>
<td>135.0</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>75.0</td>
<td>138.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>55.0</td>
<td>138.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19.0</td>
<td>140.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48.0</td>
<td>153.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29.0</td>
<td>154.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>76.0</td>
<td>158.0</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>58.0</td>
<td>162.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45.0</td>
<td>170.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>74.0</td>
<td>171.5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25.0</td>
<td>176.0</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>64.0</td>
<td>178.0</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46.0</td>
<td>186.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>60.0</td>
<td>187.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>82.0</td>
<td>189.5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37.5</td>
<td>194.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80.0</td>
<td>203.5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42.0</td>
<td>211.0</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>53.0</td>
<td>214.5</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>94.0</td>
<td>214.5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>63.5</td>
<td>217.0</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>75.0</td>
<td>221.0</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>64.0</td>
<td>224.5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25.0</td>
<td>226.0</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
G) Volcanic Ejecta

i) 1935 (n = 46)

<table>
<thead>
<tr>
<th>Value</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>42.2</td>
</tr>
<tr>
<td>37.5</td>
<td>35.2</td>
</tr>
<tr>
<td>53</td>
<td>27.7</td>
</tr>
<tr>
<td>44</td>
<td>25.7</td>
</tr>
<tr>
<td>50</td>
<td>19.5</td>
</tr>
<tr>
<td>42</td>
<td>18.2</td>
</tr>
<tr>
<td>45.5</td>
<td>8.0</td>
</tr>
<tr>
<td>43</td>
<td>5.5</td>
</tr>
<tr>
<td>48</td>
<td>3.9</td>
</tr>
<tr>
<td>53</td>
<td>358.5</td>
</tr>
<tr>
<td>47</td>
<td>355</td>
</tr>
<tr>
<td>51</td>
<td>350.8</td>
</tr>
<tr>
<td>51.5</td>
<td>347</td>
</tr>
<tr>
<td>60</td>
<td>337.5</td>
</tr>
<tr>
<td>54.5</td>
<td>335.7</td>
</tr>
<tr>
<td>56.5</td>
<td>333.2</td>
</tr>
<tr>
<td>60</td>
<td>331</td>
</tr>
<tr>
<td>54</td>
<td>330</td>
</tr>
<tr>
<td>53</td>
<td>325</td>
</tr>
<tr>
<td>56</td>
<td>318</td>
</tr>
<tr>
<td>50</td>
<td>316.3</td>
</tr>
<tr>
<td>52</td>
<td>312</td>
</tr>
<tr>
<td>47.5</td>
<td>308</td>
</tr>
<tr>
<td>55.5</td>
<td>304.2</td>
</tr>
<tr>
<td>55</td>
<td>290</td>
</tr>
<tr>
<td>60.1</td>
<td>289</td>
</tr>
<tr>
<td>60.2</td>
<td>282.4</td>
</tr>
<tr>
<td>62</td>
<td>277.5</td>
</tr>
<tr>
<td>60.05</td>
<td>271.0</td>
</tr>
<tr>
<td>60.5</td>
<td>265.5</td>
</tr>
<tr>
<td>57</td>
<td>262.5</td>
</tr>
<tr>
<td>60</td>
<td>260.4</td>
</tr>
<tr>
<td>70</td>
<td>251.0</td>
</tr>
<tr>
<td>60</td>
<td>249.5</td>
</tr>
<tr>
<td>55</td>
<td>233.0</td>
</tr>
<tr>
<td>44</td>
<td>223.0</td>
</tr>
<tr>
<td>44.5</td>
<td>221</td>
</tr>
<tr>
<td>47.5</td>
<td>216.6</td>
</tr>
<tr>
<td>45</td>
<td>214.0</td>
</tr>
<tr>
<td>48</td>
<td>209.5</td>
</tr>
<tr>
<td>46.5</td>
<td>199.2</td>
</tr>
<tr>
<td>43.5</td>
<td>197.5</td>
</tr>
<tr>
<td>48.5</td>
<td>191.0</td>
</tr>
<tr>
<td>41.0</td>
<td>182.2</td>
</tr>
<tr>
<td>39.5</td>
<td>176.5</td>
</tr>
<tr>
<td>32</td>
<td>152.5</td>
</tr>
</tbody>
</table>
ii) 1937 (n = 139)

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>37.0</td>
<td>91.2</td>
<td>20</td>
</tr>
<tr>
<td>47.1</td>
<td>91.1</td>
<td>148.0</td>
</tr>
<tr>
<td>60.0</td>
<td>91.0</td>
<td>39</td>
</tr>
<tr>
<td>47.0</td>
<td>87.1</td>
<td>269.0</td>
</tr>
<tr>
<td>50.0</td>
<td>87.0</td>
<td>239.0</td>
</tr>
<tr>
<td>44.0</td>
<td>82.5</td>
<td>235.5</td>
</tr>
<tr>
<td>53.0</td>
<td>80.0</td>
<td>235.2</td>
</tr>
<tr>
<td>47.0</td>
<td>77.5</td>
<td>235.0</td>
</tr>
<tr>
<td>52.0</td>
<td>77.2</td>
<td>230.0</td>
</tr>
<tr>
<td>60.0</td>
<td>77.1</td>
<td>220.0</td>
</tr>
<tr>
<td>55.0</td>
<td>75.0</td>
<td>217.0</td>
</tr>
<tr>
<td>61.0</td>
<td>69.0</td>
<td>209.0</td>
</tr>
<tr>
<td>51.0</td>
<td>64.0</td>
<td>194.0</td>
</tr>
<tr>
<td>48.0</td>
<td>62.0</td>
<td>178.0</td>
</tr>
<tr>
<td>51.0</td>
<td>62.1</td>
<td>170.0</td>
</tr>
<tr>
<td>54.0</td>
<td>60.0</td>
<td>166.0</td>
</tr>
<tr>
<td>59.0</td>
<td>58.0</td>
<td>162.1</td>
</tr>
<tr>
<td>51.0</td>
<td>56.1</td>
<td>157.0</td>
</tr>
<tr>
<td>53.0</td>
<td>56.2</td>
<td>152.0</td>
</tr>
<tr>
<td>58.0</td>
<td>55.0</td>
<td></td>
</tr>
<tr>
<td>62.0</td>
<td>55.1</td>
<td></td>
</tr>
<tr>
<td>63.0</td>
<td>55.2</td>
<td></td>
</tr>
<tr>
<td>54.0</td>
<td>53.1</td>
<td></td>
</tr>
<tr>
<td>60.0</td>
<td>53.0</td>
<td></td>
</tr>
<tr>
<td>65.0</td>
<td>45.1</td>
<td></td>
</tr>
<tr>
<td>70.0</td>
<td>45.0</td>
<td></td>
</tr>
<tr>
<td>67.0</td>
<td>43.0</td>
<td></td>
</tr>
<tr>
<td>67.0</td>
<td>41.0</td>
<td></td>
</tr>
<tr>
<td>50.0</td>
<td>39.0</td>
<td></td>
</tr>
<tr>
<td>69.0</td>
<td>36.5</td>
<td></td>
</tr>
<tr>
<td>63.0</td>
<td>35.0</td>
<td></td>
</tr>
<tr>
<td>57.0</td>
<td>34.0</td>
<td></td>
</tr>
<tr>
<td>69.0</td>
<td>32.0</td>
<td></td>
</tr>
<tr>
<td>65.0</td>
<td>30.1</td>
<td></td>
</tr>
<tr>
<td>67.0</td>
<td>30.0</td>
<td></td>
</tr>
<tr>
<td>70.0</td>
<td>29.0</td>
<td></td>
</tr>
<tr>
<td>66.0</td>
<td>27.1</td>
<td></td>
</tr>
<tr>
<td>68.0</td>
<td>27.0</td>
<td></td>
</tr>
<tr>
<td>71.0</td>
<td>26.9</td>
<td></td>
</tr>
<tr>
<td>68.0</td>
<td>25.5</td>
<td></td>
</tr>
<tr>
<td>65.0</td>
<td>23.0</td>
<td></td>
</tr>
<tr>
<td>63.0</td>
<td>21.1</td>
<td></td>
</tr>
</tbody>
</table>
### 1938 (n = 169)

<table>
<thead>
<tr>
<th>Value</th>
<th>Mean</th>
<th>SD</th>
<th>Value</th>
<th>Mean</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>72.5</td>
<td>63</td>
<td>0.65</td>
<td>60</td>
<td>340.6</td>
<td>0.63</td>
</tr>
<tr>
<td>74.5</td>
<td>62</td>
<td>0.1</td>
<td>82</td>
<td>340.7</td>
<td>0.63</td>
</tr>
<tr>
<td>79.0</td>
<td>63</td>
<td>0.15</td>
<td>88</td>
<td>340.3</td>
<td>0.63</td>
</tr>
<tr>
<td>77.0</td>
<td>70</td>
<td>0.12</td>
<td>68</td>
<td>338.6</td>
<td>0.63</td>
</tr>
<tr>
<td>74.5</td>
<td>72</td>
<td>0.56</td>
<td>69</td>
<td>338.5</td>
<td>0.63</td>
</tr>
<tr>
<td>73.0</td>
<td>76</td>
<td>0.51</td>
<td>83</td>
<td>338.4</td>
<td>0.63</td>
</tr>
<tr>
<td>70.5</td>
<td>102</td>
<td>0.50</td>
<td>85</td>
<td>337.0</td>
<td>0.63</td>
</tr>
<tr>
<td>66.0</td>
<td>60</td>
<td>0.01</td>
<td>89</td>
<td>337.2</td>
<td>0.63</td>
</tr>
<tr>
<td>67.5</td>
<td>62</td>
<td>0.1</td>
<td>64</td>
<td>336.1</td>
<td>0.63</td>
</tr>
<tr>
<td>70.0</td>
<td>63</td>
<td>0.15</td>
<td>66</td>
<td>336.0</td>
<td>0.63</td>
</tr>
<tr>
<td>71.5</td>
<td>97</td>
<td>0.12</td>
<td>83</td>
<td>335.9</td>
<td>0.63</td>
</tr>
<tr>
<td>79.0</td>
<td>70</td>
<td>0.12</td>
<td>86</td>
<td>335.8</td>
<td>0.63</td>
</tr>
<tr>
<td>69.0</td>
<td>78</td>
<td>0.9</td>
<td>91</td>
<td>335.7</td>
<td>0.63</td>
</tr>
<tr>
<td>71.0</td>
<td>69</td>
<td>0.9</td>
<td>61</td>
<td>335.0</td>
<td>0.63</td>
</tr>
<tr>
<td>70.0</td>
<td>73</td>
<td>0.6</td>
<td>63</td>
<td>334.0</td>
<td>0.63</td>
</tr>
<tr>
<td>75.0</td>
<td>62</td>
<td>0.7</td>
<td>91</td>
<td>334.1</td>
<td>0.63</td>
</tr>
<tr>
<td>82.0</td>
<td>64</td>
<td>0.7</td>
<td>63</td>
<td>333.5</td>
<td>0.63</td>
</tr>
<tr>
<td>73.0</td>
<td>66</td>
<td>0.4</td>
<td>65</td>
<td>333.4</td>
<td>0.63</td>
</tr>
<tr>
<td>71.0</td>
<td>72</td>
<td>0.7</td>
<td>85</td>
<td>333.6</td>
<td>0.63</td>
</tr>
<tr>
<td>73.5</td>
<td>75</td>
<td>0.6</td>
<td>60</td>
<td>332.5</td>
<td>0.63</td>
</tr>
<tr>
<td>72.0</td>
<td>78</td>
<td>0.6</td>
<td>64</td>
<td>332.6</td>
<td>0.63</td>
</tr>
<tr>
<td>74.0</td>
<td>81</td>
<td>0.5</td>
<td>73</td>
<td>323.5</td>
<td>0.63</td>
</tr>
<tr>
<td>79.0</td>
<td>63</td>
<td>0.6</td>
<td>75</td>
<td>323.4</td>
<td>0.63</td>
</tr>
<tr>
<td>76.0</td>
<td>67</td>
<td>0.6</td>
<td>59</td>
<td>310</td>
<td>0.63</td>
</tr>
<tr>
<td>70.0</td>
<td>69</td>
<td>0.6</td>
<td>55</td>
<td>306.1</td>
<td>0.63</td>
</tr>
<tr>
<td>75.0</td>
<td>78</td>
<td>0.6</td>
<td>63</td>
<td>306.0</td>
<td>0.63</td>
</tr>
<tr>
<td>71.0</td>
<td>65</td>
<td>0.5</td>
<td>54</td>
<td>303.5</td>
<td>0.63</td>
</tr>
<tr>
<td>83.5</td>
<td>70</td>
<td>0.5</td>
<td>59</td>
<td>303.4</td>
<td>0.63</td>
</tr>
<tr>
<td>72.0</td>
<td>71</td>
<td>0.5</td>
<td>64</td>
<td>303.2</td>
<td>0.63</td>
</tr>
<tr>
<td>90.0</td>
<td>72</td>
<td>0.5</td>
<td>55</td>
<td>302.1</td>
<td>0.63</td>
</tr>
<tr>
<td>67.0</td>
<td>76</td>
<td>0.5</td>
<td>57</td>
<td>302.0</td>
<td>0.63</td>
</tr>
<tr>
<td>63.0</td>
<td>93</td>
<td>0.5</td>
<td>64</td>
<td>302.2</td>
<td>0.63</td>
</tr>
<tr>
<td>71.0</td>
<td>55</td>
<td>0.2</td>
<td>56</td>
<td>300.1</td>
<td>0.63</td>
</tr>
<tr>
<td>73.0</td>
<td>60</td>
<td>0.2</td>
<td>60</td>
<td>300.0</td>
<td>0.63</td>
</tr>
<tr>
<td>79.0</td>
<td>75</td>
<td>0.2</td>
<td>54</td>
<td>298.0</td>
<td>0.63</td>
</tr>
<tr>
<td>66.0</td>
<td>57</td>
<td>0.2</td>
<td>62</td>
<td>298.1</td>
<td>0.63</td>
</tr>
<tr>
<td>70.0</td>
<td>63</td>
<td>0.2</td>
<td>70</td>
<td>298.2</td>
<td>0.63</td>
</tr>
<tr>
<td>82.0</td>
<td>70</td>
<td>0.2</td>
<td>54</td>
<td>297.0</td>
<td>0.63</td>
</tr>
<tr>
<td>71.0</td>
<td>55</td>
<td>0.2</td>
<td>60</td>
<td>297.1</td>
<td>0.63</td>
</tr>
<tr>
<td>95.0</td>
<td>57</td>
<td>0.2</td>
<td>62</td>
<td>297.2</td>
<td>0.63</td>
</tr>
<tr>
<td>96.0</td>
<td>60</td>
<td>0.2</td>
<td>67</td>
<td>297.3</td>
<td>0.63</td>
</tr>
<tr>
<td>107.0</td>
<td>68</td>
<td>0.2</td>
<td>65</td>
<td>296.0</td>
<td>0.63</td>
</tr>
<tr>
<td>60.0</td>
<td>71</td>
<td>0.2</td>
<td>68</td>
<td>1.87</td>
<td>0.63</td>
</tr>
<tr>
<td>72.5</td>
<td>90</td>
<td>0.2</td>
<td>73</td>
<td>1.86</td>
<td>0.63</td>
</tr>
<tr>
<td>75.0</td>
<td>71</td>
<td>0.2</td>
<td>75</td>
<td>1.91</td>
<td>0.63</td>
</tr>
<tr>
<td>80.0</td>
<td>69</td>
<td>0.2</td>
<td>82</td>
<td>1.92</td>
<td>0.63</td>
</tr>
<tr>
<td>81.0</td>
<td>61</td>
<td>0.2</td>
<td>94</td>
<td>1.90</td>
<td>0.63</td>
</tr>
<tr>
<td>93.0</td>
<td>63</td>
<td>0.2</td>
<td>59</td>
<td>0.7</td>
<td>0.63</td>
</tr>
<tr>
<td>97.0</td>
<td>72</td>
<td>0.2</td>
<td>80</td>
<td>344.0</td>
<td>0.63</td>
</tr>
<tr>
<td>114.0</td>
<td>59</td>
<td>0.2</td>
<td>82</td>
<td>344.2</td>
<td>0.63</td>
</tr>
<tr>
<td>75.0</td>
<td>67</td>
<td>0.2</td>
<td>83</td>
<td>344.3</td>
<td>0.63</td>
</tr>
<tr>
<td>95.5</td>
<td>61</td>
<td>0.2</td>
<td>85</td>
<td>344.1</td>
<td>0.63</td>
</tr>
<tr>
<td>104.0</td>
<td>64</td>
<td>0.2</td>
<td>58</td>
<td>343.2</td>
<td>0.63</td>
</tr>
<tr>
<td>62.0</td>
<td>76</td>
<td>0.2</td>
<td>62</td>
<td>343.0</td>
<td>0.63</td>
</tr>
<tr>
<td>63</td>
<td>82</td>
<td>0.2</td>
<td>58</td>
<td>340.5</td>
<td>0.63</td>
</tr>
<tr>
<td>65</td>
<td>80</td>
<td>0.2</td>
<td>57</td>
<td>340.0</td>
<td>0.63</td>
</tr>
<tr>
<td>67</td>
<td>77</td>
<td>0.2</td>
<td>56</td>
<td>340.5</td>
<td>0.63</td>
</tr>
</tbody>
</table>
Appendix XII

Numerical results for $\psi - \kappa$ optimisation

A variety of simulations of $M(\mu_0, \kappa + \psi \tau)$ models were carried out to assess the numerical accuracy of direct maximisation using a library routine. Having failed to obtain reasonable results using a simple routine (NAG: E04JAF), we then used a more sophisticated optimisation routine which required the specification of first derivatives. Both routines are based on a Modified Newton method. Table 2 below, gives details of the results found for the parameter range: $0 < \kappa, \psi < 40$.

Figure 47 also displays the contour map of the likelihood surface for $\psi - \kappa$ near the maximum.

The surface appears relatively flat and this supports the difficulty in obtaining precise estimates suggested in Table 2.
Figure 47

$\psi - \kappa$ Optimisation

a) point map example of simulation with $k = 8, \psi = 2, \mu_0 = 120^\circ, \kappa = 100$

b) likelihood surface contour map
Table 2
Optimisation of \( M(\mu_0, k + \psi) \) models on NAG using E4KAF

<table>
<thead>
<tr>
<th>k</th>
<th>( \psi )</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>5</th>
<th>20</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.216</td>
<td>0.149</td>
<td>1.795</td>
<td>0.237</td>
<td>1.896</td>
<td>0.195</td>
<td>1.968</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>0.901</td>
<td>2.236</td>
<td>0.877</td>
<td>2.292</td>
<td>0.871</td>
<td>2.240</td>
<td>1.122</td>
<td>2.195</td>
</tr>
<tr>
<td>2</td>
<td>0.019</td>
<td>1.094</td>
<td>0.892</td>
<td>2.048</td>
<td>2.773</td>
<td>6.839</td>
<td>21.50</td>
<td>44.3</td>
</tr>
<tr>
<td>3</td>
<td>1.900</td>
<td>2.176</td>
<td>1.932</td>
<td>2.125</td>
<td>1.901</td>
<td>2.175</td>
<td>1.859</td>
<td>2.146</td>
</tr>
<tr>
<td>4</td>
<td>0.00</td>
<td>0.636</td>
<td>2.227</td>
<td>3.337</td>
<td>7.461</td>
<td>24.841</td>
<td>47.95</td>
<td>*NC</td>
</tr>
<tr>
<td>5</td>
<td>3.043</td>
<td>2.110</td>
<td>2.958</td>
<td>2.125</td>
<td>2.656</td>
<td>2.169</td>
<td>2.436</td>
<td>2.134</td>
</tr>
<tr>
<td>6</td>
<td>0.000</td>
<td>0.213</td>
<td>1.073</td>
<td>2.794</td>
<td>4.073</td>
<td>24.579</td>
<td>41.974</td>
<td>*NC</td>
</tr>
<tr>
<td>7</td>
<td>5.988</td>
<td>2.082</td>
<td>5.075</td>
<td>2.109</td>
<td>5.103</td>
<td>2.118</td>
<td>4.679</td>
<td>2.143</td>
</tr>
<tr>
<td>8</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>1.1775</td>
<td>0.636</td>
<td>23.979</td>
<td>*NC</td>
</tr>
<tr>
<td>10</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>1.024</td>
<td>16.747</td>
<td>*NC</td>
</tr>
<tr>
<td>11</td>
<td>35.967</td>
<td>2.100</td>
<td>36.221</td>
<td>2.100</td>
<td>36.477</td>
<td>2.100</td>
<td>36.67</td>
<td>2.101</td>
</tr>
</tbody>
</table>

Simulation for \( n = 100, \mu_0 = 2.09 \) (120°)

The entries \( a, b, c \) are the estimates of \( \kappa, \psi, \mu_0 \) for the simulation specified
(a) \( \text{(b) (c)} \)
(* NC not converging)
FORTRAN PROGRAM LISTINGS FOR APPENDIX I
File _$255$DUA2:[MC.T.AL]JJW.FOR;243 (39,14,0), last revised on 14-MAY-1990 11:53, is a 60 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 69 bytes.

Job JJW (980) queued to SYS$PRINT on 6-NOV-1990 09:24 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 6-NOV-1990 09:24 from queue LASER.
implicit real*8 (x,f,a,p,g,r,t,s,e)
character*20 fname, tname
character*20 wname
character*20 zname
character*2 codey(2)
character*4 code(8)
real*8 hesl(5), hesd(5)
real*8 hinf(3,3), h(3), hwinf(2,2), hwinv(2,2)
real*8 hthinv(3,3)
real*8 x(5), g(5)
real*8 wik(40)
real*8 f,w(9), work(100)
real*8 wkth(18), wkw(10)
real*8 wk(36), fnorm
real*8 a(10), b(10)
real*8 prob(20,50), res(20,50)
real*8 ansc(20,50)
real*8 zed(2000), rcell(20,50), thcell(20,50)
real*8 resp(1000)
real*8 thclass(50), rclass(50)
real*8 ncex(20,50)
real*8 dblin, cres
real*8 fun, ath, bth
real*8 alam, del, akap, psi, pmu5
real*8 khat
integer ncob(20,50)
integer iy(2000)
integer it(2000)
integer iw(1)
integer iwork(10), nsrch
integer iworl(5)
integer icode(4,4)
real*8 mmbsi0, al
integer istate(3), iw(3), liw, lw
external fun, ath, bth
external fweib, f2parm, fgamma
external funct1
common/s1/pi
common/s3/s1, s2, s3, s4, s5, s6, s7, s8, s9, s10
common/s4/rmax
common/s5/pl, pd
common/s6/pk, ps, pmu
data(icode(1,j), j=1,4)/1,4,5,6/
data(icode(2,j), j=1,4)/2,0,0,0/
data(icode(3,j), j=1,4)/3,0,0,0/
data(icode(4,j), j=1,4)/7,8,0,0/
common/s7/sath, sbth
common/s8/st(2)
common/s9/idor
common/s10/imarg
common/s36/rbarl
common/s37/khat
common/s40/alam, del, akap, psi, pmu5
common/s42/itype
common/s50/fh(0:100,0:100)
common/s60/idismod
common/s70/cconst
common/s80/nreg, rreg(1000), threg(1000), popreg(1000)
data codey/'75', '78'/
data code/'HEXX','LTPU','RAXX','HEFR'
+ , 'HEPS','HETR','LETR','LERF'/
    write (6,*), 'enter filename:'
    read(5,88)fname
88  format(a20)
    type*, 'is the data split by factors:yes(1),no(2)' 
    read(5,*)iopr 
    if(iopr.eq.1)then 
      type*, 'are the factors:years(1) or species types(2)' 
      type*, 'or special combination(3)' 
      read(5,*)iozp 
    endif 
    open(l,file=fname,status='old') 
    read(l,*)np 
    if(iopr.eq.2)then 
        do i=l,np 
            read(l,*)r(i),th(i) 
            write(6,*)r(i),th(i) 
        enddo 
    else 
        c now read split on factors 
        c 
        if(iozp.eq.1)then 
          type*, 'which year:75(1) or 78(2)' 
          read(5,*)imz 
          npn=0 
          do i=l,np 
              read(l,*)rn(i),thn(i),iyear(i),itype(i) 
              if(iyear(i).eq.imz)then 
                npn=npn+l 
                r(npn)=rn(i) 
                th(npn)=thn(i) 
                iy(npn)=iyear(i) 
                endif 
          enddo 
          np=npn 
        endif 
        npn=np 
        type*, 'no of points is now:',npn 
        else 
          if(iozp.eq.2)then 
            type*, 'which species:' 
            type*, 'Heb(1),2,3,4,5,6,7,8' 
            read(5,*)iws 
            npn=0 
            do i=l,np 
                read(l,*)rn(i),thn(i),iyear(i),itype(i) 
                if(itype(i).eq.iws)then 
                  npn=npn+l 
                  r(npn)=rn(i) 
                  th(npn)=thn(i) 
                  iy(npn)=iyear(i) 
                  endif 
            enddo 
            np=npn 
            type*, 'no of points is:',np 
            endif 
          endif 
        endif 
      c special data combination 
      c 
      if(iozp.eq.3)then 
        type*, 'heb(1),lact(2),ram(3),lecc(4)' 
        read(5,*)iou 
        type*, 'do you want all species(l) or sub-species(2)' 
      endif
read(5,*)iop
if(iop.eq.2) then
  type*, '1,2,3,4,5,6,7,8'
read(5,*)ice
else
  ice=0
endif

read(5,*)iove
npn=0

do i=1,np
  read(1,*)rn(i),thn(i),iyear(i),itype(i)
  do j=1,4
    if(itype(i).eq.icode(iou,j)) then
      if(iyear(i).eq.iove) then
        npn=npn+1
        r(npn)=rn(i)
        th(npn)=thn(i)
        iy(npn)=iyear(i)
        it(npn)=itype(i)
      endif
    endif
  enddo
else
  if(itype(i).eq.ice) then
    if(iyear(i).eq.iove) then
      npn=npn+1
      r(npn)=rn(i)
      th(npn)=thn(i)
      iy(npn)=iyear(i)
      it(npn)=itype(i)
    endif
  endif
endif
enddo

np=npn
endif
endif

type*, 'no of points is:', np

type*, 'output to file(l:yes;0:no)'
read(5,*)irepp

if(irepp.eq.1) then
  type*, 'enter filename'
read(5,88)tname
open(23,file=tname,status='new')
write(23,*)np
  do i=1,np
    write(23,*)r(i),th(i)
  enddo
close(23)
endif

calculate max radius

rmax=0

do i=1,np
  if(r(i).gt.rmax) then
    rmax=r(i)
  endif
enddo

type*, 'data r max value:', rmax

type*, 'own max rad value(l) or default(2),'
read(5,*)iseo
if(iseo.eq.1)

  type*, 'enter your rad value:'
  read(5,*) rm
  else
  rm=1.1*rmax
  endif
  pi=3.141592
  pi2=2.*pi
  rdf=2.*pi/360.
  conv=360./(2.*pi)
  rml=0
  do i=1,np
  r(i)=r(i)*(2.5/rm)
  if(r(i).gt.rml)then
    rml=r(i)
  endif
  enddo
  rmax=2.5
  type*, 'rml=',rml
  type*, 'rmax=',rmax
  endif
  rdf=0.0174532
  type*, 'do you want graphics:yes(1) or no(2)'
  read(5,*) iut
  if(iut.eq.1)
    do i=1,np
      xcart(i)=r(i)*cos(th(i)*rdf)
      ycart(i)=r(i)*sin(th(i)*rdf)
    enddo
    call z(xcart,ycart,np)
  endif
  s=0
  rs=0.0
  c=0
  rc=0
  do i=1,np
    s=s+sin(th(i)*rdf)
    rs=rs+r(i)*sin(th(i)*rdf)
    c=c+cos(th(i)*rdf)
    rc=rc+r(i)*cos(th(i)*rdf)
  enddo
  rbar=sqrt(s*s+c*c)/np
  write (6,*) 'rbar=',rbar
  sa=s/c
  if(c.lt.0)then
    pmu=pi+atan(sa)
  else
    if(s.lt.0.and.c.gt.0.0)then
      pmu=2*pi+atan(sa)
    else
      pmu=atan(sa)
    endif
  endif
  write (6,*), 'xbar=', pmu
  write (6,*), 'mean angle(degrees)=', pmu*67.296
  endif
  type*, 'type of analysis:'
  type*, 'vm: 3 parm (0)'
  type*, 'distance (1)'
  now choose form of analysis
c

type*, 'full mult (2)'
type*, 'full additive (3)'
type*, 'case control (4)'
type*, 'pop/smr:(5)'
type*, 'r/theta model(mult):(6)'
type*, 'r/theta model(add):(7)'
type*, 'mixed model: additive (8)'
type*, 'mixed model: multi (9)'
read(5,*), idor
if(idor.ge.2.and.idor.le.7)then
type*, 'type of distance model'
type*, 'weib(0), or Gamma(1)'
read(5,*), idismod
endif

c here we do vm 3parm

c
if(idor.eq.0)then
est1=(1./6.0)*rbar*(12+6*rbar**2.+5*rbar**4.)
est2=0.1
est3=pmu
write (6,*), 'suggested initial estimates of parameters:'
write (6,*), 'kappa, psi, mu'
write (6,*), est1, est2, est3
write (6,*), 'input your estimates'
read(5,*), x(1), x(2), x(3)
liw=5
lw=100

c write (6,*) 'enter lower and upper bounds for each parameter'
do i=1,3
a(i)=0.0
enddo
b(1)=50.0
b(2)=50.0
b(3)=7.0
type*, 'bounded(0) or unbounded(1) run'
read(5,*), ib
n2=3
ifail=1
call e04kaf(n2,ib,a,b,x,f,g,iworl,liw,work,lw,ifail)
if(ifail.ne.0)then
write (6,*), 'failure due to', ifail
endif
write(6,*), x(1), x(2), x(3)
type*, 'max lik value:', -f
type*, 'aic=', 2*f+8

c now calling hessian evaluation

c
call hess2(n2,x,hesl,3,hesd)
write (6,*), 'Hessian:'
write (6,*), 'Diagonal:', hesd(1), hesd(2), hesd(3)
write (6,*), 'psik, pmuk, pmupsi:', hesl(1), hesl(2), hesl(3)
do i=1,3
do j=1,i
if(i.eq.j) then
hinf(i,j)=-hesd(i)
else
hinf(i,j)=-hesl(j+(i-1)*(i-2)/2)
hinf(j,i)=hinf(i,j)
endif
enddo
enddo

c
Hinf is now the neg of hessian ie. I
Now invert I to give var/cov matrix

inverting Hessian

```
idgt=4
Id 1inv2f(hinf,3,3,hthinv,idgt,wkth,ier)
type*, 'hessian inverse:angular ses'
do i=1,3
write(6,*)(hthinv(i,j),j=1,3)enddo
type*, 'standard errors'
do i=1,3
type*, 'se', i, ')' = sqrt(hthinv(i,i))
endo
type*, 'do you want sum of squares :yes(1) or no(2)'
read(5,*) idlw
if(idlw.eq.1) then
write (6,*) 'now trying sums of squares'
write (6,*) 'estimates: original(l) or from found min(2)'
read(5,*) ifop
if(ifop.eq.1) then
x(1)=est1
x(2)=est2
x(3)=est3
endif
lw=100
m2=3
n2=3
liw=1
ifail=1
call e04fdm(m2,n2,x,fsum,wi1,liw,work,1w,ifail)
if(ifail.ne.0) then
write (6,*) 'ifail=', ifail
endif
write (6,*) 'parms=', x(1), x(2), x(3)
endif
pk=x(1)
ps=x(2)
mmu=x(3)
endif
```

c end of vm 3 parm estimation
c select dist estimation here

```
if(idor.eq.1) then
iob=1
if(iob.eq.1) then
type*, 'distance estimation'
type*, 'weibull(1) or 2 parameter(2)'
type*, 'or Gamma(3)'
read(5,*) ider
x(2)=1
sl=0.0
do i=1,np
sl=sl+r(i)**x(2)
endo
x(1)=(np/sl)**(1/x(2))
type*, 'parms=', x(1), x(2)
nsig=3
maxfn=500
n=2
```
nh=n*(n+1)/2
nsrch=9
a(1)=0.0001
a(2)=1.0
b(1)=200.0
b(2)=200.0
if(ider.eq.1)then
  iopt=2
  ib=0
  liw=5
  n2=2
  lw=100
call e04kaf(n2,ib,a,b,x,f,g,iwork,liw,work,lw,ifail)
if(ifail.ne.0)then
  write(6,*),'failure due to:', ifail
endif
endif
if(ider.eq.2)then
  iopt=2
  call zxmwd(f2parm,n,nsig,a,b,nsrch,x,f,wik,iwork,ier)
endif
if(ider.eq.3)then
  iopt=2
  call zxmin(fgamma,n,nsig,maxfn,iopt,x,hesl,g,h,w,ier)
endif
endif
endf

end of distance estimation

do full multiplicative model here

    if(idor.eq.2.or.idor.eq.3.or.idor.eq.4.or.idor.eq.5)then
      do full add-mult-casec model here
do i=1,5
x(i)=1.01
enddo
x(2)=1.01
do i=1,5
a(i)=0.001
enddo
a(2)=1.0
do i=1,4
b(i)=20.0
enddo
b(2)=30.0
b(5)=6.283
type*, 'bounded(0) or not(1):'
read(5,*)ib
if(idor.eq.4.or.idor.eq.5)then
type*, 'enter filename:'
read(5,88)wname
open(26, file=wname, status='old')
read(26,*)npnp
if(idor.eq.4)then
do ik=1,npnp
read(26,*)caser(ik),caseth(ik)
enddo
do ik=1,npnp
snca(ik)=1.
endo
type*, 'smoother:'
read(5,*)hsm
else
if(idor.eq.5)then
type*, 'values at points(0) or interpolated(1),'
read(5,*)interp
do ik=1,npnp
read(26,*)caser(ik),caseth(ik),snca(ik)
endo
type*, 'smoother:'
read(5,*)hsm
endif
endif
do smoothing here
c
n2=20
m2=20
h1=rmax/n2
hx1=pi2/m2
bot=2.*hsm*hsm
do i=0,n2
s=i*h1
ab=s/(pi2*npnp*hsm*hsm)
do j=0,m2
phi2=j*hx1
c1=0
fh(i,j)=dense(s,phi2,hsm,caser,caseth,snca,npnp)
endo
endo
type*, 'done interpolation'
sS3=0
do i=1,np
s=r(i)
ph2=th(i)*0.01745
if(idor.eq.5.and.interp.eq.0)then
at1=snca(i)
else
at1=dense(s,ph2,hsm,caser,caseth,snca,npnp)
endif
s53=s53+log(at1)
enddo
cconst=s53
type*, 'cconst=', cconst
endif
n2=5
ifail=1
lw=100
liw=10
l1w=10
stype*, 'zxmwd(l) or e04jaf(2)'
read(5,*)isti
if(isti.eq.1)then
nsr=40
nsig=3
call zxmwd(functl,n2,nsig,a,b,nsr,x,f,work,iwork,ier)
else
call e04jaf(n2,ib,a,b,x,f,iwork,liw,work,lw,ifail)
endif
type*, 'final estimates:', x(1), x(2), x(3), x(4), x(5)
if(ifail.ne.0)then
type*, 'error=', ifail
endif
type*, 'ests:', x(1), x(2), x(3), x(4), x(5)
a1=-f+cconst
type*, 'max 1=', a1
endif
if(idor.eq.6.or.idor.eq.7)then
do i=1,5
x(i)=1.01
enddo
do i=1,5
a(i)=0.0001
enddo
do i=1,4
b(i)=30.0
enddo
b(5)=6.283
type*, 'bounded(0) or not(1):'
read(5,*)ib
n2=5
ifail=1
lw=100
liw=10
nsr=40
nsig=3
type*, 'zxmwd(l) or e04jaf(2)'
read(5,*)isti
if(isti.eq.1)then
call zxmwd(functl,n2,nsig,a,b,nsr,x,f,work,iwork,ier)
else
call e04jaf(n2,ib,a,b,x,f,iwork,liw,work,lw,ifail)
endif
type*, 'final estimates:', x(1), x(2), x(3), x(4), x(5)
type*, 'final max 1:', -f
endif
if(idor.eq.8.or.idor.eq.9)then
type*, 'enter filename:'
read(5,88)zname
open(19,file=zname,status='old')
read(19,*)nreg
do j=1,nreg
  read(19,*)rreg(j),threg(j),popreg(j)
enddo
close(19)
do i=1,5
  x(i)=1.01
enddo
do i=1,5
  a(i)=0.0001
enddo
  a(2)=1.0
  do i=1,4
    b(i)=30.0
  enddo
  b(5)=6.283
type*,'bounded(0) or not(1):'
  read(5,*)ib
n2=5
ifail=1
lw=100
liw=10
nsr=40
nsig=3
call e04jaf(n2,ib,a,b,x,f,iwork,liw,work,lw,ifail)
type*,'final estimates:',x(1),x(2),x(3),x(4),x(5)
type*,'fina max l:',-f
endif
c counts of observations

type*,'input no of angle classes'
type*,'input no of distance classes'
read(5,*)nth,nr
thwid=6.283185/float(nth)
rwid=2.5/float(nr)
thclass(1)=0.0
rclass(1)=0.0
do i=2,nth+1
thclass(i)=thclass(i-1)+thwid
enddo
do j=2,nr+1
rclass(j)=rclass(j-1)+rwid
endo
do j=1,nr
  do k=1,nth
    nrob(j,k)=0
  enddo
endo
do i=1,np
do j=2,nr+1
  if(r(i).le.rclass(j).and.r(i).gt.rclass(j-1))then
    do k=2,nth+1
      aw=th(i)*0.01745
      if(aw.lt.thclass(k).and.aw.ge.thclass(k-1))then
        nrob(j-1,k-1)=nrob(j-1,k-1)+1
      endif
    enddo
  endif
endo
do k=1,nth
endo
do j=1,nr
  do k=1,nth
write(6,*)'number in cell:',j,k,'is ',nrob(j,k)
endo
enddo

type*, 'no of points=', np

type*, 'finished binning the data'

type*, 'do you want whole fit(0) or marginal r fit(1):'

read(5,*) imarg

c now calculate expected values

c

type*, 'parms:', pl, pd, pk, ps, pmu

sl0=0.0

sl1=0.0

s12=0.0

s13=0.0

s14=0.0

aerr=0.0001

npn=0

do j=1, nr

do k=1, nth

npn=npn+1

xcart(npn)=(rclass(j)+rwid/2.)*cos(thclass(k)+thwid/2.)

ycart(npn)=(rclass(j)+rwid/2.)*sin(thclass(k)+thwid/2.)

ar=rclass(j)

sath=thclass(k)

br=rclass(j+1)

sbth=thclass(k+1)

cres=dblin(fun, ar, br, ath, bth, aerr, error, ier)

prob(j, k)=cres

ncex(j, k)=np*cres

if(ncex(j, k).eq.0.)then

ncex(j, k)=0.00001

dendif

res(j, k)=ncob(j, k)-ncex(j, k)

resp(npn)=res(j, k)

ansc(j, k)=res(j, k)/sqrt(ncex(j, k))

s10=s10+res(j, k)

s11=s11+prob(j, k)*(.prob(j, k))

s13=s13+(res(j, k)*res(j, k)/ncex(j, k))

if(ncob(j, k).gt.0)then

s14=s14+ncob(j, k)*log(ncob(j, k)/ncex(j, k))

dendif

enddo

enddo

c writing to file

c

type*, 'info to file: yes(1) or no(2)'

read (5,*) idpe

if(idpe.eq.1)then

type*, 'enter filename:'

read(5,88)tname

open(20, file=tname, status='new')

write(20,*) nr, nth

do j=1, nr

do k=1, nth

write(20,*) ncob(j, k), ncex(j, k), res(j, k), ansc(j, k)

endo

enddo

do i=1, npn

write(20,*) xcart(i), ycart(i)

endo

close(20)

dendif

type*, 'resids and info printed:yes(1) or no(2)'

read(5,*) iov
if iov.eq.1 then
  do j=1,nr
    write(6,*)'rcalass:',j
    write(6,*) 'observed expected residual anscombe'
  do k=1,nth
    write(6,*) 'angle class:',k
    write(6,*)ncob(j,k),ncex(j,k),res(j,k),ansc(j,k)
  endo
endo
dendif

c residual test
c do kl=1,nr*nth-1
  do il=kl+1,nr*nth
    il=1+(kl-1)/nth
    jl=kl-(il-1)*nth
    i2-il-(i2-1)*nth
    sl2=sl2+prob(il,jl)*prob(i2,j2)
  enddo
endo
nreg=nr*nth
ptl=s10/nreg
pt2=(sll-2.*s12)/nreg

c here we replace test with chi squared
c testlr=2*s14
  testx2=s13
  type*, 'res test is:',testlr,testx2
  sfd=nr*nth-6
  call mdch(testlr,sfd,pr,ier)
type*, 'chi-squared prob:',pr
type*, 'for lr test'
type*, 'with df:',sfd
call mdch(testx2,sdf,pr,ier)
type*, 'chi-squared prob:',pr
type*, 'for x2 test'
type*, 'with df:',sdf

c graphics for surfaces
c con=2.*pi/360.
npn=0
type*, 'residual plot now follows'
type*, 'cntrl-shift f6+1'
read(5,*)igls
call grz(xcart,ycart,resp,nreg)
type*, 'do you want surfaces:yes(1) or no(2)'
read(5,*)iug
if(iug.eq.1)then
  do j=1,nr
    do k=1,nth
      rcell(j,k)=rclass(j)+rwid/2.
      thcell(j,k)=thclass(k)+thwid/2.
      npn=npn+1
      xcart(npn)=rcell(j,k)*cos(thcell(j,k))
      ycart(npn)=rcell(j,k)*sin(thcell(j,k))
      zed(npn)=ncob(j,k)
    endo
  endo
ncont=17
ism=1
iview=0

read(5,*)isd

if(isd.eq.1)then
    call gino
call t4010
else
call gino
call scale(20.0)
call call044
dendif

call raniso(npn,xcart,ycart,zed,iview)
type*, 'cntrl-sh f6'
read(5,*)ish
call rancon(npn,xcart,ycart,zed,ncont,ism)
call ginend
call devend
dendif
stop
dendif

end

c functions and subs
c

subroutine grz(x,y,rs,n)
real x(n),y(n),r(500),th(500)
real rs(n)
real*8 pi
integer n
common/sl/pi
rscreen=2.5
rm=0.0

do i=1,n
    r(i)=sqrt(x(i)*x(i)+y(i)*y(i))
    if(x(i).eq.0.and.y(i).lt.0)then
        th(i)=4. 7123
    else
        if(x(i).eq.0.and.y(i).gt.0)then
            th(i)=1.5708
        else
            h=atan(y(i)/x(i))
            if(x(i).lt.0)then
                th(i)=h+pi
            else
                if(y(i).le.0)then
                    th(i)=h+2*pi
                else
                    th(i)=h
            endif
        endif
    endif
    continue
    if(r(i).ge.rm)then
        rm=r(i)
    endif
endo

do i=1,n
    r(i)=r(i)*(rscreen/rrn)
endo

c functions and subs
c

subroutine grz(x,y,rs,n)
real x(n),y(n),r(500),th(500)
real rs(n)
real*8 pi
integer n
common/sl/pi
rscreen=2.5
rm=0.0

do i=1,n
    r(i)=sqrt(x(i)*x(i)+y(i)*y(i))
    if(x(i).eq.0.and.y(i).lt.0)then
        th(i)=4. 7123
    else
        if(x(i).eq.0.and.y(i).gt.0)then
            th(i)=1.5708
        else
            h=atan(y(i)/x(i))
            if(x(i).lt.0)then
                th(i)=h+pi
            else
                if(y(i).le.0)then
                    th(i)=h+2*pi
                else
                    th(i)=h
            endif
        endif
    endif
    continue
    if(r(i).ge.rm)then
        rm=r(i)
    endif
endo

do i=1,n
    r(i)=r(i)*(rscreen/rrn)
endo

iview=0

type*, 'screen(1) or plotter(2)'
read(5,*)isd

if(isd.eq.1)then
    call gino
call t4010
else
call gino
call scale(20.0)
call call044
dendif

call raniso(npn,xcart,ycart,zed,iview)
type*, 'cntrl-sh f6'
read(5,*)ish
call rancon(npn,xcart,ycart,zed,ncont,ism)
call ginend
call devend
dendif
stop
dendif

end

c functions and subs
c

subroutine grz(x,y,rs,n)
real x(n),y(n),r(500),th(500)
real rs(n)
real*8 pi
integer n
common/sl/pi
rscreen=2.5
rm=0.0

do i=1,n
    r(i)=sqrt(x(i)*x(i)+y(i)*y(i))
    if(x(i).eq.0.and.y(i).lt.0)then
        th(i)=4. 7123
    else
        if(x(i).eq.0.and.y(i).gt.0)then
            th(i)=1.5708
        else
            h=atan(y(i)/x(i))
            if(x(i).lt.0)then
                th(i)=h+pi
            else
                if(y(i).le.0)then
                    th(i)=h+2*pi
                else
                    th(i)=h
            endif
        endif
    endif
    continue
    if(r(i).ge.rm)then
        rm=r(i)
    endif
endo

do i=1,n
    r(i)=r(i)*(rscreen/rrn)
endo

iview=0

type*, 'screen(1) or plotter(2)'
read(5,*)isd

if(isd.eq.1)then
    call gino
call t4010
else
call gino
call scale(20.0)
call call044
dendif

call raniso(npn,xcart,ycart,zed,iview)
type*, 'cntrl-sh f6'
read(5,*)ish
call rancon(npn,xcart,ycart,zed,ncont,ism)
call ginend
call devend
dendif
stop
dendif

end
call t4010
call units(25.0)
else
call gino
call call044
call scale(20.0)
endif
call chapos(20.0,20.0)
call piccle
rsn=rscreen+0.3
call shift2(rsn,rsn)
call movto2(-rsn,0.0)
call arcto2(0.0,0.0,-rsn,-0.001,0)
call movto2(0.0,0.0)
call symbol(5)
do i=1,n
ax=x(i)-0.2
ay=y(i)-0.1
call movto2(ax,ay)
call chaint(rs(i),3)
enddo
type*, 'cntrl-sht f6 1'
read(5,*)neft
call piccle
call devend
call ginend
type*, 'more graphics(l:yes)'
read(5,*)inore
if(inore.eq.1)then
  goto 5678
endif
return
end
c derivatives used here for eo4kcf
c subroutine funct1(n,xc,fc)
implicit real*8 (a-h,o-z)
real*8 xc(n),fc
real*8 mmbsi0,mmbsil
real*8 alam,del,akap,psi,pmu5
real anorm1
common/s1/pi
common/s4/rmax
common/s9/idor
common/s40/alam,del,akap,psi,pmu5
common/s70/cconst
rdf=0.01745
it=1
if(idor.eq.2)then
  it=1
endif
if(idor.eq.3)then
  it=2
endif
s20=0
if(idor.eq.6)then
  it=6
endif
if(idor.eq.7)then
  it=7
endif
if(idor.eq.9)then
  it=1
endif
if(idor.eq.8)then
it=2
endif
alam=xc(1)
del=xc(2)
akap=xc(3)
psi=xc(4)
mu5=xc(5)
type*, 'parms:', xc(1), xc(2), xc(3), xc(4), xc(5)
do i=l, np
thc=th(i)*rdf
as=tense(alam, del, akap, psi, mu5, r(i), thc, it)
if(as.eq.0.0)then
as=0.0001
endif
s20=s20+log(as)
enddo
ar=anorml(rmax, alam, del, it)
type*, 'norm const:', ar
if(ar.eq.0.0)then
ar=0.0001
endif
if(idor.eq.4.or.idor.eq.5)then
fc=-s20-cconst+np*log(ar)
else
fc=-s20+np*log(ar)
endif
type*, 'current lik value=', -fc
return
end
subroutine funct2(n, xc, fc, gc)
implicit real*8 (x, f, a, p, g, r, t, s)
integer n
real*8 xc(n), fc, gc(n)
real*8 mmbsi0, mmbsil
real*8 alam, del, akap, psi, mu5
real anorml
common/s1/pi
common/s4/rmax
common/s9/idor
common/s40/alam, del, akap, psi, mu5
if(idor.eq.0)then
xl=xc(l)
x2=xc(2)
x3=xc(3)
io=1
s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
s7=0
s8=0
rdf=0.01745
write (6,*) 'parm values:', xl, x2, x3
do i=l, np
s1=s1+cos(th(i)*rdf-x3)
s3=s3+r(i)*cos(th(i)*rdf-x3)
s2=s2+sin(th(i)*rdf-x3)
s4=s4+r(i)*sin(th(i)*rdf-x3)
ao=x1+x2*r(i)
a2 = mmbsi0 (io, ao, ier)
a1 = mmbsil (io, ao, ier) / a2
s5 = s5 + a1
s6 = s6 + r(i) * a1
s7 = s7 + log(a2)
s8 = s8 + log(rmax * r(i))
enddo
correct = -s8
gc(1) = -s1 + s5
gc(2) = -s3 + s6
gc(3) = -x1 * s2 - x2 * s4
write (6, *) 'gradients:', gc(1), gc(2), gc(3)
fc = -x1 * s1 - x2 * s3 + s7
type*, 'lik=' , -fc
type*, 'correction (uniform r):' , correct
else
c distance estimation
c
if (idor.eq.1) then
x1 = xc(1)
x2 = xc(2)
s1 = 0
s2 = 0
s3 = 0
do i = 1, np
s1 = s1 + r(i) ** x2
s2 = s2 + log(r(i))
s3 = s3 + log(r(i)) * r(i) ** x2
enddo
correct = -np * 1.8378
a1 = x1 * rmax ** x2
a2 = exp(-a1)
a3 = a1 * log(rmax)
a4 = 1 - a2
gc(1) = -((np/x1) - s1 - np * rmax ** x2 * a2 / (a4))
gc(2) = -((np/x2) + s2 - x1 * s3 - np * a3 * a2 / (a4))
write (6, *) 'gradients:', gc(1), gc(2)
type*, 'parms=' , x1, x2
fc = -np * log(x1 * x2) + (x2 - 2) * s2 - x1 * s1 - np * log(1 - exp(-a1))
type*, 'lik=' , -fc
type*, 'correction (uniform angle):' , correct
endif
endif
if (idor.eq.2 .or. idor.eq.3) then
if (idor.eq.2) then
it = 1
endif
if (idor.eq.3) then
it = 2
endif
s20 = 0
alam = xc(1)
del = xc(2)
akap = xc(3)
psi = xc(4)
pmu5 = xc(5)
do i = 1, np
as = tense (alam, del, akap, psi, pmu5, r(i), th(i), it)
s20 = s20 + log(as)
enddo
c gradients here in gc(n)
c
do i = 1, 5
```fortran

gc(i) = grad(i, it)
enddo
ar = anorml(rmax, alam, del, it)
type*, 'norm const:', ar
type*, 'number of points:', np
fc = s20 + np*log(ar)
endif
return
end

real function dense(s, ph2, hsm, r, th, cn, np)
implicit real*8 (a-h, o-z)
ingear np
real*8 s, ph2, hsm
real*8 r(np), th(np)
real*8 cn(1000)
pi = 3.145
pi2 = 2*pi
rdf = 0.01745
ab = s/(pi2*np*hsm*hsm)
bot = 2.*hsm*hsm
c1 = 0.0

do il = 1, np
r5 = r(il)
r6 = th(il)
rh = 2.*s*r5*cos(ph2 - r6*rdf)
c1 = c1 + cn(il)*exp(-(s*s + r5*r5 - rh)/bot)
enddo
dense = c1*ab
return
end

function tense(alam, del, akap, psi, pmu, r, th, itype)
implicit real*8 (x, f, a, p, g, r, t, s)
real*8 mmbsiO
real*8 alam, del, akap, psi, pmu, r, th
real aqw, adel
integer itype
common/s9/idor
common/s4/rmax
common/s60/idismod
pi = 3.14159
pi2 = 2*pi
iO = 1
if(itype.ne.6.or.itype.ne.7) then
adel = del
aqw = alam*r**adel
a = alam*adel*(r**(adel - 1.))*exp(-aqw)
enddo
a5 = alam*r
a6 = exp(-a5)*r**adel
gb = akap + psi*r
rdf = 0.01745
ser = gb*cos(th - pmu)
gc = exp(ser)
iO = 1
d = pi2*mmbsiO(iO, gb, ier)
type*, 'a, gc, r, d:', a, gc, r, d
if(r.eq.0.0) then
art = 0.00001
else
art = r
endif
if(idismod.eq.1) then
a = a6
endif
dint = a*gc/(art*d)
```
if(itype.eq.1) then
  tense=dint
else
  if(itype.eq.2) then
    tense=l.+dint
  endif
endif
else
  c note akap=dell, psi=kappa
  adel=del+akap*cos(th-pmu)
  al=alam*(adel)*(r**(adel-2.))
  a2=exp(-alam*r**(adel))
  a3=exp(psi*cos(th-pmu))
  a4=pi2*mmbis10(i0,psi,ier)
  abl=exp(-alam*r)
  ab2=r**(adel-1.)
  if(idismod.eq.1) then
    al=abl
    a2=ab2
  endif
  dint=al*a2*a3/(a4)
  if(itype.eq.6) then
    tense=dint
  endif
  if(itype.eq.7) then
    tense=l.0+dint
  endif
endif
return
end

real function ay(r)
ay=0.0001
return
end

real function by(r)
by=6.28319
return
end

real function funten(r,th,il,jl)
implicit real (a-h,o-z)
real *8 alam,del,akap,psi,pmu5
common/s40/alam,del,akap,psi,pmu5
common/s9/idor
common/s42/itype
common/s50/fh(0:100,0:100)
rdf=0.01745
pi2=6.28319
xl=r*cos(th)
yl=r*sin(th)
if(idor.eq.4.or.idor.eq.5) then
  fun5=fh(il,jl)
  funten=fun5*r*tense(alam,del,akap,psi,pmu5,r,th,itype)
else
  funten=r*tense(alam,del,akap,psi,pmu5,r,th,itype)
endif
return
end

real function anorm1(rmxl,all,dell,itype)
implicit real*8 (x,f,a,p,g,r,t,s)
integer itype
real*8 alam,del,akap,psi,pmu5
real*8 rmxl,all,dell
real ax,bx,ayl,byl,awe
common/s9/idor
common/s80/nreg,rreg(1000),threg(1000),popreg(1000)
common/s40/alam,del,akap,psi,pmu5
external funten
external ay,by
pi=3.14159
pi2=2*pi
area=pi*rmxl*rmxl
aint=1.-exp(-all*rmxl**del)
if(idor.eq.8.or.idor.eq.9)then
  rmax=-20
  do j=1,nreg
    if(rreg(j).gt.rmax)then
      rmax=rreg(j)
    endif
  enddo
  s0=0
  sl=0
  do j=1,nreg
    r=rreg(j)/rmax
    th=threg(j)
  type*, 'lamda,del,akap,psi,pmu5,r,th'
  c type*,alam,del,akap,psi,pmu5,r,th
  s0=s0+tense(alam,del,akap,psi,pmu5,r,th,itype)
  enddo
if(idor.eq.8)then
  anorml=s0
else
if(idor.eq.9)then
  anorml=nreg+s0
endif
endif
else
if(idor.ge.2.and.idor.le.7)then
  ax=0.0001
  type*, 'rmxl=',rmxl
  bx=rmxl
  aerr=0.0001
  ay1=0.0001
  by1=6.28319
  m2=20
  n2=20
  call doblin(ax,bx,ay1,by1,m2,n2,awe)
  aint=awe
  type*, 'integral=',awe
endif
if(itype.eq.1)then
  anorml=aint
else
if(itype.eq.2)then
  anorml=area+aint
endif
endif
endif
endif
type*, 'anorml=',anorml
return
end
real function grad(ione,it)
implicit real*8 (x,f,a,p,g,r,t,s)
real*8 x(10)
real*8 alam,del,akap,psi,pmu5
real anorml
integer ione,it
common/s40/alam,del,akap,psi,pmu5  
common/s4/rmax  
h=0.01  
x(1)=alam  
x(2)=del  
x(3)=akap  
x(4)=psi  
x(5)=pmu5  
s20=0  
do i=l,np  
at=tense(x(1),x(2),x(3),x(4),x(5),r(i),th(i),it)  
s20=s20+log(at)  
enddo  
ah=anorm1(rmax,alam,del,it)  
type*, 'norm const:', ah  
type*, 'no of points:', np  
l1=-s20+np*log(ah)  
x(ione)=x(ione)+h  
s20=0  
do i=l,np  
ak=tense(x(1),x(2),x(3),x(4),x(5),r(i),th(i),it)  
s20=s20+log(ak)  
enddo  
aw=anorm1(rmax,x(1),x(2),it)  
l2=-s20+np*log(aw)  
grad=(l2-l1)/h  
return  
end  
subroutine lsfun1(m,n,xc,fvecc)  
imPLICIT real*8 (p,r,t,s,x,a,f)  
integer m,n  
real*8 xc(n),fvecc(m)  
real*8 rnmbsiO,rnmbsil  
cornmon/sl/pi  
x1=xc(1)  
x2=xc(2)  
x3=xc(3)  
io=1  
s1=0  
s2=0  
s3=0  
s4=0  
s5=0  
s6=0  
s7=0  
rdf=0.01745  
write (6,*) 'parm values:',x1,x2,x3  
do i=l,np  
s1=s1+cos(th(i)*rdf-x3)  
s3=s3+r(i)*cos(th(i)*rdf-x3)  
s2=s2+sin(th(i)*rdf-x3)  
s4=s4+r(i)*sin(th(i)*rdf-x3)  
ao=x1*x2*r(i)  
a2=mmbsi0(io,ao,ier)  
al=mmbsil(io,ao,ier)/a2  
s5=s5+al  
s6=s6+r(i)*al  
s7=s7+log(a2)  
enddo  
fvecc(1)=s1-s5  
fvecc(2)=s3-s6  
fvecc(3)=x1*s2+x2*s4  
write (6,*) 'resids=',fvecc(1),fvecc(2),fvecc(3)
subroutine hess2(n, xc, hl, lh, hd)
  implicit real*8 (p, r, t, s, x, a)
  integer lh, n
  real*8 mmbsi0, mmbsil
  real*8 hd(n), hl(lh), xc(n)
  common/s1/pi
  common/s3/s1, s2, s3, s4, s5, s6, s7, s8, s9, s10
  io=1
  rdf=0.01745
  x1=xc(1)
  x2=xc(2)
  x3=xc(3)
  s1=0
  s2=0
  s3=0
  s4=0
  s5=0
  s6=0
  s7=0
  s8=0
  s9=0
  s10=0
  do i=1, np
    ao=x1+x2*r(i)
    a1=mmbsi1(io, ao, ier)
    a2=a1/mmbsi0(io, ao, ier)
    a3=1.-a2*a2-a2/ao
    sl=sl+cos(th(i)*rdf-x3)
    s2=s2+sin(th(i)*rdf-x3)
    s3=s3+r(i)*cos(th(i)*rdf-x3)
    s4=s4+r(i)*sin(th(i)*rdf-x3)
    s5=s5+a1
    s6=s6+r(i)*a1
    s7=s7+log(a2)
    s8=s8+r(i)*r(i)*a3
    s9=s9+r(i)*a3
    s10=s10+a3
  enddo
  hd(1)=-s10
  hd(2)=-s8
  hd(3)=-x1*s1-x2*s3
  hl(1)=-s9
  hl(2)=s2
  hl(3)=-s4
  return
end
subroutine z(x, y, n)
  real*8 x(n), y(n)
  integer n
  r=2.5
type*, 'screen(1) or plotter (2)'
read(5, *) iaw
if(iaw.eq.1)then
call gino
call t4010
call units(25.0)
call piccle
else
call gino
call call044
call scale(20.0)
endif

call shift2(r,r)
call movto2(-r,0.0)
call arcto2(0.0,0.0,-r,-0.001,0)
call movto2(0.0,0.0)
call symbol(5)
do i=1,n
call movto2(x(i),y(i))
call symbol(3)
enddo
call devend
call ginend
type*, 'cont:1 return'
read(5,*)nht
return
c

hessian of weibull

subroutine whess(pl,pd,h)
this routine returns values of -I for Weibull

implicit real*8 (a,b,c,d,e,f,g,s,t,r,p)
real*8 h(3)
common/s4/rmax
s1=0.0
s2=0.0
s3=0.0
do i=1,n
s1=s1+(r(i)**pd)*log(r(i))**2.
s2=s2+(r(i)**pd)*log(r(i))
enddo
a1=np/(p1*p1)
a2=pl*rmax**pd
a5=exp(-a2)
abx=log(1.*nr)+(2*1.*nr)*log(rmax)-a2-2*log(1.-exp(-a2»
h(1)=a1-exp(abx)
b1=np/(pd*pd)
b2=a2**log(rmax)
b3=a2*(log(rmax)**2.)
b4=b3-b2*b2/(1.-exp(-a2))
b5=1.-a2/(1.-exp(-a2))
h(2)=b1+pl*(s1+(np*a5/(1.-a5))*b4)
h(3)=s2+(np*(rmax**pd)*log(rmax)/(1.-exp(-a2)))**b5
h(1)=-h(1)
h(2)=-h(2)
h(3)=-h(3)
return
end

subroutine fweib(n,xc,fc)
implicit real*8 (x,f,a,s,r,t,p)
real*8 xc(n),fc
common/s1/pi
common/s4/rmax
rdf=2.*pi/360.
s1=0.0
s2=0.0
do i=1,n
s1=s1+log(r(i))
s2=s2+r(i)**xc(2)
enddo
az=xc(1)
as=xc(2)
ab=(np*log(az*as)+(as-1)*sl-
laz*s2)
fc=-ab+np*log(1.-exp(-az*rmax**as))
type*, 'fc value=', fc
type*, 'parms=', xc(1), xc(2)
return
end

2 parm likelihood

subroutine f2parm(n,xc,fc)
implicit real*8 (x,f,a,s,r,t,p)
real*8 xc(n),fc
common/s1/pi
common/s4/rmax
rdf=2.*pi/360.
s1=0.0
s2=0.0
s3=0.0
do i=1,np
sl=sl+log(1.0+r(i))
s2=s2+log((1.0+r(i))*xc(2)-xc(1))
s3=s3+r(i)
enddo
ac=(xc(1)-1.)*sl
ad=xc(2)*s3
ae=np*log(1.-(1.+rmax)**xc(1))*exp(-
lxc(2)*rmax))
fcc=-ac+s2-ad-ae
return
end

Gamma likelihood

subroutine fgamma(n,xc,fc)
implicit real*8 (x,f,a,s,r,t,p)
real*8 xc(n),fc
dcadre,gam
common/s4/rmax
common/s8/st(2)
s1=0.0
s2=0.0
a=0.0
b=xc(2)*rmax
do i=1,2
st(i)=xc(i)
endo
do i=1,np
sl=sl+log(r(i))
s2=s2+r(i)
endo
rerr=0.
aerr=1.0e-5
g=dcadre(gam,a,b,aerr,rerr,error,ier)
ab=np*log((xc(2)**xc(1))/g)
fcc=-(xc(1)-1)*sl-xc(2)*s2+ab
return
end
function gam(y)
implicit real*8 (s)
real*8 y
common/s8/st(2)
gam=(y**(st(l)-l))*exp(-y)
return
end

function fun(r,th)
implicit real*8 (r,t,p,b,c,a)
real*8 r,th
real*8 mmbsi0
common/s5/pl,pd
common/s4/rmax
common/s6/pk,ps,pmu
common/s10/imarg
a=(pl*pd*r**(pd-1.))*exp(-pl*r**pd)
b=1.-exp(-pl*rmax**pd)
c2=pk+ps*r
c1=exp(c2*cos(th-pmu))
c3=mmbsi0(1,c2,ier)
c4=c1/(c3*6.28318)
c5=a/b
if(imarg.eq.1)then
fun=c5/6.28318
else
fun=a*c4/b
endif
return
end

function ath(r)
implicit real*8 (s)
real*8 r
common/s7/sath,sbth
ath=sath
return
end

function bth(r)
implicit real*8 (s)
real*8 r
common/s7/sath,sbth
bth=sbth
return
end
File _$255$DUA2: [MC.T.AL]VMLIK.FOR;19 (9835,4,0), last revised on 14-MAY-1990 11:45, is a 13 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 57 bytes.

Job VMLIK (622) queued to SYS$PRINT on 24-AUG-1990 13:05 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:20 from queue LASER.
character*10 fname
real 11,12,13,14,15
real g(3), x(3)
real hest(3)
real hesd(3)
real hinf(2,2), hthinv(2,2)
real stparm(30,8)
real bu(3), bl(3), delta(3), hesl(6)
real f, w(9), work(40)
real wk(36), fnorm
real a(3), b(3)
real w1(3), w2(3), w3(3), w4(3), w5(4), w6(4,3)
real x02aae
real xw(2), xt(3)
integer iwork(3), nsrch
integer iworl(5)
real mmbs10, al
real mmbs11
real par(3)
integer istate(3), iw(3), liw, lw
real lap, moore
external fvonm, fweib, fcard, f2parm
external fvmn
external fvmtest
external vm2parm
external monit
external charfunc
common/s1/pi
common/s3/rmax
common/s4/covp(10,2000), nc
common/s5/psi, pmu, kap, delt
common/s6/s, rs, c, rc
common/s8/n
common/s9/fmax, x1, x2, x3
common/s10/rbar, sa
common/s11/npa, nsa, nsb
common/s12/ixe3
common/s13/maxp
ifl2=0
if(ifl2.eq.0)then
  type*, 'enter filename:'
  read(5,88)fname
  format(a10)
  open(1, file=fname, status='old')
  read(1,*) np
  do i=1, np
    read(1,*) r(i), th(i)
    write(5,*) r(i), th(i)
  enddo
endif

calculate max radius
rmax=0
do i=1, np
  if(r(i).gt.rmax)then
    rmax=r(i)
  endif
enddo
  type*, 'enter own max radius(l) or default(2),'
  read(5,*) ise0
  if(iseo.eq.2)then


```fortran
rm=1.1*rmax
else
type*,'enter radius value:'
read(5,*)rm
endif
pi=3.141592
rdf=2.*pi/360.
conv=360./(2.*pi)
do i=1,np
r(i)=r(i)*(2.5/rm)
enddo
rmax=rm

M L estimators
ifl4=1
if(ifl4.eq.1)then
s=0
rs=0
c=0
rc=0
do i=1,np
s=s+sin(th(i)*rdf)
rs=rs+r(i)*sin(th(i)*rdf)
c=c+cos(th(i)*rdf)
rc=rc+r(i)*cos(th(i)*rdf)
enddo
rbar=sqrt(s*s+c*c)/np
type*, 's=',s
type*, 'rs=', rs
type*, 'c=', c
type*, 'rc=', rc
type*, 'rbar=', rbar
endif

initialise parameters

x(1)=(1/6)*rbar*(12+6*rbar**2.+5*rbar**4.)
x(2)=0.1
sa=s/c
type*, 'initial estimates of parameters'
type*, 'psi=', x(1)
type*, 'kappa=', x(2)
type*, 'sa=', sa
if(c.lt.0)then
x(3)=pi+atan(sa)
else
x(3)=atan(sa)
endif
type*, 'mean angle(radians):', x(3)
xdeg=x(3)*360/(2*pi)
type*, 'mean angle(degrees):', xdeg
type*, 'enter your initial estimates:psi,mu:'
x(2)=x(3)
read(5,*)x(1),x(2)

start of opt choices:   iopl=1,2,3

iopl=3
n=2
type*, 'enter accuracy: no of sig digits:'
```
read(5,*)nsig
type*, 'max function evaluations:'
read(5,*)maxfn
nh=n*(n+1)/2
iopt=0
nsrch=13
npa=np
nsa=1
nsb=np
type*, 'enter constraints:'
type*, 'enter 2 lower values and then 2 upper values'
type*, 'for k, and mu (radians)'
read(5,*) (a(i), i=1,2)
read(5,*) (b(i), i=1,2)
call zxmwd(fvrntest,n,nsig,a,b,nsrch,x,f,work,iwork,ier)
type*, 'parm estimates:', x(1), x(2)
type*, 'max lik value:', -f
s62=0
do i=1, np
s62=s62+log(r(i)*rmax)
enddo
cconstl=-s62
type*, 'lik const (uniform r):', cconstl
type*, 'aic =', 2*f + 6
type*, 'asymp variance of k is:'
iopt=1
ak=mmb511(iopt,x(l),ier)/mmb510(iopt,x(l),ier)
vmu=1./(np*x(l)*ak)
vk=1./(np*(1.-ak*ak-ak/x(l)))
type*, vk
type*, 'asymp variance of mu is:'
type*, vmu
n2=2
call hess2(n2,x,hesl,1,hesd)
write (5,*) 'Hessian:
write (5,*) 'Diagonal:', hesd(1), hesd(2)
write(5,*) 'pmuk:', hesl(1)
do i=1, 2
do j=1, i
if (i.eq.j) then
hinf(i,j)=-hesd(i)
else
hinf(i,j)=-hesl(j+(i-1)*(i-2)/2)
hinf(j,i)=hinf(i,j)
endif
endo
dendo
c Hinf is now the neg of hessian ie. I
Now invert I to give var/cov matrix
inverting Hessian
idgt=4
call linv2f(hinf,2,2,hthinv,idgt,wkth,ier)
type*, 'hessian inverse: angular ses'
do i=1, 2
write(6,*)(hthinv(i,j),j=1,2)
endo
stop
end
functions and subroutines

subroutine hess2(n, xc, hl, lh, hd)
integer lh, n
real mmbsi0, mmbsil
real hd(n), hl(lh), xc(n)
common/s1/pi
common/s3/s1, s2, s3, s4, s5, s6, s7, s8, s9, s10
io=1
rdf=0.01745
x1=xc(1)
x2=0.0
x3=xc(2)
s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
s7=0
s8=0
s9=0
s10=0
do i=1, np
  ao=x1
  a1=mmbsi0(io, ao, ier)
  a2=a1/mmbsil(io, ao, ier)
  a3=1.-a2*a2-a2/ao
  sl=sl+cos(th(i)*rdf-x3)
  s2=s2+sin(th(i)*rdf-x3)
  s5=s5+a1
  s7=s7+log(a2)
  s10=s10+a3
enddo
hd(1)=-s10
hd(2)=-x1*s1
hl(1)=s2
return
end

numerical integration for 10 Bessel function

subroutine simp(x0, xf, ni, sk, abes)
h=(xf-x0)/(float(ni+ni))
hx=x0
hy=exp(sk*cos(x0))
do i=1, ni
  x0i=hx
  x1i=x0i+h
  x2i=x1i+h
  y0i=hy
  y1i=exp(sk*cos(x1i))
  y2i=exp(sk*cos(x2i))
  hx=x2i
  hy=y2i
  abes=abes+h*(y0i+4*y1i+y2i)/3
enddo
return
end
Von Mises likelihood

subroutine fvonrm(n,x,f)

real x(n), f
real al, s1, s2
real xbes
real rdf
common/s1/pi
common/s3/rmax
common/s9/fmax,x1,x2,x3
common/s12/ixe3
rdf=2.*pi/360.
xbes=0.0
abes=0.0
al=0.0
type*, 'params at entry=', x(1), x(2), x(3)
s1=0.0
s2=0.0
do i=1, np
al=0.0
xbes=0.0
if(ixe3.eq.1)then
al=x(1)*(1.+x(2)*r(i))
else
al=x(1)+x(2)*r(i)
endif
s1=s1+al*cos(th(i)*rdf)-x(3)
if(al.lt.0.0)then
al=-al
endif
if(al.eq.0.0)then
xbes=1.0
else
call simp(0.0, 2.0*pi, 32, al, abes)
xbes=abes/(2.*pi)
endif
s2=s2+alog(xbes)
enddo
f=-(s1-s2)
type*, 'sl,s2=', s1, s2
ac=s1-s2
type*, 'funct=', -f
type*, 'parameters=', x(1), x(2), x(3)
return
end

simple von mises likelihood

subroutine fvmtest(n,x,f)
real x(n), f
real s1
real mmbsi0
common/s1/pi
common/s3/rmax
common/s11/npa,nsa,nsb
a=0.0
al=0.0
xbes=0.0
rdf=2.*pi/360.
IOPT = 1
SL = 0.0
Do i = NSA, NSB
SL = SL + COS((TH(i) * RDF) - X(2))
Enddo
A = X(1) * SL
AL = X(1)
If (AL .LT. 0.) then
    AL = -AL
Endif
If (AL .EQ. 0.) then
    XBES = 1.
Else
    XBES = MMBSI0(IOPT, X(1), IER)
Endif
B = NPA * (ALOG(2. * PI * XBES))
F = -(A - B)
C Type*, 'funct=', -F
Return
End
File__$255$DUA2:[MC.T.AL]NEWKP2.FOR;71 (16242,171,0), last revised on 27-JUN-1990 13:46, is a 17 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 62 bytes.

Job NEWKP2 (624) queued to SYS$PRINT on 24-AUG-1990 13:05 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:25 from queue LASER.
dimension znew(500,1),xg(500,1),yg(500,1)
real hat(500,500),hat2(500)
real tau(500,1),rmat(500,500),rminv(500,500)
real glinp(500),z(500,1),r(500),th(500),x(500,1)
real wei(500,1)
real y(500,1),kn(500,500),kninv(500,500)
real wklp(30)
real sspt(20,20),ssp2(20,20)
real resu(500,1),gmode(500,1)
real u(500,1)
dimension at(100,500),c(500,50)
dimension be(500,100)
real dtu(20,1),binc(20,1),dta(20,500),ssp(20,20)
real xpar(10),apar(10),bpar(10),gpar(10)
real work5(100)
integer iwork5(10)
real wk55(500),sing(500)
real a(500,20)
real ga(l),gb(l)
real xv(l)
real ghat(500,1)
real kmn(500,500)
real fn(500,20),fm(500,20),fnt(20,500),wk(20500)
real mul1(20,500),mul2(20,20),minv(20,20)
real mul3(20,1),mul4(500,1),res(500,1)
real kns(20100),knins(20100)
real mul2s(210),mul3inv(210)
real klu(20100),rl(500,20),r2(500,20),r2kn(20000)
real s2(500),c1(20),h(20),ip(20)
real xln(500,1),yln(500,1)
real res2(500),ajinv(500,500),w12(90000)
real par(l)
real gtrend(500)
real gtrl(500,1)
real am(6,6),ai(6,6),ainv(6,6)
real bl(6),bu(6),xp(6),w(100)
integer iw(10)
integer betem(500,100),betold(500,100)
integer n,m,p
integer nf(500)
character*20 fname,fname2
common/s1/alpha
common/s2/rmax
common/s3/b(20,1)
Iinv=0
type*, 'enter region data filename:'
read(5,10)fname
10 format(a20)
s15=0.0
open(15,file=fname,status='old')
read(15,*)nreg
do i=1,nreg
read(15,*)r(i),th(i),wei(i,1),nf(i)
s15=s15+wei(i,1)
if(wei(i,1).eq.0.0)then
wei(i,1)=0.00001
endif
if(nf(i).eq.0)then
tau(i,1)=0
else
tau(i,1)=-log(wei(i,1))
endif
enddo
wetot=s15
close(15)
rmax=0.0
do i=1,nreg
if(r(i).gt.rmax)then
  rmax=r(i)
endif
enddo
pi=3.141593
pi2=pi*2.
do i=1,nreg
  r(i)=r(i)/rmax
  x(i,1)=r(i)*cos(th(i))
  y(i,1)=r(i)*sin(th(i))
  wei(i,1)=pi*wei(i,1)/wetot
enddo

pi='enter grand mean:'
read(5,*)b(1,1)

pi='enter parameter estimates:'

pi='r,lnr,cos,sin,rcos,rsin'
read(5,*)b(2,1),b(3,1),b(4,1),b(5,1),b(6,1),b(7,1)

pi='alpha, sigma2'
read(5,*)alpha,sigma2

pi='enter tolerance:'
read(5,*)efac

do i=1,7
  betold(i,1)=int(b(i,1)*efac)
enddo

nt=7

nt='no of parameters=7'

nt='now setting up covariances'
do i=1,nreg-1
do j=i+1,nreg
  kn(i,j)=cov(x(i,1),y(i,1),x(j,1),y(j,1))*sigma2
  kn(j,i)=kn(i,j)
enddo
kn(i,i)=sigma2
enddo

kn(nreg,nreg)=sigma2

kl=0
do i=1,nreg
  do j=1,i
    k1=kl+1
    kns(k1)=kn(i,j)
  enddo
enddo
call ludecp(kns,k1u,nreg,d1,d2,ier)
call linv2p(klu,nreg,knins,idgt,d1,d2,wk,ier)
if(ier.ne.0)then
  type*, 'Kn lu:ier=',ier
  iinv=2
  type*, 'enter tolerance(usually 0.0)'
  read(5,*)tol
  ia=500
  iav=500
  call lginv(kn,ia,nreg,nreg,tol,kninv,iav,sing,wk55,ier)
  if(ier.ne.0)then
    type*, 'gen inv error:',ier
do i=1,nreg
  knin(i,i)=knin(i,i)+0.01
enddo
goto 562
dendif
goto 5550
dendif
kl=1
do i=1,nreg
do j=1,i
  knin(i,j)=knins(kl)
  knin(j,i)=knin(i,j)
  kl=kl+1
endo
type*, 'diag inv cov:', knin(i,i)
endo
5550  continue
do i=1,nreg-1
do j=i+1,nreg
  rmat(i,j)=knin(i,j)
  rmat(j,i)=rmat(i,j)
endo
  rmat(i,i)=knin(i,i)+1.
endo
  rmat(nreg,nreg)=knin(nreg,nreg)+1.
do i=1,nreg
  if(nf(i).eq.0)then
    rmat(i,i)=knin(i,i)
  endif
endo
call trend(fn,r,th,glinp,nreg)
type*, 'trend set up'
c transpose and mult matrices
c generalised inverse here
c
call vmulfm(fn,knin,nreg,nt,nreg,500,500,mull,20,ier)
call vmulff(mull,fn,nt,nreg,nt,20,500,ssp,20,ier)
c insert here
cia=500
ia=500
tol=0.0
5467  continue
call lginf(rmat,ia,nreg,nreg,tol,rminv,ia,
sing,wk55,ier)
  if(ier.ne.0)then
    type*, 'error in inverse:', ier
    do i=1,nreg
      rmat(i,i)=rmat(i,i)+0.1
    enddo
    goto 5467
  endif
call vmulff(mull,rminv,nt,nreg,nreg,20,500,dta,20,ier)
call vmulfp(dta,mull,nt,nreg,nt,20,20,ssp2,20,ier)
do i=1,nt
do j=1,nt
  sspt(i,j)=ssp(i,j)+ssp2(i,j)
endo
enddo
call vmulff(dta,tau,nt,nreg,1,20,500,dtu,20,ier)
ia=20
iav=20
lw=30
tol=5.0e-4
call f04jae(nt,nt,sspt,20,dtu,tol,asig,ir,wk55,lw,ifa)
type*, 'rmse:', asig
asig2=asig*asig
type*, 'mean square error:', asig2
c end of calc:: now display estimates
c
5251 type*, 'beta values:'
do i=1,nt
write(6,*)'beta(',i,'value=',dtu(i,1))
endo
call lginf(sspt,ia,nt,nt,tol,minv,iav,sing,wk55,ier)
snl=0
type*, 'variances:'
do i=1,nt
type*, minv(i,1)
endo
type*, 'var-cov of parms'
do i=1,nt
write(nt,1501)(minv(i,j),j=1,nt)
endo

c predictions
c
now doing g- mode values
c
call vmulfm(mull,dtu,nt,nreg,1,20,20,resu,500,ier)
do i=1,nreg
resu(i,1)=resu(i,1)+tau(i,1)
endo
call vmulff(rminv,resu,nreg,nreg,1,500,500,gmode,500,ier)
type*, 'ord and shrunk estimates'
do i=1,nreg
ghat(i,1)=tau(i,1)
if(nf(i).eq.l)then
type*, ghat(i,1),gmode(i,1)
endif
endo
c end of gmode calculation
c
s53=0
s52=0
do i=1,nreg
s53=s53+gmode(i,1)
s52=s52+exp(gmode(i,1))
endo
alm=s53-s52-asig2
type*, 'postl_max=', alm
type*, 'residual file(1:yes):'
read(5,*)irfile
if(irfile.eq.l)then
otype*, 'filename:'
read(5,10)fname2
open(23,file=fname2,status='new')
call vmulff(fn,dtu,nreg,nt,1,500,500,mu4,500,ier)
call vmulff(minv,dtu,nt,nreg,20,20,fnt,20,ier)
call vmulff(fn,fnt,nreg,nt,nreg,500,20,ajinv,500,ier)
do i=1,nreg
  hat2(i)=ajinv(i,i)
  if(nf(i).eq.l)then
    type*, 'lev(',i,')',hat2(i)
  endif
endo
s5=0
  type*, 'res1(ghat-gmode) res2(ghat-fna)'
do i=1,nreg
  ab=ghat(i,1)-mu4(i,1)
  ac=ghat(i,1)-gmode(i,1)
  res(i,1)=ab
  if(nf(i).eq.l)then
    type*,ac,ab
  endif
endo
  s5=s5+res(i,1)*res(i,1)
endo
  type*, 'np=',nreg
  amser=s5/(float(nreg)-7.)
  type*, 'sum of squared errors:',s5
  type*, 'mean squared error=',amser
  do i=1,nreg
    if(nf(i).eq.l)then
      res(i,1)=(ghat(i,1)-mu4(i,1))/sqrt(amser*(1.-hat2(i))
      ordres=ghat(i,1)-mul4(i,1)
      if(irfile.eq.l)then
        write(23,*)r(i),th(i),res(i,1),hat2(i),mu4(i,1),ordres
      endif
      type*, 'r,th,stan res:',r(i),th(i),res(i,1)
    endif
  enddo
  close(23)
  stop
end

c functions and subs

real function cov(xl,yl,x2,y2)
real xl,yl,x2,y2
common/s1/alpha
common/s2/rmax
ad=dist(xl,yl,x2,y2)
cov=exp(-alpha*ad/rmax)
return
end
real function dist(xl,yl,x2,y2)
real x1,y1,x2,y2
a=(x1-x2)**2.
b=(y1-y2)**2.
dist=sqrt(a+b)
return
end
subroutine trendfn(tf,r,th,glinp,np)
real tf(SOO,20),r(np),th(np)
integer np
real glinp(np)
common/s3/b(20,1)
s1=0
do i=1,np
  if(r(i).eq.0)then
r(i)=0.0001
endif
tf(i,1)=r(i)
tf(i,2)=log(r(i))
tf(i,3)=cos(th(i))
tf(i,4)=sin(th(i))
tf(i,5)=r(i)*cos(th(i))
tf(i,6)=r(i)*sin(th(i))
do j=1,6
sl=sl+tf(i,j)
endo
glinp(i)=sl
endo
return
end

subroutine transp(a,at,ner,nper)
real a(500,20),at(100,500)
integer ner,nper
do i=1,ner
  do j=1,nper
    at(j,i)=a(i,j)
  enddo
endo
return
end

subroutine rnult(a,be,n,rn,n2,e)
real a(500,20),be(500,100),e(500,50)
integer n,rn,n2
do i=1,n
  do j=1,n2
    sl=0
    do k=1,rn
      sl=sl+a(i,k)*be(k,j)
    enddo
    e(i,j)=sl
  enddo
endo
return
end

subroutine trend(tf,r,th,glinp,nreg)
real tf(500,20),r(nreg),th(nreg)
greal glinp(nreg)
integer nreg
common/s3/b(20,1)
do i=1,nreg
  tf(i,1)=1.
tf(i,2)=r(i)
  if(r(i).eq.0.0)then
tf(i,3)=log(0.0000001)
else
tf(i,3)=log(r(i))
endif
tf(i,4)=cos(th(i))
tf(i,5)=sin(th(i))
tf(i,6)=r(i)*cos(th(i))
tf(i,7)=r(i)*sin(th(i))
sl=0
  do j=1,7
    sl=sl+b(j,1)*tf(i,j)
  enddo
endo
glinp(i)=sl
endo
return
end
File $255$DUA2:[MC.T.AL]NEWKP2.FOR;71 (16242,171,0), last revised on 27-JUN-1990 13:46, is a 17 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 62 bytes.

Job NEWKP2 (625) queued to SYS$PRINT on 24-AUG-1990 13:05 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:26 from queue LASER.
dimension znew(500,1),xg(500,1),yg(500,1)
real hat1(500,500),hat2(500)
real tau(500,1),rmat(500,500),rminv(500,500)
real glinp(500),z(500,1),r(500),th(500),x(500,1)
real wei(500,1)
real y(500,1),kn(500,500),kninv(500,500)
real wklp(30)
real sspt(20,20),ssp2(20,20)
real resu(500,1),gmode(500,1)
real u(500,1)
dimension at(100,500),c(500,50)
dimension be(500,100)
real dtu(20,1),binc(20,1),dta(20,500),ssp(20,20)
real xpar(10),apar(10),bpar(10),gpar(10)
real work5(100)
integer iwork5(10)
real wk55(500),sing(500)
real a(500,20)
real ga(1),gb(1)
real xv(1)
real ghat(500,1)
real krm(500,500)
real fn(500,20),fm(500,20),fnt(20,500),wk(20500)
real mul1(20,500),mul2(20,20),minv(20,20)
real mul3(20,1),mul4(500,1),res(500,1)
real kns(20100),knins(20100)
real mul2s(210),mul2sinv(210)
real klus(21000),r1(500,20),r2(500,20),r2kn(20000)
real s2(500),c1(20),h(20),ip(20)
real xin(500,1),yln(500,1)
real res2(500),ajinv(500,500),w12(90000)
real par(1)
real gtrend(500)
real gtrl(500,1)
real am(6,6),ai(6,6),ainv(6,6)
real bl(6),bu(6),xp(6),w(100)
integer iw(10)
integer betem(500,100),betold(500,100)
integer n,m,p
integer nf(500)
character*20 fname,fname2
common/s1/a1pha
common/s2/rmax
common/s3/b(20,1)
1inv=0

10 type*,'enter region data filename:'
read(5,10)fname
format(a20)
s15=0.0
open(15,fi1e=fname,status='old')
read(15,*)nreg
do i=1,nreg
read(15,*)r(i),th(i),wei(i,1),nf(i)
s15=s15+wei(i,1)
if(wei(i,1).eq.0.0)then
wei(i,1)=0.0001
endif
if(nf(i).eq.0)then
tau(i,1)=0
else
tau(i,1)=-log(wei(i,1))
endif
enddo
wetot=s15
close(15)
rmax=0.0
do i=1,nreg
  if(r(i).gt.rmax)then
    rmax=r(i)
  endif
enddo
pi=3.141593
pi2=pi*2.
do i=1,nreg
  r(i)=r(i)/rmax
  x(i,1)=r(i)*cos(th(i))
  y(i,1)=r(i)*sin(th(i))
  wei(i,1)=pi*wei(i,1)/wetot
enddo
type*, 'enter grand mean:'
read(5,*)b(1,1)
type*, 'enter parameter estimates:'
read(5,*)b(2,1),b(3,1),b(4,1),b(5,1),b(6,1),b(7,1)
type*, 'alpha, sigma^2'
read(5,*)alpha,sigma2
type*, 'must be a multiple of 10'
read(5,*)efac
do i=1,7
  betold(i,1)=int(b(i,1)*efac)
endo
c set up covariances
c no of parameters = 7
  nt=7
type*, 'now setting up covariances'
do i=1,nreg-1
  do j=i+1,nreg
    kn(i,j)=cov(x(i,1),y(i,1),x(j,1),y(j,1))*sigma2
    kn(j,i)=kn(i,j)
  enddo
  kn(i,i)=sigma2
enddo
kn(nreg,nreg)=sigma2
c now invert kn matrix
c kl=0
  do i=1,nreg
    do j=1,i
      kl=kl+1
      kns(kl)=kn(i,j)
    enddo
  enddo
call ludecp(kns,klu,nreg,dl,d2,ier)
call linv2p(klu,nreg,knins,idgt,dl,d2,wk,ier)
  if(ier.ne.0)then
type*, 'Kn lu: ier=',ier
  end if
  linv=2
  type*, 'enter tolerance (usually 0.0)'
  read(5,*)tol
  ia=500
  iav=500
  call 19inf(kn,ia,nreg,nreg,tol,kninv,iav,sing,wk55,ier)
  if(ier.ne.0)then
type*, 'gen inv error: ',ier
  end if
do i=1,nreg
kn(i,i)=kn(i,i)+0.01
enddo
goto 562
endif
goto 5550
endif
kl=1
do i=1,nreg
  do j=1,i
    kninv(i,j)=knins(kl)
    kninv(j,i)=kninv(i,j)
    kl=kl+1
  enddo
type*,'diag inv cov:',kninv(i,i)
enddo
5550 continue
do i=1,nreg-1
  do j=i+1,nreg
    rmat(i,j)=kninv(i,j)
    rmat(j,i)=rmat(i,j)
  enddo
  rmat(i,i)=kninv(i,i)+1.
enddo
rmat(nreg,nreg)=kninv(nreg,nreg)+1.
do i=1,nreg
  if(nf(i).eq.0)then
    rmat(i,i)=kninv(i,i)
  endif
enddo
call trend(fn,r,th,glinp,nreg)
type*,'trend set up'
c transpose and mult matrices
c
generalised inverse here
c
call vmulffm(fn,kninv,nreg,nt,nreg,500,500,mull,20,ier)
call vmulff(mull,fn,nt,nreg,nt,20,500,ssp,20,ier)
c insert here
c
ia=500
iav=500
tol=0.0
5467 continue
call lginf(rmat,ia,nreg,nreg,tol,rminv,iav,
  sing,wk55,ier)
  if(ier.ne.0)then
type*,'error in inverse:',ier
  do i=1,nreg
    rmat(i,i)=rmat(i,i)+0.1
  enddo
goto 5467
endif
call vmulff(mull,rminv,nt,nreg,nreg,20,500,dta,20,ier)
call vmulffp(dta,rnull,nt,nreg,nt,20,ssp2,20,ier)
do i=1,nt
  do j=1,nt
    sspt(i,j)=ssp(i,j)+ssp2(i,j)
  enddo
enddo
call vmulff(dta,tau,nt,nreg,1,20,500,dtu,20,ier)
ia=20
iav=20
lw=30
tol=5.0e-4
call f04jae(nt,nt,sspt,20,dtu,tol,asig,ir,wklp,lw,ifa)
type*, 'rmse:', asig
asig2=asig*asig
type*, 'mean square error:', asig2
c end of calc:: now display estimates
c c set up beta values
c 5251 type*, 'beta values:'
do i=l,nt
write(6,*)'beta(', i, , 'value=', dtu(i, l))
enddo
call lginf(sspt,ia,nt,nt,tol,minv,iav,sing,wk55,ier)
nsi=0
type*, 'variances:'
do i=l,nt
 type*, i, minv(i, i)
enddo
type*, 'var-cov of parms'
do i=l,nt
write(nt,150l)(minv(i,j), j=1,nt)
1501 format(7f10.5)
 enddo
c c predictions
c c now doing g- mode values
c c

call vmulffm(mull,dtu,nt,nreg,1,20,20,resu,500,ier)
do i=l,nreg
resu(i, 1)=resu(i, 1)+tau(i, 1)
enddo
call vmulff(rminv,resu,nreg,nreg,1,500,500,gmode,500,ier)
type*, 'ord and shrunk estimates'
do i=l,nreg
ghat(i, 1)=tau(i, 1)
if(nf(i).eq.1) then
type*, ghat(i, 1), gmode(i, 1)
endif
dendo
c e nd of gmode calculation
c c
s53=0
s52=0
do i=l,nreg
s53=s53+gmode(i, 1)
s52=s52+exp(gmode(i, 1))
enddo
alm=s53-s52-asig2
type*, 'postl_max=', alm
type*, 'residual file(1:yes):'
read(5,*)irfile
if(irfile.eq.1) then
 type*, 'filename:'
read(5,10)fname2
open(23,file=fname2,status='new')
endif
call vmullff(fn,dtu,nreg,nt,1,500,500,mul4,500,ier)
call vmullff(minv,dta,nt,nreg,20,20,fnt,20,ier)
call vmullff(fn,fnt,nreg,nt,nreg,500,20,ajinv,500,ier)
do i=1,nreg
hat2(i)=ajinv(i,i)
if(nf(i).eq.1)then
type*, 'lev(',i,')',hat2(i)
endif
enddo
s5=0
type*, 'resl(ghat-gmode) res2(ghat-fna)'
do i=1,nreg
ab=ghat(i,1)-mul4(i,1)
ac=ghat(i,1)-gmode(i,1)
res(i,1)=ab
if(nf(i).eq.1)then
    type*,ac,ab
endif
s5=s5+res(i,1)*res(i,1)
enddo
type*, 'np=', nreg
amser=s5/(float(nreg)-7.)
type*, 'sum of squared errors:', s5
if(nf(i).eq.1)then
    write(23,*) r(i), th(i), res(i,1), hat2(i), mul4(i,1), ordres
endif
enddo
close(23)
stop
end

c functions and subs
real function cov(x1,y1,x2,y2)
real x1,y1,x2,y2
common/s1/alpha
common/s2/rmax
ad=dist(x1,y1,x2,y2)
cov=exp(-alpha*ad/rmax)
return
end
real function dist(x1,y1,x2,y2)
real x1,y1,x2,y2
a=(x1-x2)**2.
b=(y1-y2)**2.
dist=sqrt(a+b)
return
end
subroutine trendfn(tf,r,th,glinp,np)
real tf(500,20),r(np),th(np)
integer np
real glinp(np)
common/s3/b(20,1)
s1=0
do i=1,np
if(r(i).eq.0)then
r(i)=0.0001
endif
tf(i,1)=r(i)
tf(i,2)=log(r(i))
tf(i,3)=cos(th(i))
tf(i,4)=sin(th(i))
tf(i,5)=r(i)*cos(th(i))
tf(i,6)=r(i)*sin(th(i))
do j=1,6
s1=s1+tf(i,j)
enddo
glinp(i)=s1
enddo
return
end
subroutine transp(a,at,ner,nper)
real a(500,20),at(100,500)
integer ner,nper
do i=1,ner
  do j=1,nper
    at(j,i)=a(i,j)
  enddo
endo
to return
end
subroutine mult(a,b,c)
real a(500,20),b(500,100),c(500,50)
integer n,m,n2
ndo i=1,n
do j=1,m
  sl=0
  do k=1,n2
    sl=sl+a(i,k)*b(k,j)
  enddo
  c(i,j)=sl
endo
to return
endo
subroutine trend(tf,r,th,glinp,nreg)
real tf(500,20),r(nreg),th(nreg)
real glinp(nreg)
integer nreg
ndo i=1,nreg
  tf(i,1)=1.
tf(i,2)=r(i)
if(r(i).eq.0.0)then
tf(i,3)=log(0.0000001)
else
tf(i,3)=log(r(i))
endif
tf(i,4)=cos(th(i))
tf(i,5)=sin(th(i))
tf(i,6)=r(i)*cos(th(i))
tf(i,7)=r(i)*sin(th(i))
s1=0
ndo j=1,7
  s1=s1+b(j,1)*tf(i,j)
enddo
glinp(i)=s1
endo
to return
endo

File _$255$DUA2: [MC.T.AL)NEWKRIGE2.FOR;62 (16972,44,0), last revised on 7-JUN-1990 12:10,
is a 16 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length
with implied (CR) carriage control. The longest record is 60 bytes.

Job NEWKRIGE2 (627) queued to SYS$PRINT on 24-AUG-1990 13:05 by user MCTAL, UIC [MCT,MCTAL],
under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:28 from
queue LASER.
implicit real (a-h,o-z)
dimension znew(500,1),xg(500,1),yg(500,1)
dimension at(100,500),c(500,50)
dimension be(500,100)
real dtu(20,1),dta(20,500)
real tau(500,1),rmat(500,500),rminv(500,500)
real sspt(20,20),ssp2(20,20)
real dexp(500)
integer betold(500,100)
integer betem(500,100)
real a(1000,20)
real kn(500,500),kninv(500,500)
real ghat(500,1)
real knm(500,500)
real fn(500,20),fm(500,20),fnt(20,500),wk(20500)
real mul(20,500),mul2(20,20),minv(20,20)
real mul3(20,1),mul4(500,1),res(500,1)
real kns(20100),knins(20100)
real mul2s(210),mul2sinv(210)
real klu(20100),rl(500,20),r2(500,20),r2kn(20000)
real rtr(20,500),ssp(20,20),sinv(20,20)
real s2new(500),cl(20),h(20),ip(20)
real xln(500,1),yln(500,1)
real par(1)
real w12(90000)
real gtrend(500)
real gtrl(500,1)
real sing(500),wk55(1000)
real am(6,6),ai(6,6),ainv(6,6)
real bl(6),bu(6),xpar(6),w(100)
real glinp(500),z(500,1)
real r(2000),th(2000),x(500,1),y(500,1)
real sl,s2,s3
real wklp(30)
real resu(500,1),gmode(500,1)
integer nreg,ind1,ind2

integer iw(10)
in integer n,m,p
character*20 fname
character*20 fname2
character*20 tname
common/s1/alpha
common/s2/rmax
common/s3/b(20,1)
Inv=0

type*, 'enter region data filename:'
read(5,10)fname

format(a20)
type*, 'enter expected deaths file:'
read(5,10)tname

open(17,file=tname,status='old')
read(17,*)nreg
do i=1,nreg
read(17,*)dexp(i)
endo
close(17)
open(15,file=fname,status='old')
read(15,*)nreg
read(15,*)xcen,ycen
do i=1,nreg
read(15,*)x(i,1),y(i,1),z(i,1)
if(z(i,1).eq.0.0)then
z(i,1)=0.0001
tau(i,1)=z(i,1)*log(z(i,1)/dexp(i))
endif

close(15)
pi=3.141593
pi2=pi*2.
do i=1,nreg
ax=x(i,1)-xcen
ay=y(i,1)-ycen
r(i)=sqrt(ax*ax+ay*ay)
if(ax.eq.0)then
  if(ay.gt.0)then
    the(i)=1.570796
  else
    if(ay.eq.0)then
      th(i)=0.0
    else
      th(i)=4.7124
    endif
  endif
  goto 1678
endif
has=atan(ay/ax)
if(ax.lt.0.0)then
  th(i)=has+pi
else
  if(ay.gt.0.0)then
    th(i)=has+pi2
  else
    th(i)=has
  endif
endif
1678 continue
do i=1,nreg
r(i)=r(i)/rm1
enddo
type*, 'enter grand mean:'
read(5,*)b(1,1)
type*, 'enter parameter estimates:'
type*, 'for r,lnr,cos,sin,rcos,rsin'
read(5,*)b(2,1),b(3,1),b(4,1),b(5,1),b(6,1),b(7,1)
type*, 'range,sigma2'
read(5,*)rmax,sigma2
alpha=1
type*, 'enter iter tolerance(no of sig figs)'
read(5,*)tol
efac=10**tol
do j=1,6
  betold(j,1)=int(b(j,1)*efac)
endo
c set up covariances
c no of parameters=7
nt=7
type*, 'now setting up covariances'
do i=1,nreg-1
  do j=i+1,nreg
    kn(i,j)=cov(x(i,1),y(i,1),x(j,1),y(j,1))*sigma2
    kn(j,i)=kn(i,j)
  enddo
  kn(i,i)=sigma2
  kn(nreg,nreg)=sigma2
enddo

! now invert kn matrix!

kl=0
  do i=1,nreg
    do j=1,i
      kl=kl+1
      kns(kl)=kn(i,j)
    enddo
  enddo
  call ludecp(kns,klu,nreg,dl,d2,ier)
call linv2p(klu,nreg,knins,idgt,dl,d2,wk,ier)
c full matrix inversion here

563 idgt=0
  call linvlf(kn,nreg,500,kninv,idgt,wk,ier)
  if(ier.ne.0) then
    type*, 'Kn lu:ier=', ier
    type*, 'enter tolerance(usually 0.0):'
    read(5,*)tol
    ia=500
    iav=500
    call lglnf(kn,ia,nreg,nreg,tol,kninv,iav,sing,wk55,ier)
    if(ier.ne.0) then
      type*, 'gen inf error:', ier
      do i=1,nreg
        kn(i,i)=kn(i,i)+0.01
      enddo
      goto 563
    endif
  goto 5550
  else
    type*, 'done inversion ok'
    endif
  endif
5550 continue

do i=1,nreg-1
  do j=i+1,nreg
    rmat(i,j)=kninv(i,j)
    rmat(j,i)=rmat(i,j)
  enddo
  rmat(i,i)=kninv(i,i)+z(i,1)
  type*, 'rmat,kninv=', rmat(i,i),kninv(i,i)
enddo

n9=nreg
  rmat(n9,n9)=kninv(n9,n9)+z(n9,1)

! now set up trend values!
call trend(fn,r,th,glinp,nreg)
type*, 'fn values,glinp'
do i=1,20
  write(6,*) (fn(i,j),j=1,nt),glinp(i)
enddo
c transpose and mult matrices
call vmulfm(fn, kninv, nreg, nt, nreg, 500, 500, mull, 20, ier)
call vmulff(mull, fn, nt, nreg, nt, 20, 500, ssp, 20, ier)
type*, 'ssp matrix'
do i=1, nt
write(6, *)(ssp(i, j), j=1, nt)
endo
ia=500
iav=500
tol=0.0
call lginf(rmat, ia, nreg, nreg, tol, rminv, iav,
sing, wk55, ier)
if(ier.ne.0)then
type*, 'gen inv error:', ier
endif
call vmulff(mull, rminv, nt, nreg, nreg, 20, 500, dta, 20, ier)
call vmulfp(dta, mull, nt, nreg, nt, 20, 20, ssp2, 20, ier)
type*, 'ssp2 matrix'
do i=1, nt
write(6, *)(ssp2(i, j), j=1, nt)
endo
do i=1, nt
do j=1, nt
sspt(i, j)=ssp(i, j)+ssp2(i, j)
endo
do i=1, nt
call vmulff(dta, tau, nt, nreg, 1, 20, 500, dtu, 20, ier)
ia=20
iav=20
lw=30
call f04jau(nt, nt, sspt, 20, dtu, tol, asig, ir, wk5p, lw, ifa)
type*, 'rmse:', asig
asig2=asig*asig
type*, 'mean square error:', asig2
type*, 'parm ests:'
do i=1, nt
type*, i, dtu(i, l)
endo
call lginf(sspt, ia, nt, nt, tol, minv, iav, sing, wk55, ier)
type*, 'parms ', ' vars'
do i=1, nt
type*, i, minv(i, i)
endo
type*, 'var-cov matrix'
do i=1, nt
write(6, *)(minv(i, j), j=1, nt)
endo
format(7f10.5)
call vmulfm(mull, dtu, nt, nreg, 1, 20, 20, resu, 500, ier)
do i=1, nreg
resu(i, l)=resu(i, l)+tau(i, l)
endo
call vmulff(rminv, resu, nreg, nreg, 1, 500, 500, gmode, 500, ier)
typ* ', 'ord and shrunk estimates'
do i=1, nreg
ghat(i, l)=tau(i, l)
type*, ghat(i, l), gmode(i, l)
endo
now do residuals
s53=0
s52=0
do i=1, nreg
sS3=sS3+gmode(i,l)
sS2=sS2+exp(gmode(i,l))
enddo
alm=sS3-sS2-asig2
type*, 'post l-max:', alm
type*, 'residual file(l:yes):'
read(5,*)irfile
if(irfile.eq.l)then
type*, 'filename: '
read(5,10)fname2
open(23, file=fname2, status='new')
endif
call VIn1.llff(fn,dtu,nreg,nt,l,500,500,mul4,500,ier)
do i=l,nreg
res(i,l)=ghat(i,1)-mul4(i,1)
if(irfile.eq.l)then
write(23,*)x(i,1),y(i,1),res(i,l)
endif
type*, res(i,l)
enddo
close(23)
stop
end

functions and subs

real function dist(xl,yl,x2,y2)
real xl,yl,x2,y2
a=(xl-x2)**2.
b=(yl-y2)**2.
dist=sqrt(a+b)
return
end

real function cov(xl,yl,x2,y2)
implicit real (a-h,o-z)
real xl,yl,x2,y2
common/sl/alpha
common/s2/rmax
ad=dist(xl,yl,x2,y2)
cov=exp(-alpha*ad/rmax)
return
end

subroutine transp(a,at,ner,nper)
real a(SOO,20),at(20,500)
integer ner,nper
do i=1,ner
do j=1,nper
at(j,i)=a(i,j)
enddo
enddo
return
end

subroutine mult(a,be,n,m,n2,c)
real a(500,500),be(500,500),c(500,500)
integer n,m,n2
do i=1,n
do j=1,n2
sl=0
do k=1,m
sl=sl+a(i,k)*be(k,j)
enddo
c(i,j)=sl
enddo
enddo
subroutine trendfn(tf,r,th,glinp,np)
  implicit real (a-h,o-z)
  real tf(500,20),r(np),th(np)
  integer np
  real glinp(np)
  common/s3/b(20,1)
  s1=0
  do i=1,np
    if(r(i).eq.0)then
      r(i)=0.0001
    endif
    tf(i,1)=r(i)
    tf(i,2)=log(r(i))
    tf(i,3)=cos(th(i))
    tf(i,4)=sin(th(i))
    tf(i,5)=r(i)*cos(th(i))
    tf(i,6)=r(i)*sin(th(i))
    do j=1,6
      sl=sl+tf(i,j)
    enddo
    glinp(i)=sl
  enddo
  return
end

subroutine trend(tf,r,th,glinp,nreg)
  implicit real (a-h,o-z)
  real tf(500,20),r(nreg),th(nreg)
  real glinp(nreg)
  integer nreg
  common/s3/b(20,1)
  do i=1,nreg
    if(r(i).eq.0)then
      r(i)=0.0001
    endif
    tf(i,1)=1.
    tf(i,2)=r(i)
    tf(i,3)=log(r(i))
    tf(i,4)=cos(th(i))
    tf(i,5)=sin(th(i))
    tf(i,6)=r(i)*cos(th(i))
    tf(i,7)=r(i)*sin(th(i))
    do j=1,7
      sl=sl+b(j,1)*tf(i,j)
    enddo
    glinp(i)=sl
  enddo
  return
end
File _$255$DUA2:[MC.T.AL]POWER.FOR;210 (24957,5,0), last revised on 24-JUL-1990 15:22, is a 45 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 61 bytes.

Job POWER (612) queued to SYS$PRINT on 24-AUG-1990 13:02 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:02 from queue LASER.
This program generates weibull-Von-Mises rate functions

```fortran
program test
  double precision dseed
  double precision dsl
  character*20 fname,tname
  character*20 zname,wname
  real a,b,xc,yc,r
  real prob(5)
  real prob1(5)
  real psi,kap,pmu
  real lap,spac,zed,urao
  real bc(10)
  real moore
  real lincor,wlik
  real crit(15,20)
  real c0,c1,c2,pi
  real stat(15)
  real chialfl(7),chialf2(7),zedalf(7)
  real mooralf(15,7),watalf(15,7)
  real hed(15)
  real khat
  real rh(1)
  real la
  real montc(11,2000)
  real ala(100),thetm(2000),radm(2000)
  integer ira(100)
  integer ncount(15,5)
  integer ncountl(15,5)
  integer ipx(100000)
  external f
  common/s1/r
  common/s2/c0,c1,c2
  common/s3/rnum,thnum
  common/s4/pi
  common/s5/dseed
  common/s6/rbar1
  common/s7/khat
  common/s8/nr2000,theta(2000)
  common/s9/rmax
  common/s10/lor
  common/s11/ust,la,f5a
  common/s12/s1,s2,s3,s4
  common/s13/umard
  common/s20/rbar5,rend
  common/s30/sigma2,alp
  common/s40/xcen,ycen
  common/s50/b1,b2,b3,b4,b5
  data(chialfl(i),i=1,5)/3.841,5.024,6.635,7.879,10.83/
  data(chialf2(i),i=1,5)/5.991,7.378,9.21,10.597,13.815/
  data(zedalf(i),i=1,5)/1.6449,1.96,2.326,2.576,3.09/
  data(mooralf(1,i),i=1,4)/1.048,1.144,1.248,1.313/
  data(mooralf(2,i),i=1,4)/1.024,1.127,1.247,1.329/
  data(mooralf(3,i),i=1,4)/1.010,1.117,1.242,1.331/
  data(mooralf(4,i),i=1,4)/1.004,1.112,1.240,1.329/
  data(mooralf(5,i),i=1,4)/0.999,1.109,1.239,1.329/
  data(watalf(1,i),i=1,4)/0.182,0.213,0.254,0.283/
```
data(watalf(2,i),i=1,4)/0.185,0.217,0.261,0.293/
data(watalf(3,i),i=1,4)/0.186,0.220,0.265,0.299/
data(watalf(4,i),i=1,4)/0.186,0.221,0.266,0.301/
data(watalf(5,i),i=1,4)/0.187,0.221,0.267,0.302/
data(mooralf(6,i),i=l,4)/1.016,1.123,1.245,1.334/
data(watalf(6,i),i=1,4)/0.185,0.219,0.263,0.296/
ntest=13
pi=3.14159
a=5.5
b=5.5
xc=0
yc=0
dseed=801129.d0
r=a/2.0
power test or otherwise

type*, 'power(1) or stats(0)' read(5,*)ipow
  if(ipow.eq.1)then
    type*, 'batch or not(1/0):'
    READ(5,*)ibatch
    if(ibatch.eq.1)then
      read(5,*)nflag
      READ(5,*)np
      nps=np
      if(nflag.eq.2.or.nflag.eq.3)then
        read(5,*)beth
        read(5,*)aeth
        if(nflag.eq.3)then
          read(5,*)amu
          read(5,*)vs2
        endif
        read(5,*)nangt
        if(nangt.eq.2)then
          read(5,*)akap
        endif
      else
        read(5,*)cl,c2,psi,kap,pmu
      endif
      read(5,*)igauss
      READ(5,*)niter
      READ(5,10)zname
      read(5,10)zname
      open(30,file=zname,status='new')
    ELSE
      type*, 'hom or non-hom process(1,0):'
      type*, 'ordinary simulation(uniform(2), normal(3))'
      read(5,*)nflag
      type*, 'enter number of points:'
      read(5,*)np
      nps=np
do special fit here

if(nflag.eq.2.or.nflag.eq.3)then
  type*, 'ord simulation: a+bcos(ph)+(1-b)u'
  type*, 'enter mixing parameter:'
  read(5,*)beth
  type*, 'enter alpha:'
  read(5,*)aeth
  if(nflag.eq.3)then
    type*, 'normal'
    type*, 'enter mean:'
read(5,*)amu
   type*, 'enter variance:'
read(5,*)vs2
endif
   type*, 'angular form: uniform(1), von Mises(2)'
read(5,*)nangt
if(nangt.eq.2)then
   type*, 'enter kappa:'
   read(5,*)akap
endif
else
   type*, 'enter: shape, scale, psi, kap, pmu (degrees)':'
   read(5,*)c1, c2, psi, kap, pmu
endif
   type*, 'gaussian(1) or not(0):'
read(5,*)igauss
   type*, 'no of iterations required: '
   read(5,*)niter
   type*, 'enter output filename: '
   read(5,10)tname
   type*, 'enter monte carlo filename: '
   read(5,10)zname
   open(30,file=zname,status='new')
endif
format(a20)
do i = 1, 4
   crit(1,i)=zedalf(i)
   crit(2, i)=zedalf(i)
   crit(3, i)=zedalf(i)
   crit(4, i)=zedalf(i)
   if(np.eq.10) then
      crit(5, i)=mooralf(1, i)
      crit(6, i)=watalf(1, i)
   else
      if(np.eq.20) then
         crit(5, i)=mooralf(2, i)
         crit(6, i)=watalf(2, i)
      else
         if(np.eq.30) then
            crit(5, i)=mooralf(6, i)
            crit(6, i)=watalf(6, i)
         else
            if(np.eq.50) then
               crit(5, i)=mooralf(3, i)
               crit(6, i)=watalf(3, i)
            else
               if(np.eq.100) then
                  crit(5, i)=mooralf(4, i)
                  crit(6, i)=watalf(4, i)
               else
                  crit(5, i)=mooralf(5, i)
                  crit(6, i)=watalf(5, i)
               endif
            endif
         endif
      endif
   endif
endif
crit(7, i)=chialf2(1)
crit(8, i)=zedalf(i)
crit(9, i)=chialfl(i)
crit(10, i)=chialf2(1)
crit(11, i)=zedalf(i)
crit(12, i)=zedalf(i)
crit(13,i)=chialf2(i)
enddo
do j=1,ntest
do k=1,4
ncount(j,k)=0
ncountl(j,k)=0
enddo
enddo

c start of simulation loop
c
write(30,*),niter,ntest
if(igauss.eq.1)then
if(ibatch.ne.1)then
type*, 'enter cv and range:'
endif
read(5,*),cv,range
endif

nps=nps-npn=1-kp=np
nr=np-10
dsl=9745631.0d0
do il=1,niter
if(igauss.eq.1)then
if(il.eq.1)then
bc(1)=c1
bc(2)=c2
bc(3)=psi
bc(4)=kap
bc(5)=pmu
call gauss2(bc,5,cv,range,nps,radn,thn)
do i=1,nps
rad(i)=radn(i)
theta(i)=S7.296*thn(i)
enddo
else
call ud(dsl,kp,nr,ipx)
do ihj=1,nr
rad(ihj)=radn(ipx(ihj))
theta(ihj)=thn(ipx(ihj))*S7.296
enddo
np=nr
endif

type*, 'r,th coords:'
type*, 'n=',np
do i=1,np
type*,rad(i),theta(i)
enddo
else
 type*, 'loop count=',il
if(nflag.eq.0)then
ds=12347.0d0
do i=1,np
rnum=ggubfs(dseed)
ab=rnum*(1.-exp(-c2*r**c1))
rad(i)=(-alog(1.-ab)/c2)**(1/c1)
rad(i)=rad(i)/r
ai=psi+kap*rad(i)
call ggvms(ds,ai,1,rh)
h=rh(l)
endif
if(h.lt.0)then

h=2.0*pi+h
endif
theta(i)=(h*360/(2.*pi)+pmu
if(theta(i).gt.360.)then
theta(i)=theta(i)-360.
endif
enddo
else
if(nflag.eq.1)then
do i=l,np
rnum=ggubfs(dseed)
thnum=ggubfs(dseed)
theta(i)=2.*pi*thnum
rad(i)=sqrt(r**2)*rnum)
rad(i)=rad(i)/r
theta(i)=theta(i)*360./(2.*pi)
enddo
else
if(nflag.eq.2.or.nflag.eq.3)then
rmax=0.0
if(nflag.eq.1)then
rnum=ggubfs(dseed)
else
ggnqf(dseed)
rnum=amu+ab*sqrt(vs2)
endif
if(nangt.eq.1)then
thnum=ggubfs(dseed)
else
call ggvms(dseed,akap,l,rh)
h=rh(1)
if(h.1t.0)then
h=2.*pi+h
endif
theta(i)=57.29577*h
endif
cnv=0.01745329
rad(i)=aeth+beth*cos(theta(i)*cnv)+(l.-beth)*rnum
if(rad(i).gt.rmax)then
rmax=rad(i)
endif
doi=l,np
rad(i)=rad(i)/rmax
enddo
diendif
dendif
dendif
dendif
dendif
dendif
dendif
dendif
call sort(rnew,np)
ro5=1.0
hed(1)=lap(rnew,ro5,np)
hed(2)=spac(rnew,ro5,np)
hed(3)=zed(rnew,ro5,np)
hed(12)=weib(rnew,ro5,np)
hed(4)=urao(thnew,np)
call sortboth(rad,np,theta)
hed(5)=moore(rad,theta,np)
hed(6)=ust
acor=corr(rad,rad,np)
acor=lincor(rad,rad,np)
if(acor.gt.0.98)then
  acor=0.98
endif
hed(7)=-np*log(1.-acor)
hed(8)=wscore(rad,theta,np)
hed(13)=2.*rbarl*rbarl/float(np)
hed(9)=wlik(rad,theta,np)
hed(10)=umard
hed(11)=fishlee(rad,theta,np)
do j=1,ntest
do k=1,4
if(abs(hed(j)).ge.crit(j,k))then
  ncount(j,k)=ncount(j,k)+1
endif
if(hed(j).ge.crit(j,k))then
  ncountl(j,k)=ncountl(j,k)+1
endif
enddo
montc(j,il)=hed(j)
enddo
write(30,*)(montc(j,il),j=1,ntest)
enddo

c end of simulation loop here
c
do j=1,ntest
do k=1,4
prop=float(ncount(j,k))/float(niter)
enddo
endo
open(17,file=tname,status='new')
write(17,*)'input parameters'
write(17,*)'shape,scale,k,psi,pmu'
write(17,*)'nflag=' ,nflag
write(17,*)'no of points=' ,nps
write(17,*)'no of iterations=' ,niter
write(17,*)'counts for 2-tail test ',j
write(17,*)(ncount(j,k),k=1,4)
write(17,*)'counts for 1-tail test ',j
write(17,*)(ncountl(j,k),k=1,4)
do k=1,4
prob(k)=float(ncount(j,k))/float(niter)
probl(k)=float(ncountl(j,k))/float(niter)
endo
write(17,*)'prob for 2-tail test : ',j
write(17,*)(prob(k),k=1,4)
write(17,*)'prob for 1-tail test ',j
write(17,*)(probl(k),k=1,4)
endo
close(17)
else

do statistics on ord data here

type*, 'enter filename:'
read(5,10) fname
open(25, file=fname, status='old')
read(25, *) np
rmax=-10.0
do i=1, np
read(25, *) rad(i), theta(i)
if(rad(i).gt.rmax) then
rmax=rad(i)
endif
enddo
close(25)
type*, 'enter scaling for r:'
read(5, *) rsca
rsca=1.0
do i=1, np
rad(i)=(rad(i)/rmax)*rsca
rnew(i)=rad(i)
theta(i)=theta(i)
enddo
type*, 'do oyu want own rmax(l:tyes),
type*, 'max r value is:', rmax
read(5, *) irm
if(irm.eq.1) then
type*, 'enter r max:'
read(5, *) rb
r=rb/rmax
endif
r=1
call sort(rnew, np)
a6=lap(rnew, r, np)
b6=spac(rnew, r, np)
c6=zed(rnew, r, np)
d6=urao(theta, np)
call sortboth(rad, np, theta)
e6=moore(rad, theta, np)
f6=ust
acor=lincor(rad, theta, np)
g6=-np*log(1.-acor)
h6=wscore(rad, theta, np)
p6=wlik(rad, theta, np)
r6=umard
s6=fishlee(rad, theta, np)
z6=weib(rnew, r, np)
type*, 'lap, spac, zed, rao, watson'
type*, a6, b6, c6, d6, f6
type*, 'moore, angcor, score, lik, rmar, fishlee'
type*, e6, g6, h6, p6, r6, s6
type*, 'weib=', z6
type*, 'rbar: ', rbarl
y6=2.*rbarl*rbarl/float(np)

c now doing monte carlo

type*, 'enter your ml estimates(lam, del, kap, mu),'
read(5, *) c2, c1, akap, pmu
ds=12347.0d0
do il=1, 99
do i=1, np
rnu=ggubfs(dseed)

thu=ggubfs(dseed)
radm(i)=rnu*(1.-exp(-c2*r**c1))
ai=aka?
call ggvmgs(ds,ai,1,rh)
h=rh(1)
if(h.lt.0.0)then
  h=2.0*pi+h
endif
thetm(i)=(360/(2*pi))*h+pmu
if(thetm(i).gt.360)then
  thetm(i)=thetm(i)-360
endif
rnew(i)=radm(i)
thenew(i)=thetm(i)
enddo
call sort(rnew,np)
montc(1,1)=lap(rnew,r,np)
montc(2,1)=spac(rnew,r,np)
montc(3,1)=zed(rnew,r,np)
montc(4,1)=urao(thnew,np)
call sortboth(radm,np,thetm)
montc(5,1)=moore(radm,thetm,np)
montc(6,1)=ust
acor=lincor(radm,thetm,np)
montc(7,1)=-np*log(1-acor)
montc(8,1)=wscore(radm,thetm,np)
montc(9,1)=wlik(radm,thetm,np)
montc(10,1)=umard
montc(11,1)=fishlee(radm,thetm,np)
montc(12,1)=weib(rnew,r,np)
montc(13,1)=2.*rbar1*rbar1/float(np)
enddo
montc(1,100)=a6
montc(2,100)=b6
montc(3,100)=c6
montc(4,100)=d6
montc(5,100)=e6
montc(6,100)=f6
montc(7,100)=g6
montc(8,100)=h6
montc(9,100)=p6
montc(10,100)=r6
montc(11,100)=s6
montc(12,100)=z6
montc(13,100)=y6
type*, 'done all tests: now sorting'
c now sorting vectors
do ik=1,13
do i=1,100
  ira(i)=i
  ala(i)=montc(ik,i)
enddo
lapa=100
call vsrtr(ala,lapa,ira)
type*, 'vec sorted'
do j=1,100
  if(ira(j).eq.100)then
    dart=real(j)
  type*, 'rank=',j
  pr=dart/100.
  opr=1.-pr
  type*, 'exact MC tail prob of test=',opr
goto 5231
endif
enddo

5231 continue
enddo
endif
stop
end

* Functions and Subroutines *

subroutine ud(dsl,kp,nb,ipx)
integer kp,nb,ipx(nb)
double precision dsl
do i=1,nb
ub=ggubfs(dsl)
ipx(i)=1+int(ub*kp)
endo
return
end

real function f(x)
real x,b,mnrad,pi
common/s1/r
common/s2/c0,cl,c2
common/s3/rnum,thnum
common/s4/pi

mnrad=(c0/cl)*x-(l/cl)*exp(cl*r)+(l/cl)
ba=(x-(l/cl)*exp(cl*x)-(c0/c2)*mnrad*rnum+(l/cl)
f=abs(ba)
return
end

real function weib(r,r0,n)
real r(n),r0
real lhat
external funk5
common/s20/rbar5,rend
s0=0
s1=0
s2=0
s3=0

do i=1,n
s0=s0+r(i)
s1=s1+log(r(i))
s2=s2+r(i)*log(r(i))
s3=s3+r(i)*(log(r(i))**2.)
endo
rend=r0
rbar=s0/n
rbar5=s0/n
a1=-20
b1=20.0
tol=0.1
call zxgsn(funk5,a1,b1,tol,lhat,ier)
type*, 'lamda est=',lhat
a=exp(lhat*r0)-1.
b=r0*log(r0)
c=log(r0)
d=exp(lhat*r0)
top=n+sl-lhat*s2-n*(lhat*b/a)
fishdd=n+1+lhat*s3+n*(lhat*b*c/a-(lhat**2.)*b*b*d/(a*a))
fishdl=s2+n*(b/a-lhat*r0*b*d/(a*a))
fishll=n/(lhat*lhat)-n*(r0*r0*d/(a*a))
botl=(fishdd-fishdl/fishl)
if(botl.le.0)then
bot=0.00001
else
bot=sqrt(bot1)
endif
weib=top/bot
return
end
real function funk5(x)
real x
common/s20/rbar5,rend
a=exp(x*rend)-1.
b=rend/a
funk5=abs(1./x-b-rbar5)
return
end
real function cord(f,x.nr)
real f(4),x(4),ch(4)
integer ir(4),nsgn(4)
integer nr
la=4
do i=1,4
ir(i)=i
enddo
call vsrtr(f,la,ir)
do i=1,4
enddo
ch(1)=x(ir(1»-x(ir(2)
ch(2)=x(ir(2»-x(ir(3)
ch(3)=x(ir(3»-x(ir(4)
ch(4)=x(ir(4»-x(ir(1)
nsl=0
do i=1,4
if(ch(i).lt.0.0)then
nsgn(i)=-1
else
if(ch(i).ge.0.0)then
nsgn(i)=1
endif
endif
nsl=nsl+nsgn(i)
enddo
nost=0
if(nsgn(1).ne.nsgn(2).and.nsgn(2).ne.nsgn(3))then
nost=1
endif
if(nsl.eq.0.and.nost.eq.1)then
cord=0.0
else

cord=1.0
endif
return
end
real function var(x,xbar,m)
real x(500),xbar
integer m
sl=0.0
do i=1,m
sl=sl+(x(i)-xbar)**2.0
enddo
var=sl/m
return
end
real function fishlee(rad, theta, np)
real rad(np), theta(np), f(4), x(4), g(500)
integer ir(4), np
double precision dseed4
m=200
kl=np
dseed4=857011.d0
sl=0.0
do k=1, m
nr=4
call ggud(dseed4, kl, nr, ir)
do il=1, nr
f(il)=theta(ir(il))
x(il)=rad(ir(il))
enddo
g(k)=cord(f, x, nr)
sl=sl+g(k)
enddo
alm=sl/m
g2=var(g, alm, m)
if(g2.1t.0.001)then
g2=0.001
endif
fishlee=(alm-0.666)/(sqrt(g2/m))
c type*, 'fish, var, alm', fishlee, g2, alm
return
end

subroutine funct2(n, xc, fc, gc)
real xc(n), fc, gc(n)
real mmbsi0, mmbsi1
integer n
common/s4/pi
common/s9/rmax
common/s10/idor
iO=1
sl=0
s2=0
s3=0
s4=0
s5=0
s6=0
s7=0
rd=0.01745
if(idor.eq.0)then
x1=xc(1)
x2=xc(2)
x3=xc(3)
do i=1, np
s1=sl+cos(theta(i)*rd-x3)
s3=s3+rad(i)*cos(theta(i)*rd-x3)
s2=s2+sin(theta(i)*rd-x3)
s4=s4+rad(i)*sin(theta(i)*rd-x3)
a0=x1+x2*rad(i)
a2=mmbsi0(i0, a0, ier)
a1=mmbsi1(i0, a0, ier)/a2
s5=s5+a1
s6=s6+rad(i)*a1
s7=s7+log(a2)
enddo
gc(1)=s1+s5
\[ gc(2) = -s3 + s6 \]
\[ gc(3) = -x1s2 - x2s4 \]
\[ fc = -x1s1 - x2s3 + s7 \]

else
\[ x1 = xc(1) \]
\[ x2 = xc(2) \]
do 
\[ i = 1, np \]
\[ s1 = s1 + \cos(\theta(i)*rd - x2) \]
\[ s2 = s2 + \sin(\theta(i)*rd - x2) \]
enddo
\[ a2 = \text{mmbsi0}(i0, x1, ier) \]
\[ a1 = \text{mmbsil}(i0, x1, ier)/a2 \]
\[ gc(1) = -s1 + np*a1 \]
\[ gc(2) = -x1s2 \]
\[ fc = -x1s1 + np*\log(a2) \]
endif
return
end

subroutine sort(ar,ni)
real ar(ni)
do 
\[ i = 1, ni - 1 \]
\[ j = ni - i \]
mark = 0
do 
\[ k = 1, j \]
kpl = k + 1
if(ar(k) > ar(kpl)) then
\[ b = ar(kpl) \]
\[ ar(kpl) = ar(k) \]
\[ ar(k) = b \]
mark = 1
endif
dendo
if(mark.eq.0) goto 30
dendo
countinue
return
end

subroutine rank(ar,n,ir)
real ar(n)
integer n,ir(n)
do 
\[ i = 1, n \]
\[ ir(i) = i \]
dendo
do 
\[ i = 1, n - 1 \]
\[ j = n - i \]
mark = 0
do 
\[ k = 1, j \]
kpl = k + 1
if(ar(k) > ar(kpl)) then
\[ b = ar(kpl) \]
\[ bl = ir(kpl) \]
\[ ar(kpl) = ar(k) \]
\[ ir(kpl) = ir(k) \]
\[ ar(k) = b \]
\[ ir(k) = bl \]
mark = 1
endif
dendo
if(mark.eq.0) goto 40
dendo
continue
countinue
return
end
real function lap(ri,ro,n)
real ri(n)
rx=0
rde=ro**2
do i=1,n
rx=rx+(ri(i)**2)
enddo
lap=((rx/n)-0.5*rde)/(rde*sqrt(1.0/(12.0*n)))
return
end
real function spac(ri,ro,n)
real ri(n)
real ai(2001)
real si(2000)
b=0
ai(1)=ri(1)**2
do i=2,n
ai(i)=ri(i)**2-ri(i-1)**2
endo
ai(n+1)=ro**2-ri(n)**2
call sort(ai,n+1)
ax=0
do i=1,n+1
si(i)=ai(i)/(ro**2)
ax=ax+i*si(i)
endo
sp=2.0*(n+1)-2.0*ax
ser=sp-0.5*n
spac=(sp-0.5*n)/sqrt(float(n)/12.0)
return
end
real function zed(ri,ro,n)
real ri(n)
al=0
do i=1,n
al=al+a*log(ro**2/(ri(i)**2))
endo
rap=2.0*al
zed=(rap-2.0*n)/sqrt(4.0*n)
return
end
real function wlik(r,th,np)
real r(np),th(np)
real x(3),a(3),b(3),g(3),work(40)
real mmbsi0
real khat
real 11,12
integer np,iworl(5),liw,lw
common/s10/idor
common/s4/pi
common/s7/khat
common/s12/s1,s2,s3,s4
c=0.01745
x(1)=-0.001
x(2)=0.01
cm=atan(s2/s1)
if(s1.le.0)then
 pmu=pi+cm
else
 if(s2.le.0.and.s1.gt.0)then
  pmu=2*pi+cm
 else
  pmu=cm
endif
endif
x(3)=pmu
ib=0
do i=1,3
a(i)=0.0
enddo
b(1)=30.0
b(2)=30.0
b(3)=6.4
liw=5
lw=40
n2=3
ifail=1
idor=0

c now call 3 parm von mises

call e04kae(n2,ib,a,b,x,f,g,iworl,liw,work,lw,ifail)
if(ifail.ne.0)then
write(6,*)'failure(3) due to:',ifail
endif
sl=0
s2=0
s3=0
s4=0
do i=1,np
sl=sl+cos(th(i)*c-x(3))
s2=s2+r(i)*cos(th(i)*c-x(3))
a0=mmbsi0(1,x(1)+x(2)+r(i),ier)
s3=s3+log(a0)
enddo
bll=x(1)*sl+x(2)*s2-s3

c now do ordinary von mises
	x(1)=0.01
x(2)=pmu
do i=1,2
a(i)=0.0
enddo
b(1)=30.0
b(2)=6.4
n2=2
ifail=1
idor=1
call e04kae(n2,ib,a,b,x,f,g,iworl,liw,work,lw,ifail)
if(ifail.ne.0)then
write(6,*)'failure(2)=' ,ifail
endif
sl=0
s2=0
s3=0
s4=0
do i=1,np
sl=sl+cos(th(i)*c-x(2))
enddo
a0=log(mmbsi0(1,x(1),ier))
bll=x(1)*sl-np*a0
wlik=2*(bll-b12)
return
end

real function urao(fie,n)
real
real fie(n)
real fi2(2000)
real la
common/sll/ust,la,f5a
call sort(fie,n)
s1=0
s2=0
s3=0
do i=1,n
v(i)=fie(i)/360.
s2=s2+v(i)*v(i)
s1=s1+v(i)
c(i)=2.*i-1.
s3=s3+c(i)*v(i)/n
endo
do vbar=s1/n
ust=s2-s3+n*((1./3.)-(vbar-0.5)**2.)
do i=1,n-1
fi2(i)=fie(i+1)-fie(i)
dendo
fi2(n)=360-(fie(n)-fie(1))
f1=0
f5=0
do i=1,n
f1=f1+abs(fi2(i)-(360/n))
endo
f5a=(f5/n-180)/(360*sqrt(1.0/(n**2)))
l1=0.5*f1
urao=sqrt(1.0*n)*(l1-132.435)/87.4612
return
end
real function moore(ra,theta,n)
c *** note ra and theta are already sorted(based on ra)
c
real ra(n),theta(n),rwk(2000),it(2000),ir(2000)
integer n,iwk(2000)
cornmon/s13/umard
eps=0.00001
call nmrank(ra,n,eps,iwk,rwk,ir,sl,tl)
call nmrank(theta,n,eps,iwk,rwk,it,s2,t2)
c=0.01745
as=0
ac=0
bc=0
bs=0
do i=1,n
ac=ac+ir(i)*cos(theta(i)*c)
as=as+ir(i)*sin(theta(i)*c)
ang=6.2832*it(i)/n
bc=bc+i*cos(ang)
bs=bs+i*sin(ang)
dendo
rs=ac**2+as**2
smard=bc**2+bs**2
umard=24.0*smard/(n*n*(n+1))
denom=n*(n+1)*(2*n+1)/12.0
al=rs/denom
r=sqrt(rs)
rstar=r/sqrt(float(n*n*n))
moore=rstar
return
end
real function lincor(r,th,np)
real r(np),th(np)
real thc(1000),ths(1000)
real corr 
common/s4/pi 
c=0.0174532 
do i=l,np 
thc(i)=cos(th(i)*c) 
ths(i)=sin(th(i)*c) 
enddo 
s1=0 
s2=0 
do i=l,np 
s1=s1+thc(i) 
s2=s2+ths(i) 
enddo 
cm=atan(s2/s1) 
if(s1.le.0)then 
  pmu=pi+cm 
else 
  if(s2.le.0.and.s1.gt.0)then 
    pmu=2*pi+cm 
  else 
    pmu=cm 
  endif 
endif 
do i=l,np 
thc(i)=cos(th(i)*c-pmu) 
ths(i)=sin(th(i)*c-pmu) 
enddo 
a=corr(r,thc,np) 
b=corr(r,ths,np) 
c=corr(thc,ths,np) 
lincor=(a*a+b*b-2*a*b*c)/(1.-c*c) 
return 
end 

real function wscore(r,th,np) 
real r(np),th(np) 
real khat 
external funk 
common/s6/rbar1 
common/s7/khat 
common/s12/s1,s2,s3,s4 
c=0.0174532 
s1=0 
s2=0 
s3=0 
s4=0 
s5=0 
s6=0 
do i=l,np 
s1=s1+cos(th(i)*c) 
s2=s2+sin(th(i)*c) 
s3=s3+r(i)*cos(th(i)*c) 
s4=s4+r(i)*sin(th(i)*c) 
s5=s5+r(i) 
s6=s6+r(i)*r(i) 
enddo 
rbar1=(sqrt(s1*s1+s2*s2)/np 
r2=sqrt(s3*s3+s4*s4) 
ssr=s6-(s5*s5)/np 
if(ssr.eq.0)then 
  ssr=0.00001 
endif 
a1=0.0 
b1=60.0 
tol=0.1
call zxgsn(funk,al,bl,tol,khat,ier)
if(khat.le.0.0)then
  khat=0.00001
endif
stat=r2-s5*rbarl
aderv=1.-rbarl/khat-rbarl*rbarl
if(aderv.eq.0.)then
  aderv=0.0001
endif
wscore=stat/((float(np)/(float(np)-1))*aderv*ssr)
return
end
real function funk(x)
real x
real mmbsi0,mmbsil
common/s6/rbarl
iO=1
a=mmbsi0(i0,x,ier)
b=mmbsil(i0,x,ier)
funk=abs(rbarl-b/a)
return
end
real function corr(x,y,n)
real x(n),y(n)
integer n
s1=0
s2=0
s3=0
s4=0
s5=0
do i=1,n
  s1=s1+x(i)
  s2=s2+x(i)*x(i)
  s3=s3+y(i)
  s4=s4+y(i)*y(i)
  s5=s5+x(i)*y(i)
enddo
a=s5-s3*s1/n
b=s2-s1*s1/n
c=s4-s3*s3/n
bc=b*c
if(bc.le.0.)then
  bc=0.0001
endif
corr=a/sqrt(bc)
return
end

****
ROUTINE GENERATES VON MISES VARIATES H WITH PARAMETER C SJ.
USES BEST AND FISHER ENVELOPE REJECTION METHOD,
WITH A WRAPPED CAUCHY TARGET DISTRIBUTION.
VERSION EMPLOYS PRETEST TO AVOID LOG EVALUATIONS.
COSINE IS EVALUATED VIA POLAR METHOD.
****
SUBROUTINE BFISHC(SJ,H,JC)
double precision dseed
common/s5/dseed
SK= SJ
IF(JC.GT.1)GOTO 24
T=1.+SQRRT(1.+4.*SK*SK)
RHO=(T-SQRRT(T+T))/(SK+SK)
R=(1.+RHO*RHO)/(RHO+RHO)
V=ggubfs(dseed)-.5
W=ggubfs(dseed)-.5

D=V*V
E=W*W
if(E.eq.0.0)goto 24
SUM=4*(D+E)
IF(SUM.GT.1.)GOTO 24
TR=D/E
Z=(1.-TR)/(1.+TR)
F=(1.+R*Z)/(R+Z)
IF(F.GT.1.)F=1.
IF(F.LT.-1.)F=-1.
C=SK*(R-F)
U2=SUM
IF(C*(2.-C)-U2.GT.0)GOTO 4
IF(ALOG(C/U2)+1.-C.LT.0)GOTO 24
H=ACOS(F)
U3=ggubfs(dseed)-.5
IF(U3.LT.0)H=-H
RETURN
END

subroutine sortboth(ar,ni,thr)
c this routine sorts r and th based on permutation of r
c
real ar(ni),thr(ni)
do i=1,ni-1
j=ni-i
mark=0
do k=1,j
kpl=k+1
if(ar(k).gt.ar(kpl))then
b=ar(kpl)
bl=thr(kpl)
ar(kpl)=ar(k)
thr(kpl)=thr(k)
ar(k)=b
thr(k)=bl
mark=1
endif
enddo
if(mark.eq.0)goto 30
enddo
continue
return
end
common/s4/pi

mnrad=(c0/c1)*((r-(1/c1))*exp(c1*r)+(1/c1))
ba=(x-(1/c1))*exp(c1*x)-(c1/c0)*mnrad*rnum+(1/c1)
f=abs(ba)
return
end

subroutine sort(ar,ni)
real ar(ni)
do i=1,ni-1
  j=ni-i
  mark=0
  do k=1,j
    kpl=k+1
    if(ar(k).gt.ar(kpl))then
      b=ar(kpl)
ar(kpl)=ar(k)
ar(k)=b
      mark=1
    endif
  enddo
  if(mark.eq.0)goto 30
enddo
continue
return
end

subroutine rank(ar,n,ir)
real ar(n)
integer n,ir(n)
do i=1,n
  ir(i)=i
enddo
do i=1,n-1
  j=n-i
  mark=0
  do k=1,j
    kpl=k+1
    if(ar(k).gt.ar(kpl))then
      b=ar(kpl)
      bl=ir(kpl)
ar(kpl)=ar(k)
      ir(kpl)=ir(k)
ar(k)=b
      ir(k)=bl
      mark=1
    endif
  enddo
  if(mark.eq.0)goto 40
enddo
continue
return
end

real function lap(ri,ro,n)
real ri(n)
rx=0
rde=ro**2
do i=1,n
  rx=rx+(ri(i)**2)
enddo
lap=((rx/n)-0.5*rde)/(rde*sqrt(1.0/(12.0*n)))
type*, 'lap=',lap
return
end

real function spac(ri,ro,n)
real ri(n) 
real ai(2001) 
real si(2000) 
b=0 
ai(1)=ri(1)**2 
do i=2,n 
ai(i)=ri(i)**2-ri(i-1)**2 
enddo 
ai(n+1)=ro**2-ri(n)**2 
call sort(ai,n+1) 
ax=0 
do i=1,n+1 
si(i)=ai(i)/(ro**2) 
ax=ax+i*si(i) 
enddo 
sp=2.0*(n+1)-2.0*ax 
ser=sp-0.5*n 
spac=(sp-0.5*n)/sqrt(float(n)/12.0) 
type*, 'spac=', spac 
type*, 'ser=', ser 
return 
end 
real function zed(ri,ro,n) 
real ri(n) 
al=0 
do i=1,n 
al=al+alog(ro**2/(ri(i)**2)) 
enddo 
rap=2.0*al 
zed=(rap-2.0*n)/sqrt(4.0*n) 
type*, 'zed=', zed 
return 
end 
real function urao(fie,n) 
real fie(n) 
real fi2(2000) 
call sort(fie,n) 
s1=0 
s2=0 
s3=0 
do i=1,n 
v(i)=fie(i)/360. 
s2=s2+v(i)*v(i) 
s1=s1+v(i) 
c(i)=2.*i-1. 
s3=s3+c(i)*v(i)/n 
enddo 
vbar=s1/n 
ust=s2-s3+n*((1./3.)-(vbar-0.5)**2.) 
type*, 'watsons U squared =', ust 
do i=1,n-1 
fi2(i)=fie(i+1)-fie(i) 
enddo 
fi2(n)=360-(fie(n)-fie(1)) 
fl=0 
f5=0 
do i=1,n 
fl=fl+abs(fi2(i)-(360/n)) 
f5=f5+fie(i) 
enddo 
f5a=(f5/n-180)/(360*sqrt(1.0/(12.0*n))) 
type*, 'laplace: angle=', f5a 
l=0.5*fl
type*, 'raos original test=' , la
urao=sqrt(1.0*n)*(la-132.435)/87.4612
return
end
real function moore(ra,theta,n)
real ra(n),theta(n)
integer n,ir(2000)
call rank(ra,n,ir)
c=0.0174532
as=0
ac=0
do i=1,n
ac=ac+i*cos(theta(ir(i))*c)
as=as+i*sin(theta(ir(i))*c)
enddo
r=sqrt(ac**2+as**2)
denom=n*(n+1)*(2*n+1)/12.0
moore=r/sqrt(denom)
return
end
real function corr(x,y,n)
real x(n),y(n)
integer n
s1=0
s2=0
s3=0
s4=0
s5=0
do i=1,n
s1=s1+x(i)
s2=s2+x(i)**2
s3=s3+y(i)
s4=s4+y(i)**2
s5=s5+x(i)*y(i)
enddo
a=s5-s3*s1/n
b=s2-s1*s1/n
c=s4-s3*s3/n
corr=a/sqrt(b*c)
return
end

C
ROUTINE GENERATES VON MISES VARIATES H WITH PARAMETER
C SJ. USES BEST AND FISHER ENVELOPE REJECTION METHOD,
C WITH A WRAPPED CAUCHY TARGET DISTRIBUTION.
C VERSION EMPLOYS PRETEST TO AVOID LOG EVALUATIONS.
C COSINE IS EVALUATED VIA POLAR METHOD.
C
******
SUBROUTINE BFISHC(SJ,H,JC)
double precision dseed
common/s5/dseed
SK= SJ
IF(JC.GT.1)GOTO 24
T=1.+SQRT(1.+4*SK*SK)
RHO=(T-SQRT(T+T))/(SK+SK)
R=(1.+RHO*RHO)/(RHO+RHO)
V=ggu bfs(dseed)-.5
W=ggu bfs(dseed)-.5
D=V*V
E=W*W
SUM=4*(D+E)
IF(SUM.GT.1.)GOTO 24
TR=D/E
Z=(1.-TR)/(1.+TR)
\[
F = \frac{1 + RZ}{R + Z}
\]

IF (F .GT. 1.) F = 1.
IF (F .LT. -1.) F = -1.
C = \text{SK} \times (R - F)
U2 = \text{SUM}
IF (C \times (2. - C) - U2 .GT. 0) GOTO 4
IF (\text{ALOG}(C / U2) + 1. - C .LT. 0) GOTO 24
H = ACOS(F)
U3 = gubfs(dseed) - .5
IF (U3 .LT. 0) H = -H
RETURN
END

```fortran
subroutine sortboth(ar, ni, thr)
  c
  this routine sorts r and th based on permutation of r
  c
  real ar(ni), thr(ni)
do i = 1, ni - 1
  j = ni - i
  mark = 0
  do k = 1, j
    kpl = k + 1
    if (ar(k) .gt. ar(kpl)) then
      b = ar(kpl)
      bl = thr(kpl)
      ar(kpl) = ar(k)
      thr(kpl) = thr(k)
      ar(k) = b
      thr(k) = bl
      mark = 1
    endif
  enddo
enddo
if (mark .eq. 0) goto 30
enddo
continue
return
end
```
File $_$255$DUA2:[MC.T.A]SIMREG.FOR;186 (21786,22,0), last revised on 27-JUN-1990 14:41, is a 52 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 61 bytes.

Job SIMREG (615) queued to SYS$PRINT on 24-AUG-1990 13:02 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:06 from queue LASER.
double precision dseed
real pr(1000,18), pm(9000), wk1(9000), wk2(9000)
real wkvec(50000), rvec(1,50000), rmv(50000)
real var(10000)
real pf(9000), rate(20)
real tpop(1000)
real mrate(9), frate(9), x(9000), y(9000)
real reg(9000), hed(20)
real bp, ap, kp, ip, mup
real probl(6)
real regcon(1000), regex(1000)
real pml(9000), pfl(9000)
real stat(15), chialfl(7), zedalf(7)
real corralf(7)
real stonalf(7)
real crit(15, 20)
real nce(1000, 2)
real lincor, wscore, scor, scoth, wlik, umard
real scorst, scothst
real prob(5), montc(10, 2000)
real kest
real wor(12), xk(1), a(1), b(1)
integer iwor(1)
integer num(9), nuf(9), ia(9000), nm(9000), nf(9000)
integer tp(1000)
integer irl(1000), ir2(1000), ib(9000)
integer ncon(1000, 20)
integer nb(9000), np(9000)
integer nde(1000, 2)
integer ncount1(15, 5)
real ncex(1000, 20)
real texb(1000), texp(1000)
integer mpop, fpop
character*20 fname, wname, tname, zname
external funk
external funk5
common/s1/pi, pi2, c
common/s2/r1, nt, xbar
common/s3/tex(9000), r(1000), th(1000), ncell
common/s4/kest
common/s10/sigma2, alpha
common/s20/idor
common/s25/idrt
common/s30/nd(9000)
common/s50/tml, tm2, tm3, t13
common/s51/wml, wm2, w12
data(stonalf(i), i=1, 3)/0.0, 0.0, 0.0/
data(corralf(i), i=1, 3)/0.0, 0.0, 0.0/
data(chialfl(i), i=1, 3)/3.841, 6.635, 10.827/
data(zedalf(i), i=1, 3)/1.645, 2.326, 3.090/
do i=1, 3
crit(1, i)=chialfl(i)
crit(2, i)=chialfl(i)
crit(3, i)=corralf(i)
crit(4, i)=chialfl(i)
crit(5, i)=chialfl(i)
crit(6, i)=stionalf(i)
enddo
pi=3.14159
pi2=6.283185
c=0.017453
type*, 'power(0) or stats(1)'
read(5, *) idrt
if(idrt.eq.0)then
type*,'note: 20 slots are provided for male(1-9)'
type*,'and female (10-18)'
type*,'enter parameters:'
type*,'enter shape(>=1),scale(>0),conc(>=0)'
type*,'inter(>=0),mu(radians)'
read(5,*)bp,ap,kp,ip,mup

'you must enter popn and exp rates for 18 age*sex'
type*,'groups from 2 files'
type*,'enter total male and female pop of all regions:'
read(5,*)mpop,fpop

type*,'enter no of radial cells:'
read(5,*)nr

'type*,'enter no of angle cells:'
read(5,*)nth
ncell-nr*nth

'type*,'no of cells=:',ncell
'type*','age:sex structure now follows'
'type*','the files must have <ncell> units each'
'type*','with an age:sex <18> structure'
'type*','option(1): all regions same age:sex'
'type*','option(2): different structure in each'
type*,'input option:'
read(5,*)nopt

if(nopt.eq.1)then
'type*,'enter file with 18 age:sex cells:'
'type*,'enter filename:'
read(S,lOO)fname

format(a20)
open(15,file=fname,status='old')
do i=1,9
read(15,*)num(i)
enddo
do i=1,9
read(15,*)nuf(i)
enddo

ncell=nr*nth
close(15)
do i=1,ncell
do j=1,9
ncon(i,j)=num(j)
enddo
do j=10,18
il=j-9
ncon(i,j)=nuf(il)
enddo

else
'type*,'enter filename:'
read(5,100)fname
open(15,file=fname,status='old')
do i=1,ncell
read(15,*)ncon(i,j),j=1,18)
enddo
close(15)

endif

'type*,'now enter file for 18 expected rates(/1000):'
'type*,'if no expected rates are needed use file of 1s'
read(5,100)fname

open(16,file=wname,status='old')
do i=1,18
read(16,*)rate(i)
rate(i)=rate(i)/1000.
enddo

close(16)

'type*,'enter output filename:'
read(5,100) tname
    type*, 'enter monte carlo filename:'
read(5,100) zname
    open(18, file=zname, status='new')

now calculate expected values
    do i=1,ncell
        do j=1,18
            nceex(i,j)=ncon(i,j)*rate(j)
        enddo
        s1=0
        s2=0
        do j=1,9
            s1=s1+nceex(i,j)
            j1=j+9
            s2=s2+nceex(i,j1)
        enddo
        nce(i,1)=s1
        nce(i,2)=s2
    enddo

calculate r and th of cells
    should enter rmax here, but i have set it to 1
    rmax=1
    k=0
    do i=1,nr
        a5=(float(i)/float(nr))*rmax
        do j=1,nth
            k=k+1
            b5=(float(j)/float(nth))*6.283
            r(k)=a5
            th(k)=b5
        enddo
    enddo
    type*, 'bayesian(1) or not(0)'
    read(5,*) ibayes
    if(ibayes.eq.1) then
        type*, 'enter alpha and sigma2:'
        read(5,*) alpha, sigma2
        k2=0
        do i=1,ncell
            do j=1,i
                k2=k2+1
                x1=r(i)*cos(th(i))
                y1=r(i)*sin(th(i))
                x2=r(j)*cos(th(j))
                y2=r(j)*sin(th(j))
                var(k2)=cov(x1,y1,x2,y2)
            enddo
        enddo
dseed=8429109.d0
    nr=1
    ir=1
    wkvec(1)=0.0
    call ggnsm(dseed, nr, ncell, var, ir, rvec, wkvec, ier)
    if(ier.ne.0) then
        type*, 'error=', ier
    endif
    type*, 'enter no of simulations:'
    read(5,*) nsim
    type*, 'press cntrl shift f6 and 1'
read(5,*) dert

do calculation of norm constant

   sl=0
   do i=1,ncell
      if(ibayes.ne.1)then
         rmv(i)=tense(r(i),th(i),bp,ap,kp,ip,mup)
      else
         a23=log(tense(r(i),th(i),bp,ap,kp,ip,mup))
         b23=a23+rvec(1,i)
         rmv(i)=exp(b23)
      endif
   enddo
   s0=0
   do j=1,18
      s0=s0+ncex(i,j)*rmv(i)
   enddo
   s1=s1+s0
   enddo
   cnorm=s1

calc prob for each region

   s15=0
   s16=0
   k1=0
   k2=0
   pm(0)=0.0
   pf(0)=0.0
   do i=1,ncell
      do j=1,9
         pr(i,j)=ncex(i,j)*rmv(i)/cnorm
         k1=k1+1
         pm(k1)=pr(i,j)
         s15=s15+pm(k1)
      enddo
      do j=10,18
         pr(i,j)=ncex(i,j)*rmv(i)/cnorm
         k2=k2+1
         pf(k2)=pr(i,j)
         s16=s16+pf(k2)
      enddo
   enddo
   dseed=123457.d0
   ia(1)=-1
   ib(1)=-1

iteration loop starts here

   ntest=6
   write(18,*) nsim, ntest
   do kl=1,nsim
      type*, 'iteration:', kl
      type*, 'prob total: male:', s15, 'female:', s16
   c**** do generation now

   nr1=mpop
   nr2=fpop
   ndmp=ncell*9
   ndfp=ndmp
type*, 'nr1,nr2,ndmp,ndfp'
type*, nr1, nr2, ndmp, ndfp
ncell9 = ncell*9
do i=1, ncell9
pml(i) = pm(i)/s15
pf1(i) = pf(i)/s16
enddo
call ggda(dseed, nr1, ndmp, pm1, ia, wk1, ir1)
call ggda(dseed, nr2, ndfp, pf1, ib, wk2, ir2)
type*, 'male pops'
s1 = 0
s2 = 0
do i=1, nr1
   c type*, 'i=', i, 'cell picked:', ir1(i)
   if(ir1(i).ne.0)then
      s1 = s1 + 1
   endif
endo
type*, 'female pops'
do i=1, nr2
   c type*, 'i=', i, 'cell picked:', ir2(i)
   if(ir2(i).ne.0)then
      s2 = s2 + 1
   endif
endo
type*, 'total male=', s1, 'total female=', s2

c**** binning data

do i=1, ndmp
   nm(i) = 0
endo
do i=1, ndfp
   nf(i) = 0
endo
do i=1, nr1
   in = ir1(i)
   nm(in) = nm(in) + 1
endo
do i=1, nr2
   in = ir2(i)
   nf(in) = nf(in) + 1
endo

nm > deaths> male
nf > deaths> female
pr > regional prob
rate > age specific rates

c**** do statistics here

k = 0
s5 = 0
do i=1, ncell
   s3 = 0
   s4 = 0
   do j=1, 9
      k = k + 1
      s3 = s3 + rm(k)
      s4 = s4 + nf(k)
      enddo
endo
nde(i,1) = s3
nde(i,2) = s4
tp(i)=s3+s4
s5=s5+tp(i)
tex(i)=nce(i,1)+nce(i,2)
nd(i)=tp(i)
enddo

type*, 'total number=', s5

s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
do i=1, ncell
c0=cos(th(i))
ci=sin(th(i))
s1=s1+nd(i)*co
s2=s2+nd(i)*ci
s3=s3+nd(i)
enddo
rl=sqrt(s1*s1+s2*s2)
h=atan(s2/s1)
if(s1.lt.0.0) then
  xbar=h+pi
else
  if(s2.lt.0.0) then
    xbar=h+pi2
  else
    xbar=h
  endif
endif
al=0.0
bl=60.0
tol=0.1
nt=s3
npar=1
nsig=3
a(1)=0.0
b(1)=70.0
nsr=20

call zxmwd(funk5,npar,nsig,a,b,nsr,xk,f,wor,iwor,ier)
kest=xk(1)
if(ier.ne.0) then
  type*, 'error(ier)=', ier
endif

type*, 'xbar,kest:', xbar,kest

hed(1)=scorst(tp,r,tex,ncell)
hed(2)=scothst(tp,th,tex,ncell)
acor=lincor(tp,r,th,tex,ncell)
type*, 'corr=', acor
hed(3)=acor
hed(4)=wlik(tp,r,th,tex,ncell)
kest=wml
xbar=wm2
hed(5)=wscore(tp,r,th,tex,ncell)
type*, 'ml estimates'
type*, '3-model:k,psi,mu,lik'
type*, tml, tm2, tm3, t13

type*, '2-model:k,mu,lik'
type*, wml, wm2, w12

mardia test not used here
stone's test instead
hed(6)=stone(r,tp, tex, ncell)

type*, 'tests(r,th,corr,lik,score,stone)
type*, (hed(j), j=1,6)
do j=1,ntest
do k=1,3
if(hed(j).ge.crit(j,k))
ncountl(j,k)=ncountl(j,k)+1
endif
enddo
montc(j,kl)=hed(j)
enddo
write(18,*)(montc(j,kl), j=1,ntest)

c end of simulation loop here
doendo
open(17, file=tname, status='new')
write(17,*)'input parameters'
do j=1,ntest
write(17,*)'count for one-tail test:', j
write(17,*)(ncountl(j,k), k=1,4)
do k=1,4
probl(k)=float(ncountl(j,k))/float(nsim)
endo
write(17,*)'one-tail test prob', j
write(17,*)(probl(k), k=1,4)
doendo
do easygraph graphics here
doendo
type*, 'end of file writing'
type*, 'press csf6 1'
read(5,*)igt
call grz(r,th,tp,ncell)

c statistics are here
c else
type*, 'enter age*sex rates file:'
read(5,100)fname
open(15, file=fname, status='old')
do j=1,9
read(15,*)mrate(j), frate(j)
endo
close(15)
type*, 'enter region file'
read(5,100)tname

type*, 'x-y(1) or r-th(2) file'
read(5,*)ifile
type*, 'one(1) or two(2) diseases on file:'
read(5,*)iont
type*, 'enter x,y coords of centre'
read(5,*)xcen, ycen
type*, 'which disease: bronch(1) pneum(2)'
read(5,*)idis
open(25, file=tname, status='old')
read(25,*)ncell
rmax=0
do i=1,ncell
s30=0
if(int.eq.1)then
read(25,*)reg(i),x(i),y(i),nb(i),
   (num(j),nuf(j),j=1,9)
np(i)=nb(i)
elself
read(25,*)reg(i),x(i),y(i),nb(i),np(i),
   (num(j),nuf(j),j=1,9)
endif

do j=1,9
s30=s30+num(j)+nuf(j)
endo
tpop(i)=s30
if(idis.eq.1)then
nd(i)=nb(i)
elself
nd(i)=np(i)
endif
if(ifile.eq.2)then
r(i)=x(i)
th(i)=y(i)
if(r(i).gt.rmax)then
rmax=r(i)
endif
endif
sl=0.0
endo
do j=1,9
ncex(i,j)=num(j)*mrate(j)/1000.
sl=sl+ncex(i,j)
endo
s2=0.0
endo
do j=10,18
jm=j-9
ncex(i,j)=nuf(jm)*frate(jm)/1000.
s2=s2+ncex(i,j)
endo
if(idis.eq.1)then
texb(i)=sl+s2
else
texp(i)=sl+s2
endif
tex(i)=sl+s2
if(ifile.ne.2)then
if(ycen.lt.20.or.ycen.gt.980)then
if(y(i).gt.800)then
y(i)=1000.-y(i)
endif
endif
xa=x(i)-xcen
ya=y(i)-ycen
r(i)=sqrt(xa*xa+ya*ya)
if(r(i).gt.rmax)then
rmax=r(i)
endif
if(xa.eq.0.0)then
if(ya.lt.0.0)then
th(i)=4.7124
else
th(i)=1.5708
endif
goto 2345
endif
h=atan(ya/xa)
if(xa.lt.0.0)then
th(i)=h+pi
endif
else
  if(ya.lt.0) then
    th(i) = h + pi/2
  else
    th(i) = h
  endif
endif
endif
continue
enddo

2345

if(iyes.eq.1) then
  type*, 'filename:'
  read(5,100) zname
  open(30, file=zname, status='new')
  write(30, *) ncell
  do i=1,ncell
    if(idis.eq.1) then
      write(30, *) i, r(i), th(i), nb(i), tpop(i), texb(i)
    else
      write(30, *) i, r(i), th(i), np(i), tpop(i), texp(i)
    endif
  enddo
  close(30)
endif

do i=1,ncell
  r(i) = r(i) * 2.5 / rmax
enddo

c basic calculations
c

sl = 0
s2 = 0
s3 = 0
s4 = 0
s5 = 0
s6 = 0

sl = sl + nd(i) * co
s2 = s2 + nd(i) * si
s3 = s3 + nd(i)
enddo
rl = sqrt(sl*sl + s2*s2)

h = atan(s2/sl)
if(sl.lt.0.0) then
  xbar = h + pi/2
else
  if(s2.lt.0.0) then
    xbar = h + pi
  else
    xbar = h
  endif
endif

type*, 'simple mean angle:', xbar
al = 0.0
bl = 60.0
tol = 0.1
nt = s3
npar = 1
nsig=3
a(1)=0.0
b(1)=-70.0
nsr=20
call zxmd(funk5,npar,nsig,a,b,nsr,xk,f,wor,iwor,ier)
call zxgsn(funk,falu1,b1,tol,kest,ier)
kest=xk(1)
if(ier.ne.0)then
  type*, 'error(ier)=' ,ier
endif
  type*, 'initial k estimate:' ,kest

now have deaths, polar coords, and exp deaths

acor=lincor(nd,r,th,tex,ncell)
type*, 'acor:' ,acor
ston=stone(r,nd,tex,ncell)
type*, 'stone test:' ,ston
hed(6)=scorst(nd,r,tex,ncell)
type*, 'result: r test=' ,hed(6)
hed(7)=scothst(nd,th,tex,ncell)
type*, 'result: th test=' ,hed(7)
hed(9)=wlik(nd,r,th,tex,ncell)
kest=wml
xbar=wm2
type*, 'do you want own parm estimates(l:yes), read(S,*)iden
if(iden.eq.1)then
  type*, 'kappa: ' 
  read(S,*)kest
  type*, 'pmu: '
  read(S,*)xbar
endif
hed(8)=wscore(nd,r,th,tex,ncell)
if(hed(9).le.0.0)then
  hlik=0.0
else
  hlik=sqrt(hed(9))
endif
if(hed(8).le.0.0)then
  hscor=0.0
else
  hscor=sqrt(hed(8))
endif
type*, 'scor and lik as N(0,1) variates'
type*, hscor,hlik
type*, 'results: r, th, int tests'
type*, hed(6),hed(7),hed(8),hed(9)
type*, 'ml estimates of k, psi, mu and lik'
type*, tm1,tm2,tm3,t13
  type*,'ml estimates of k, mu and lik2'
type*, wml,wm2,w12
  endif

end of stats section

stop
end

functions and subs

real function stone(r,ncob,tex,n)
real r(n),tex(n)
real alla(500)
integer ncobn(n), n
integer ira(500)
la=n
do i=1,n
ira(i)=i
enddo
call vsrtr(r, la, ira)
s0=0
sl=0
tmax=0
do i=1,n
s0=s0+ncobn(ira(i))
sl=sl+tex(ira(i))
tm=s0/sl
if(tm.gt.tmax)then
  tmax=tm
  ntn=i
endif
enddo
stone=tmax
return
end
real function cov(a,b,c,d)
real a,b,c,d
common/s10/sigma2, alpha
sqd= ((a-c)**2.0) + ((b-d)**2.0)
dist=sqrt(sqd)
cov=sigma2*exp(-alpha*dist)
return
end
real function trend(r, th, b1, b2, b3, b4, b5)
ar=r
alr=log(r)
ass=(b3+b4*ar)*cos(th-b5)
trend=-b1*ar+b2*alr+ass
return
end
real function umard(tp, r, th, tex, ncell)
real r(ncell), th(ncell), tex(ncell)
integer tp(ncell), ncell
return
end
real function funk(x)
real x
common/s2/r1, nt, xbar
common/s3/tex(9000), r(1000), th(1000), ncell
s1=0
s2=0
s3=0
do i=1,ncell
  cox=cos(th(i)-xbar)
  alam=tex(i)*exp(x*cox)
  blam=cox*alam
  s1=s1+cox
  s2=s2+alam
  s3=s3+blam
endo
funk=abs(r1/nt-s3/s2)
type*, 'kappa, funk'
type*, x, funk
return
end
real function lam(pk, ps, pm, pl, pd, r, th)
real pk,ps,pm,pl,pd,r,th
common/sl/pi,pi2,c
ab=exp((pk+ps*r)*cos(th-pm))
if(pl.eq.0.0)then
  lam=ab
else
  ac=pl*pd*r**(pd-1.)*exp(-pl*r**pd)
  lam=ab*ac
endif
return
end

real function lan(pk,pm,pl,pd,r,th)
real pk,pm,pl,pd,r,th
common/sl/pi,pi2,c
ab=exp(pk*cos(th-pm)
if(pl.eq.0.0)then
  lan=ab
else
  ac=pl*pd*r**(pd-1.)*exp(-pl*r**pd)
  lan=ab*ac
endif
return
end

real function corr(nd,x,y,ncell)
real x(ncell),y(ncell)
integer nd(ncell)
integer ncell
sl=0
s2=0
s4=0
s5=0
s6=0
do i=1,ncell
  sl=sl+nd(i)*x(i)
  s2=s2+nd(i)*x(i)*x(i)
  s3=s3+nd(i)*y(i)
  s4=s4+nd(i)*y(i)*y(i)
  s5=s5+nd(i)*y(i)*x(i)
  s6=s6+nd(i)
endo
da=s5-s3*sl/s6
b=s2-sl*sl/s6
c=s4-s3*s3/s6
bc=b*c
if(bc.lt.0)then
  bc=0.0001
endif
  corr=a/sqrt(bc)
return
end

real function lincor(nd,r,th,tex,ncell)
real r(ncell),th(ncell)
real tex(ncell)
real smr(1000)
integer nd(ncell)
integer ncell
real corr
common/sl/pi,pi2,c
sl=0
s2=0
do i=1,ncell
  thc(i)=cos(th(i))
end
ths(i)=sin(th(i))
s1=s1+nd(i)*thc(i)
s2=s2+nd(i)+ths(i)
endo
cm=atan(s2/s1)
if(s1.1e.0)then
  pmu=cm+pi
else
  if(s2.1e.0)then
    pmu=pi2+cm
  else
    pmu=cm
  endif
endif
doi=1,ncell
thc(i)=cos(th(i)-pmu)
ths(i)=sin(th(i)-pmu)
if(tex(i).eq.0.0)then
  srnr(i)=0.0
else
  srnr(i)=nd(i)/tex(i)
endif
endo
a=corr(snr,r,thc,ncell)
b=corr(snr,r,ths,ncell)
c=corr(snr,thc,ths,ncell)
lincor=(a*a+b*b-2*a*b*c)/(1.-c*c)
return
end
real function scorst(nd,r,tex,ncell)
real r(ncell),tex(ncell)
integer nd(ncell),ncell
s0=0.0
s1=0.0
s2=0.0
s3=0.0
s4=0.0
do i=1,ncell
  s0=s0+r(i)*tex(i)
  s1=s1+r(i)*nd(i)
  s2=s2+r(i)*r(i)*tex(i)
  s3=s3+tex(i)
  s4=s4+nd(i)
endo
ratio=s4/s3
ab=(s1-ratio*s0)**2.
ac=ratio*(s2-s0*s0/s3)
sorst=ab/ac
return
end
real function scothst(nd,th,tex,ncell)
real th(ncell),tex(ncell)
integer nd(ncell),ncell
common/s1/pi,pi2,c
s1=0.0
s2=0.0
do i=1,ncell
  s1=s1+sin(th(i))*nd(i)
  s2=s2+cos(th(i))*nd(i)
endo
h=atan(s1/s2)
if(s2.1t.0.0)then
  emu=h+pi
else
if(s1.lt.0.0) then
  emu=h+pi2
else
  emu=h
endif
endif
sl0=0.0
sl1=0.0
sl2=0.0
sl3=0.0
sl4=0.0
do i=1,ncell
  s10=s10+tex(i)*cos(th(i)-emu)
  s11=s11+tex(i)
  s12=s12+tex(i)*(cos(th(i)-emu)**2.)
  s13=s13+nd(i)
  s14=s14+nd(i)*cos(th(i)-emu)
enddo
ratio=s13/s11
ab=(s14-ratio*s10)**2.
ac=ratio*(s12-s10*s10/s11)
scothst=ab/ac
return
end
real function wlik(nd,r,th,tex,ncell)
real r(ncell),th(ncell),tex(ncell)
integer nd(ncell),ncell
real x(3),a(3),b(3),g(3),work(40)
real kest
integer iworl(5),liw,lw
external functl
common/s20/idor
common/s4/kest
common/s1/pi,pi2,c
common/s2/r1,nt,xbar
common/s50/tml,tm2,tm3,t13
common/s51/wml,wm2,w12
x(1)=kest
x(2)=0.01
x(3)=xbar
ib=0
do i=1,3
  a(i)=0.0
enddo
b(1)=20.0
b(2)=20.0
b(3)=6.283
liw=5
lw=40
n2=3
ier=1
idor=0
n=3
npar=3
nsig=3
nsr=25
call e04kae(n2,ib,a,b,x,f,g,iewl,liw,work,lw,ier)
call zxmwd(functl,npar,nsig,a,b,nsr,x,f,work,iewl,ier)
if(ier.ne.0) then
  write(6,*)'fail(3)=' ,ier
endif
type*,'parms=' ,x(1),x(2),x(3)
tml=x(1)
tm2=x(2)
tm3=x(3)
s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
do i=1,ncell
  s1=s1+nd(i)*log(tex(i))
  s2=s2+nd(i)*cos(th(i)-x(3))
  s3=s3+nd(i)*r(i)*cos(th(i)-x(3))
  s4=s4+nd(i)
  s5=s5+tex(i)*exp((x(1)+x(2)*r(i))*cos(th(i)-x(3)))
enddo
s2=x(1)*s2
s3=x(2)*s3
bw1=s1+s2+s3-s4*log(s5)
t13=bw1

c now do ord von Mises
c

x(1)=kest
x(2)=xbar
do i=1,2
  a(i)=0.0
enddo
b(1)=20.0
b(2)=6.283
n2=2
ifail=1
idor=1
np=2
nsig=3
nsr=10
call e04kae(n2,ib,a,b,x,f,g,ivar,liw,work,ifail)
call zxmwd(functl,nsig,a,b,nsr,x,f,ivar,ier)
if(ier.ne.0)then
  type*, 'fail(2)=', ier
  type*, 'parms(2-model):', x(1), x(2)
endif
wm1=x(1)
wml=x(2)
s1=0
s2=0
s3=0
s4=0
s5=0
do i=1,ncell
  s1=s1+nd(i)*log(tex(i))
  s2=s2+nd(i)*cos(th(i)-x(2))
  s3=s3+nd(i)
  s4=s4+tex(i)*exp(x(1)*cos(th(i)-x(2)))
enddo
s2=x(1)*s2
cwl=s1+s2+s3-s4*log(s4)
w12=cwl
wlik=2*(bw1-cwl)
return
end
subroutine functl(n,xc,fc)
real fc,gc,lest,delest
real xl,x2,x3,x4,x5
real lam,lan
real x(2)
end
common/s3/tex(9000),r(1000),th(1000),ncell
common/s30/nd(9000)
common/s20/idor
s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
s7=0
s8=0
s9=0
s10=0
sll=0
if(idor.eq.0)then
xl=xe(1)
x2=xe(2)
x3=xe(3)
do i=1, neell
skap=xl+x2*r(i)
cosx=cos(th(i)-x3)
sinx=sin(th(i)-x3)
s1=s1+nd(i)*cosx
s2=s2+nd(i)*r(i)*cosx
s3=s3+nd(i)
s4=s4+exp(skap*cosx)
s5=s5+nd(i)*sinx
s6=s6+nd(i)*r(i)*sinx
lest=0.0
delest=1.0
ay=lam(xl,x2,x3,lest,delest,r(i),th(i))
ay=ay*tex(i)
s7=s7+ay
s8=s8+skap*ay
s9=s9+r(i)*cosx*ay
s10=s10+cosx*ay
s11=s11+nd(i)*log(tex(i))
enddo
a=sll+x1*s1+x2*s2-s3*log(s7)
gc(1)=-(s1-s3*s10/s7)
gc(2)=-(s2-s3*s9/s7)
gc(3)=-(x1*s5+x2*s6-s3*s8/s7)
f=-a
else
2 variable model
dx1=xe(1)
dx2=xe(2)
do i=1, neell
ab=cos(th(i)-x2)
ac=sin(th(i)-x2)
lest=0.0
delest=1.0
az=lam(xl,x2,lest,delest,r(i),th(i))
s1=s1+nd(i)*ab
s2=s2+nd(i)*log(tex(i))
s3=s3+nd(i)
s4=s4+tex(i)*az
s5=s5+tex(i)*ab*az
s6=s6+tex(i)*ac*az
s7=s7+nd(i)*ac
enddo
all=s2+x1*s1-s3*log(s4)
gc(1) = -(s1 - s3*(s5/s4))  
gc(2) = -(x1*s7 - x1*s3*(s6/s4))  
fcc = -a11  
endif  
return  
end  

subroutine funk5(n,x,f)  
real x(n)  
common/s2/rl,nt,xbar  
common/s3/tex(9000),r(1000),th(1000),nce11  
s1 = 0  
s2 = 0  
s3 = 0  
do i = 1, nce11  
cox = cos(th(i) - xbar)  
alam = tex(i)*exp(x(1)*cox)  
blam = cox*lam  
s1 = s1 + cox  
s2 = s2 + lam  
s3 = s3 + blam  
enddo  
f = 0.0  
f = abs(rl/nt - s3/s2)  
return  
end  

subroutine funct2(n,xc,fc,gc)  
real fc,gc(n),xc(n)  
real xl,x2,x3,x4,x5  
real lam,lan  
real lest,delest  
common/s3/tex(9000),r(1000),th(1000),nce11  
common/s30/nd(9000)  
common/s20/idor  
s1 = 0  
s2 = 0  
s3 = 0  
s4 = 0  
s5 = 0  
s6 = 0  
s7 = 0  
s8 = 0  
s9 = 0  
s10 = 0  
s11 = 0  
if(idor.eq.0)then  
x1 = xc(1)  
x2 = xc(2)  
x3 = xc(3)  
doi = 1, nce11  
kap = x1 + x2*r(i)  
cosx = cos(th(i) - x3)  
sinx = sin(th(i) - x3)  
s1 = s1 + nd(i)*cosx  
s2 = s2 + nd(i)*r(i)*cosx  
s3 = s3 + nd(i)  
s4 = s4 + exp(kap*cosx)  
s5 = s5 + nd(i)*sinx  
s6 = s6 + nd(i)*r(i)*sinx  
lest = 0.0  
delest = 1.0  
ay = lam(x1,x2,x3,lest,delest,r(i),th(i))  
ay = ay*tex(i)  
s7 = s7 + ay  
s8 = s8 + kap*sinx*ay
\begin{verbatim}
s9=s9+r(i)*cosx*ay
s10=s10+cosx*ay
s11=s11+nd(i)*log(tex(i))
enddo
\end{verbatim}
\begin{verbatim}
a=-s11+x1*s1+x2*s2-s3*log(s7)
gc(1)=-(s1-s3*s10/s7)
gc(2)=-(s2-s3*s9/s7)
gc(3)=-(x1*s5+x2*s6-s3*s8/s7)
f_{c}=-a
else
\end{verbatim}
\begin{verbatim}
c 2 variable model
x1=xc(1)
x2=xc(2)
do i=1,ncell
ab=cos(th(i)-x2)
ac=sin(th(i)-x2)
lest=0.0
delest=1.0
az=lan(x1,x2,lest,delest,r(i),th(i))
s1=s1+nd(i)*ab
s2=s2+nd(i)*log(tex(i))
s3=s3+nd(i)
s4=s4+tex(i)*az
s5=s5+tex(i)*ab*az
s6=s6+tex(i)*ac*az
s7=s7+nd(i)*ac
enddo
\end{verbatim}
\begin{verbatim}
a1l=s2+x1*s1.s3*log(s4)
gc(1)=-(s1-s3*(s5/s4))
gc(2)=-(x1*s7-x1*s3*(s6/s4))
f_{c}=-a1l
endif
\end{verbatim}
\begin{verbatim}
return
end
\end{verbatim}

\begin{verbatim}
real function wscore(nd,r,th,tex,ncell)
real r(ncell),th(ncell)
real tex(ncell)
integer nd(ncell)
integer ncell
real ikk,imm,ipp,ikm,ill
real ipk,ipm,ilm,ilk
real imk
real lest,delest,kest
real lam,lan
common/sl/pi,pi2,c
common/s2/rl,nt,xbar
common/s25/idrt
common/s4/kest
s2=0.0
s3=0
s4=0
s5=0
s6=0
do i=1,ncell
thc(i)=cos(th(i))
ths(i)=sin(th(i))
endo
doi=1,ncell
s2=s2+nd(i)*ths(i)
s3=s3+nd(i)*thc(i)
s4=s4+nd(i)*r(i)*ths(i)
\end{verbatim}
s5 = s5 + nd(i) * r(i) * thc(i)
enddo
ab = s2 + s4
ac = s3 + s5
h = atan(ab/ac)
if(ac .lt. 0) then
  xbar1 = h + pi
else
  if(ab .lt. 0.0) then
    xbar1 = h + pi2
  else
    xbar1 = h
  endif
endif
if(idrt .eq. 1) then
  type*, 'xbar and xbar1'
ten*,xbar,xbar1
endif
cz = cos(xbar)
sz = sin(xbar)
do i = 1, ncell
  thc(i) = cos(th(i) - xbar)
ths(i) = sin(th(i) - xbar)
endo
d9 = 0
d10 = 0
d11 = 0
d12 = 0
d13 = 0
d14 = 0
d15 = 0
d16 = 0
d17 = 0
d18 = 0
d19 = 0
d20 = 0
d21 = 0
d22 = 0
leste = 0.0
dellest = 1.0
do i = 1, ncell
  abz = lan(kest, xbar, lest, dellest, r(i), th(i))
  abz = tex(i) * abz
  d9 = d9 + nd(i)
d10 = d10 + abz
d11 = d11 + thc(i) * thc(i) * abz
d12 = d12 + thc(i) * abz
d13 = d13 + ths(i) * ths(i) * abz
d14 = d14 + ths(i) * abz
d15 = d15 + r(i) * thc(i) * abz
d16 = d16 + r(i) * thc(i) * thc(i) * abz
d17 = d17 + thc(i) * ths(i) * abz
d18 = d18 + r(i) * ths(i) * abz
d19 = d19 + r(i) * thc(i) * ths(i) * abz
d20 = d20 + r(i) * r(i) * thc(i) * thc(i) * abz
d21 = d21 + nd(i) * r(i) * thc(i) * abz
d22 = d22 + nd(i) * r(i) * thc(i) * abz
endo
dratio = d9/d10
ipp = ratio * (d20 - (d15 * d15)/d10)
ipk = ratio * (d16 - (d12 * d15)/d10)
ikk = ratio * (d11 - (d12 * d12)/d10)
ipp = ipp + ratio * (s18 + kest * (s19 - (s14 * s15)/s10))
imm = kest * (cz * s5 - cz * s4)

& +ratio*(kest*kest*(sl3-(sl4*sl4)/s10)-kest*sl2)
  ikm=sz*s3-cz*s2
& +ratio*(sl4+kest*(s17-(sl4*sl4)/s10))
  imk=ikm
  a=ipk*imm-ipm*ikm
  b=-ipk*imk+ipm*ikk
  az=a*ipk+b*ipm
  cz=ikk*imm-ikm*ikm
  denom=ipp-az/cz
  aws=s21-ratio*s15
  wscore=aws*aws/denom
  return
end
real function tense(ra,tha,b,a,k,ps,mu)
real ra,tha,b,a,k,ps,mu
a1=ra**b
a2=exp(-a*ra)
a3=exp((k+ps*ra)*cos(tha-mu))
tense=a1*a2*a3
return
end

subroutine grz(r,th,nt,n)
real r(n),th(n),x(500),y(500)
integer nt(n),n
rscreen=2.5
rm=0.0
do i=1,n
  if(r(i).ge.rm)then
    rm=r(i)
  endif
enddo
do i=1,n
  r(i)=r(i)*(rscreen/rm)
  x(i)=r(i)*cos(th(i))
  y(i)=r(i)*sin(th(i))
enddo
5678 type*, 'screen(0) or plotter(1)'
read(5,*)idet
if(idet.eq.0)then
call gino
call t4010
call units(25.0)
else
call gino
call call044
call scale(20.0)
endif
call chapos(20.0,20.0)
call piccle
rsn=rscreen+0.3
call shift2(rsn,rsn)
call movt02(-rsn,0.0)
call arcto2(0.0,0.0,-rsn,-0.001,0)
call movt02(0.0,0.0)
call symbol(5)
do i=1,n
ax=x(i)-0.2
ay=y(i)-0.1
call movt02(ax,ay)
call chaint(nt(i),3)
endo
type*, 'cntrl-sht f6 l'
read(5,*)neft
call piccle
call devend
call ginend
type*, 'more graphics(l:yes)'
read(5,*)inore
if(inore.eq.l)then
goto 5678
endif
return
end
real function scoR(nd,r,ncex,nsex,ncell)
real r(ncell),ncex(ncell,nsex)
integer nsex,ncell,nd(ncell,nsex)
s6=0
s7=0
do i=l,ncell
s4=0
s5=0
do j=l,nsex
s0=0
s1=0
s2=0
enddo
s4=s4+nd(i,j)*(r(i)-s0/s1)
s5=s5+nd(i,j)*(s2/sl-(sO/sl)**2)
enddo
s6=s6+s4
s7=s7+s5
enddo
scoR=s6*s6/s7
else
s6=0
s7=0
endif

real function scoTH(nd,th,ncex,nsex,ncell)
real th(ncell),ncex(ncell,nsex)
integer nsex,ncell,nd(ncell,nsex)
s3=0
s4=0
s1=0
s2=0
s1=sl+s1+sin(th(i))nd(i,j)
s2=sl+s2+cos(th(i))nd(i,j)
enddo
emu=atan(s3/s4)
s6=0
s7=0
endif
s6=s6+ncex(i,j)*cos(th(il)-emu)
s1=s1+ncex(i,j)
s2 = s2 + (cos(th(il) - emu)**2) * ncex(il, j)
enddo
s4 = s4 + nd(i, j) * (cos(th(i) - emu) - s0/s1)
sp = s5 + nd(i, j) * (s2/s1 - (s0/s1)**2)
enddo
s6 = s6 + s4
s7 = s7 + s5
enddo
scoTH = s6*s6/s7
print *, 'mean angle:', emu
print *, 'test for cone:', scoTH
return
end
File _$255$DUA2:[MC.T.AL]SIMREG2.FOR;63 (18730,78,0), last revised on 24-JUL-1990 15:41, is a 56 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 61 bytes.

Job SIMREG2 (616) queued to SYS$PRINT on 24-AUG-1990 13:02 by user MCT, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:11 from queue LASER.
double precision dseed
double precision dsl
real pr(1000,18),pm(9000),wk1(9000),wk2(9000)
real wkvec(50000),rvec(1,50000),rmv(50000)
real statr(1000)
integer irstat(1000)
real var(10000)
real pf(9000),rate(20)
real tpop(1000)
real mpopl(1000),fpopl(1000)
real res(1000)
real mrate(9),frate(9),x(9000),y(9000)
real reg(9000),hed(20)
real rnl(1000),thnl(1000),txnl(1000)
real bp,ap,kp,ip,mup
real prob(6)
real regcon(1000),regex(1000)
real pml(9000),pf1(9000)
real stat(15),chialfl(7),zedalf(7)
real corralfl(7)
real stonalf(7)
real crit(15,20)
real nce(1000,2)
real lincor,wscore,scor,sooth,wlik,umard
real scorst,soothst
real prob(5),montc(10,2000)
real kest
real wor(12),xk(1),a(1),b(1)
integer iwor(1)
integer num(9),nuf(9),ia(9000),nm(9000),nf(9000)
integer tp(1000)
integer ir1(1000),ir2(1000),ib(9000)
integer ncon(1000,20)
integer nb(9000),np(9000)
integer ndnl(1000),ipx(100000)
integer nde(1000,2)
integer ncount1(15,5)
integer ndgen(9000)
real nce(1000,20)
real texb(1000),texp(1000)
integer mp,fpop
integer mp,fp
character*20 fname,wname,tname,zname
external funk
external funk5
common/s1/pi,pi2,c
common/s2/rl,nt,xbar
common/s3/tex(9000),r(1000),th(1000),ncell
common/s4/kest
common/s10/sigma2,alpha
common/s20/idor
common/s25/idrt
common/s30/nd(9000)
common/s50/tm1,tm2,tm3,tl3
common/s51/wm1,wm2,tl2
data(stonalf(i),i=1,3)/0.0,0.0,0.0/  
data(corralf(i),i=1,3)/0.0,0.0,0.0/  
data(chialfl(i),i=1,3)/3.841,6.635,10.827/  
data(zedalf(i),i=1,3)/1.645,2.326,3.090/  
do i=1,3  
crit(1,i)=chialfl(i)  
crit(2,i)=chialfl(i)  
crit(3,i)=zedalf(i)  
crit(4,i)=chialfl(i)
crit(5,i)=chialfl(i)
crit(6,i)=stonalf(i)
enddo
pi=3.14159
pi2=6.283185
c=0.017453
type*, 'power(0) or stats(1) or residual test(2):'
read(5,*)idrt
if(idrt.eq.0)then
type*, 'note: 20 slots are provided for male(1-9)'
type*, 'and female (10-18)'
type*, 'enter parameters:'
type*, 'enter shape(>=1),scale(>0),conc(>=0)'
type*, 'inter(>=0),mu(radians)'
read(5,*)bp,ap,kp,ip,mup
type*, 'you must enter popn and exp rates for 18 age*sex'
type*, 'groups from 2 files'
type*, 'enter total male and female pop of all regions:'
read(5,*)mpop,fpop
type*, 'enter no of radial cells:'
read(5,*)nr
type*, 'enter no of angle cells:'
read(5,*)nth
ncell=nr*nth
type*, 'no of cells=:',ncell
type*, 'age:sex structure now follows'
type*, 'the files must have <ncell> units each'
type*, 'with an age:sex <18> structure'
type*, 'option(1): all regions same age:sex'
type*, 'option(2): different structure in each'
type*, 'input option:'
read(5,*)nopt
if(nopt.eq.1)then
    type*, 'enter file with 18 age:sex cells:'
    type*, 'enter filename:'
    read(5,100)fname
    format(a20)
    open(15,file=fname,status='old')
    do i=1,9
        read(15,*)num(i)
    enddo
do i=1,9
    read(15,*)nuf(i)
enddo
    close(15)
do i=1,ncell
do j=1,9
    ncon(i,j)=num(j)
    enddo
do j=10,18
    il=j-9
    ncon(i,j)=nuf(il)
    enddo
derift else
    type*, 'enter filename:'
    read(5,100)fname
    open(15,file=fname,status='old')
    do i=1,ncell
        read(15,*)(ncon(i,j),j=1,18)
    enddo
    close(15)
endif
type*, 'now enter file for 18 expected rates(/1000):'
type*, 'if no expected rates are needed use file of ls'
read(5,100) wname
open(16, file=wname, status='old')
do i=1,18
read(16,*) rate(i)
rate(i)=rate(i)/1000.
dndo
close(16)

now calculate expected values

do i=1,ncell
do j=1,18
ncex(i,j)=ncon(i,j)*rate(j)
dndo
s1=0
s2=0
do j=1,9
s1=s1+ncex(i,j)
j1=j+9
s2=s2+ncex(i,j)
j1
enddo
ncex(i,1)=s1
ncex(i,2)=s2
dndo

calculate rand th of cells

should enter rmax here, but i have set it to 1
rmax=1
k=0
do i=1,nr
a5=(float(i)/float(nr))*rmax
do j=1,nth
k=k+1
b5=(float(j)/float(nth))*6.283
r(k)=a5
th(k)=b5
dndo
dndo

bayesian(1) or not(0)'
read(5,*) ibayes
if(ibayes.eq.1)then
type*, 'enter alpha and sigma2:'
read(5,*) alpha, sigma2
k2=0

do i=1,ncell
do j=1,i
k2=k2+1
xl=r(i)*cos(th(i))
yl=r(i)*sin(th(i))
x2=r(j)*cos(th(j))
y2=r(j)*sin(th(j))
var(k2)=cov(xl,yl,x2,y2)
dndo
dndo
dseed=8429109.d0
nr=1
ir=1
wkvec(1)=0.0

call ggnsm(dseed,nr,ncc11, var,ir,rvec,wkvec,iér)
if(iér.ne.0)then
  type*,'error=',iér
endif
endif

if(iér.ne.0)then
  type*,'enter no of simulations:'
  read(5,*)nsim
  type*,'press cntrl shift f6 and l'
  read(5,*)dert
endif

do calculation of norm constant

c

c

c
c
c
c
c
c
c

sl=0
do i=1,ncc11
  if(ibayes.ne.1)then
    rmv(i)=tense(r(i),th(i),bp,ap,kp,ip,mup)
  else
    a23=log(tense(r(i),th(i),bp,ap,kp,ip,mup))
    b23=a23+rvec(1,i)
    rmv(i)=exp(b23)
  endif
enddo
s0=0
do j=1,18
  s0=s0+ncex(i,j)*rmv(i)
enddo
sl=sl+s0
dseed=123457.d0
ia(l)=-l
ib(l)=-l
nreg9=ncc11*9
do i=1,nreg9
  pm(i)=pm(i)/s15
  pf(i)=pf(i)
enddo
dseed=123457.d0
ia(l)=-l
ib(l)=-l
nreg9=ncc11*9
do i=1,nreg9
  pm(i)=pm(i)/s15
  pf(i)=pf(i)/s16
enddo
iteration loop starts here

ntest=6
write(18,*)nsim,ntest
do kl=1,nsim
type*, 'iteration:', kl
type*, 'prob total: male:', sl5, 'female:', sl6

do generation now
nreg9=ncell*9
mp=mpop
fp=fpop
call cellgen(sl5, sl6, pm, pf, nreg9, mp, fp, ia, ib, ndgen &
, nm, nf, ds1)

num > death> male
nf > death> female
pr > regional prob
nd > total deaths

do statistics here
k=0
s5=0
do i=1, ncell
s3=0
s4=0
do j=1, 9
k=k+1
s3=s3+num(k)
s4=s4+nf(k)
enddo
nde(i,1)=s3
nde(i,2)=s4
tp(i)=s3+s4
s5=s5+tp(i)
tex(i)=nce(i,1)+nce(i,2)
nd(i)=tp(i)
enddo
type*, 'total number=', s5

s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
do i=1, ncell
co=cos(th(i))
si=sin(th(i))
s1=s1+nd(i)*co
s2=s2+nd(i)*si
s3=s3+nd(i)
enddo
r1=sqrt(s1*s1+s2*s2)
h=atan(s2/s1)
if(s1.lt.0.0)then
xbar=h+pi
else
if(s2.lt.0.0)then
xbar=h+pi2
else
xbar=h
endif
endif
a=0.0
b=60.0
tol=0.1
nt=s3
np=1
ns=3
a(1)=0.0
b(1)=70.0
nsr=20
call zxmwd(funk5, np, ns, a, b, nsr, xk, f, wor, iwor, ier)
kest=xk(1)
if(ier.ne.0)then
type*, 'error(ier)=' ,ier
endif
type*, 'xbar,kest:' ,xbar,kest
c
hed(1)=scorst(tp, r, tex, ncell)
ked(2)=scothst(tp, th, tex, ncell)
acor=linacor(tp, r, th, tex, ncell)
type*, 'corr=' ,acor

bootstrap here

s=0
s2=0
ds=88565783.d0
type*, 'bootstrapping'
k=ncell
nb=ncell-20
do i=1,50
call ud(ds1, k, nb, ipx)
do i=1,nb
  type*, 'k, nb, ipx:' ,k, nb, ipx
  ndn1(i)=tp(ipx(i))
  rni1(i)=r(ipx(i))
  th1(i)=th(ipx(i))
  tex1(i)=tex(ipx(i))
endo
ab=linacor(ndn1, rni1, th1, tex1, nb)
s1=s+s
s2=s2+s*ab
endo
cor=sl/50.0
varbb=(s2-nb*s)*corb/(nb-1)
sd=sqrt(varbb)
zcor=(acor-corb)/sd3
type*, 'zcor=' ,zcor
ked(3)=zcor
ked(4)=wlk(tp, r, th, tex, ncell)
kest=wm1
xbar=wm2
ked(5)=wscore(tp, r, th, tex, ncell)
type*, 'ml estimates'
type*, '3-mode: k, psi, mu, lik'
type*, '2-mode: k, mu, lik'
type*, 'wm1, wm2, w12'
c
mardia test not used here

c
c
stone's test instead
hed(6)=stone(r, tp, tex, ncell)
type*, 'tests(r,th,corr,lik,score,stone)'
type*, (hed(j), j=1,6)
do j=1,ntest
do k=1,3
if(hed(j) .ge. crit(j,k)) then
ncount1(j,k) = ncount1(j,k) + 1
endif
enddo
montc(j,k) = hed(j)
enddo
write(18,*) (montc(j,k), j=1,ntest)

end of simulation loop here
endo
open(17, file=tname, status='new')
write(17,*) 'input parameters'
do j=1,ntest
write(17,*) 'count for one-tail test:', j
write(17,*) (ncount1(j,k), k=1,4)
do k=1,4
prob1(k) = float(ncount1(j,k))/float(nsim)
enddo
write(17,*) 'one-tail test prob', j
write(17,*) (prob1(k), k=1,4)
do easygraph graphics here
endo
c statistics are here
c
else
type*, 'enter age*sex rates file:'
read(5,100) fname
open(15, file=fname, status='old')
do j=1,9
read(15,*) mrate(j), frate(j)
endo
close(15)
type*, 'enter region file'
read(5,100) tname
type*, 'x-y(1) or r-th(2) file'
read(5,*) ifile
type*, 'one(1) or two(2) diseases on file:'
read(5,*) iont
type*, 'enter x,y coords of centre'
read(5,*) xcen, ycen
type*, 'which disease: bronch(1) pneum(2)'
read(5,*) idis
open(25, file=tname, status='old')
read(25,*) ncell
rmax=0
s56=0
s57=0
do i=1,ncell
s31=0
s33=0
s30=0
if(iont .eq. 1) then
read(25,*)reg(i),x(i),y(i),nb(i),
    (num(j),nuf(j),j=1,9)
np(i)=nb(i)
elseread(25,*)reg(i),x(i),y(i),nb(i),np(i),
    (num(j),nuf(j),j=1,9)
&
    endif
do j=1,9
s30=s30+num(j)+nuf(j)
s31=s31+num(j)
s33=s33+nuf(j)
enddo
tpop(i)=s30
mpopl(i)=s31
fpop1(i)=s33
s56=s56+mpopl(i)
s57=s57+fpop1(i)
if(idis.eq.1)then
    nd(i)=nb(i)
else
    nd(i)=np(i)
endif
if(ifile.eq.2)then
    r(i)=x(i)
th(i)=y(i)
x(i)=r(i)*cos(th(i))
y(i)=r(i)*sin(th(i))
c note th(i) massumed to be in radians here
endif
if(r(i).gt.rmax)then
    rmax=r(i)
endif
endif
sl=0.0
do j=1,9
ncex(i,j)=num(j)*mrate(j)/1000.
sl=sl+ncex(i,j)
enddo
s2=0.0
do j=10,18
jm=j-9
ncex(i,j)=nuf(jm)*frate(jm)/1000.
s2=s2+ncex(i,j)
enddo
if(idis.eq.1)then
texb(i)=sl+s2
else
texp(i)=sl+s2
endif
tex(i)=sl+s2
if(ifile.eq.2)then
    if(ycen.lt.20.or.ycen.gt.980)then
        if(y(i).gt.800)then
            y(i)=1000.-y(i)
        endif
    endif
    xa=x(i)-xcen
    ya=y(i)-ycen
    r(i)=sqrt(xa*xa+ya*ya)
    if(r(i).gt.rmax)then
        rmax=r(i)
    endif
    if(xa.eq.0.0)then
        if(ya.lt.0.0)then
th(i)=4.7124
else
th(i)=1.5708
endif

goto 2345
endif

h=atan(ya/xa)
if(xa.lt.0)then
th(i)=h+pi
else
if(ya.lt.0)then
th(i)=h+pi2
else
th(i)=h
endif
endif
endif

continue
enddo

mpop=s56
fpop=s57

type*, 'male and female pops:', mpop, fpop

type*, 'do you want to write data to file (1:yes)'
read(5,*)iyes
if(iyes.eq.1)then

type*, 'filename:'
read(5,100)zname
open(30, file=zname, status='new')
write(30,*)ncell

do i=1,ncell
if(idis.eq.1)then
write(30,*)i,r(i),th(i),nb(i),tpop(i),texb(i)
else
write(30,*)i,r(i),th(i),np(i),tpop(i),texp(i)
endif
endo

close(30)
endif

do i=1,ncell
r(i)=r(i)*2.5/rmax
endo

c basic calculations

c
if(idrt.eq.2)then

c now do residual analysis

c

type*, 'enter residual filename:'
read(5,100)tname
open(23, file=tname, status='old')
read(23,*)nreg
rmax=0

do i=1,nreg
read(23,*)r(i),th(i),res(i)
endo
type*, 'x-y file (1) or r-thfile (2):'
type*, 'th must be in radians'
read(5,*)ifile3
if(ifile3.eq.1)then
type*, 'enter centre coords (x,y):'
read(5,*)xcen,ycen
endif
do i=1,nreg
if(ifile3.eq.1)then
x(i)=r(i)
y(i)=th(i)
al=x(i)-xcen
a2=y(i)-ycen
r(i)=sqrt(al*a1+a2*a2)
if(al.eq.0.0)then
if(a2.lt.0.0)then
th(i)=4.7124
else
th(i)=1.5708
endif
goto 2341
endif
h=a2/a1
h=atan(h)
if(a1.lt.0)then
th(i)=h+pi
else
if(a2.lt.0)then
th(i)=h+pi2
else
th(i)=h
endif
endif
continue
else
x(i)=r(i)*cos(th(i))
y(i)=r(i)*sin(th(i))
endif
enddo
do i=l,nreg
if(r(i).gt.rmax)then
rmax=r(i)
endif
enddo
r(i)=r(i)*2.5/rmax
type*, 'ML ests: shape, scale, kappa, psi, pmu'
read(5,*) bp, ap, kp, ip, mup
s1=0
do i=l,nreg
rmv(i)=tense(r(i), th(i), bp, ap, kp, ip, mup)
s0=0
do j=1,18
s0=s0+ncex(i,j)*rmv(i)
enddo
s1=s1+s0
enddo
cnorm=s1
type*, 'norm const:', cnorm
s15=0
s16=0
k1=0
k2=0
pm(0)=0
pf(0)=0
do i=l,nreg
do j=1,9
pr(i,j)=ncex(i,j)*rmv(i)/cnorm
k1=k1+1
pm(k1)=pr(i,j)
s15=s15+pm(k1)
enddo
do  j=10,18
pr(i,j)=ncex(i,j)*rmv(i)/cnorm
k2=k2+1
pf(k2)=pr(i,j)
s16=s16+pf(k2)
enddo
enddo
dseed=176542.d0
dsl=482745.d0
ia(l)=-l
ib(l)=-l
nreg9=nreg*9
do  i=1,nreg9
pm(i)=pm(i)/s15
pf(i)=pf(i)/s16
enddo
c now doing generation
c
do 1=1,99
nreg9=nreg*9
mp=mpop
fp=fpop
call cellgen(s15,s16,pm,pf,nreg9,mp,fp,ia,ib,
&  ndgen,nm,nf,dsl)
type*,'gen ended'
s6=0
k=0
do  i=1,nreg
s81=0
s82=0
do  j=1,9
k=k+1
s81=s81+pm(k)
s82=s82+pf(k)
enddo
az=mp*s81+fp*s82
res(i)=ndgen(i)-az
s6=s6+az
enddo
type*,'res set up for loop:',1
c now call moran
call moran2(x,y,res,nreg,ai,ain,air)
statr(l)=ai
type*,'moran stat:',statr(l)
enddo
type*,'enter moran value:'
read(5,*)statr(100)
do ik=1,100
irstat(ik)=ik
enddo
do sorting and find mc value
call vsrtr(statr,100,irstat)
rank=0
do  l=1,100
if(irstat(l).eq.100)then
rank=float(l)
goto 6666
endif
enddo
6666 continue
prr=1. - rank/100.
type*, 'one tail prob(MC):', prr

else
s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
do i=1, ncell
co=cos(th(i))
si=sin(th(i))
s1=s1+nd(i)*co
s2=s2+nd(i)*si
s3=s3+nd(i)
dendo
r1=sqrt(s1*s1+s2*s2)
type*, 'stats:nc, ns, nt, rl'
type*, s1, s2, s3, r1
h=atan(s2/s1)
if(s1. lt. 0.0) then
  xbar=h+pi
else
  if(s2. lt. 0.0) then
    xbar=h+pi2
  else
    xbar=h
  endif
endif
type*, 'simple mean angle:', xbar
al=0.0
bl=60.0
tol=0.1
nt=s3
npar=1
nsig=3
a(1)=0.0
b(1)=-70.0
nsr=20
call zxmd(funk5,npar,nsig,a,b,nsr,xk,f,wor,iwor,ier)
call zxgsn(funk,al,bl,tol,kest,ier)
kest=xk(1)
if(ier. ne. 0) then
  type*, 'error(ier)=' , ier
endif
type*, 'initial k estimate:', kest

now have deaths, polar coords, and exp deaths

acor=lincor(nd,r,th,tex,ncell)
type*, 'acor:', acor
ston=stone(r,nd,tex,ncell)
type*, 'stone test:', ston
hed(6)=scorst(nd,r,tex,ncell)
type*, 'result: r test=', hed(6)
hed(7)=scothst(nd,th,tex,ncell)
type*, 'result: th test=', hed(7)
ked=wm1
xbar=wm2
type*, 'do you want own parm estimates(1:yes)'
read(5,*), iden
if(iden. eq. 1) then
type* , 'kappa: '
read(5,*) kest

read(5,*) xbar
endif

dif(hed(9).le.0.0)then
hlik=0.0
else
hlik=sqrt(hed(9))
endif

if(hed(8).le.0.0)then
hscor=0.0
else
hscor=sqrt(hed(8))
endif
type*, 'scor and lik as N(0,1) variates'
type*, hscor, hlik
type*, 'results: r, th, int tests'
type*, hed(6), hed(7), hed(8), hed(9)
type*, 'ml estimates of k, psi, mu and lik'
type*, tml, tm2, tm3, t13

type*, 'ml estimates of k, mu and lik2'
type*, wml, wm2, w12
endif
endif

end of stats section

stop

c*** functions and subs

subroutine ud(dsl,kp,nb,ipx)
integer kp,nb,ipx(nb)
double precision dsl
do i=1,nb
  ub=ggubfs(dsl)
  ipx(i)=l+int(ub*kp)
type*, 'ipx(i)=', ipx(i)
endo
return
end

real function stone(r,ncob,tex,n)
real r(n), tex(n)
real ala(500)
integer ncob(n), n
integer ira(500)
la=n
do i=1,n
  ira(i)=i
endo
call vsrtr(r,la,ira)
s0=0
sl=0
tmax=0

do i=1,n
  s0=s0+ncob(ira(i))
  sl=sl+tex(ira(i))
  tm=s0/sl
  if(tm.gt.tmax)then
    tmax=tm
  endif
ntn=i
endif
enddo
stone=tmax
return
end
real function cov(a,b,c,d)
real a,b,c,d
common/s10/sigma2,alpha
sqd=((a-c)**2.0+((b-d)**2.0)
dist=sqrt(sqd)
cov=sigma2*exp(-alpha*dist)
return
end
real function trend(r,th,b1,b2,b3,b4,b5)
ar=r
alr=log(r)
ass=(b3+b4*ar)*cos(th-b5)
trend=-b1*ar+b2*alr+ass
return
end
real function umard(tp,r,th,tex,ncell)
real r(ncell),th(ncell),tex(ncell)
integer tp(ncell),ncell
return
end
real function funk(x)
real x
common/s2/r1,nt,xbar
common/s3/tex(9000),r(1000),th(1000),ncell
s1=0
s2=0
s3=0
do i=1,ncell
cox=cos(th(i)-xbar)
alam=tex(i)*exp(x*cox)
blam=cox*alam
s1=s1+cox
s2=s2+alam
s3=s3+blam
endo
defsl=abs(r1/nt-s3/s2)
type*, 'kappa', funk'
type*, x, funk
return
end
real function lam(pk,ps,pm,pl,pd,r,th)
real pk,ps,pm,pl,pd,r,th
common/s1/pi,p12,c
ab=exp((pk+ps*r)*cos(th-pm))
if(pl.eq.0.0)then
lam=ab
else
ac=pl*pd*r**(pd-1.)*exp(-pl*r**pd)
lam=ab*ac
endif
return
end
real function lan(pk,pm,pl,pd,r,th)
real pk,pm,pl,pd,r,th
common/s1/pi,p12,c
ab=exp(pk*cos(th-pm))
if(pl.eq.0.0)then
lan=ab
else
  ac=pl*pd*r**(pd-1.)*exp(-pl*r**pd)
  lan=ab*ac
endif
return
end
real function corr(nd,x,y,ncell)
  real x(ncell),y(ncell)
  integer nd(ncell)
  integer ncell
  s1=0
  s2=0
  s3=0
  s4=0
  s5=0
  s6=0
  do i=1,ncell
    sl=sl+nd(i)*x(i)
    s2=s2+nd(i)*x(i)*x(i)
    s3-s3+nd(i)*y(i)
    s4=s4+nd(i)*y(i)*y(i)
    sS=sS+nd(i)*y(i)*x(i)
    s6=s6+nd(i)
  enddo
  a=sS-s3*sl/s6
  b=s2-s1*sl/s6
  c=s4-s3*s3/s6
  bc=b*c
  if(bc.1e.0)then
    bc=O.OOO1
  endif
  corr=a/sqrt(bc)
return
end
real function lincor(nd,r,th,text,ncell)
  real r(ncell),th(ncell)
  real tex(ncell)
  real smr(1000)
  integer nd(ncell)
  integer ncell
  real corr
  double precision ds1
  common/s1/pi,pi2,c
  sl=0
  s2=0
  do i=1,ncell
    thc(i)=cos(th(i))
    ths(i)=sin(th(i))
    sl=sl+nd(i)*thc(i)
    s2=s2+nd(i)+ths(i)
  enddo
  crn=atan(s2/s1)
  if(sl.le.0)
    prnu=crn+pi
  else
    if(s2.le.0)
      prnu=pi2+crn
    else
      prnu=crn
    endif
  endif
  do i=1,ncell
    thc(i)=cos(th(i)-prnu)
  enddo
  cm=atan(s2/s1)
  if(s1.1e.0)then
    pmu=cm+pi
  else
    if(s2.1e.0)then
      pmu=pi2+cm
    else
      pmu=cm
    endif
  endif
  end
ths(i)=sin(th(i)-pmu)
if(tex(i).eq.0.0)then
   smr(i)=0.0
else
   smr(i)=nd(i)/tex(i)
endif
enddo
a=corr(smr,r,thc,ncell)
b=corr(smr,r,ths,ncell)
c=corr(smr,thc,ths,ncell)
lincor=(a*a+b*b-2*a*b*c)/(1.-c*c)
return
end
real function scorst(nd,r,tex,ncell)
real r(ncell),tex(ncell)
integer nd(ncell),ncell
sO=0.0
s1=0.0
s2=0.0
s3=0.0
s4=0.0
do i=1,ncell
   sO=sO+r(i)*tex(i)
   s1=s1+r(i)*nd(i)
   s2=s2+r(i)*r(i)*tex(i)
   s3=s3+tex(i)
   s4=s4+nd(i)
endo
dratio=s4/s3
ab=(s1-ratio*s0)**2.
ac=ratio*(s2-s0*s0/s3)
scorst=ab/ac
return
end
real function scothst(nd,th,tex,ncell)
real th(ncell),tex(ncell)
integer nd(ncell),ncell
common/s1/pi,pi2,c
s1=0.0
s2=0.0
do i=1,ncell
   s1=s1+sin(th(i))*nd(i)
   s2=s2+cos(th(i))*nd(i)
endo
h=atan(s1/s2)
if(s2.lt.0.0)then
   emu=h+pi
else
   if(s1.lt.0.0)then
      emu=h+pi2
   else
      emu=h
   endif
endif
s10=0.0
s11=0.0
s12=0.0
s13=0.0
s14=0.0
do i=1,ncell
   s10=s10+tex(i)*cos(th(i)-emu)
   s11=s11+tex(i)
   s12=s12+tex(i)*(cos(th(i)-emu)**2.)
   s13=s13+nd(i)
endo
s14 = s14 + nd(i)*cos(th(i) - emo)
endo
tratio = s13/s11
ab = (s14 - ratio*s10)**2.
ac = ratio*(s12 - s10*s10/s11)
scothst = ab/ac
return
end
real function wlik(nd,r,th,tex,ncell)
real r(ncell),th(ncell),tex(ncell)
integer nd(ncell),ncell
real x(3),a(3),b(3),g(3),work(40)
real kest
integer iworl(5),liw,lw
external functl
common/s20/idor
common/s4/kest
common/s1/pi,pi2,c
common/s2/r1,nt,xbar
common/s50/tm1,tm2,tm3,t13
common/s51/wml,wm2,w12
x(1) = kest
x(2) = 0.01
x(3) = xbar
ib = 0
do i = 1,3
a(i) = 0.0
endo
b(1) = 20.0
b(2) = 20.0
b(3) = 6.283
liw = 5
lw = 40
n2 = 3
ier = 1
idor = 0
n = 3
npars = 3
nsig = 3
nsr = 25
call zxmwd(func1,npars,nsig,a,b,nsr,x,f,work,iewl,ier)
if(ier.ne.0)then
write(6,*)'fail(3)=',ier
endif
type*, 'parms=', x(1),x(2),x(3)
tm1 = x(1)
tm2 = x(2)
tm3 = x(3)
s1 = 0
s2 = 0
s3 = 0
s4 = 0
s5 = 0
s6 = 0
do i = 1,ncell
sl = sl + nd(i)*log(tex(i))
s2 = s2 + nd(i)*cos(th(i) - x(3))
s3 = s3 + nd(i)*r(i)*cos(th(i) - x(3))
s4 = s4 + nd(i)
s5 = s5 + ex(i)*exp((x(1)+x(2)*r(i))*cos(th(i)-x(3)))
endo
s2 = x(1)*s2
s3 = x(2)*s3
bwl = sl + s2 + s3 - s4*log(s5)
t13=bw1

c now do ord von Mises

c
\begin{verbatim}
  x(1)=kxest
  x(2)=xbar
  do i=1,2
    a(i)=0.0
  enddo
  b(1)=20.0
  b(2)=6.283
  n2=2
  ifail=1
  idor=1
  npar=2
  nsig=3
  nsr=10
  call zxrnwd(funct1,npar,nsig,a,b,nsr,x,f,work,iwor1,ier)
  if (ier.ne.0) then
    type*, 'fail(2)=', ier
    type*, 'parms(2-model):', x(1), x(2)
  endif
  wml=x(1)
  wml=x(2)
  s1=0
  s2=0
  s3=0
  s4=0
  s5=0
  do i=1,ncell
    s1=s1+nd(i)*log(tex(i))
    s2=s2+nd(i)*cos(th(i)-x(2))
    s3=s3+nd(i)
    s4=s4+tex(i)*exp(x(1)*cos(th(i)-x(2)))
  enddo
  s2=x(1)*s2
  cw1=s1+s2-s3*log(s4)
  wlik=2*(bw1-cw1)
  return
end
subroutine funct1(n,xc,fc)
  real fc,gc(5000),xc(n)
  real x1,x2,x3,x4,x5
  real lam,lan
  real krest,delest
  common/s3/tex(9000),r(1000),th(1000),ncell
  common/s30/nd(9000)
  common/s20/idor
  s1=0
  s2=0
  s3=0
  s4=0
  s5=0
  s6=0
  s7=0
  s8=0
  s9=0
  s10=0
  s11=0
  if(idor.eq.0) then
    x1=xc(1)
    x2=xc(2)
    x3=xc(3)
end
\end{verbatim}

c
\begin{verbatim}
  x(1)=kxest
  x(2)=xbar
  do i=1,2
    a(i)=0.0
  enddo
  b(1)=20.0
  b(2)=6.283
  n2=2
  ifail=1
  idor=1
  npar=2
  nsig=3
  nsr=10
  call zxrnwd(funct1,npar,nsig,a,b,nsr,x,f,work,iwor1,ier)
  if (ier.ne.0) then
    type*, 'fail(2)=', ier
    type*, 'parms(2-model):', x(1), x(2)
  endif
  wml=x(1)
  wml=x(2)
  s1=0
  s2=0
  s3=0
  s4=0
  s5=0
  do i=1,ncell
    s1=s1+nd(i)*log(tex(i))
    s2=s2+nd(i)*cos(th(i)-x(2))
    s3=s3+nd(i)
    s4=s4+tex(i)*exp(x(1)*cos(th(i)-x(2)))
  enddo
  s2=x(1)*s2
  cw1=s1+s2-s3*log(s4)
  wlik=2*(bw1-cw1)
  return
end
subroutine funct1(n,xc,fc)
  real fc,gc(5000),xc(n)
  real x1,x2,x3,x4,x5
  real lam,lan
  real krest,delest
  common/s3/tex(9000),r(1000),th(1000),ncell
  common/s30/nd(9000)
  common/s20/idor
  s1=0
  s2=0
  s3=0
  s4=0
  s5=0
  s6=0
  s7=0
  s8=0
  s9=0
  s10=0
  s11=0
  if(idor.eq.0) then
    x1=xc(1)
    x2=xc(2)
    x3=xc(3)
end
\end{verbatim}

do i=1,nce11
skap=x1+x2*r(i)
cox=cos(th(i)-x3)
sinx=sin(th(i)-x3)
s1=s1+nd(i)*cox
s2=s2+nd(i)*r(i)*cox
s3=s3+nd(i)
s4=s4+exp(skap*cox)
s5=s5+nd(i)*sinx
s6=s6+nd(i)*r(i)*sinx
lest=0.0
dest=1.0
ay=lam(x1,x2,x3,lest,dest,r(i),th(i))
ay=ay*tex(i)
s7=s7+ay
s8=s8+skap*sinx*ay
s9=s9+r(i)*cox*ay
s10=s10+cox*ay
s11=s11+nd(i)*log(tex(i))
endo
a=s11+x1*s1+x2*s2-s3*log(s7)
gc(1)=-(s1-s3*s10/s7)
gc(2)=-(s2-s3*s9/s7)
gc(3)=-(x1*s5+x2*s6-s3*s8/s7)
f=-a
else

2 variable model

x1=xc(1)
x2=xc(2)
do i=1,nce11
ab=cos(th(i)-x2)
ac=sin(th(i)-x2)
lest=0.0
dest=1.0
az=lan(x1,x2,lest,dest,r(i),th(i))
s1=s1+nd(i)*ab
s2=s2+nd(i)*log(tex(i))
s3=s3+nd(i)
s4=s4+tex(i)*az
s5=s5+tex(i)*ab*az
s6=s6+tex(i)*ac*az
s7=s7+nd(i)*ac
endo
all=s2+x1*s1-s3*log(s4)
gc(1)=-(s1-s3*(s5/s4))
gc(2)=-x1*s7-xl*s3*(s6/s4)
f=-all
endif
return
end
subroutine funk5(n,x,f)
real x(n)
common/s2/r1,nt,xbar
common/s3/tex(9000),r(1000),th(1000),nce11
s1=0
s2=0
s3=0
do i=1,nce11
cox=cos(th(i)-xbar)
alamin=tex(i)*exp(x(1)*cox)
bamin=cox*alam
s1=s1+cox
s2 = s2 + alam
s3 = s3 + blam
enddo
f = 0.0
f = abs(r1/nt - s3/s2)
return
end

subroutine funct2(n, xc, fc, gc)
real f, gc(n), xc(n)
real x1, x2, x3, x4, x5
real lam, lan
real lest, delest

common/s3/tex(9000), r(1000), th(1000), ncell
common/s30/nd(9000)
common/s20/idor
s1 = 0
s2 = 0
s3 = 0
s4 = 0
s5 = 0
s6 = 0
s7 = 0
s8 = 0
s9 = 0
s10 = 0
s11 = 0

if (idor.eq.0) then
x1 = xc(1)
x2 = xc(2)
x3 = xc(3)
do i = 1, ncell
  skal = x1 + x2*r(i)
  cosx = cos(th(i) - x3)
  sinx = sin(th(i) - x3)
  s1 = s1 + nd(i)*cosx
  s2 = s2 + nd(i)*r(i)*cosx
  s3 = s3 + nd(i)
  s4 = s4 + exp(skap*cosx)
  s5 = s5 + nd(i)*sinx
  s6 = s6 + nd(i)*r(i)*sinx
  lest = 0.0
  delest = 1.0
  ay = lam(x1, x2, x3, lest, delest, r(i), th(i))
  ay = ay*tex(i)
  s7 = s7 + ay
  s8 = s8 + skal*sinx*ay
  s9 = s9 + r(i)*cosx*ay
  s10 = s10 + cosx*ay
  s11 = s11 + nd(i)*log(tex(i))
endo
da = s11 + x1*s1 + x2*s2 - s3*log(s7)
ge(1) = -(s1 - s3*s10/s7) = -(s2 - s3*s9/s7)
ge(3) = -(x1*s5 + x2*s6 - s3*s8/s7)
fc = a
else
endif

2 variable model
xl = xc(1)
x2 = xc(2)
do i = 1, ncell
ab = cos(th(i) - x2)
ac = sin(th(i) - x2)
llest=0.0
delest=1.0
az=lan(x1,x2,llest,delest,r(i),th(i))
s1=s1+nd(i)*ab
s2=s2+nd(i)*log(tex(i))
s3=s3+nd(i)
s4=s4+tex(i)*az
s5=s5+tex(i)*ab*az
s6=s6+tex(i)*ac*az
s7=s7+nd(i)*ac
enddo
all=s2+x1*s1-s3*log(s4)
gc(1)=-(s1-s3*(s5/s4))
gc(2)=-(x1*s7-x1*s3*(s6/s4))
fc=-all
endif
return
end
real function wscore(nd,r,th,tex,ncell)
real r(ncell),th(ncell)
real tex(ncell)
integer nd(ncell)
integer ncell
real ikk,imm,ipp,ikm,ill
real ipk,ipm,ilm,ilk
real imk
real lest,delest,kest
real lam,
common/sl/pi,pi2,c
common/s2/rl,nt,xbar
common/s25/idrt
common/s4/kest
s2=0.0
s3=0
s4=0
s5=0
s6=0
do i=1,ncell
thc(i)=cos(th(i))
ths(i)=sin(th(i))
enddo
do i=1,ncell
s2=s2+nd(i)*ths(i)
s3=s3+nd(i)*thc(i)
s4=s4+nd(i)*r(i)*ths(i)
s5=s5+nd(i)*r(i)*thc(i)
enddo
ab=s2+s4
ac=s3+s5
h=atan(ab/ac)
if(ac.lt.0)then
xbar1=h+pi
else
if(ab.lt.0.0)then
xbar1=h+pi2
else
xbar1=h
endif
endif
if(idrt.eq.1)then
type*,xbar and xbar1'
type*,xbar,xbar1
endif
$cz = \cos(xbar)$

$sz = \sin(xbar)$

do $i = 1, ncell$

$thc(i) = \cos(th(i) - xbar)$

$ths(i) = \sin(th(i) - xbar)$

dendo

$s9 = 0$

$s10 = 0$

$s11 = 0$

$s12 = 0$

$s13 = 0$

$s14 = 0$

$s15 = 0$

$s16 = 0$

$s17 = 0$

$s18 = 0$

$s19 = 0$

$s20 = 0$

$s21 = 0$

$lest = 0.0$

$delest = 1.0$

do $i = 1, ncell$

$abz = lan(kest, xbar, lest, delest, r(i), th(i))$

$abz = tex(i) * abz$

$s9 = s9 + nd(i)$

$s10 = s10 + abz$

$s11 = s11 + thc(i) * thc(i) * abz$

$s12 = s12 + thc(i) * abz$

$s13 = s13 + ths(i) * ths(i) * abz$

$s14 = s14 + ths(i) * abz$

$s15 = s15 + r(i) * thc(i) * abz$

$s16 = s16 + r(i) * ths(i) * thc(i) * abz$

$s17 = s17 + thc(i) * ths(i) * abz$

$s18 = s18 + r(i) * ths(i) * abz$

$s19 = s19 + r(i) * thc(i) * abz$

$s20 = s20 + r(i) * r(i) * thc(i) * thc(i) * abz$

$s21 = s21 + nd(i) * r(i) * thc(i)$

dendo

$ratio = s9 / s10$

$ipp = ratio * (s20 - (s15 * s15) / s10)$

$ipk = ratio * (s16 - (s12 * s15) / s10)$

$ikk = ratio * (s11 - (s12 * s12) / s10)$

$ipm = sz * s5 - cz * s4$

$imm = kest * (cz * s3 + sz * s2)$

$ikm = sz * s3 - cz * s2$

$imm = kest * (cz * s3 + sz * s2) - kest * s12$

$ikm = sz * s3 - cz * s2$

&

$imm = kest * (cz * s3 + sz * s2) - kest * s12$

$ikm = sz * s3 - cz * s2$

$imm = kest * (cz * s3 + sz * s2) - kest * s12$

$ikm = sz * s3 - cz * s2$

$imm = kest * (cz * s3 + sz * s2) - kest * s12$

$ikm = sz * s3 - cz * s2$

$a = ipk * imm - ipm * ikm$

$b = -ipk * imm + ipm * ikk$

$az = a * ipk + b * ipm$

$cz = ikk * imm - ikm * ikm$

$denom = ipp - az / cz$

$aws = s21 - ratio * s15$

$wscore = aws * aws / denom$

$return$

end

real function tense(ra, tha, b, a, k, ps, mu)
real ra, tha, b, a, k, ps, mu
a1 = ra ** b

$a2 = \exp(-a * ra)$

$a3 = \exp((k + ps * ra) * \cos(tha - mu))$
tense=a1*a2*a3
return
end

real function scoR(nd, r, ncex, nsex, ncell)
real r(ncell), ncex(ncell, nsex)
integer nsex, ncell, nd(ncell, nsex)
s6=0
s7=0
do i=1, ncell
s4=0
s5=0
do j=1, nsex
s0=0
s1=0
s2=0
do il=1, ncell
s0=s0+r(il)*ncex(il, j)
s1=s1+ncex(il, j)
s2=s2+r(il)*r(il)*ncex(il, j)
endo
do s4=nd(i, j)*(r(i)-s0/s1)
s5=nd(i, j)*(s2/s1-(s0/s1)**2)
endo
s6=s6+s4
s7=s7+s5
endo
scoR=s6*s6/s7
type*, 'test for r=', scoR
return
end

real function scoTH(nd, th, ncex, nsex, ncell)
real th(ncell), ncex(ncell, nsex)
integer nsex, ncell, nd(ncell, nsex)
s3=0
s4=0
do i=1, ncell
s1=0
s2=0
do j=1, nsex
s1=s1+sin(th(i))*nd(i, j)
s2=s2+cos(th(i))*nd(i, j)
endo
s3=s3+s1
s4=s4+s2
endo
emu=atan(s3/s4)
s6=0
s7=0
do i=1, ncell
s4=0
s5=0
do j=1, nsex
s0=0
s1=0
do il=1, ncell
s0=ncex(il, j)*cos(th(il)-emu)
s1=ncex(il, j)
s2=cos(th(il)-emu)**2)*ncex(il, j)
endo
s4=nd(i, j)*(cos(th(i)-emu)-s0/s1)
s5=nd(i, j)*(s2/s1-(s0/s1)**2)
endo
s6=s6+s4
s7 = s7 + s5
enddo
scoTH = s6 * s6 / s7
write *, 'mean angle:', emu
write *, 'test for conc:', scoTH
return
end
M M CCCC TTTTT AAA L  
MM MM C T A A L  
M M M C T A A L  
M M C T A A L  
M M C T AAAA L  
M M C T A A L  
M M CCCC T A A LLLLL  

SSSS III M M BBBB AAA TTTTT CCCC H H  
S I MM MM B B A A T C H H  
S I MM MM B B A A T C H H  
SSS I M M BBBB A A T C HHHHH  
S I M M B B AAAA A T C H H  
S I M M B B A A T C H H  
SSSS III M M BBBB A A T CCCC H H  

FFFFF OOO RRRR ;; 15555  
F 0 0 R R ;; 11 5  
F 0 0 R R 1555  
FFFFF O 0 RRRR ;; 15 5  
F 0 0 R R ;; 15 5  
.. F 0 0 R R ; 15 5  
.. F 000 R R ; 111 555  

File _$255$DUA2: [MC.T.AL]SIMBATCH.FOR;15 (1111,8,0), last revised on 26-JUN-1990 09:38, is a 49 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 61 bytes.  
Job SIMBATCH (617) queued to SYS$PRINT on 24-AUG-1990 13:03 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:14 from queue LASER.
double precision dseed
real pr(1000,18),pm(9000),wk(9000),wk2(9000)
real wkvec(50000),rmvec(1,50000),rmv(50000)
real var(10000)
real prf(9000),rate(20)
real tpop(1000)
real mrate(9),frate(9),x(9000),y(9000)
real reg(9000),hed(20)
real bp,ap,kp,ip,mup
real probl(6)
real regcon(1000),regex(1000)
real pmr(9000),pfr(9000)
real stat(15),chialfl(7),zedalf(7)
real corralf(7)
real stonalf(7)
real crit(15,20)
real nce(1000,2)
real lincor,wscore,scor,scoth,wlik,umard
real scorst,scothst
real prob(5),montc(10,2000).
real kest
real wor(12),xk(1),a(1),b(1)
integer iwor(1)
integer num(9),nuf(9),ia(9000),nm(9000),nf(9000)
integer tp(1000)
integer irl(1000),ir2(1000),ib(9000)
integer ncon(1000,20)
integer nb(9000),np(9000)
integer nde(1000,2)
integer ncountl(15,5)
real ncex(1000,20)
real texb(1000),texp(1000)
integer mpop,fpop
character*20 fname,wname,tname,zname
external funk
external funk5
common/s1/pi,pi2,c
common/s2/r1,nt,xbar
common/s3/tex(9000),r(1000),th(1000),ncell
common/s4/kest
common/s10/sigma2,alpha
common/s20/idor
common/s25/idrt
common/s30/nd(9000)
common/s50/tml,tm2,tm3,t13
common/s51/wml,wm2,w12
data(stonalf(i),i=1,3)/0.0,0.0,0.0/
data(corralf(i),i=1,3)/0.0,0.0,0.0/
data(chialfl(i),i=1,3)/3.841,6.635,10.827/
data(zedalf(i),i=1,3)/1.645,2.326,3.090/
do i=1,3
    crit(1,i)=chialfl(i)
crit(2,i)=chialfl(i)
crit(3,i)=corralf(i)
crit(4,i)=chialfl(i)
crit(5,i)=chialfl(i)
crit(6,i)=stonalf(i)
endo
dpi=3.14159
dpi2=6.283185
c=0.017453
read(S,*)idrt
if(idrt.eq.0)then
    read(S,*)bp,ap,kp,ip,mup
read(5,*)mpop,fpop
read(5,*)nr
read(5,*)nth
ncell=nr*nth
read(5,*)nopt
if(nopt.eq.1)then
  read(S,100)fname
  format(a20)
  open(15,file=fname,status='old')
  do i=1,9
    read(15,*)num(i)
  enddo
  do i=1,9
    read(15,*)nuf(i)
  enddo
  close(15)
  do i=1,ncell
    do j=1,9
      ncon(i,j)=num(j)
    enddo
    do j=10,18
      il=j-9
      ncon(i,j)=nuf(il)
    enddo
  enddo
else
  read(S,100)fname
  open(15,file=fname,status='old')
  do i=1,ncell
    read(15,*)(ncon(i,j),j=1,18)
  enddo
  close(15)
endif
read(S,100)wname
open(16,file=wname,status='old')
read(16,*)rate(i)
rate(i)=rate(i)/1000.
close(16)
read(S,100)tname
read(S,100)zname
open(18,file=zname,status='new')
c**** now calculate expected values

do i=1,ncell
do j=1,18
  ncex(i,j)=ncon(i,j)*rate(j)
endo
s1=0
s2=0
do j=1,9
  s1=s1+ncex(i,j)
  ji=j+9
  s2=s2+ncex(i,ji)
endo
nce(i,1)=s1
nce(i,2)=s2
endo
c**** calculate r and th of cells
c should enter rmax here, but i have set it to 1
rmax=1
k=0
do i=1,nr
a5=(float(i)/float(nr))*rmax
do j=1,nth
k=k+1
b5=(float(j)/float(nth))*6.283
r(k)=a5
th(k)=b5
enddo
doo
read(5,*)ibayes
if(ibayes.eq.1)then
read(5,*)alpha,sigma2
k2=0
do i=1,ncell
do j=1,i
k2=k2+1
x1=r(i)*cos(th(i))
y1=r(i)*sin(th(i))
x2=r(j)*cos(th(j))
y2=r(j)*sin(th(j))
vara(k2)=cov(x1,y1,x2,y2)
enddo
doo
dseed=8429109.d0
nr=1
ir=1
wkvec(1)=0.0
call ggnsm(dseed,nr,ncell,vara,ir,rvec,wkvec,ier)
if(ier.ne.0)then
endif
doo
read(5,*)nsim
type*, 'now enter l'
read(5,*)dert

do calculation of norm constant
do s1=0
do i=1,ncell
if(ibayes.ne.1)then
rmv(i)=tense(r(i),th(i),bp,ap,kp,ip,mup)
else
a23=log(tense(r(i),th(i),bp,ap,kp,ip,mup))
b23=a23+rvec(1,i)
rmv(i)=exp(b23)
endif
do s0=0
do j=1,18
s0=s0+ncex(i,j)*rmv(i)
enddo
s1=s1+s0
doo
calc prob for each region
s15=0
s16=0
kl=0
k2=0
pm(0)=0.0
pf(0)=0.0
do i=1,ncell
  do j=1,9
    pr(i,j)=ncex(i,j)*rmv(i)/cnorm
    kl=kl+1
    pm(kl)=pr(i,j)
    s15=s15+pm(kl)
  enddo
  do j=10,18
    pr(i,j)=ncex(i,j)*rmv(i)/cnorm
    k2=k2+1
    pf(k2)=pr(i,j)
    s16=s16+pf(k2)
  enddo
enddo
dseed=123457.d0
ia(l)=-1
ib(l)=-1

* iteration loop starts here

ntest=6
write(18,*)nsim,ntest
do kl=1,nsim
  type*, 'iteration: ', kl
  type*, 'prob total: male: ', s15, 'female: ', s16
enddo

do generation now

nrl=mop
nr2=fpop
ndmp=ncell*9
ndfp=ndmp
ncell9=ncell*9
do i=1,ncell9
  pml(i)=pm(i)/s15
  pf1(i)=pf(i)/s16
enddo
call ggda(dseed, nrl, ndmp, pml, ia, wk1, ir1)
call ggda(dseed, nr2, ndfp, pf1, ib, wk2, ir2)
type*, 'male pops'
s1=0
s2=0
do i=1,nrl
  if(ir1(i).ne.0)then
    s1=s1+1
  endif
enddo
type*, 'female pops'
do i=1,nr2
  if(ir2(i).ne.0)then
    s2=s2+1
  endif
enddo
type*, 'total male=', s1, 'total female=', s2

* binning data

do i=1,ndmp
  nm(i)=0
enddo
do i=1,ndfp
nf(i)=0
endif
do i=1,nrl
in=irl(i)
nm(in)=nm(in)+1
endif
do i=1,nr2
in=ir2(i)
nf(in)=nf(in)+1
endif

*** do statistics here

k=0
s5=0
do i=1,ncell
s3=0
s4=0
do j=1,9
k=k+1
s3=s3+nm(k)
s4=s4+nf(k)
enddo
nde(i,1)=s3
nde(i,2)=s4
tp(i)=s3+s4
s5=s5+tp(i)
tex(i)=nce(i,1)+nce(i,2)
nd(i)=tp(i)
enddo
type*, 'total number=', s5

sl=0
s2=0
s3=0
s4=0
s5=0
s6=0
do i=1,ncell
co=cos(th(i))
si=sin(th(i))
s1=s1+nd(i)*co
s2=s2+nd(i)*si
s3=s3+nd(i)
enddo
rl=sqrt(s1*s1+s2*s2)
h=atan(s2/s1)
if(s1.lt.0.0)then
  xbar=h+pi1
else
  if(s2.lt.0.0)then
    xbar=h+pi2
  else
    xbar=h
  endif
endif
al=0.0
bl=60.0
tol=0.1
nt=s3
npar=1
nsig=3
a(1)=0.0
b(1)=70.0
nsr=20
call zxmwd(funk5,npar,nsig,a,b,nsr,xk,f,wor,iwor,ier)
kest=xk(1)
if(ier.ne.0)then
type*, 'error(ier)=' ,ier
endif
type*, 'kest=' ,kest
hed(1)=scorst(tp,r,tex,ncell)
hed(2)=scothst(tp,th,tex,ncell)
acor=lincor(tp,r,th,tex,ncell)
type*, 'acor=' ,acor
hed(3)=acor
hed(4)=wlik(tp,r,th,tex,ncell)
kest=wml
xbar=wm2
hed(5)=wscore(tp,r,th,tex,ncell)
type*, 'wml estimates'
type*, '3-model: k,psi,mu,lik'
type*, '2-model: k,mu, lik'
type*, 'wml,wm2,w12'
c mardia test not used here
c stone's test instead
hed(6)=stone(r,tp,tex,ncell)
type*, 'tests(r,th,corr,lik,score,stone)'
type*(,hed(j),j=1,6)
do j=1,ntest
do k=1,3
if(hed(j).ge.crit(j,k))then
ncountl(j,k)=ncountl(j,k)+1
endif
endo
montc(j,kl)=hed(j)
endo
write(18,*)(montc(j,kl),j=1,ntest)
c end of simulation loop here
do open(17,file=tname,status='new')
write(17,*) 'input parameters'
do j=1,ntest
write(17,*) 'count for one-tail test:' ,j
write(17,*)(ncountl(j,k),k=1,4)
do k=1,4
probl(k)=float(ncountl(j,k))/float(nsim)
dendo
write(17,*)'one-tail test prob',j
write(17,*)(probl(k),k=1,4)
c do easygraph graphics here
do enddo
c close(17)
type*, 'end of file writing'
statistics are here

else
    type*, 'enter age*sex rates file:'
    read(5,100)fname
    open(15,file=fname,status='old')
    do j=1,9
        read(15,*)mrate(j),frate(j)
    enddo
    close(15)
    type*, 'enter region file'
    read(5,100)tname
    type*, 'x-y(1) or r-th(2) file'
    read(5,*)ifile
    type*, 'one(1) or two(2) diseases on file:'
    read(5,*)iont
    type*, 'enter x,y coords of centre'
    read(5,*)xcen,ycen
    type*, 'which disease: bronch(1) pneum(2)'
    read(5,*)idis
    open(25,file=tname,status='old')
    read(25,*)ncell
    rmax=0
    do i=1,ncell
        s30=0
        if(iont.eq.1) then
            read(25,*)reg(i),x(i),y(i),nb(i),
                (num(j),nuf(j),j=1,9)
            np(i)=nb(i)
        else
            read(25,*)reg(i),x(i),y(i),nb(i),np(i),
                (num(j),nuf(j),j=1,9)
        endif
        do j=1,9
            s30=s30+num(j)+nuf(j)
        enddo
        tpop(i)=s30
        if(idis.eq.1) then
            nd(i)=nb(i)
        else
            nd(i)=np(i)
        endif
        if(ifile.eq.2) then
            r(i)=x(i)
            th(i)=y(i)
            if(r(i).gt.rmax) then
                rmax=r(i)
            endif
        endif
    enddo
    sl=0.0
    do j=1,9
        nce(i,j)=num(j)*mrate(j)/1000.
        sl=sl+nce(i,j)
    enddo
    s2=0.0
    do j=10,18
        jm=j-9
        nce(i,j)=nuf(jm)*frate(jm)/1000.
        s2=s2+nce(i,j)
    enddo
if(idis.eq.1)then
texb(i)=s1+s2
else
texp(i)=s1+s2
endif
tex(i)=s1+s2
if(ifile.ne.2)then
if(ycen.lt.20.or.ycen.gt.980)then
if(y(i).gt.800)then
  y(i)=1000.-y(i)
endif
endif
xa=x(i)-xcen
ya=y(i)-ycen
r(i)=sqrt(xa*xa+ya*ya)
if(r(i).gt.rmax)then
  rmax=r(i)
endif
if(xa.eq.0.0)then
  if(ya.lt.0.0)then
    th(i)=4.7124
  else
    th(i)=1.5708
  endif
  goto 2345
endif
h=atan(ya/xa)
if(xa.lt.0)then
  th(i)=h+pi
else
  if(ya.lt.0)then
    th(i)=h+pi2
  else
    th(i)=h
  endif
endif
endif
continue
enddo
type*, 'do you want to write data to file(l:yes),'
read(5,*)iyes
if(iyes.eq.1)then
type*, 'filename:'
read(5,100)zname
open(30,file=zname,status='new')
write(30,*)ncell
do i=1,ncell
  if(idis.eq.1)then
    write(30,*)i,r(i),th(i),nb(i),tpop(i),texb(i)
  else
    write(30,*)i,r(i),th(i),np(i),tpop(i),texp(i)
  endif
enddo
close(30)
endif
do i=1,ncell
  r(i)=r(i)*2.5/rmax
enddo
c  basic calculations
c  ...
  s1=0
  s2=0
  s3=0
\begin{verbatim}
  s4=0
  s5=0
  s6=0
  do i=1,ncell
    co=cos(th(i))
    si=sin(th(i))
    s1=s1+nd(i)*co
    s2=s2+nd(i)*si
    s3=s3+nd(i)
  enddo
  rl=sqrt(s1*s1+s2*s2)
  type*, 'stats:nc,ns,nt,rl'
  type*, s1,s2,s3,rl
  h=atan(s2/s1)
  if(s1.lt.0.0) then
    xbar=h+pi
  else
    if(s2.lt.0.0) then
      xbar=h+pi2
    else
      xbar=h
    endif
  endif
  type*, 'simple mean angle:',xbar
  a1=0.0
  b1=60.0
  tol=0.1
  nt=s3
  npar=1
  nsig=3
  a(1)=0.0
  b(1)=70.0
  nsr=20
  call zxmd(funk5,npar,nsig,a,b,nsr,xk,f,wor,iwor,ier)
  call zxgsn(funk,al,bl,tol,kest,ier)
  kest=xk(1)
  if(ier.ne.0) then
    type*, 'error(ier)=',ier
  endif
  type*, 'initial k estimate:',kest
  acor=lincor(nd,r,th,tex,ncell)
  type*, 'acor:',acor
  ston=stone(r,nd,tex,ncell)
  type*, 'stone test:',ston
  hed(6)=scorst(nd,r,tex,ncell)
  type*, 'result: r test=',hed(6)
  hed(7)=scothst(nd,th,tex,ncell)
  type*, 'result: th test=',hed(7)
  hed(9)=wlik(nd,r,th,tex,ncell)
  kest=wml
  xbar=wm2
  type*, 'do you want own parm estimates(l:yes),'
  read(5,*)iden
  if(iden.eq.l) then
    type*, 'kappa: '
    read(5,*)kest
    type*, 'pmu: '
    read(5,*)xbar
  endif
  hed(8)=wscore(nd,r,th,tex,ncell)
  if(hed(9).le.0.0) then
\end{verbatim}
hlik=0.0
else
  hlik=sqrt(hed(9))
endif
if(hed(8).le.0.0)then
  hscor=0.0
else
  hscor=sqrt(hed(8))
endif
type*, 'scor and lik as N(0,1) variates'
type*, hscor, hlik
type*, 'results: r, th, int tests'
type*, hed(6), hed(7), hed(8), hed(9)
type*, 'ml estimates of k, psi, mu and lik'
type*, tml, tm2, tm3, tl3
type*, 'ml estimates of k, mu and lik2'
type*, wml, wm2, w12
endif
c end of stats section

stop
end
c*** functions and subs

c
real function stone(r,ncob,tex,n)
real r(n), tex(n)
real ala(500)
integer ncob(n), n
integer ira(500)
l=n
do i=l, n
  ira(i)=i
endo
call vsrtr(r, l, ira)
s0=0
sl=0
tmax=0
do i=l, n
  s0=s0+ncob(ira(i))
  sl=sl+tex(ira(i))
  tm=s0/sl
if(tm.gt.tmax)then
  tmax=tm
  ntn=i
endif
endo
stone=tmax
return
end
real function cov(a,b,c,d)
real a,b,c,d
common/s10/sigma2, alpha
sqd=((a-c)**2.)+((b-d)**2.)
dist=sqrt(sqd)
cov=sigma2*exp(-alpha*dist)
return
end
real function trend(r,th,b1,b2,b3,b4,b5)
ar=r
alr=log(r)
ass=(b3+b4*ar)*cos(th-b5)
\[
\text{trend} = -b1 \cdot a + b2 \cdot a + r + s + \text{ass}
\]

return
end

real function umard(tp, r, th, tex, ncell)
real r(ncell), th(ncell), tex(ncell)
integer tp(ncell), ncell
return
end

real function funk(x)
real x
common/s2/rl, nt, xbar
common/s3/tex(9000), r(1000), th(1000), ncell
s1 = 0
s2 = 0
s3 = 0
do i = 1, ncell
cox = cos(th(i) - xbar)
alam = tex(i) * exp(x * cox)
blam = cox * alam
s1 = s1 + cox
s2 = s2 + alam
s3 = s3 + blam
enddo
funk = abs(rl/nt - s3/s2)
type*, 'kappa, funk'
type*, x, funk
return
end

real function lam(pk, ps, pm, pl, pd, r, th)
real pk, ps, pm, pl, pd, r, th
common/s1/pi, pi2, c
ab = exp((pk+ps*r)*cos(th-pm))
if(pl.eq.0.0)
  lam = ab
else
  ac = pl*pd*r**(pd-1.)*exp(-pl*r**pd)
  lam = ab*ac
endif
return
end

real function lan(pk, pm, pl, pd, r, th)
real pk, pm, pl, pd, r, th
common/s1/pi, pi2, c
ab = exp(pk*cos(th-pm))
if(pl.eq.0.0)
  lan = ab
else
  ac = pl*pd*r**(pd-1.)*exp(-pl*r**pd)
  lan = ab*ac
endif
return
end

real function corr(nd, x, y, ncell)
real x(ncell), y(ncell)
ingest nd(ncell)
ingest ncell
s1 = 0
s2 = 0
s3 = 0
s4 = 0
s5 = 0
s6 = 0
do i = 1, ncell
s1 = s1 + nd(i) * x(i)
\[ s_2 = s_2 + n_d(i) \cdot x(i) \cdot x(i) \]
\[ s_3 = s_3 + n_d(i) \cdot y(i) \]
\[ s_4 = s_4 + n_d(i) \cdot y(i) \cdot y(i) \]
\[ s_5 = s_5 + n_d(i) \cdot y(i) \cdot x(i) \]
\[ s_6 = s_6 + n_d(i) \]
\[ \text{enddo} \]
\[ a = s_5 - s_3 \cdot s_1 / s_6 \]
\[ b = s_2 - s_1 \cdot s_3 / s_6 \]
\[ c = s_4 - s_3 \cdot s_3 / s_6 \]
\[ bc = b \cdot c \]
\[ \text{if}(bc < 0.) \text{then} \]
\[ bc = 0.0001 \]
\[ \text{endif} \]
\[ \text{corr} = a / \sqrt{bc} \]
\[ \text{return} \]
\[ \text{end} \]

\text{real function lincor(nd, r, th, tex, ncell)}
\text{real r(ncell), th(ncell), tex(ncell)}
\text{real smr(1000)}
\text{integer nd(ncell)}
\text{integer ncell}
\text{real thc(2000), ths(2000)}
\text{real corr}
\text{common/s1/pi, pi2, c}
\text{s1 = 0}
\text{s2 = 0}
\text{do i = 1, ncell}
\text{thc(i) = cos(th(i))}
\text{ths(i) = sin(th(i))}
\text{s1 = s1 + nd(i) \cdot thc(i)}
\text{s2 = s2 + nd(i) \cdot ths(i)}
\text{enddo}
\text{cm = atan(s2/s1)}
\text{if(s1.1e.0) then}
\text{pmu = cm + pi}
\text{else}
\text{if(s2.1e.0) then}
\text{pmu = pi2 + cm}
\text{else}
\text{pmu = cm}
\text{endif}
\text{endif}
\text{do i = 1, ncell}
\text{thc(i) = cos(th(i) - pmu)}
\text{ths(i) = sin(th(i) - pmu)}
\text{if(tex(i).eq.0.0) then}
\text{smr(i) = 0.0}
\text{else}
\text{smr(i) = nd(i) / tex(i)}
\text{endif}
\text{enddo}
\text{a = corr(smr, r, thc, ncell)}
\text{b = corr(smr, r, ths, ncell)}
\text{c = corr(smr, thc, ths, ncell)}
\text{lincor = (a*a + b*b - 2*a*b*c) / (1. - c*c)}
\text{return}
\text{end}

\text{real function scorst(nd, r, tex, ncell)}
\text{real r(ncell), tex(ncell)}
\text{integer nd(ncell), ncell}
\text{s0 = 0.0}
\text{s1 = 0.0}
\text{s2 = 0.0}
s3=0.0
s4=0.0

do i=1,ncell
  s0=s0+r(i)*tex(i)
  s1=s1+r(i)*nd(i)
  s2=s2+r(i)*r(i)*tex(i)
  s3=s3+tex(i)
  s4=s4+nd(i)
enddo
ratio=s4/s3
ab=(sl-ratio*s0)**2.
ac=ratio*(s2-s0*s0/s3)
scothst=ab/ac
return
end

real function scohst(nd,th,tex,ncell)
real th,nd,ncell
integer nd,ncell
common/sl/pi,pi2,c
sl=0.0
s2=0.0

do i=1,ncell
  sl=sl+sin(th(i)*nd(i)
  s2=s2+cos(th(i)*nd(i)
enddo
h=atan(sl/s2)
if(s2.1t.0.0)then
  emu=h+pi
else
  if(sl.lt.O.O)then
    emu=h+pi2
  else
    emu=h
  endif
endif
s10=0.0
s11=0.0
s12=0.0
s13=0.0
s14=0.0

do i=1,ncell
  s10=s10+tex(i)*cos(th(i)-emu)
  s11=s11+tex(i)
  s12=s12+tex(i)*cos(th(i)-emu)**2.
  s13=s13+nd(i)
  s14=s14+nd(i)*cos(th(i)-emu)
enddo
ratio=s13/s11
ab=(s14-ratio*s10)**2.
ac=ratio*(s12-s10*s10/s11)
scothst=ab/ac
return
end

real function wlik(nd,r,th,tex,ncell)
real r,th,tex,ncell
integer nd,ncell
real x(3),a(3),b(3),g(3),work(40)
real kest
integer iworl(5),liw,lw
external funct1
common/s20/idor
common/s4/kest
common/s1/pi,pi2,c
common/s2/rl,nt,xbar
common/s50/tm1,tm2,tm3,t13
common/s51/wm1,wm2,wm2
x(1)=kest
x(2)=0.01
x(3)=xbar
ib=0
do i=1,3
  a(i)=0.0
enddo
b(1)=20.0
b(2)=20.0
b(3)=6.283
liw=5
lw=40
n2=3
ier=1
idor=0
n=3
npar=3
nsig=3
nsr=25
call e04kae(n2,ib,a,b,x,f,g,iworl,liw,work,lw,ier)
call zxmwd(functl,npar,nsig,a,b,nsr,x,f,work,iworl,ier)
if(ier.ne.0)then
  write(6,*)'fail(3)=' ,ier
endif
  type*, 'parms=' ,x(1),x(2),x(3)
  tml=x(1)
  tm2=x(2)
  tm3=x(3)
  s1=0
  s2=0
  s3=0
  s4=0
  s5=0
  s6=0
do i=1,ncell
  s1=s1+nd(i)*log(tex(i))
  s2=s2+nd(i)*cos(th(i)-x(3))
  s3=s3+nd(i)*r(i)*cos(th(i)-x(3))
  s4=s4+nd(i)
  s5=s5+tex(i)*exp«x(1)+x(2)*r(i»*cos(th(i)-x(3»)
enddo
s2=x(1)*s2
s3=x(2)*s3
bwl=s1+s2+s3-s4*log(s5)
t13=bwl
c
  now do ord von Mises
  x(1)=kest
  x(2)=xbar
  do i=1,2
    a(i)=0.0
  enddo
  b(1)=20.0
  b(2)=6.283
  n2=2
  ifail=1
  idor=1
  npar=2
  nsig=3
  nsr=10
call e04kae(n2,ib,a,b,x,f,g,iworl,liw,work,lw,ifail)
call zxmwd(functl,npar,nsig,a,b,nsr,x,f,work,iworl,ier)
if(ier.ne.0)then
    type*, 'fail(2)=' ,ier
    type*, 'parms(2-model):' ,x(1),x(2)
endif
wml=x(1)
wml2=x(2)
s1=0
s2=0
s3=0
s4=0
s5=0
do i=1,ncell
   sl=sl+nd(i)*log(tex(i))
   s2=s2+nd(i)*cos(th(i)-x(2))
   s3=s3+nd(i)
   s4=s4+tex(1)*exp(x(1)*cos(th(i)-x(2)))
enddo
s2=x(1)*s2
cll=sl+s2-s3*log(s4)
wl2=cll
wlk=2*(bwl-cll)
return
end
subroutine functl(n,xc,fc)
real fc,gc(5000),xc(n)
real xl,x2,x3,x4,x5
real lam,lan
real lest,delest
common/s3/tex(9000),r(1000),th(1000),ncell
common/s30/nd(9000)
common/s20/idor
sl=0
s2=0
s3=0
s4=0
s5=0
s6=0
s7=0
s8=0
s9=0
s10=0
s11=0
if(idor.eq.0)then
   xl=xc(l)
x2=xc(2)
x3=xc(3)
do i=1,ncell
   skap=x1+x2*r(i)
   cosx=cos(th(i)-x3)
   sinx=sin(th(i)-x3)
   sl=sl+nd(i)*cosx
   s2=s2+nd(i)*r(i)*cosx
   s3=s3+nd(i)
   s4=s4+exp(skap*cosx)
   s5=s5+nd(i)*sinx
   s6=s6+nd(i)*r(i)*sinx
   lest=0.0
   lest=lest+1.0
   ay=lam(x1,x2,x3,lest,delest,r(i),th(i))
   ay=ay*txtex(i)
   s7=s7+ay
   s8=s8+skap*sinx*ay
   s9=s9+r(i)*cosx*ay
\begin{verbatim}
s10 = s10 + \cos x \cdot ay
s11 = s11 + nd(i) \cdot \log(\text{tex}(i))
enddo
a = s11 + x1 \cdot s1 + x2 \cdot s2 - s3 \cdot \log(s7)
gc(1) = -(s1 - s3 \cdot s10 / s7)
gc(2) = -(s2 - s3 \cdot s9 / s7)
gc(3) = -(x1 \cdot s5 + x2 \cdot s6 - s3 \cdot s8 / s7)
f c = -a
else
else
 2 variable model
  \text{xc}(1)
  \text{xc}(2)
  do i = 1, ncell
  ab = \cos(\text{th}(i) - x2)
  ac = \sin(\text{th}(i) - x2)
  lest = 0.0
  delest = 1.0
  az = \text{lan}(x1, x2, lest, delest, r(i), \text{th}(i))
  sl = sl + nd(i) \cdot ab
  s2 = s2 + nd(i) \cdot \log(\text{tex}(i))
  s3 = s3 + nd(i)
  s4 = s4 + \text{tex}(i) \cdot az
  s5 = s5 + \text{tex}(i) \cdot ab \cdot az
  s6 = s6 + \text{tex}(i) \cdot ac \cdot az
  s7 = s7 + nd(i) \cdot ac
endo
all = s2 + x1 \cdot sl - s3 \cdot \log(s4)
gc(1) = -(sl - s3 \cdot (sS / s4))
gc(2) = -(x1 \cdot s7 - x1 \cdot s3 \cdot (s6 / s4))
f c = -all
endif
return
end
subroutine funk5(n, x, f)
real x(n)
common/s2/r1, nt, xbar
common/s3/tex(9000), r(1000), th(1000), ncell
sl = 0
s2 = 0
s3 = 0
do i = 1, ncell
  cox = \cos(\text{th}(i) - xbar)
  alam = \text{tex}(i) \cdot \text{exp}(x(1) \cdot cox)
  blam = cox \cdot alam
  sl = sl + cox
  s2 = s2 + alam
  s3 = s3 + blam
endo
f = 0.0
f = abs(r1/nt - s3/s2)
return
end
subroutine funct2(n, xc, fc, gc)
real fc, gc(n), xc(n)
real x1, x2, x3, x4, x5
real lam, lan
real lest, delest
common/s3/tex(9000), r(1000), th(1000), ncell
common/s30/nd(9000)
common/s20/idor
s1 = 0
s2 = 0
\end{verbatim}
s3=0
s4=0
s5=0
s6=0
s7=0
s8=0
s9=0
s10=0
s11=0
if(idor.eq.0)then
x1=xc(1)
x2=xc(2)
x3=xc(3)
do i=1,ncell
  skap=x1+x2*r(i)
  cosx=cos(th(i)-x3)
  sinx=sin(th(i)-x3)
  s1=s1+nd(i)*cosx
  s2=s2+nd(i)*r(i)*cosx
  s3=s3+nd(i)
  s4=s4+exp(skap*cosx)
  s5=s5+nd(i)*sinx
  s6=s6+nd(i)*r(i)*sinx
  lest=0.0
delest=1.0
  ay=lam(x1,x2,x3,lest,delest,r(i),th(i))
  ay=ay*tex(i)
  s7=s7+ay
  s8=s8+skap*sinx*ay
  s9=s9+r(i)*cosx*ay
  s10=s10+cosx*ay
  s11=s11+nd(i)*log(tex(i))
endo
da=s11+x1*s1+x2*s2-s3*log(s7)
gc(1)=-(s1-s3*s10/s7)
gc(2)=-(s2-s3*s9/s7)
gc(3)=-(x1*s5+x2*s6-s3*s8/s7)
f=-a
else
endif
return

c
2 variable model
c
x1=xc(1)
x2=xc(2)
do i=1,ncell
  ab=cos(th(i)-x2)
  ac=sin(th(i)-x2)
  lest=0.0
delest=1.0
  az=lan(x1,x2,lest,delest,r(i),th(i))
  s1=s1+nd(i)*ab
  s2=s2+nd(i)*log(tex(i))
  s3=s3+nd(i)
  s4=s4+tex(i)*az
  s5=s5+tex(i)*ab*az
  s6=s6+tex(i)*ac*az
  s7=s7+nd(i)*ac
endo
da=s2+x1*s1-s3*log(s4)
gc(1)=-(s1-s3*(s5/s4))
gc(2)=-(x1*s7-x1*s3*(s6/s4))
f=-a
endif
return
end
real function wscore(nd,r,th,tex,ncell)
real r(ncell),th(ncell)
real tex(ncell)
integer nd(ncell)
integer ncell
real ikk,imm,ipp,ikm,ill
real ipk,ipm,ilm,ilk
real imk
real lest,delest,kest
real lam,lan
common/s1/pi,pi2,c
common/s2/r1,nt,xbar
common/s25/idrt
common/s4/kest
s2=0.0
s3=0
s4=0
s5=0
s6=0
do i=1,ncell
thc(i)=cos(th(i))
ths(i)=sin(th(i))
enddo
do i=1,ncell
s2=s2+nd(i)*ths(i)
s3=s3+nd(i)*thc(i)
s4=s4+nd(i)*r(i)*ths(i)
s5=s5+nd(i)*r(i)*thc(i)
enddo
ab=s2+s4
ac=s3+s5
h=atan(ab/ac)
if(ac.lt.0)then
xbar1=h+pi
else
if(ab.lt.0.0)then
xbar1=h+pi2
else
xbar1=h
endif
endif
if(idrt.eq.1)then
type*, 'xbar and xbarl'
type*,xbar,xbar1
endif
cz=cos(xbar)
sz=sin(xbar)
do i=1,ncell
thc(i)=cos(th(i)-xbar)
ths(i)=sin(th(i)-xbar)
enddo
s9=0
s10=0
s11=0
s12=0
s13=0
s14=0
s15=0
s16=0
s17=0
s18=0
s19=0
s20=0
s21=0
s22=0
lest=0.0
delest=1.0
do i=1,ncell
  abz=lan(kest,xbar,lest,delest,r(i),th(i))
  abz=tex(i)*abz
  s9=s9+nd(i)
  s10=s10+abz
  s11=s11+thc(i)*thc(i)*abz
  s12=s12+thc(i)*abz
  s13=s13+ths(i)*ths(i)*abz
  s14=s14+ths(i)*abz
  s15=s15+r(i)*thc(i)*abz
  s16=s16+r(i)*ths(i)*abz
  s17=s17+thc(i)*abz
  s18=s18+r(i)*ths(i)*abz
  s19=s19+r(i)*thc(i)*abz
  s20=s20+r(i)*ths(i)*abz
  s21=s21+nd(i)*r(i)*thc(i)
enddo
ratio=s9/s10
ipp=ratio*(s20-(sl5*sl5)/sl0)
ipk=ratio*(s16-(s12*sl5)/sl0)
ikk=ratio*(sll-(s12*s12)/sl0)
ipm=sz*sS-cz*s4
  & +ratio*(s18+kest*(s19-(s14*sl5)/sl0))
imm=kest*(cz*s3+sz*s2)
  & +ratio*(kest*kest*(s13-(s14*s14)/sl0)-kest*s12)
ikm=sz*s3-cz*s2
  & +ratio*(s14+kest*(s17-(s14*s12)/sl0))
imk=ikm
  a=ipk*imm-ipm*ikm
  b=-ipk*imk+ipm*ikk
  az=a*ipk+b*ipm
  cz=ikk*imm-ikm*ikm
  denom=ipp-az/cz
  aws=s21-ratio*sl5
wscore=aws*aws/denom
return
end
real function tense(ra,tha,b,a,k,ps,mu)
real ra,tha,b,a,k,ps,mu
al=ra**b
a2=exp(-a*ra)
a3=exp((k+ps*ra)*cos(tha-mu))
tense=al*a2*a3
return
end

subroutine grz(r,th,nt,n)
real r(n),th(n),x(500),y(500)
integer nt(n),n
rscreen=2.5
rm=0.0
do i=1,n
  if(r(i).ge.rm)then
    rm=r(i)
  endif
enddo
do i=1,n
  r(i)=r(i)*(rscreen/rm)
  x(i)=r(i)*cos(th(i))
\begin{verbatim}
y(i)=r(i)*sin(th(i))
enddo

5678  \texttt{type*, 'screen(0) or plotter(1)'}
read(5,*) idet
if(idet.eq.0) then
  call gino
call t4010
call units(25.0)
else
  call gino
call call044
call scale(20.0)
endif
  call chapos(20.0,20.0)
call piccle
rsn=rscreen+0.3
call shift2(rsn,rsn)
call movto2(-rsn,0.0)
call arcto2(0.0,0.0,-rsn,-0.001,0)
call movto2(0.0,0.0)
call symbol(5)
do i=1,n
  ax=x(i)-0.2
  ay=y(i)-0.1
call movto2(ax,ay)
call chaint(nt(i),3)
endo
type*, 'cntrl-sht f6 1'
read(5,*) neft
call piccle
call devend
call ginend
type*, 'more graphics(1:yes)'
read(5,*) inore
if(inore.eq.1) then
go to 5678
endif
return
end

real function scoR(nd,r,ncex,nsex,ncell)
  real r(ncell),ncex(ncell,nsex)
  integer nsex,ncell,nd(ncell,nsex)
s6=0
s7=0
do i=1,ncell
  s4=0
  s5=0
do j=1,nsex
    s0=0
    s1=0
    s2=0
do il=1,ncell
      s0=s0+r(il)*ncex(il,j)
      s1=s1+ncex(il,j)
      s2=s2+r(il)*r(il)*ncex(il,j)
endo
  s4=s4+nd(i,j)*(r(i)-s0/s1)
  s5=s5+nd(i,j)*(s2/s1-(s0/s1)**2)
endo
  s6=s6+s4
  s7=s7+s5
endo
  scoR=s6*s6/s7
  type*, 'test for r=', scoR
\end{verbatim}
real function scoTH(nd, th, ncex, nsex, ncell)
real th(ncell), ncex(ncell,nsex)
integer nsex, ncell, nd(ncell,nsex)
s3=0
s4=0
do i=1,ncell
s1=0
s2=0
do j=1,nsex
s1=s1+sin(th(i)*nd(i,j))
s2=s2+cos(th(i)*nd(i,j))
enddo
s3=s3+s1
s4=s4+s2
enddo
emu=atan(s3/s4)
s6=0
s7=0
do i=1,ncell
s4=0
s5=0
do j=1,nsex
s1=0
s0=0
do il=1,ncell
s0=s0+ncex(il,j)*cos(th(il)-emu)
s1=s1+ncex(il,j)
s2=s2+(cos(th(il)-emu)**2)*ncex(il,j)
enddo
s4=s4+nd(i,j)*(cos(th(i)-emu)-s0/s1)
s5=s5+nd(i,j)*(s2/s1-(s0/s1)**2)
enddo
s6=s6+s4
s7=s7+s5
enddo
scoTH=s6*s6/s7
print *, 'mean angle:', emu
print *, 'test for conc:', scoTH
return
end
File _$255$DUA2:[MC.T.AL]PPGEN.FOR;159 (21532,2,0), last revised on 16-JAN-1990 12:34, is a 20 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 61 bytes.

Job PPGEN (613) queued to SYS$PRINT on 24-AUG-1990 13:02 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:05 from queue LASER.
This program generates Weibull-Von-Mises rate functions

```fortran
program test
  double precision dseed
  character*10 fname, tname
  real a, b, x, y, r, theta(2000), rad(2000)
  real lap, spac, zed, urao
  real moore
  real c0, c1, c2, pi
  real 11, 12, 13, 14
  real stat(10)
  real chialf(7), zedalf(7)
  real rh(1)
  integer ncount(10, 5)
  external f
  common/s1/r
  common/s2/c0, c1, c2
  common/s3/rnum, thnum
  common/s4/pi
  common/s5/dseed
  data(chialf(i), i=1, 7)/5.991, 7.378, 7.824, 9.21, 10.597, 13.815/
  & 15.20/
  data(zedalf(i), i=1, 7)/1.6449, 1.96, 2.054, 2.326, 2.576, 3.09, 3.2905/
  type*, 'hom or non-hom process(1,0):'
  read(5, *) nflag
  con=0.01745
  pi=3.14159
  pi2=2.*pi
  a=5.5
  b=5.5
  xc=0
  yc=0
  r=a/2.0
  r=2.5
  type*, 'enter number of points:'
  read(5, *) np
  type*, 'enter: shape, scale, psi, kap, pmu (degrees):'
  read(5, *, *) cl, c2, psi, kap, pmu
  pmur=pmu*0.01745
  dseed=801129.d0

c
  power test or otherwise

c
  type*, 'power(1) or not(0)'
  read(5, *, *) ipow
  if(ipow.eq.1)then
    do j=1, 5
      do k=1, 3
        ncount(j, k)=0
      enddo
    enddo
    do il=1, 1000
      type*, 'loop count=', il
      if(nflag.eq.0)then
        do i=1, np
          rnum=ggubfs(dseed)
          ab=rnum*(1.-exp(-c2*r**c1))
          rad(i)=(-alog(1.-ab)/c2)**(1/c1)
          ai=psi+kap*rad(i)
        enddo
      endif
    enddo
  endif
end program test
```
if(i.eq.1) then
  call bfishc(ai,h,0)
else
  call bfishc(ai,h,2)
endif
if(h.lt.0) then
  h=2.0*pi+h
endif
theta(i)=(h*360/(2.*pi)+pmu
if(theta(i).gt.360.) then
  theta(i)=theta(i)-360.
endif
endif
do i=1, n
  rnum=ggubfs(dseed)
  thnum=ggubfs(dseed)
  theta(i)=2.*pi*thnum
  rad(i)=sqrt(r**2)*rnum
  theta(i)=theta(i)*360./(2.*pi)
enddo
endif
do i=1, n
  rnew(i)=rad(i)
  thnew(i)=theta(i)
endo
call sort(rnew,n)
hed(1)=lap(rnew,r,n)
hed(2)=spac(rnew,r,n)
hed(3)=zed(rnew,r,n)
hed(4)=urao(thnew,n)
call sortboth(rad,n,theta)
hed(5)=moore(rad,theta,n)
do j=1, 5
  do k=1, 3
    if(abs(hed(j)).ge.zedalf(k)) then
      ncount(j,k)=ncount(j,k)+1
    endif
endo
doo
do j=1, 5
  do k=1, 3
    type*, 'count for', j, 'and', k, 'are', ncount(j,k)
    prop=ncount(j,k)/1000.0
    type*, 'prob=', prop
  enddo
doo
c else
  do i=1, n
    rnum=ggubfs(dseed)
    thnum=ggubfs(dseed)
    theta(i)=2.*pi*thnum
    rad(i)=sqrt(r**2)*rnum
    theta(i)=theta(i)*360./(2.*pi)
  enddo
call sort(rnew,n)
  hed(1)=lap(rnew,r,n)
  hed(2)=spac(rnew,r,n)
  hed(3)=zed(rnew,r,n)
  hed(4)=urao(thnew,n)
call sortboth(rad,n,theta)
  hed(5)=moore(rad,theta,n)
do j=1, 5
  do k=1, 3
    if(abs(hed(j)).ge.zedalf(k)) then
      ncount(j,k)=ncount(j,k)+1
    endif
  enddo
doo
do j=1, 5
  do k=1, 3
    type*, 'count for', j, 'and', k, 'are', ncount(j,k)
    prop=ncount(j,k)/1000.0
    type*, 'prob=', prop
  enddo
doo
else
  c ordinary visuals
  type*, 'coordinates of points in area:', a
  if(nflag.eq.0) then
  c choice of model
  type*, 'ord(1) or r-given-theta model(2)' 
  read(5,*) imod
  if(imod.eq.2) then
    type*, 'interaction must be zero'
    type*, 'enter shape regression parameter'
    read(5,*) dell
do i=1,np
rn=ggubfs(dseed)
call ggvms(dseed,psi,l,rh)
h=rh(i)
if(h.lt.0)then
  h=h+pi2
endif
theta(i)=(h*360/pi2)+pmu
if(theta(i).gt.360)then
  theta(i)=theta(i)-360
endif
vn=theta(i)
delpl=c1+dell*cos(vn*con-pmur)
anor=1.-exp(-c2*r**delpl)
anl=log(1.-rn*anor)
rad(i)=(-anl/c2)**(1./delpl)
type*,rad(i),theta(i)
enddo
else
  do i=1,np
    rnum=ggubfs(dseed)
    ab=rnum*(1.-exp(-c2*r**c1))
    rad(i)=(-alog(1.-ab)/c2)**(1/c1)
    ai=psi+kap*rad(i)
    if(i.eq.l)then
      call bfishc(ai,h,O)
    else
      call bfishc(ai,h,2)
    endif
    if(h.lt.0)then
      h=2.0*pi+h
    endif
    theta(i)=(h*360/(2.*pi)) + pmu
    if(theta(i).gt.360.)then
      theta(i)=theta(i)-360.
    endif
    type*,rad(i),theta(i)
  enddo
  endif
else
  do i=1,np
    rnum=ggubfs(dseed)
    thnum=ggubfs(dseed)
    theta(i)=2*pi*thnum
    rad(i)=sqrt(r**2)*rnum)
    theta(i)=theta(i)*360/(2*pi)
    type*,rad(i),theta(i)
  enddo
endif
open(l,file=fname,status='new')
do i=1,np
  write(l,*)xcart(i),ycart(i)
enddo

format(a10)
do i=1,np
xcart(i)=rad(i)*cos(theta(i)*2.*pi)/360)
ycart(i)=rad(i)*sin(theta(i)*2.*pi)/360)
enddo

read(5,*)ijse
if(ijse.eq.l)then
  type*, 'enter filename:
read(5,88)fname
else
  type*, 'enter graphics filename(gra):
read(5,88)fname
enddo
read(5,88) tname
open(3, file=tname, status='new')
write(3,*) np
do i=1, np
write(3,*) rad(i), theta(i)
enddo
close(3)
endif
do i=1, np
rnew(i)=rad(i)
thnew(i)=theta(i)
enddo
call sort(rnew, np)
ll=lap(rnew, r, np)
l2=spac(rnew, r, np)
l3=zed(rnew, r, np)
l4=urao(thnew, np)
call sortboth(rad, np, theta)
amo=moore(rad, theta, np)
type*, 'moores test:', amo
type*, 'statistics results'
type*, ll, l2, l3, l4
type*, 'lap, spac, zed, urao'
type*,
    
type*,
    
type*,
    
type*,
    
type*,
3256   type*, 'screen(1) or plotter(2):'
read(5,*) ifqw
if(ifqw.eq.1)then
    type*, 'press cntrl-shift f6 and 1 return'
read(5,*) iger
call gino
call t4010
call units(25.)
call piccle
else
call gino
call call044
call scale(20.0)
endif
call shift2(r, r)
call movto2(-r, 0.0)
call arcto2(0.0, 0.0, -r, -0.001, 0)
call movto2(0.0, 0.0)
call symbol(5)
do i=1, np
call movto2(xcart(i), ycart(i))
call symbol(3)
enddo
call devend
call ginend
endif
type*, 'more graphics(1:yes)'
read(5,*) ider
if(ider.eq.1)then
    goto 3256
endif
stop
end
real function f(x)
real x, b, mnrad, pi
common/s1/r
common/s2/c0, c1, c2
common/s3/rnum, thnum
File _$255$DUA2:[MC.T.AL]JJW.FOR;243 (39,14,0), last revised on 14-MAY-1990 11:53, is a 60
block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with
implied (CR) carriage control. The longest record is 69 bytes.

Job JJW (611) queued to SYS$PRINT on 24-AUG-1990 13:02 by user MCTAL, UIC [MCT,MCTAL], under
account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 13:59 from queue
LASER.
implicit real*8 (x,f,a,p,g,r,t,s,e)
character*20 fname,tname
character*20 wname
character*20 zname
character*2 codey(2)
character*4 code(8)
real*8 hesl(5),hesd(5)
real*8 hinfinf(3,3),h(3),hwinf(2,2),hwinv(2,2)
real*8 hthinv(3,3)
real*8 x(5),g(5)
real*8 wik(40)
real*8 f,w(9),work(100)
real*8 wkth(18),wkth(10)
real*8 wk(36),fnorm
real*8 a(10),b(10)
real*8 prob(20,50),res(20,50)
real*8 ansc(20,50)
real*8 zed(2000),rcell(20,50),thcell(20,50)
real*8 resp(1000)
real*8 thclass(50),rclass(50)
real*8 ncex(20,50)
real*8 dblin,cre
real*8 fun,ath,bth
real*8 alam,del,akap,psi,pmu5
real*8 khat
integer ncob(20,50)
integer iy(2000)
integer it(2000)
integer iw(1)
integer iwork(10),nsrch
integer iwork(5)
integer icode(4,4)
real*8 mmbs10,a1
integer istate(3),iw(3),liw,lw
external fun,ath,bth
external fweib,f2parm,fgamma
external funct1
common/s1/pi
common/s3/s1,s2,s3,s4,s5,s6,s7,s8,s9,s10
common/s4/rmax
common/s5/p1,pm
common/s6/pk,ps,pmu
data(icode(1,j),j=1,4)/1,4,5,6/
data(icode(2,j),j=1,4)/2,0,0,0/
data(icode(3,j),j=1,4)/3,0,0,0/
data(icode(4,j),j=1,4)/7,8,0,0/
common/s7/sath,sbth
common/s8/st(2)
common/s9/idor
common/s10/imarg
common/s36/rbar1
common/s37/khat
common/s40/alam,del,akap,psi,pmu5
common/s42/itype
common/s50/fh(0:100,0:100)
common/s60/idismod
common/s70/cccost
common/s80/nreg,rreg(1000),threg(1000),popreg(1000)
data codey/'75', '78'/
data code/'HEXX','LTPU','RAXX','HEFR',
+ 'HEPS','HETR','LETR','LERF'/
write (6,'*) 'enter filename:'
read(5,88)fname
format(a20)
type*, 'is the data split by factors:yes(1) ,no(2)'
read(5,*)iopr
if(iopr.eq.1)then
type*, 'are the factors:years(1) or species types(2)'
type*, 'or special combination(3)'
read(5,*)iozp
dendif
ome(open,l,file=fname,status='old')
read(l,*)np
if(iopr.eq.2)then
doi=1,np
read(l,*)r(i),th(i)
write(6,*)r(i),th(i)
dendothen
endif
c now read split on factors
c
if(iozp.eq.1)then
type*, 'which year:75(1) or 78(2)'
read(5,*)imz
npn=0
doi=1, np
read(l,*)rn(i),thn(i),iyear(i),itype(i)
if(iyear(i).eq.imz)then
npn=npn+1
r(npn)=rn(i)
th(npn)=thn(i)
it(npn)=itype(i)
dendif
enddo
np=npn
type*, 'no of points is now: ',npn
eelse
if(iozp.eq.2)then
type*, 'which species:'
type*, 'Heb(1),2,3,4,S,6,7,8'
read(5,*)iws
npn=0
doi=1, np
read(l,*)rn(i),thn(i),iyear(i),itype(i)
if(iyear(i).eq.iws)then
npn=npn+1
r(npn)=rn(i)
th(npn)=thn(i)
iy(npn)=iyear(i)
dendif
enddo
np=npn
type*, 'no of points is: ',np
endif
eendif
c special data combination
c
if(iozp.eq.3)then
type*, 'heb(1),lact(2),ram(3),lecc(4)' 
read(5,*)iou
type*, 'do you want all species(1) or sub-species(2)'
read(5,*),iop
if(iop.eq.2) then
  type*, '1,2,3,4,5,6,7,8'
read(5,*),ice
else
  ice=0
endif

type*, 'enter year: 75(1) or 78(2)'
read(5,*),iove
npn=0
do i=1,np
  read(1,*),rn(i),thn(i),iyear(i),itype(i)
  if(ice.eq.0) then
    do j=1,4
      if(itype(i).eq.icode(iou,j)) then
        if(iyear(i).eq.iove) then
          npn=npn+1
          r(npn)=rn(i)
          th(npn)=thn(i)
          iy(npn)=iyear(i)
          it(npn)=itype(i)
        endif
      endif
    enddo
  else
    if(itype(i).eq.ice) then
      if(iyear(i).eq.iove) then
        npn=npn+1
        r(npn)=rn(i)
        th(npn)=thn(i)
        iy(npn)=iyear(i)
        it(npn)=itype(i)
      endif
    endif
  endif
enddo
np=npn
endif
endif
type*, 'no of points is:', np

type*, 'output to file(l:yes;0:no)'
read(5,*) irepp
if(irepp.eq.1) then
  type*, 'enter filename'
  read(5,88), tname
  open(23,file=tname,status='new')
  write(23,*) np
  do i=1,np
    write(23,*) r(i), th(i)
  enddo
  close(23)
endif

calculate max radius

rmax=0
do i=1,np
  if(r(i).gt.rmax) then
    rmax=r(i)
  endif
enddo
type*, 'data r max value:', rmax

type*, 'own max rad value(1) or default(2)', read(5,*) iseo
if(iseo.eq.1) then
    type*, 'enter your rad value:'
    read(5,*) rm
    else
        rm=1.1*rmax
    endif
pi=3.141592
pi2=2.*pi
rdf=2.*pi/360.
conv=360./(2.*pi)
rm=0
do i=1,np
    r(i)=r(i)*(2.5/rm)
    if(r(i).gt.rml) then
        rml=r(i)
    endif
enddo
rmax=2.5
plot*, 'rml=', rml
plot*, 'rmax=', rmax

graphics

rdf=0.0174532
plot*, 'do you want graphics:yes(1) or no(2)'
read(5,*) iut
if(iut.eq.1) then
    do i=1,np
        xcart(i)=r(i)*cos(th(i)*rdf)
        ycart(i)=r(i)*sin(th(i)*rdf)
    enddo
    call z(xcart,ycart,np)
endif

s=0
rs=0.0
rc=0

do i=1,np
    s=s+sin(th(i)*rdf)
    rs=rs+r(i)*sin(th(i)*rdf)
    c=c+cos(th(i)*rdf)
    rc=rc+r(i)*cos(th(i)*rdf)
enddo
rbar=sqrt(s*s+c*c)/np
write (6,*) 'rbar=', rbar
sa=s/c
if(c.lt.0) then
    pmu=pi+atan(sa)
else
    if(s.lt.0.and.c.gt.0.0) then
        pmu=2*pi+atan(sa)
    else
        pmu=atan(sa)
    endif
endif
write (6,*), 'xbar=', pmu
write (6,*), 'mean angle(degrees)=', pmu*67.296

now choose form of analysis

plot*, 'type of analysis:
plot*, 'vm: 3 parm (0)
plot*, 'distance (1)
type*, 'full mult (2)'
type*, 'full additive (3)'
type*, 'case control(4)'
type*, 'pop/smr:(5)'
type*, 'r/theta model(mult):(6)'
type*, 'r/theta model(add):(7)'
type*, 'mixed model:additive(8):'
type*, 'mixed model:multi(9):'
read(5,*)idor
if(idor.ge.2.and.idor.le.7)then
type*, 'type of distance model'
type*, 'weib(O), or Gamma(l),'
read(5,*)idismod
endif

here we do vm 3parm

if(idor.eq.0)then
est1=(1./6.0)*rbar*(12+6*rbar**2.+5*rbar**4.)
est2=0.1
est3=pmu
write (6,*) 'suggested initial estimates of parameters:'
write (6,*) 'kappa,psi,mu'
write (6,*) est1,est2,est3
write (6,*) 'input your estimates'
read(S,*)x(1),x(2),x(3)
liw=5
lw=100

write (6,*) 'enter lower and upper bounds for each parameter'
do i=1,3
a(i)=0.0
enddo
b(1)=50.0
b(2)=50.0
b(3)=7.0
type*, 'bounded(0) or unbounded(1) run'
read(5,*)ib
n2=3
ifail=1
call e04kaf(n2,ib,a,b,x,f,g,iworl,liw,work,lw,ifail)
if(ifail.ne.0)then
write (6,*) 'failure due to', ifail
endif
write(6,*)x(1),x(2),x(3)
type*, 'max lik value: ', -f
write(6,*) 'aic= ', 2*f+8

now calling hessian evaluation

call hess2(n2,x,hesl,hesd)
write (6,*) 'Hessian:'
write (6,*) 'Diagonal: ', hesd(1),hesd(2),hesd(3)
write (6,*) 'psik,pmuk,pmupsi: ', hesl(1),hesl(2),hesl(3)
do i=1,3
do j=1,i
if(i.eq.j) then
hinf(i,j)=-hesd(i)
ext else
hinf(i,j)=-hesl(j+(i-1)*(i-2)/2)
hinf(j,i)=hinf(i,j)
endif
enddo
enddo
Hinf is now the neg of hessian i.e. $I$
Now invert $I$ to give var/cov matrix

Inverting Hessian

```fortran
idgt=4
call linv2f(hinf,3,3,hthinv,idgt,wkth,ier)
type*, 'hessian inverse: angular ses'
do i=1,3
write(6,*) (hthinv(i,j),j=1,3)
enddo
type*, 'standard errors'
do i=1,3
type*, 'se( ,i ,)=', sqrt(hthinv(i,i))
enddo
type*, 'do you want sum of squares : yes(1) or no(2)'
read(5,*) idlw
if(idlw.eq.1) then
write (6,*) 'now trying sums of squares'
write (6,*) 'estimates: original(1) or from found min(2)'
read(5,*) ifop
if(ifop.eq.1) then
x(1)=est1
x(2)=est2
x(3)=est3
endif
lw=100
m2=3
n2=3
liw=1
ifail=1
call e04fdf(m2,n2,x,fsum,iw1,liw,work,lw,ifail)
if(ifail.ne.0) then
write (6,*) 'ifail=', ifail
endif
write (6,*) 'parms=', x(1), x(2), x(3)
endif
pk=x(1)
ps=x(2)
pmu=x(3)
endif

end of vm 3 parm estimation

Select dist estimation here

if(idor.eq.1) then
io=1
endif
if(ioq.eq.1) then
type*, 'distance estimation'
type*, 'weibull(1) or 2 parameter(2)'
type*, 'or Gamma(3)'
read(5,*) ider
x(2)=1
sl=0.0
do i=1,np
sl=sl+r(i)**x(2)
enddo
x(1)=(np/sl)**(1/x(2))
type*, 'parms=', x(1), x(2)
nsig=3
maxfn=500
n=2
```
nh=n*(n+1)/2
nsrch=9
a(1)=0.0001
a(2)=1.0
b(1)=200.0
b(2)=200.0
if(ider.eq.1)then
  iopt=2
endif
if(ider.eq.2)then
  iopt=2
endif
if(ider.eq.3)then
  iopt=2
endif
if(ifail.ne.0)then
  write(6,*) 'failure due to:', ifail
endif
type*, 'max lik-', -f
if(ider.eq.2)then
  call zxmwd(f2parm,n,nsig,a,b,nsrch,x,f,wik,iwork,ier)
endif
if(ider.eq.3)then
  call zxmin(fgamma,n,nsig,maxfn,iopt,x,hesl,g,h,w,ier)
endif
type*, 'final estimates'
type*, 'scale(lambda) and shape(delta),
write(6,*)x(1),x(2)
pl=x(1)
pd=x(2)
call whess(pl,pd,h)
hwinf(1,1)=h(1)
hwinf(2,1)=h(3)
hwinf(2,2)=h(2)
hwinf(1,2)=hwinf(2,1)
idgt=4
call linv2f(hwinf,2,2,hwinv,idgt,wkw,ier)
type*, 'hessian inverse: radial ses'
do i=1,2
  write(6,*) (hwinv(i,j),j=1,2)
enddo
type*, 'standard errors'
do i=1,2
  type*, 'se(',i,')=' ,sqrt(hwinv(i,i))
enddo
type*, 'hessian:'
type*, (h(j),j=1,3)
endif

c c end of distance estimation

e ndif

c c do full multiplicative model here

c if(idor.eq.2.or.idor.eq.3.or.idor.eq.4.or.idor.eq.5)then

c c do full add-mult-casec model here

c
do i=1,5
x(i)=1.01
enddo
x(2)=1.01
do i=1,5
a(i)=0.001
enddo
a(2)=1.0
do i=1,4
b(i)=20.0
enddo
b(2)=30.0
b(5)=6.283
type*, 'bounded(0) or not(1):'
read(5,*)ib
if(idor.eq.4.or.idor.eq.5)then
  type*, 'enter filename:'
  read(5,88)wname
  open(26, file=wname, status='old')
  read(26,*)nnpnp
  if(idor.eq.4)then
    do ik=1,nnpnp
      read(26,*)caser(ik),caseth(ik)
    enddo
    do ik=1,nnpnp
      snca(ik)=1.
    enddo
    type*, 'smoother:'
    read(5,*)hsm
  else
    if(idor.eq.5)then
      type*, 'values at points(0) or interpolated(1),'
      read(5,*)interp
      do ik=1,nnpnp
        read(26,*)caser(ik),caseth(ik),snca(ik)
      enddo
      type*, 'smoother:'
      read(5,*)hsm
    endif
  endif
endif
do smoothing here

c n2=20
m2=20
hl=rmax/n2
hx1=pi2/m2
bot=2.*hsm*hsm
do i=0,n2
s=i*hl
ab=s/(pi2*nnpnp*hsm*hsm)
do j=0,m2
phi2=j*hx1
cl=0
fh(i,j)=dense(s,phi2,hsm,caser,caseth,snca,nnpnp)
enddo
dono
enddo

type*, 'done interpolation'
s53=0
do i=1,np
s=r(i)
ph2=th(i)*0.01745
if(idor.eq.5.and.interp.eq.0)then
  atl=snca(i)
else
  atl = dense(s, ph2, hsm, caser, caseth, snca, npnp)
endif
sS3 = sS3 + log(atl)
enddo
c const = sS3
type*, 'c const=', c const
endif
n2 = 5
ifail = 1
lw = 100
liw = 10
type*, 'zxmwd(1) or e04jaf(2)'
read(5, *), isti
if(isti.eq.1)then
  nsr = 40
  n sig = 3
  call zxmwd(funcl, n2, nsig, a, b, nsr, x, f, work, iwork, ier)
else
  call e04jaf(n2, ib, a, b, x, f, iwork, liw, work, lw, ifail)
endif
type*, 'final ests:', x(1), x(2), x(3), x(4), x(S)
if(ifail.ne.0)then
type*, 'error=', ifail
endif
type*, 'ests:', x(1), x(2), x(3), x(4), x(S)
al = -f + c const
stype*, 'max 1=', al
endif
if(idor.eq.6.or.idor.eq.7)then
do i = 1, S
  x(i) = 1.01
endo
do i = 1, S
  a(i) = 0.0001
endo
do i = 1, 4
  b(i) = 30.0
endo
b(5) = 6.283
type*, 'bounded(0) ormnot(1):'
read(5, *) ib
n2 = 5
ifail = 1
lw = 100
liw = 10
nsr = 40
nsig = 3
type*, 'zxmwd(1) or e04jaf(2)'
read(5, *), isit
if(isit.eq.1)then
  call zxmwd(funcl, n2, nsig, a, b, nsr, x, f, work, iwork, ier)
else
  call e04jaf(n2, ib, a, b, x, f, iwork, liw, work, lw, ifail)
endif
type*, 'final estimates:', x(1), x(2), x(3), x(4), x(5)
type*, 'final max 1:', -f
endif
if(idor.eq.8.or.idor.eq.9)then
type*, 'enter filename:'
read(5, 88) zname
open(19, file=zname, status='old')
read(19, *) nreg
do j=1,nreg
read(19,*)rreg(j),threg(j),popreg(j)
enddo
close(19)
do i=1,5
x(i)=1.01
dendo
do i=1,5
a(i)=0.0001
dendo
a(2)=1.0
do i=1,4
b(i)=30.0
dendo
b(5)=6.283
type*, 'bounded(0) or not(1):'
read(5,*)ib
n2=5
ifail=1
lw=100
liw=10
nsr=40
nsig=3
call e04jaf(n2,ib,a,b,x,f,iwork,liw,work,lw,ifail)
type*, 'final estimates:',x(1),x(2),x(3),x(4),x(5)
type*, 'fina max 1:',-f
dendif
c counts of observations
c
write(6,*)'number in cell:',j,k,'is',ncob(j,k)
enddo
enddo

type*, 'no of points=', np

type*, 'finished binning the data'

type*, 'do you want whole fit(0) or marginal r fit(l):'

read(S,*) imarg

c now calculate expected values

c

type*, 'parms:', pl, pd, pk, ps, pmu

sl0=0.0

sl1=0.0

sl2=0.0

sl3=0.0

sl4=0.0

aerr=0.0001

npn=0

do j=1,nr

do k=1,nth

npn=npn+1

xcart(npn)=(rclass(j)+rwid/2.)*cos(thclass(k)+thwid/2.)

ycart(npn)=(rclass(j)+rwid/2.)*sin(thclass(k)+thwid/2.)

ar=rclass(j)

sath=thclass(k)

br=rclass(j+1)

sbth=thclass(k+1)

cres=dblin(fun,ar,br,ath,bth,aerr,error,ier)

prob(j,k)=cres

ncex(j,k)=np*cres

if(ncex(j,k).eq.0.)then

ncex(j,k)=0.00001
endif

res(j,k)=ncob(j,k)-ncex(j,k)

resp(npn)=res(j,k)

ansc(j,k)=res(j,k)/sqrt(ncex(j,k))

s10=s10+res(j,k)

s11=s11+prob(j,k)*(1.-prob(j,k))

s13=s13+(res(j,k)*res(j,k)/ncex(j,k))

if(ncob(j,k).gt.0)then

s14=s14+ncob(j,k)*log(ncob(j,k)/ncex(j,k))
endif

enddo

ing writing to file

c

type*, 'info to file: yes(l) or no(2)'

read (5,*) idpe

if(idpe.eq.1)then


type*, 'enter filename:'

read(5,88) tname

open(20, file=tname, status='new')

write(20,*) nr, nth

do j=1,nr

do k=1,nth

write(20,*) ncob(j,k), ncex(j,k), res(j,k), ansc(j,k)

enddo

enddo

do i=1,npn

write(20,*) xcart(i), ycart(i)

enddo

close(20)

dendif

type*, 'resids and info printed:yes(1) or no(2)'

read(5,*) iov
if(iov.eq.1) then
  do j=1,nr
    write(6,*)'rcalass:',j
    write(6,*)'observed expected residual anscombe'
  do k=1,nth
    write(6,*)'angle class:',k
    write(6,*)ncob(j,k),ncex(j,k),res(j,k),ansc(j,k)
  enddo
  enddo
endif

c residual test

do kl=1,nr*nth-1
  do il=kl+1,nr*nth
    j1=kl-(il-1)*nth
    i1=il-(j1-1)*nth
    s12=s12+prob(il,j1)*prob(i1,j2)
  enddo
enddo
nreg=nr*nth
pt1=s10/nreg
pt2=(s11-2.*s12)/nreg

c here we replace test with chi squared

testlr=2*s14
testx2=s13
  type*, 'res test is:', testlr, testx2
sdf=nr*nth-6
  call mdch(testlr,sdf,pr,ier)
  type*, 'chi-squared prob:', pr
  type*, 'for lr test'
  type*, 'with df:', sdf
  call mdch(testx2,sdf,pr,ier)
  type*, 'chi-squared prob:', pr
  type*, 'for x2 test'
  type*, 'with df:', sdf

c graphics for surfaces

con=2.*pi/360.
npn=0
  type*, 'residual plot now follows'
  type*, 'cntrl-shift f6+1'
  read(5,*)iigls
  call grz(xcart,ycart,resp,nreg)
  type*, 'do you want surfaces:yes(1) or no(2)'
  read(5,*)iug
  if(iug.eq.1) then
    do j=1,nr
      do k=1,nth
        rcell(j,k)=rclass(j)+rwid/2.
        thcell(j,k)=thclass(k)+thwid/2.
        npn=npn+l
        xcart(npn)=rcell(j,k)*cos(thcell(j,k))
        ycart(npn)=rcell(j,k)*sin(thcell(j,k))
        zed(npn)=ncob(j,k)
      enddo
      enddo
    ncont=17
    ism=1
iview=0

read(5,*)isd
if(isd.eq.1)then
    call gino
call t4010
else
    call gino
call scale(20.0)
call call044
endif

call raniso(npn,xcart,ycart,zed,iview)
type*, 'cntrl-sh f6'
read(5,*)ish
call rancon(npn,xcart,ycart,zed,ncont,ism)
call ginend
call devend
endif
stop
end

c functions and subs

subroutine grz(x,y,rs,n)
real x(n),y(n),r(500),th(500)
real rs(n)
real*8 pi
integer n
common/sl/pi
rscreen=2.5
rm=0.0
    do i=1,n
        r(i)=sqrt(x(i)*x(i)+y(i)*y(i))
        if(x(i).eq.0.and.y(i).lt.0)then
            th(i)=pi
            goto 456
        endif
        if(x(i).eq.0.and.y(i).gt.0)then
            th(i)=3.1415926
            goto 456
        endif
        if(x(i).lt.0)then
            th(i)=pi+atan(y(i)/x(i))
            if(x(i).lt.0)then
                th(i)=h+pi
            else
                th(i)=h+2*pi
            endif
        else
            th(i)=h
        endif
        continue
    if(r(i).ge.rm)then
        rm=r(i)
    endif
    enddo
    do i=1,n
    r(i)=r(i)*(rscreen/rm)
    enddo

5678 type*, 'screen(0) or plotter(1)'
read(5,*)idet
if(idet.eq.0)then
    call gino
call t4010
call units(25.0)
else
call gino
call cal1044
call scale(20.0)
endif
call chapos(20.0,20.0)
call pickle
rsn=rscreen+0.3
call shift2(rsn,rsn)
call movto2(-rsn,0.0)
call arcto2(0.0,0.0,-rsn,-0.001,0)
call movto2(0.0,0.0)
call symbol(5)
doi=1,n
ax=x(i)-0.2
ay=y(i)-0.1
call movto2(ax,ay)
call chaint(rs(i),3)
endo
type*, 'cntrl-sht f6 l'
read(5, *)neft
call pickle
call devend
call ginend
type*, 'more graphics(l:yes)'
read(5, *)inore
if (inore.eq.1) then
goto 5678
endif
return
dend

c derivatives used here for eo4kcf

c subroutine functl(n,xc,fc)
implicit real*8 (a-h,o-z)
real*8 xc(n),fc
real*8 mmbisle9, mmbisle1
real*8 alam, del, akap, psi, pmu5
real anorm1
common/s1/pi
common/s4/rmax
common/s9/idor
common/s40/alam,del,akap,psi,pmu5
common/s70/cconst
rdf=0.01745
it=1
if (idor.eq.2) then
it=1
endif
if (idor.eq.3) then
it=2
endif
s20=0
if (idor.eq.6) then
it=6
endif
if (idor.eq.7) then
it=7
endif
if (idor.eq.9) then
it=1
endif
if(idor.eq.8)then
  it=2
endif
alam=xc(1)
del=xc(2)
akap=xc(3)
psi=xc(4)
pmu5=xc(5)
type*, 'parms:', xc(1), xc(2), xc(3), xc(4), xc(5)
do i=l, np
  thc=th(i)*rdf
  as=tense(alam, del, akap, psi, pmu5, r(i), thc, it)
  if(as.eq.0.0)then
    as=0.0001
  endif
  s20=s20+log(as)
endo
ar=anorm1(rmax, alam, del, it)
type*, 'norm const:', ar
if(ar.eq.0.0)then
  ar=0.0001
endif
if(idor.eq.4.or.idor.eq.5)then
  fc=-s20-cconst+np*log(ar)
else
  fc=-s20+np*log(ar)
endif
type*, 'current lik value=', -fc
return
end
subroutine funct2(n, xc, fc, gc)
  implicit real*8 (x, f, a, p, g, r, t, s)
  integer n
  real*8 xc(n), fc, gc(n)
  real*8 mmbsi0, mmbsi1
  real*8 alam, del, akap, psi, pmu5
  real anorm1
  common/s1/p1
  common/s4/rmax
  common/s9/idor
  common/s40/alam, del, akap, psi, pmu5
  if(idor.eq.0)then
    x1=xc(1)
    x2=xc(2)
    x3=xc(3)
    io=1
    s1=0
    s2=0
    s3=0
    s4=0
    s5=0
    s6=0
    s7=0
    s8=0
    rdf=0.01745
    write (6, *) 'parm values:', x1, x2, x3
    do i=1, np
      s1=s1+cos(th(i)*rdf-x3)
      s3=s3+r(i)*cos(th(i)*rdf-x3)
      s2=s2+sin(th(i)*rdf-x3)
      s4=s4+r(i)*sin(th(i)*rdf-x3)
      ao=x1+x2*r(i)
      write (6,*) ao, s1, s2, s3, s4
    enddo
  endif
end
a2=mmbst0(io,ao,ier)
a1=mmbstl(io,ao,ier)/a2
s5=s5+a1
s6=s6+r(i)*a1
s7=s7+log(a2)
s8=s8+log(rmax*r(i))
enddo
correct=-s8
gc(1)=-s1+s5
gc(2)=-s3+s6
gc(3)=-x1*s2-x2*s4
write(6,'(gradients:','gc(1),gc(2),gc(3)
fcc=-x1*s1-x2*s3+s7
type*,'lik=',-fc
type*,'correction(uniform r):',correct
c c distance estimation
if(idor.eq.1)then
x1=xc(1)
x2=xc(2)
s1=0
s2=0
s3=0
do i=1,np
s1=s1+r(i)**x2
s2=s2+log(r(i))
s3=s3+log(r(i))*r(i)**x2
endo
correct=-np*1.8378
a1=x1*rmax**x2
a2=exp(-a1)
a3=a1*log(rmax)
a4=1-a2
gc(1)=-(np/x1-s1-np*rmax**x2*a2/(a4))
gc(2)=-(np/x2)+s2-xl*s3-np*a3*a2/(a4))
write(6,'(graedients:','gc(1),gc(2)
type*,'parms=','x1,x2
fc=-(np*log(x1*x2)+(x2-2)*s2-xl*sl-np*log(1.-exp(-a1)))
type*,'lik=','-fc
type*,'correction(uniform angle):',correct
eendif
endif
if(idor.eq.2.or.idor.eq.3)then
if(idor.eq.2)then
it=1
endif
if(idor.eq.3)then
it=2
endif
s20=0
alam=xc(1)
del=xc(2)
akap=xc(3)
psi=xc(4)
muS=xc(5)
do i=1,np
as=tense(alam,del,akap,psi,muS,r(i),th(i),it)
s20=s20+log(as)
endo
c gradients here in gc(n)
c

do i=1,5


```fortran
gc(i) = grad(i, it)
enddo
ar = anorm1(rmax, alam, del, it)
type*, 'norm const:', ar
type*, 'number of points:', np
fc = -s20 + np*log(ar)
endif
return
end

real function dense(s, ph2, hsm, r, th, cn, np)
implicit real*8 (a-h,o-z)
integer np
real*8 s, ph2, hsm
real*8 r(np), th(np)
real*8 cn(l000)
pi = 3.1415
pi2 = 2*pi
rdf = 0.01745
ab = s / (pi2 * np * hsm * hsm)
bot = 2. * hsm * hsm
c1 = 0.0
do il = 1, np
r5 = r(il)
r6 = th(il)
rh = 2. * s * r5 * cos(ph2 - r6 * rdf)
c1 = c1 + cn(il) * exp(-(s * s + r5 * r5 - rh) / bot)
enddo
dense = c1 * ab
return
end

function tense(alam, del, akap, psi, pmu, r, th, itype)
implicit real*8 (x,f,a,p,g,r,t,s)
real*8 mmbsi0
real*8 alam, del, akap, psi, pmu, r, th
real aqw, adel
integer itype
common/s9/idor common/s4/rmax common/s60/idismod
pi = 3.14159
pi2 = 2*pi
i0 = 1
if (itype .ne. 6 .or. itype .ne. 7) then
adel = del
aqw = alam * r ** adel
a = alam * adel * (r ** (adel - 1.)) * exp(-aqw)
a5 = alam * r
a6 = exp(-a5) * r ** adel
gb = akap + psi * r
rdf = 0.01745
ser = gb * cos(th - pmu)

c gc = exp(ser)
i0 = 1
d = pi2 * mmbsi0(i0, gb, ier)
type*, 'a, gc, r, d:', a, gc, r, d
if (r .eq. 0.0) then
art = 0.00001
else
art = r
endif
if (idismod .eq. 1) then
a = a6
endif
dint = a * gc / (art * d)
```

if(itype.eq.1) then
tense=dint
else
if(itype.eq.2) then
tense=1.0+dint
endif
endif

else

c note akap=delta, psi=kappa
adel=del+akap*cos(th-pmu)
al=alam*(adel)*(r**(adel-2.))
a2=exp(-alam*r**(adel))
a3=exp(psi*cos(th-pmu))
a4=pi2*mmbi0(i0,psi,ier)
ab1=exp(-alam*r)
ab2=r**(adel-1.)
if(idismod.eq.1) then
al=ab1
a2=ab2
endif
dint=al*a2*a3/(a4)
if(itype.eq.6) then
tense=dint
endif
if(itype.eq.7) then
tense=1.0+dint
endif
return
end

real function ay(r)
ay=0.0001
return
end

real function by(r)
by=6.28319
return
end

real function funten(r,th,i1,j1)
imPLICIT REAL (A-H,O-Z)
REAL *8 alam,del,akap,psi,pmu5
COMMON/S40/alam,del,akap,psi,pmu5
COMMON/S9/idor
COMMON/S42/itype
COMMON/S50/fh(0:100,0:100)
rdf=0.01745
pi2=6.28319
xl=r*cos(th)
yl=r*sin(th)
if(idor.eq.4.or.idor.eq.5) then
fun5=fh(i1,j1)
funten=fun5*r*tense(salam,del,akap,psi,pmu5,r,th,itype)
else
funten=r*tense(salam,del,akap,psi,pmu5,r,th,itype)
endif
return
end

real function anorm1(rmx1,a11,de11,itype)
imPLICIT REAL*8 (X,F,A,P,G,R,T,S)
INTEGER itype
REAL*8 alam,del,akap,psi,pmu5
REAL*8 rmx1,all,de1
REAL ax,bx,ayl,byl,awe
common/s9/idor
common/s80/nreg,rreg(1000),threg(1000),popreg(1000)
common/s40/alam,del,akap,psi,pmu5
external funten
external ay,by
pi=3.14159
pi2=2*pi
area=pi*rmxl*rmxl
aint=-exp(-all*rmxl**del)
if(idor.eq.8.or.idor.eq.9)then
rmax=-20
do j=1,nreg
if(rreg(j).gt.rmax)then
rmax=rreg(j)
endif
enddo
endif
s0=0
s1=0
do j=1,nreg
r=rreg(j)/rmax
th=threg(j)
c type*, 'lamda,del,akap,psi,pmu5,r,th'
c type*,alam,del,akap,psi,pmu5,r,th
s0=s0+tense(alam,del,akap,psi,pmu5,r,th,itype)
enddo
if(idor.eq.8)then
anorml=s0
else
if(idor.eq.9)then
anorml=nreg+s0
endif
endif
else
if(idor.ge.2.and.idor.le.7)then
ax=0.0001
type*, 'rmxl=',rmxl
bx=rmxl
aerr=0.0001
ayl=0.0001
byl=6.28319
m2=20
n2=20
call doblin(ax,bx,ayl,byl,m2,n2,awe)
aint=awe
type*, 'integral=',awe
endif
if(itype.eq.1)then
anorml=aint
else
if(itype.eq.2)then
anorml=area+aint
endif
endif
type*, 'anorml=',anorml
return
end
real function grad(ione,it)
implicit real*8 (x,f,a,p,g,r,t,s)
real*8 x(10)
real*8 alam,del,akap,psi,pmu5
real anorml
integer ione,it
common/s40/alam,del,akap,psi,pmu5
common/s4/rmax
h=0.01
x(1)=alam
x(2)=del
x(3)=akap
x(4)=psi
x(5)=pmu5
s20=0
do i=1,np
  at=tense(x(1),x(2),x(3),x(4),x(5),r(i),th(i),it)
  s20=s20+log(at)
enddo
ah=anorml(rmax,alam,del,it)
type*, 'norm const:', ah
type*, 'no of points:', np
ll=-s20+np*log(ah)
x(ione)=x(ione)+h
s20=0
do i=1,np
  ak=tense(x(1),x(2),x(3),x(4),x(5),r(i),th(i),it)
  s20=s20+log(ak)
enddo
aw=anorml(rmax,x(1),x(2),it)
l2=-s20+np*log(aw)
grad=(l2-ll)/h
return
end
subroutine lsfun1(m,n,xc,fvecc)
implicit real*8 (p,r,t,s,x,a,f)
integer m,n
real*8 xc(n),fvecc(m)
real*8 rnmbsi0,rnmbsil
cornmon/sl/pi
x1=xc(1)
x2=xc(2)
x3=xc(3)
io=1
s1=0
s2=0
s3=0
s4=0
s5=0
s6=0
s7=0
rdf=0.01745
write (6,*) 'parm values:',x1,x2,x3
do i=1,np
  s1=s1+cos(th(i)*rdf-x3)
  s3=s3+r(i)*cos(th(i)*rdf-x3)
  s2=s2+sin(th(i)*rdf-x3)
  s4=s4+r(i)*sin(th(i)*rdf-x3)
ao=x1+x2*r(i)
a2=mmbst0(io,ao,ier)
al=mmbstl(io,ao,ier)/a2
s5=s5+al
s6=s6+r(i)*al
s7=s7+log(a2)
endo
fvecc(1)=s1-s5
fvecc(2)=s3-s6
fvecc(3)=x1*s2+x2*s4
write (6,*) 'resids=',fvecc(1),fvecc(2),fvecc(3)
subroutine hess2(n, xc, hl, lh, hd)
implicit real*8 (p, r, t, s, x, a)
integer lh, n
real*8 mmbsi0, mmbsil
real*8 hd(n), hl(lh), xc(n)
common/s1/pi
common/s3/s1, s2, s3, s4, s5, s6, s7, s8, s9, s10
io = 1
rdf = 0.01745
x1 = xc(1)
x2 = xc(2)
x3 = xc(3)
s1 = 0
s2 = 0
s3 = 0
s4 = 0
s5 = 0
s6 = 0
s7 = 0
s8 = 0
s9 = 0
s10 = 0
do i = 1, np
  ao = x1 + x2 * r(i)
a1 = mmbsi1(io, ao, ier)
a2 = a1 / mmbsi0(io, ao, ier)
a3 = 1. - a2 * a2 - a2 / ao
s1 = s1 + cos(th(i) * rdf - x3)
s2 = s2 + sin(th(i) * rdf - x3)
s3 = s3 + r(i) * cos(th(i) * rdf - x3)
s4 = s4 + r(i) * sin(th(i) * rdf - x3)
s5 = s5 + a1
s6 = s6 + r(i) * a1
s7 = s7 + log(a2)
s8 = s8 + r(i) * r(i) * a3
s9 = s9 + r(i) * a3
s10 = s10 + a3
endo
dh(1) = - s10
dh(2) = - s8
dh(3) = - x1 * s1 - x2 * s3
hl(1) = - s9
hl(2) = s2
hl(3) = - s4
return
end

subroutine z(x, y, n)
real*8 x(n), y(n)
integer n
r = 2.5
type*, 'screen(1) or plotter (2)'
read(5, *) iaw
if (iaw.eq.1) then
call gino
call t4010
call units(25.0)
call piccle
else
call gino
call call1044
call scale(20.0)
end
endif

call shift2(r,r)
call movto2(-r,0.0)
call arcto2(0.0,0.0,-r,-0.001,0)
call movto2(0.0,0.0)
call symbol(5)
do i=1,n
call movto2(x(i),y(i))
call symbol(3)
enddo
call devend
call ginend
type*,'cont:1 return'
read(5,*)nht
return
end

c hessian of weibull

c subroutine whess(pl,pd,h)
c
this routine returns values of -I for Weibull

c implicit real*8 (a,b,c,d,e,f,g,s,t,r,p)
real*8 h(3)
common/s4/rmax
s1=0.0
s2=0.0
s3=0.0
do i=1,np
  s1=s1+(r(i)**pd)*log(r(i))**2.
s2=s2+(r(i)**pd)*log(r(i))
endo
da1=np/(pd*pd)
a2=a2*log(rmax)
a3=a2*(log(rmax)**2.)
a4=a2-a3*b3-b2/b2/(1. - exp(-a2))
b5=1. - a2/(1. - exp(-a2))
h(1)=a1-exp(abx)
b1=np/(pd*pd)
b2=2*log(rmax)
b3=a2*(log(rmax)**2.)
b4=b3-b2*b2/(1. - exp(-a2))
b5=1. - a2/(1. - exp(-a2))
h(2)=b1+np*(np*a5/(1. - a5))*b4
h(3)=s2+(np*(rmax**pd)*log(rmax)/(1. - exp(-a2)))*b5
h(1)=-h(1)
h(2)=-h(2)
h(3)=-h(3)
return
end

subroutine fweib(n,xc,fc)
implicit real*8 (x,f,a,s,r,t,p)
real*8 xc(n),fc
common/s1/pi
common/s4/rmax
rdf=2. *pi/360.
s1=0.0
s2=0.0
do i=1,np
  s1=s1+log(r(i))
s2=s2+r(i)**xc(2)
endo
az=xc(1)
as=xc(2)
ab=(np*log(az*as)+(as-1)*s1-
laz*s2)
fc=-ab+np*log(1.-exp(-az*rmax**as))
type*, 'fc value=', fc
type*, 'parms=', xc(1), xc(2)
return
end

c 2 parm likelihood

subroutine f2parm(n, xc, fc)
implicit real*8 (x,f,a,s,r,t,p)
real*8 xc(n), fc
common/s1/pi
common/s4/rmax
rdf=2.*pi/360.
s1=0.0
s2=0.0
s3=0.0
do i=1, np
s1=s1+log(1.0+r(i))
s2=s2+log((1.0+r(i))*xc(2)-xc(1))
s3=s3+r(i)
enddo
ac=(xc(1)-1.)*s1
ad=xc(2)*s3
ae=np*log(1.-(1.+rmax)**xc(1))*exp(-
lxc(2)*rmax))
fc=-(ac+s2-ad-ae)
return
end

c Gamma likelihood

Gamma likelihood

subroutine fgamma(n, xc, fc)
implicit real*8 (x,f,a,s,r,t,p)
real*8 xc(n), fc
dcadre,gam
common/s4/rmax
common/s8/st(2)
s1=0.0
s2=0.0
a=0.0
b=xc(2)*rmax
do i=1,2
st(i)=xc(i)
enddo
do i=1, np
s1=s1+log(r(i))
s2=s2+r(i)
enddo
rerr=0.
aerr=1.0e-5
g=dcadre(gam, a, b, aerr, rerr, error, ier)
ab=np*log((xc(2)**xc(1))/g)
fc=-(xc(1)-1)*s1-xc(2)*s2+ab)
return
end
function gam(y)
implicit real*8 (s)
real*8 y
common/s8/st(2)
gam=(y**(st(l)-l))*exp(-y)
return
end

function fun(r,th)
implicit real*8 (r,t,p,b,c,a)
real*8 r,th
real*8 mmbsi0
common/s5/pl,pd
common/s4/rmax
common/s6/pk,ps,pmu
common/s10/imarg
a=(pl*pd*r**(pd-1.))*exp(-pl*r**pd)
b=1.-exp(-pl*rmax**pd)
c2=pk+ps*r
c1=exp(c2*cos(th-pmu))
c3=mmbsi0(1,c2,ier)
c4=c1/(c3*6.28318)
c5=a/b
if(imarg.eq.1) then
fun=c5/6.28318
else
fun=a*c4/b
endif
return
end

function ath(r)
implicit real*8 (s)
real*8 r
common/s7/sath,sbth
ath=sath
return
end

function bth(r)
implicit real*8 (s)
real*8 r
common/s7/sath,sbth
bth=sbth
return
end
File _$2SS$DUA2: [MC.T.AL]KERNTEST.FOR; 82 (17418, 34, 0), last revised on 20-JUN-1990 16:33, is a 22 block sequential file owned by UIC [MCT, MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 60 bytes.

Job KERNTEST (621) queued to SYS$PRINT on 24-AUG-1990 13:04 by user MCTAL, UIC [MCT, MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:19 from queue LASER.
implicit real (a-h,o-z)
real rr(2000), rth(2000), hs(1000,5)
real a1a(1000)
integer ira(1000)
integer npc, iflip
real bh(0:100,0:100)
real px(1000), py(1000)
real*8 ds1
character*20 fname, tname, wname
external ay, by, funten
external funct
common/s1/iflip
common/s3/hsm, ak, pmu, xbar0, a1am
common/s4/rbar, rmax
common/s50/fh(0:100,0:100)
type*, 'enter filename'
read(5,10)fname
format(a20)
read(20,*) np
sl=0
do i=1, np
read(20,*) r(i), th(i)
if(r(i).gt.rmax) then
rmax=r(i)
endif
sl=sl+r(i)
enddo
rbar=sl/np
type*, 'rmax, rbar=', rmax, rbar
rnaser=rmax
type*, 'rnaser=', rnaser
type*, 'enter control filename'
read(5,10) tname
type*, 'pop/smr(1) or case control(2)'
read(5,*) itre
open(25, file=tname, status='old')
read(25,*) npc
do i=1, npc
if(itre.eq.1) then
read(25,*) cr(i), cth(i), cna(i)
else
read(25,*) cr(i), cth(i)
cna(i)=1.0
endif
enddo
type*, 'enter smoother:'
read(5,*) hsm
type*, 'enter parameters(pmu and xbar0)'
read(5,*) pmu, xbar0

c interpolation here
	n2=20
m2=20
h=rnaser/n2
hx=pi2/m2
bot=2.*hsm*hsm
do i=0, n2
s=i*h
ab=s/(pi2*npc*hsm*hsm)
do j=0,m2
phi=j*hx
c1=0.0
do il=1,npc
rh=2*s*cr(il)*cos(phi-cth(il)*rdf)
c1=c1+cna(i)*exp(-(s*s+cr(il)*cr(il)-rh)/bot)
enddo
fh(i,j)=c1*ab
enddo
enddo
type*, 'interpolation done!'
type*, 'mc(l) or not(0)'
read(5,*),imc
if(imc.eq.l)then
nreq=np
type*, 'type of test:'
type*, 'radial(l)'
type*, 'peaked(2)'
type*, 'angle(3)'
type*, 'interaction(4)'
read(5,*),ijty
if(ijty.eq.l)then
it=1
else
if(ijty.eq.2)then
it=9
else
if(ijty.eq.3)then
it=1
else
if(ijty.eq.4)then
it=15

type*, 'enter null kappa and pmu:'
read(5,*),ak,pmu

type*, 'enter null beta:'
read(5,*),beta
endif
endif
endif
endif
rd=rmax/20.
thd=18.0
bot=2.*hsm*hsm
do i=0,20
s=i*rd
ab=s/(pi2*npc*hsm*hsm)
do j=0,20
phi=j*thd
c1=0.0
do il=1,npc
rh=2*s*cr(il)*cos(phi-cth(il)*rdf)
c1=c1+cna(i)*exp(-(s*s+cr(il)*cr(il)-rh)/bot)
enddo
fh(i,j)=c1*ab
enddo
enddo
fmax=-10.0
do i=0,20
sl=i*rd
do j=0,20
ph=j*thd
bh(i,j)=fh(i,j)*sl*tensnull(al,ak,pmu,xbar0,sl,ph,ijty)
if(bh(i, j).gt.fmax) then
  fmax = bh(i, j)
end if
end do
end do

C now doing loop

dsl = 874563.d0
do ilj = l, 99
  npoin = 1000
  do i = 1, npoin
    x(i) = ggubfs(dsl)
    y(i) = ggubfs(dsl)
  end do
  kl = 0
  do k = 1, npoin
    ab = ggubfs(dsl)
    xa = x(i) - 0.5
    ya = y(i) - 0.5
    rr(i) = sqrt(xa * xa + ya * ya)
    if(xa.eq.0.0) then
      h = 1.5707963
    elseif(ya.lt.0.0) then
      h = 4.712389
    elseif(ya.eq.0.0) then
      h = 0.0
    endif
    rth(i) = h
    goto 5678
  end do
  h = atan(ya/xa)
  if(xa.lt.0.0) then
    rth(i) = h + pi
  else
    if(ya.lt.0.0) then
      rth(i) = h + pi2
    else
      rth(i) = h
    endif
  endif
  goto 5678
end if

5678 continue

s = rr(i)
phi = rth(i)

cl = 0.0
do il = 1, npc
  rh = 2.*s*cr(il)*cos(phi - cth(il)*rdf)
  cl = cl + cna(i)*exp(-(s*s+cr(il)*cr(il)-rh)/bot)
end do
value = cl*s/(pi2*npc*hsm*hsm)
al5 = value*s*tensnull(al, ak, pmu, xbar0, s, phi, it)

C now do test

crit = alm5/fmax
if(ab .le. crit) then
  kl = kl + 1
  px(kl) = x(i)
  py(kl) = y(i)
end if
if(kl.ge.noreq) goto 1275
continue
npost=k1

c

   type*, 'loop number =', ilj
   iflip=1
   ax=0.0001
   bx=rnaser
   s1=0
   s2=0
   do i=1,np
      s1=s1+rr(i)
      s2=s2+cos(rth(i)*rdf-xbar0)
   enddo
   rbar=s1/float(np)
   m=20
   n=20
   c=0
d=pi2
call doblin(ax,bx,c,d,m,n,al)
   acl=s2/float(np)
   hs(ilj,1)=rad(ax,bx,c,d,m,n,al,np)
   hs(ilj,2)=ang(ax,bx,c,d,m,n,al,acl,np)
   alam=beta
   sl=0
   do i=1,np
      sl=sl+log(rr(i))
   enddo
   acl=sl/float(np)
   hs(ilj,3)=peak(ax,bx,c,d,m,n,al,acl,np)
   hs(ilj,4)=dinter(ax,bx,c,d,m,n,al,acl,rr,rth,np)

c
endo
c

      sl=0
      s2=0
      do i=1,np
         sl=sl+r(i)
         s2=s2+cos(th(i)*rdf-pmu)
      enddo
      acl=s2/float(np)
      rbar=sl/float(np)
      iflip=1
      hs(100,1)=rad(ax,bx,c,d,m,n,al,np)
      hs(100,2)=ang(ax,bx,c,d,m,n,al,acl,np)
      alam=beta
      ac2=s3/float(np)
      hs(100,3)=peak(ax,bx,c,d,m,n,al,ac2,np)
      hs(100,4)=dinter(ax,bx,c,d,m,n,al,ac2,r,th,np)

c
now doing sorting

c
   type*, 'now sorting'
   do j=1,4
      do i=1,100
         ala(i)=hs(i,j)
      enddo
      call vsrtr(ala,100,ira)
      do i=1,100
         if(ira(i).eq.100)then
            dart=float(i)
            pr=1.-dart/100.
            type*, 'mc prob= ', pr
            goto 6785
endif
enddo

continue
enddo

c this end of sim loop here
c
c
c
c
nO, 'l
do tests
c
c
c
c
endif
c
c now do tests
c
c
c
type*, 'type of test:'
type*, 'radial(1)'
type*, 'angle(2)'
type*, 'peakedness(3)'
type*, 'interaction(4)'
read(5,*) itype
type*, 'rmax, pi2, rmaxer: ', rmax, pi2, rmaxer
ax=0.0001
bx=rmaxer
aerr=0.0001
iflip=1
m=20
n=20
c=0
d=pi2
aser=rmaxer*rmaxer*pi2/2.
type*, 'circle radius=', aser
call dobin(ax, bx, c, d, m, n, a1)
type*, 'a1: first integral=' , a1
if(itype.eq.1) then
h4=rad(ax, bx, c, d, m, n, a1, np)
endif
if(itype.eq.2) then
s1=0
do i=1, np
s1=s1+cos(th(i)*rdf-xbar0)
enddo
acl=s1/float(np)
type*, 'acl=', acl
type*, 'a1=', a1
h5=ang(ax, bx, c, d, m, n, acl, np)
endif
if(itype.eq.3) then
ab=0.0001
bb=50.0
tol=0.1
call zxgsn(funct, ab, bb, tol, beta, ier)
type*, 'null beta est: ', beta
alam=beta
s1=0
do i=1, np
s1=s1+log(r(i))
enddo
acl=s1/np
h6=peak(ax, bx, c, d, m, n, acl, np)
endif
if(itype.eq.4) then

c

c here is interaction test (TO DO)
c

c
h7=dinter(ax, bx, c, d, m, n, acl, r, th, np)
c functions

c

real function rad(ax,bx,c,d,m,n,al,np)
real ax,bx,c,d,al
integer m,n,np
common/sl/iflip
iflip=2
call doblin(ax,bx,c,d,m,n,a2)
type*,'a2: second integral=' ,a2
iflip=4
call doblin(ax,bx,c,d,m,n,a3)
type*,'a3: third integral=' ,a3
as=a2/al
at=a3/al
atop=rbar-as
type*,'as,at,atop:' ,as,at,atop
abot=sqrt((at-(as*as))/np)
wtest=atop/abot
type*,'w test:' ,wtest
rad=wtest
return
end

real function ang(ax,bx,c,d,m,n,al,acl,np)
real ax,bx,c,d,al,acl
integer m,n,np
common/sl/iflip
iflip=5
call doblin(ax,bx,c,d,m,n,a2)
type*,'a2 integ=' ,a2
iflip=7
call doblin(ax,bx,c,d,m,n,a3)
type*,'a3 integ=' ,a3
as=a2/al
at=a3/al
atop=acl-as
abot=sqrt((at-(as*as))/np)
wtest=atop/abot
type*,'w test:' ,wtest
ang=wtest
return
end

real function peak(ax,bx,c,d,m,n,al,acl,np)
real ax,bx,c,d,al,acl
integer m,n,np
common/sl/iflip
iflip=9
call doblin(ax,bx,c,d,m,n,a2)
type*,'a2 integ=' ,a2
iflip=10
call doblin(ax,bx,c,d,m,n,a3)
type*,'a3 integ=' ,a3
iflip=11
call doblin(ax,bx,c,d,m,n,a4)
type*,'a4 integ=' ,a4
iflip=14
call doblin(ax,bx,c,d,m,n,a5)
type*,'a5 integ=' ,a5
iflip=12
call doblin(ax,bx,c,d,m,n,a6)
type*,'a6 integ=' ,a6
iflip=13
call doblin(ax,bx,c,d,m,n,a7)
type*, 'a7 integ=', a7
as=a3/a2
add=np*(a6/a2-as*as)
abb=np*(a7/a2-(a5/a2)*(a5/a2))
adb=np*(a3*a5/(a2*a2)-a4/a2)
type*, 'add, abb, adb:', add, abb, adb
atop=np*(acl-as)
type*, 'stat=', atop
type*, 'lnrbar, intlnr:', acl, as
adf=add-(adb*adb)/abb
type*, 'variance:', adf
wtest=atop/(sqrt(add-adb*adb/abb))
type*, 'wtes t=', wtest
peak=wtest
return
end
real function dinter(ax,bx,c,d,m,n,al,acl,r,th,np)
real r(np),th(np)
integer np
common/sl/iflip
common/s3/hsm,ak,pmu,xbar0,alam
rdf=0.0174
iflip=15
call doblin(ax,bx,c,d,m,n,a2)
iflip=18
call doblin(ax,bx,c,d,m,n,a3)
iflip=17
call doblin(ax,bx,c,d,m,n,a4)
iflip=16
call doblin(ax,bx,c,d,m,n,a5)
iflip=19
call doblin(ax,bx,c,d,m,n,a6)
iflip=20
call doblin(ax,bx,c,d,m,n,a7)
iflip=21
call doblin(ax,bx,c,d,m,n,a8)
iflip=22
call doblin(ax,bx,c,d,m,n,a9)
iflip=23
call doblin(ax,bx,c,d,m,n,a10)
iflip=24
call doblin(ax,bx,c,d,m,n,a11)
iflip=25
call doblin(ax,bx,c,d,m,n,a12)
kap=ak
asl=a2*a2
s1=0
s2=0
s3=0
s4=0
do i=1,np
s2=s2+cos(th(i)*rdf-pmu)
s1=s1+r(i)*cos(th(i)*rdf-pmu)
b2=sin(th(i)*rdf-pmu)
s3=s3+b2
s4=s4+r(i)*b2
enddo
sco=s1-np*(a3/a2)
type*, 'score is:', sco
akk=np*(a4/a2-a5*a5/asl)
kap=np*(a6/a2-a5*a3/asl)
akk=-s3+np*(a7/a2+akap*(a8/a2-a5*a7/asl))
amk=akm
app=anp*(a9/a2-a3*a3/asl)
apm=-s4+np*(a10/a2+akap*(a11/a2-a3*a7/asl))
amm=akap*s2+np*(akap*akap*(a12/a2-a7*a7/asl)-akap*(a5/a2))
all=apk*apm*amk
a12=apk*apm*amk
a14=akk*amk-amk*amk
apot=sqrt(app-(all-2*a12+a13)/a14)
atop=sco
wtest=apot/abot
dinter=wtest
type*, 'sco test:', wtest
return
end
real function funct(xp)
implicit real (a-h,o-z)
real xp
commom/s1/iflip
commom/s3/hsm,ak,pmu,xbarO,alam
commom/s4/rbar,rmax
external funten,ay,by
alam=xp
ax=0.0001
bx=rmax
type*, 'bx=', bx
aerr=0.0001
iflip=9
m2=20
n2=20
asy=ay(l)
bsy=by(l)
type*, 'asy=', asy
type*, 'bsy=', bsy
call doblin(ax,bx,asy,bsy,m2,n2,al)
if(al.eq.O.O)then
al=0.0001
endif
iflip=14
call doblin(ax,bx,asy,bsy,m2,n2,a2)
funct=abs(rbar-a2/al)
return
end
real function funten(r,th,il,jl)
implicit real (a-h,o-z)
integer iflip
real r,th
commom/s1/iflip
commom/s3/hsm,ak,pmu,xbarO,alam
commom/s50/fh(0:100,0:100)
h=hsm
xl=r*cos(th)
yl=r*sin(th)
s0=0
rdf=0.01745
pi=3.1415
p12=2*pi
h=hsm
fun5=fh(il, j1)
funten=fun5*r*tensnull(alam,ak,pmu,xbarO,r,th,iflip)
return
end
real function tensnull(al,ak,pmu,xbarO,r,th,iflip)
implicit real (a-h,o-z)
real al,ak,pmu,xbar0,var(30)
real r,th
integer it

now set up integral kernels

var(1)=1.0
if(r.eq.0)then
  r=0.0001
endif
var(2)=r
var(3)=log(r)
var(4)=r*r
var(5)=cos(th-pmu)
var(6)=sin(th-pmu)
var(7)=var(5)*var(5)
var(8)=var(6)*var(6)
var(9)=exp(-al*r)
var(10)=var(3)*var(9)
var(11)=var(2)*var(10)
var(12)=var(3)*var(3)*var(9)
var(13)=var(4)*var(9)
var(14)=var(2)*var(9)
var(15)=exp(ak*var(5))
var(16)=var(5)*var(15)
var(17)=var(7)*var(15)
var(18)=var(16)*var(2)
var(19)=var(2)*var(17)
var(20)=var(6)*var(15)
var(21)=var(6)*var(16)
var(22)=var(4)*var(17)
var(23)=var(2)*var(20)
var(24)=var(2)*var(21)
var(25)=var(8)*var(15)
tensnull=var(it)
return
end
real function ay(r)
ay=0.0001
return
end
real function by(r)
by=6.28319
return
end
File _$255$DUA2:[MC.T.AL]PROCDEL.FOR;31 (2353,51,0), last revised on 12-JUN-1990 11:19, is a 8 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 64 bytes.

Job PROCDEL (628) queued to SYS$PRINT on 24-AUG-1990 13:05 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:28 from queue LASER.
real x(1000),y(1000),x1(1000),y1(1000),x2(1000)
real y2(1000),s(1000)
real aint(1000)
real ar(30)
real disp(1000)
real r(1000),th(1000),rl(1000),thl(1000)
integer index,ind1(1000),ind2(1000)
integer pind(1000)
character*20 fname,tname,wname,zname
character*20 pname,sname
character*20 skip
character*10 area(1000)
type*, 'enter input filename'
read(5,200)fname
format(a20)
type*, 'enter del filename:'
read(5,200)wname
otype*, 'enter dir filename:'
read(5,200)zname
otype*, 'do you want summary files:'
type*, 'no(0),dels(1),dirs(2)'
read(5,*)isumar
if(isumar.ne.0)then
otype*, 'filename:'
read(5,200)pname
open(32,file=pname,status='new')
endif
open(30,file=wname,status='new')
onen(20,file=fname,status='old')
type*, 'orde r-th file(0) or tilevertex file(1)'
type*, 'or not(2)'
read(5,*)ivert
if(ivert.eq.1.or.ivert.eq.0)then
otype*, 'enter filename:'
read(5,200)tname
open(22,file=tname,status='new')
endif
type*, 'enter x-y coords of centre of coord system:'
read(5,*)xcen,ycen
type*, 'no of data points used'
read(5,*)ndata
k=0
k=k+1
read(20,400,err=550,end=500)ncf,index,x(k),y(k),ndir
format(q,i5,2f10.5,i5)
if(ncf.gt.2)then
439 format(i5,2f10.5,i5)
write(30,439)index,x(k),y(k),ndir
kl=0
s(k)=0.0
do i=1,ndir
read(20,720)indl(i),xl(i),yl(i),ind2(i),x2(i),y2(i),area(i)
720 format(i5,2f10.5,i5,2f10.5,a10)
write(30,720)indl(i),xl(i),yl(i),ind2(i),x2(i),y2(i),area(i)
dis=500000
ag=x(k)-xl(i)
bg=y(k)-yl(i)
dis=sqrt(ag*ag+bg*bg)
if(dis.lt.dis0)then
dis0=dis
endif
disp(k)=disO
read(20,660)arean
write(30,660)arean
if(isumar.eq.1)then
write(32,*)(index,x(k),y(k),ndir,arean
endif.
goto 41
else
read(20,660)areat
write(30,660)areat
format(f15.5)
810 format(a20)
read(20,810)skip
read(20,810)skip
endif

end of triangles

end of tiles

open(35,file=zname,status='new')
k=0
30 k=k+1
read(20,400,end=800)(ncf,index,x(k),y(k),ndir
if(ncf.gt.2)then
write(35,439)(index,x(k),y(k),ndir
do i=1,ndir
read(20,700)(ind1(i),x1(i),y1(i),x2(i),y2(i)
format(i5,4f10.5)
write(35,700)(ind1(i),x1(i),y1(i),x2(i),y2(i)
enddo
read(20,660)area1
write(30,660)area1
tarea=area1/3
xa=x(k)-xcen
ya=y(k)-ycen
r(k)=sqrt(xa*xa+ya*ya)
th(k)=tanm(ya,xa)
if(ivert.eq.1)then
if(k.1e.ndata)then
nfl=1
else
nfl=0
endif
write(22,*)(r(k),th(k),tarea,nfl
do i=1,ndir
ar(i)=farea(x(k),y(k),x1(i),y1(i),x2(i),y2(i)
enddo
nfl=0
do i=2,ndir+1
tar2=(ar(i-1)+ar(i))/3
xl1=x1(i)-xcen
yl1=y1(i)-ycen
rl1(i)=sqrt(xal*xl1+yal*yal)
th1(i)=tanm(yal,xal)
write(22,*)(rl1(i),th1(i),tar2,nfl
enddo
else
if(ivert.eq.0)then
if(k.le.ndata)then
nfl=1
else
nf1=0
endif
write(22,*)r(k),th(k),tarea,nf1
endif
endif
if(isumar.eq.2)then
write(32,*)index,x(k),y(k),ndir,areal
endif
goto 30
else
read(20,660)areat
write(35,660)areat
endif
c end of triangles
c
500
800
type*, 'end of data'
type*, 'end of data'
close(20)
close(30)
close(35)
stop
end
real function farea(xk,yk,xm,ym,xl,yl)
real xk,yk,xm,ym,xl,yl
xlk=xl-xk
xmk=xm-xk
ylk=yl-yk
ymk=ym-yk
al=xlk*ymk-xmk*ylk
if(al.lt.0)
  it=0
else
  it=1
endif
farea=abs(0.5*al)
return
end
real function tanm(y,x)
real y,x
pi=3.141593
pi2=2.*pi
if(x.eq.0)
  if(y.lt.0)
    h=1.5707
  endif
  if(y.eq.0)
    h=0.0
  endif
tanm=h
goto 5768
endif
h=atan(y/x)
if(x.lt.0)
  if(y.lt.0)
    tanm=h+pi
  else
    if(y.lt.0)
      tanm=h+pi2
    else
      tanm=h
  endif
endif
continue
return
end
File §255$DUA2:[MC.T.AL]DIRINT.FOR;52 (10460,10,0), last revised on 8-MAR-1990 17:18, is a 19 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 59 bytes.

Job DIRINT (630) queued to SYS$PRINT on 24-AUG-1990 13:06 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:30 from queue LASER.
this program produces integrations over dirichlet cells in 2-dim
the cell vertices and centre points are required as input

real xvert2(40),yvert2(40),xvert3(40),yvert3(40)
real xvert4(40),yvert4(40)
real xvert(40),yvert(40)
real x(0:1000),y(0:1000),x1(0:1000),y1(0:1000),x2(0:1000)
real y2(0:1000),s(0:1000),arean(0:1000)
real parea(0:1000) real rarea(0:1000)
real aint(0:1000)
integer icut(40)
integer index,ind1(1000),ind2(1000)
integer pind(1000)
integer nvert(40)
character*20 fname,tname,wname,zname,gname
external radlow,radth,fint
common/s1/a,b,c,as,bs,cs
common/s2/xc,yc
common/s3/nlp
common/s4/plam,del,akap,psi,pmu
ivert=0
nfl=0
type*, 'enter input filename'
read(5,200)fname
format(a20)
type*, 'enter intensity model(1-6)'
type*, '1:exp(-br)'
type*, '2:distance+peak'
type*, '3:distance-von Mises'
type*, '4:distance-interaction von Mises'
type*, '5:dist+peak+von Mises'
type*, '6:dist+peak+interaction von Mises'
read(5,*)nlp
type*, 'input parameters:'
type*, 'plam,del,kappa,psi,mu'
read(5,*)plam,del,akap,psi,pmu
type*, 'enter window radius:'
read(5,*)radw
type*, 'do you want vertex output(l:yes):'
read(5,*)ivert
if(ivert.eq.l)then
type*, 'enter filename:'
read(5,200)gname
endif
open(23,file=gname,status='new')
open(20,file=fname,status='old')
k=0
k=k+1
read(20,400,end=500)ncf,index,x(k),y(k),ndir
if(ncf.gt.18)then
kl=0
s(k)=0.0
pind(index)=0
c1=0
do i=1,ndir
icut(i)=0
endo
ijk=1
  do i=1,ndir
    ijk=ijk+1
    read(20,700)indl(i),xl(i),yl(i),x2(i),y2(i)
  enddo
  format(i5,4f10.5)
  enddo
  read(20,157)arean(k)
  format(f15.5)
  if(ivert.eq.1)then
    nfl=1
    tarea=arean(k)/3
    write(23,*)(x(k),y(k),tarea,nfl)
  endif

   c now do each tile
   c
   ijk=0
   xc=x(k)
   yc=y(k)
   do i=1,ndir
     ijk=ijk+1
     if(xl(i).eq.x2(i).and.yl(i).eq.y2(i))then
       goto 5050
     endif
     if(i.eq.1)then
       x2(i-1)=x2(ndir)
       y2(i-1)=y2(ndir)
     endif
   c check for gap
   c
   c gap here
   c
   icut(i)=1
   call bothleg(xl(i),yl(i),x2(i-1),y2(i-1),xc,yc,
                radw,xv1,yv1,xv2,yv2,pargap)
   xvert(0)=xv2
   yvert(0)=yv2
   parea(0)=pargap
   cl=cl+pargap
   ijk=ijk+1
   nfl=0
   endif
   xcen=0
   ycen=0
   ad=xcen-xl(i)
   ac=ycen-yl(i)
   bd=xcen-x2(i)
   bc=ycen-y2(i)
   az=ad*ad+ac*ac
   at=bd*bd+bc*bc
   rsq=radw*radw
   if(az.lt.rsq.and.at.lt.rsq)then
     parea(ijk)=area(xl(i),yl(i),x2(i),y2(i),xc,yc)
     xvert(ijk)=xl(i)
     yvert(ijk)=yl(i)
     cl=cl+parea(ijk)
   else
     c test for intersection
     call tint(xl(i),yl(i),x2(i),y2(i),radw,xn1,yn1,xn2
     if(it.eq.-1)then
       type*, 'lines corrupt'
     endif
   endif
goto 500
endif
if(it.eq.0.or.it.eq.1.or.it.eq.2)then
c    tangential or zero intersection
    call tint(xl(i),yl(i),xc,yc,radw,axl,ayl,all,a12,izl)
c    note there can only be one intersection of legs
    call tint(x2(i),y2(i),xc,yc,radw,ax2,ay2,b12,b13,izl)
afl=area(axl,ayl,ax2,ay2,xc,yc)
af3=encang(axl,ayl,ax2,ay2,radw)
af4=arcarea(radw,af3)
parea(ijk)=afl+af4
xvert(ijk)=axl
yvert(ijk)=ayl
c1=c1+parea(ijk)
else
if(it.eq.3)then
    c    line cuts but two intersections
    if(az.gt.rsq)then
        call tint(xl(i),yl(i),xc,yc,radw,axl,ayl,all,a12,it)  ! Error: Missing argument for azimuth (az)
a1l=area(xnl,ynl,x2(i),y2(i),xc,yc)
        xvert(ijk)=axl
        yvert(ijk)=ayl
    else
        call tint(x2(i),y2(i),xc,yc,radw,axl,ayl,all,a12,it)
a1l=area(xnl,ynl,xl(i),yl(i),xc,yc)
        xvert(ijk)=xl(i)
        yvert(ijk)=yl(i)
    endif
    afl=area(axl,ayl,xnl,ynl,xc,yc)
af3=encang(axl,ayl,xnl,ynl,radw)
af4=arcarea(radw,af3)
atemp=afl+af4
    if(az.gt.rsq)then
        parea(ijk)=atemp
    else
        parea(ijk)=afl
    endif
    parea(ijk+l)=atemp
    endif
    cl=c1+parea(ijk)+parea(ijk+l)
i1k=ijk+l
    xvert(ijk)=xnl
    yvert(ijk)=ynl
else
    c    two intersections
    call tint(xl(i),yl(i),xc,yc,radw,axl,ayl,all,a12,it)
call tint(x2(i),y2(i),xc,yc,radw,bxl,byl,b11,b12,itr)
ar1=area(xn1,yn1,xn2,yn2,xc,yc)
ar2=area(axl,ayl,xn1,yn1,xc,yc)
ar3=area(bxl,byl,xn2,yn2,xc,yc)
af3=encang(axl,ayl,xn1,yn1,radw)
af4=arcarea(radw,af3)
af5=encang(bxl,byl,xn2,yn2,radw)
af6=arcarea(radw,af5)
af7=encang(xn1,yn1,xn2,yn2,radw)
af8=arcarea(radw,af7)
xvert(ijk)=axl
yvert(ijk)=ayl
parea(ijk)=ar2+af4
ijk=ijk+l
xvert(ijk)=xnl
yvert(ijk)=ynl
parea(ijk)=arl+afB
\[ ijk = ijk + 1 \]
\[ x_{\text{vert}}(ijk) = x_{n2} \]
\[ y_{\text{vert}}(ijk) = y_{n2} \]
\[ p_{\text{area}}(ijk) = a_{r3} + a_{f6} \]
\[ c_{1} = c_{1} + a_{r1} + a_{r2} + a_{r3} + a_{f4} + a_{f6} + a_{f8} \]

```fortran
5050   continue
   enddo

end of main loop

```

\[ \text{itc} = ijk \]
\[ \text{do } i = 1, \text{itc} \]
\[ \text{if } (i .eq. 1) \text{ then} \]
\[ \text{if (icut(1).eq.1) then} \]
\[ \tau_{2} = (p_{\text{area}}(1) + p_{\text{area}}(0))/3 \]
\[ \text{else} \]
\[ \tau_{2} = (p_{\text{area}}(1) + p_{\text{area}}(\text{ndir}))/3 \]
\[ \text{endif} \]
\[ \text{else} \]
\[ \tau_{2} = (p_{\text{area}}(i) + p_{\text{area}}(i-1))/3 \]
\[ \text{endif} \]
\[ \text{if (ivert.eq.1) then} \]
\[ n_{f1} = 0 \]
\[ \text{write}(23,*),x_{\text{vert}}(i),y_{\text{vert}}(i),\tau_{2},n_{f1} \]
\[ \text{endif} \]
\[ \text{enddo} \]
\[ \text{type*, 'total tile area:', } c_{1} \]
\[ r_{\text{area}}(k) = c_{1} \]
\[ \text{type*, 'true area,area:', } r_{\text{area}}(k), \text{area}(k) \]
\[ \text{goto 30} \]
\[ \text{else} \]
\[ \text{endif} \]
\[ \text{type*, 'finished calculation: list follows'} \]
\[ \text{do } il = 1, k \]
\[ \text{write}(6,*), il, r_{\text{area}}(il), \text{area}(il) \]
\[ \text{enddo} \]
\[ \text{aint(k) holds the area integral for the kth tile} \]

```

```
500   type*, 'end of data'
   close(20)
   stop
end

functions and subroutines

```

```fortran
subroutine tint(xl,yl,x2,y2,r,xn1,yn1,xn2,yn2,it)
   assumes that xl<x2
   real xl,x2, yl, y2, r, xin, yin
   real xn1, yn1, xn2, yn2
   data accy/1.0e-6/
   f=x2-xl
   g=y2-yl
   fsq=f*f
   gsq=g*g
```

```fortran
end```
fgsq=fsq+gsq
if(fgsq.lt.accy)then
  c  lines corrupt
  it=-1
  goto 5000
else
  xj0=-x1
  yj0=-y1
  fygx=f*yj0-g*xj0
  root=r*r*fgsq-fygx*fygx
  if(root.lt.-accy)then
    c line does not intersect
    it=0
    goto 5000
  else
    fxgy=f*xj0+g*yj0
    t=fxgy/fgsq
    xn1=x1+f*t
    yn1=y1+g*t
    it=1
    goto 5000
  else
    it=2
    root=sqrt(root)
    fginv=1.0/fgsq
    t1=(fxgy-root)*fginv
    t2=(fxgy+root)*fginv
    a=x1+f*t1
    b=y1+g*t1
    c=x1+f*t2
    d=y1+g*t2
    iswap=0
    if(xl.ge.x2)then
      iswap=1
      bac=xl
      x1=x2
      x2=bac
    endif
    iflip=0
    if(a.gt.xl.and.a.lt.x2)then
      xnl=a
      ynl=b
      iflip=1
    endif
    if(c.gt.xl.and.c.lt.x2)then
      if(iflip.eq.0)then
        xnl=c
        ynl=d
      else
        xnl2=c
        ynl2=d
      endif
      iflip=iflip+1
    endif
    it=it+iflip
    if(iswap.eq.1)then
      al=xnl
      bl=ynl
      xnl=xn2
      ynl=yn2
      xn2=al
      yn2=bl
    endif
real function tanm(y,x)
real y,x
pi=3.15149
pi2=2*pi
if(x.eq.0)then
  if(y.gt.0)then
    tanm=1.5707
  else
    tanm=4.7123
  endif
else
  h=atan(y/x)
  if(x.lt.0)then
    tanm=h+pi
  else
    if(y.le.0)then
      tanm=h+pi2
    else
      tanm=h
    endif
  endif
endif
return
endif
end

real function radth(th)
real th,a,b,c,as,bs,cs
common/sl/a,b,c,as,bs,cs
radth=-cs/(as*sin(th)+bs*cos(th))
return
endif
end

real function radlow()
radlow=0.0
return
endif
end

real function fint(th,r)
real r,th
common/s2/xc,yc
x=xc+r*cos(th)
y=yc+r*sin(th)n=sqrt(x*x+y*y)
thn=tanm(y,x)
fint=tense(rn,thn)*rn
return
endif
end

real function tense(rn,thn)
real rn,thn
real
a=plam
b=del
c=akap
d=psi
e=pmu
if(nlp.eq.1)then
  ains=a*exp(-a*rn)/pi2
else
if(nlp.eq.2)
  bl=a*b*rn**(b-l)
  ains=bl*exp(-a*rn**b)/pi2
else
if(nlp.eq.3)
  al=exp(c*cos(thn-e)/(pi2*mmbsi0(c))
  ains=al*bl
else
if(nlp.eq.4)
  al=exp((c+d*rn)*cos(thn-e)/(pi2*mmbsi0(c+d*rn))
  b1=a*exp (-a*rn)
  ains=al*bl
else
if(nlp.eq.5)
  al=exp(c*cos(thn-e)/(pi2*mmbsi0(c))
  cl=exp(-a*rn**(b-l))
  ains=al*bl*cl
else
if(nlp.eq.6)
  al=c+d*rn
  bl=a*exp (-a*rn)
  cl=exp(-a*rn**b)
  d1=a*b*rn**(b-l)
  ains=bl*cl*d1
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endif
endi...
end
File $255$DUA2:[MC.T.AL]SES.FOR;15 (15402,8,0), last revised on 8-MAR-1990 19:47, is a 8 block sequential file owned by UIC [MCT,MCTAL]. The records are variable length with implied (CR) carriage control. The longest record is 56 bytes.

Job SES (618) queued to SYS$PRINT on 24-AUG-1990 13:03 by user MCTAL, UIC [MCT,MCTAL], under account MCT at priority 100, started on printer _VAXC$TXB2: on 24-AUG-1990 14:17 from queue LASER.
implicit real*8 (a-h,o-z)
real*8 x(10),hforw(10),objgrd(10)
real*8 hcnt(10),hess(10,10),hinv(10,10)
real*8 wka(150),work(150),user(1)
integer iuser(1),info(10)
character*20 fname
external objfun
common/s1/idor,ich
type*, 'enter data filename'
read(5,10)fname
format(a20)
rmmax=0.0
open(20,file=fname,status='old')
read(20,*)np
do i=1,np
read(20,*)r(i),th(i)
th(i)=th(i)*0.01745
if(r(i).gt.rmax)then
rmax=r(i)
endif
enddo
doi=1,np
r(i)=r(i)/rmax
dendo
type*, 'continuous(1) or discrete(2)'
read(5,*)idor
if(idor.eq.1)then
msg=0
if(idor.eq.1)then
if(ich.eq.1)then
n=2
type*, 'enter lamda and delta'
read(5,*)x(1),x(2)
else
if(ich.eq.2)then
n=2
type*, 'enter kappa and mu'
read(5,*)x(1),x(2)
else
if(ich.eq.3)then
n=3
type*, 'enter kappa,psi,mu'
read(5,*)x(1),x(2),x(3)
else
if(ich.eq.4.or.ich.eq.5)then
n=5
type*, 'enter lamda,delta,kappa,psi,mu'
read(5,*)x(1),x(2),x(3),x(4),x(5)
else
if(ich.eq.6)then
n=5
type*, 'enter lamda, del0, del1, kappa, mu:'
read(5,*)x(1),x(2),x(3),x(4),x(5)
endif
endif
endif
endif
endif
eps=-1.0
mode=2
do i=1,n
  hforw(i)=-1.0
enddo
ifail=0
lhes=10
call e04xaf(msg,n,eps,x,mode,objfun,lhes,
  hforw, objf, objgrd, hcnt, hess, iwarn, work
,iuser, user, info, ifail)
do i=1,n
do j=1,n
  hess(i,j)=-hess(i,j)
enddo
endo
d峨=0
call linv2f(hess,n,lhes,hinv,idgt,wka,ier)
type*, 'variances'
do i=1,n
type*, 'var(parm)':'i,'='hinv(i,i)
type*, 'se(parm)':'i,'='sqrt(hinv(i,i))
endo
stop
end

subroutine objfun(mode,n,x,objf,objgrd,nst,iuser,user)
implicit real*8 (a-h,o-z)
integer mode,n,nst,iuser(l)
real*8 x(n),objf,objgrd(n),user(l)
real*8 mmbsi0
common/sl/idor,ich
if(idor.eq.1)then
i0=1
pi=3.14159
pi2=2*pi
if(ich.eq.1)then
s0=0
s1=0
do i=1,np
  s0=s0+log(r(i))
  s1=s1+r(i)**x(2)
endo
anor=1.-exp(-x(1)*rmax**x(2))
ax=(x(2).2.)*s0-x(1)*s1-np*log(anor)
objf=np*log(x(1)*x(2))+ax
else
if(ich.eq.2)then
s2=0
do i=1,np
  s2=s2+cos(th(i)-x(2))
endo
av=pi2*mmbsi0(i0,x(1),ier)
if(ich.eq.3)then
  s2=0
  s3=0
  s4=0
  do i=1,np
    ab=x(1)+x(2)*r(i)
    a=cos(th(i)-x(3))
    s2=s2+a
    s3=s3+a*r(i)
    s4=s4+log(mmbsi0(i0,ab,ier))
  enddo
  objf=x(1)*s2+x(2)*s3-s4
else
  if(ich.eq.4.or.ich.eq.5)then
    area=pi*rmax*rmax
    anor=1.-exp(-x(1)*rmax**x(2))
    s10=0
    s11=0
    do i=1,np
      ab=x(3)+x(4)*r(i)
      az=x(1)*x(2)*r(i)**(x(2)-2.)
      bz=exp(-x(1)*r(i)**x(2))
      cz=exp(ab*cos(th(i)-x(5)))
      dz=pi2*mmbsi0(i0,ab,ier)
      C type*, 'az,bz,cz,dz:' ,az,bz,cz,dz
    enddo
    if(x(1).ne.0)then
      dlam=az*bz*cz/dz
    else
      dlam=cz/dz
    endif
    s10=s10+log(1.+dlam)
    s11=s11+log(dlam)
  enddo
  if(ich.eq.5)then
    objf=s10-np*log(area+anor)
  else
    objf=s11-np*log(anor)
  endif
else
  if(ich.eq.6)then
    s0=0
    s1=0
    s2=0
    s3=0
    s4=0
    do i=1,np
      ab=x(2)+x(3)*cos(th(i)-x(5))
      s0=s0+ab
      sl=sl+(ab*2)*log(r(i))
      s2=s2+r(i)**ab
      s3=s3+cos(th(i)-x(5))
      s4=s4+log(1.-exp(-x(1)*rmax**ab))
    enddo
    ac=np*log(pi2*mmbsi0(i0,x(4),ier))
    af=np*log(x(1))+s0+s1-x(1)*s2+x(4)*s3-ac-s4
    objf=af
  endif
return
end