Spatial point process methods for linear networks with applications to road accident analysis

Greg McSwiggan
Bachelor of Engineering (Civil)
Bachelor of Science (Honours)

Department of Mathematics and Statistics, UWA

2019

This thesis is presented for the degree of Doctor of Philosophy of The University of Western Australia

THESIS DECLARATION

I, Greg McSwiggan , certify that:

This thesis has been substantially accomplished during enrolment in this degree.

This thesis does not contain material which has been submitted for the award of any other degree or diploma in my name, in any university or other tertiary institution.

In the future, no part of this thesis will be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of The University of Western Australia and where applicable, any partner institution responsible for the joint-award of this degree.

This thesis does not contain any material previously published or written by another person, except where due reference has been made in the text and, where relevant, in the Authorship Declaration that follows.

This thesis does not violate or infringe any copyright, trademark, patent, or other rights whatsoever of any person.

This thesis contains published work and/or work prepared for publication, some of which has been co-authored.

Signature:

Date: 25 OCT 2019

AUTHORSHIP DECLARATION CO-AUTHORED PUBLICATIONS

This thesis contains work that has been published or prepared for publication.

1) *Details of the work:* Unpublished paper, Greg McSwiggan, Adrian Baddeley, Gopalan Nair, "Fitting Poisson point process models to road accident data "

Location in the thesis: Sections 2.1.1, 2.2.1 to 2.2.3, and Chapter 3

Student contribution to the work: 60% (literature review, initial software implementation, draft results and text).

2) Details of the work: Published paper, Greg McSwiggan, Adrian Baddeley, Gopalan Nair, "Kernel density estimation on a linear network "

Location in the thesis: Sections 2.4.2 to 2.4.10, and Chapter 4

Student contribution to the work: 80% (main technical idea, initial software development, draft text).

3) Details of the work: Published paper, Greg McSwiggan, Adrian Baddeley, Gopalan Nair, "Estimation of relative risk for events on a linear network "

Location in the thesis: Sections 2.6.2 to 2.6.4, and Chapter 5

Student contribution to the work: 90% (original concept, literature review, software development, computer experiments, draft text).

Adrian Baddeley [redacted] Date: 25/10/2019

Goplan Nair [redacted] Date: 25/10/2019

Student Signature:

Greg McSwiggan [redacted] Date: 25/10/2019

Abstract

Motivated by problems arising in the analysis of road traffic accident data, this thesis develops statistical methodology for analysing spatial patterns of points along a network of lines.

While there is a well-developed body of methods for analysing point patterns in two-dimensional space, this does not easily extend to point patterns on a linear network. Indeed there are some substantial technical challenges.

First we describe a general statistical approach which treats an observed point pattern on a linear network as a realisation of a spatial point process on the network. We develop basic statistical theory for Poisson point process models on a linear network, and implement an approximation to maximum likelihood estimation, using the Berman–Turner device. The methodology and computational implementation are demonstrated by fitting a variety of parametric models to a dataset of traffic accidents in the Australian regional city of Geelong. We also offer a critique of the the more traditional crash-frequency approach to analysing such data.

Next we develop a new, statistically-principled approach to kernel density estimation on a linear network. Existing heuristic techniques are reviewed. We develop a kernel estimator of the probability density function, equivalently of the intensity function, on a linear network based on the heat (diffusion) kernel, which we argue is the correct analogue of the familiar Gaussian kernel in one and two dimensions. The diffusion kernel estimate can be computed rapidly by numerically solving the time-dependent heat equation on the network. This enables bandwidth selection using cross-validation. The diffusion kernel estimation method is demonstrated using road accident data.

Estimation of relative risk on a linear network presents new challenges and exigencies. Existing techniques for estimating relative risk for spatial point patterns in two-dimensional space are extended to point patterns on a linear network, using the diffusion kernel intensity estimate. Several standard methods for bandwidth selection in two dimensions are adapted and extended to linear networks, and their finite-sample performance is evaluated by simulation. In the literature it is reported that, in the one- and two-dimensional cases, the Kelsall-Diggle density-ratio cross-validation method suffered sporadic "breakdowns", selecting extreme bandwidth values and yielding unsatisfactory risk estimates. The adaptation of this method to linear networks exhibits breakdowns even more frequently. The thesis provides a theoretical explanation for the breakdown, in either context, and proposes a modification of the Kelsall–Diggle method to improve its performance.

Acknowledgements

Thank you to my supervisors, Gopal and Adrian. This thesis has been done over eight years, part-time external from Brisbane. Adrian and Gopal have doggedly and painstakingly worked along with me constantly during all that time to help me get this thesis finished. I am better person having gone through this journey with them. Hopefully, this thesis is now close to their high standards.

Thank you to Bernard Ellem and Bob Murison who taught me statistics at UNE. Thank you to the late Mr Farokzedeh who taught mathematics St Leo's. I hope this work would have pleased you.

Thank you to my great friends Peter and Amanda from Coledale for their support and friendship. Special thanks to Susan for your wonderful encouragement and interest in this project. Thank you to Dr James for the motivational talks at Albion Hotel. Thank you to my parents for all of their support.

Thank you to my children Holly and Rob, who inspire with their unselfconscious teenage intellectual curiosity, and who, unlike everyone else in my life, have never once questioned why their engineer father spends so much of his spare time working on spatial point processes.

This research was supported by an Australian Government Research Training Program (RTP) Scholarship.

Contents

1 Introduction and overview **1**

1.1 Fitting point process models 1

1.2 Kernel estimation using diffusion 4

1.3 Relative risk on a linear network 5

2 Preliminaries and literature review **8**

2.1 Point patterns on a linear network 8

2.1.1 Linear network definitions 9

2.1.2 Visualisation of a function on a linear network 10

2.1.3 Example datasets 11

2.2 Point processes on a network 14

2.2.1 Point processes on L 14

2.2.2 Homogeneous Poisson process 15

2.2.3 Inhomogeneous Poisson process 15

2.2.4 Density and intensity 17

2.3 Approaches to data analysis for events on L 17

2.3.1 Count regression 17

2.3.2 Critique of count regression 19

2.3.3 Point process approach 21

2.4 Kernel intensity estimation: basics 22

2.4.1 Kernel smoothing on a line 23

2.4.2 Kernel intensity problem statement 23

2.4.3 Naive approach intensity estimators 25

2.4.4 The Okabe Group approach 27

2.4.5 Equal-split discontinuous estimator 27

2.4.6 Properties of the equal-split discontinuous estimator 29

2.4.7 Computation of equal-split discontinuous estimator 30

2.4.8 Equal-split continuous kernel estimator 30

2.4.9 Computation of equal-split continuous estimate . . . 31
2.4.10 Critique of the equal-split discontinuous estimator . . . 32
2.5 Bandwidth selection for density and intensity estimation . . . 34
2.5.1 Rules of thumb . . . 35
2.5.2 Likelihood cross-validation . . . 36
2.5.3 Weaknesses of cross-validation . . . 37
2.6 Relative risk . . . 37
2.6.1 Relative risk function . . . 37
2.6.2 Bandwidth selection for relative risk . . . 38
2.6.3 Kelsall–Diggle density-ratio cross-validation in $\mathbb{R}$ and $\mathbb{R}^2$ . . . 39
2.6.4 Likelihood and least-squares cross-validation . . . 40

3 Fitting point process models 42
3.1 Statistical theory . . . 42
3.1.1 Likelihood . . . 43
3.1.2 Score function . . . 44
3.1.3 Fisher information . . . 44
3.1.4 Standard errors and confidence intervals . . . 45
3.1.5 Model selection . . . 46
3.2 Intensity models . . . 47
3.2.1 Constant intensity . . . 47
3.2.2 Intensity proportional to a baseline . . . 47
3.2.3 The loglinear intensity model . . . 47
3.3 Model-fitting algorithm . . . 49
3.3.1 Discretisation methods . . . 49
3.3.2 Berman–Turner device . . . 49
3.3.3 Software implementation . . . 51
3.4 Applications . . . 52
3.4.1 Constant intensity . . . 52
3.4.2 Linear model: Accident intensity proportional to traffic volume model . 53
3.4.3 Loglinear model: Cartesian coordinates with traffic volume offset . . . 55
3.4.4 Loglinear model: Distance to nearest intersection with traffic volume offset 57
3.4.5 Extension of nearest-intersection model . . . 57
3.4.6 Loglinear model: Speed limit with traffic volume offset . . . 58
3.4.7 Loglinear model: Accident rate as a power of traffic volume . . . 61
3.4.8 Loglinear model: Using the most significant covariates . . . 61
3.5 Aggregation bias . . . 64

3.6 Applicability of Poisson models . . . 65
3.6.1 Dependence . . . 65
3.6.2 Accidents at an intersection . . . 66

4 Kernel estimation using diffusion **67**
4.1 Kernel smoothing and diffusion . . . 68
4.1.1 Diffusion on the real line . . . 68
4.1.2 Diffusion on a linear network . . . 69
4.1.3 Explicit representation of heat kernel . . . 70
4.2 Properties of the diffusion estimator . . . 71
4.2.1 Basic properties . . . 71
4.2.2 Asymptotics . . . 72
4.3 Numerical solution of the heat equation . . . 73
4.3.1 Discretization of a network . . . 73
4.3.2 Finite difference approximation . . . 74
4.4 Application to the Geelong accident data . . . 76
4.5 Computation time comparisons . . . 76
4.6 Bandwidth selection . . . 78
4.6.1 Rules of thumb . . . 81
4.7 Edge effects and corrections . . . 81
4.8 Subsequent papers . . . 82

5 Relative risk on a network **85**
5.1 Introduction . . . 85
5.2 Relative risk on L . . . 87
5.2.1 Kelsall-Diggle cross-validation on linear networks . . . 87
5.3 Improved cross-validation method . . . 87
5.3.1 General derivation . . . 88
5.3.2 Derivation of Kelsall–Diggle criterion . . . 90
5.3.3 Our proposed alternative . . . 90
5.4 Approximation to leave-one-out estimator . . . 92
5.5 Simulation experiments . . . 93
5.5.1 Description of experiments . . . 93
5.5.2 Representative results . . . 94
5.5.3 Summary of performance . . . 98
5.5.4 General comments on experiments . . . 99
5.6 Examples . . . 99

5.6.1 Geelong road accidents 99
5.6.2 Dendritic spines data 101

6 Discussion **107**
6.1 Parametric model-fitting 107
6.2 Kernel estimation 108
6.3 Relative risk 109
6.4 Kernel estimates and the K–function 111
6.4.1 Connection between K–function and kernel estimate 112
6.4.2 Heat kernel K–function 112
6.4.3 Examples 113

References **113**

List of Tables

2.1 Computation times (in seconds) for Okabe group kernel estimates of accident intensity in Geelong. 32

4.1 Computation times (in seconds) for diffusion kernel and Okabe group kernel estimates of accident intensity in Geelong. 78

5.1 Fraction of outcomes of each bandwidth-selection method in which the selected bandwidth is equal to the minimum permitted bandwidth (*Minimal*) or the maximum permitted bandwidth (*Maximal*). Here h_1, h_2 are the bandwidths selected jointly without any constraint (method M2), and h is the symmetric bandwidth (method M3). Simulation experiment case $i = 2, j = 3$. 96

5.2 Automatically-selected bandwidths for the Geelong accidents separated into night and day accidents. Symmetric bandwidths selected by method M3; asymmetric bandwidth pairs by method M2; exact calculation. The symbol ∞ indicates that infinite bandwidth achieved a better cross-validation score than the selected bandwidth, $C(\infty, \infty) < C(h, h)$. 100

5.3 Automatically-selected bandwidths (fast method) for the dendritic spines, relative risk of 'thin' type against 'other' types. 102

List of Figures

1.1 High-severity traffic accidents on state declared roads in Geelong between 2009 and 2011. 2
1.2 Fitted intensity (accident rate per unit length) of Poisson point process model with intensity values represented by a colour gradient 3
1.3 Kernel estimate of the Geelong data accident rate using the kernel diffusion method, bandwidth $\sigma = 2250$ metres. Perspective plot with vertical height proportional to intensity. 4
1.4 Geelong data split into daytime (*Top*) and night-time (*Bottom*) accidents. . . . 6
1.5 Relative risk of night versus day accidents using bandwidth 5 km. A line width plot, line width proportional to relative risk. 6

2.1 Line segment ℓ_i 9
2.2 An example fitted model predicting accident intensity for the Geelong data. *Top:* Colour gradient plot *Middle:* Line width plot in the style of Xie & Yan [133] *Bottom:* Perspective view plot in the style of Okabe & Sugihara [110] 12
2.3 Colour gradient plots of functions on a linear network for Geelong data. *Top:* Speed limit zones in km/hour. *Bottom:* AADT in vehicles per day (averaged over a year) 13
2.4 Dendritic spine data. One branch of the dendritic tree of a neuron, showing the positions of dendritic spines, of "stubby" or "mushroom" type. 14
2.5 Simulated realisation of a homogeneous Poisson point process on the Geelong major road network with the same average intensity as the Geelong accident data. 16
2.6 Illustration of kernel density estimate of $\lambda(\cdot)$ Solid circles denote the locations of the point pattern; the solid curves are the Gaussian kernel shifted to each data point location; and the dashed line is the kernel intensity estimate. 24

2.7 Naive estimate of intensity. *Left:* a data point (•) on a network containing a fork. *Right:* the naive kernel estimate (2.21). The tail of the kernel k is effectively duplicated onto each of the outgoing segments of the fork, increasing the total mass of $\hat{f}$. Mass is also lost due to truncation at terminal endpoints. 26
2.8 Equal-split kernel estimate. *Left:* the algorithm begins by making a copy of the kernel function for each data point x_i, confined to the line segment containing x_i. *Right:* At each fork, the remaining tail of the kernel is split equally between the outgoing segments. If there are n outgoing segments, each outgoing segment receives a copy of the kernel tail weighted by $1/n$. 41
2.9 Equal-split continuous method. At each fork, with $m-1$ outgoing segments, each outgoing segment receives a copy of the kernel weighted by $2/m$, while the incoming segment receives a copy with the negative weight $2/m - 1$. 41

3.1 Scheme for assigning weights to quadrature points. Each line interval contains one dummy point (circles) and may contain one or more data points (squares). The weight of each quadrature point is equal to its line interval length divided by the total number of quadrature points contained in that line interval. 51
3.2 Annual Average Daily Traffic (AADT) for Geelong major roads, portrayed as a line width plot. 54
3.3 Fitted intensity of loglinear model for Geelong data using x and y coordinates with traffic volume offset. portrayed as a colour gradient (top) and a perspective plot with height proportional to fitted intensity (bottom). 56
3.4 Fitted intensity of the loglinear model for Geelong data using distance to nearest intersection and intersection type with traffic volume offset, as a colour gradient plot (top) and line width plot (bottom). 59
3.5 Speed limits for Geelong major roads, portrayed as a colour gradient plot. . . . 60
3.6 Fitted intensity (accident rate) of model depending on speed limit with a traffic volume offset, with intensity values represented by a colour gradient plot (top) and perspective plot (bottom). 62
3.7 Fitted intensity (accident rate) of model depending only on most significant covariates, with intensity values represented by a colour gradient plot (top) and a perspective plot (bottom). 63

4.1 Discretization of a network 74
4.2 Diffusion estimate of intensity for Geelong road accident data with bandwidth $\sigma = 1000$ metres. *Top:* line width plot, line width proportional to intensity, in the style of Xie & Yan [133]; *Bottom:* perspective plot in the style of Okabe & Sugihara [110] with vertical height proportional to intensity. 77

4.3 Automatic bandwidth selection for Geelong accident data. Leave-one-out cross-validation criterion $C(\sigma)$ against bandwidth σ. Right panel is close-up of left panel around optimal value. 79
4.4 Counterpart of Figure 4.2 using the bandwidth $\sigma = 2250$ metres selected by leave-one-out cross-validation. 80

5.1 Geelong data split into daytime (*Top*) and night-time (*Bottom*) accidents. . . . 86
5.2 Linear network used in the experiments. 93
5.3 Simulation experiment reported in this section. *Top:* relative risk $r(u)$. *Middle:* denominator intensity $\lambda_{\mathbf{Y}}(u) = d(u)$. *Bottom:* numerator intensity $\lambda_{\mathbf{X}}(u) = r(u)d(u)$. Line width plots in the style of [133], with line width proportional to function value. 94
5.4 Boxplots of ISE values achieved by different methods for bandwidth selection in a simulation experiment (case $i = 2, j = 3$). *Top*: bandwidths h_1, h_2 are constrained to be equal (method M3). *Bottom:* bandwidths unconstrained (method M2). Note logarithmic scale for ISE. 95
5.5 Scatterplot matrix for the bandwidth values selected by each method under the constraint $h_1 = h_2 = h$, method M3. Simulation experiment case $i = 2, j = 3$. . 97
5.6 Relative risk of night versus day accidents using bandwidth 5 km. Line width proportional to relative risk. 100
5.7 Contours of cross-validation criterion as a function of the smoothing bandwidths h_1, h_2, for the Geelong data separated into night and day accidents. *Top*: modified Kelsall-Diggle criterion (5.17). *Bottom:* negative likelihood cross-validation criterion (2.47). Geelong data, relative risk, night versus day. Symbol $\oplus$ indicates optimal symmetric bandwidth h; symbol $*$ indicates optimal joint bandwidths (h_1, h_2). 103
5.8 Relative risk of single-vehicle versus multiple-vehicle accidents in the Geelong data, using bandwidth 5 km. Line width proportional to relative risk. 104
5.9 Dendritic spine data. One branch of the dendritic tree of a neuron, showing the positions of dendritic spines, of "stubby" or "mushroom" type (*Top*) and "thin" type (*Bottom*). 104
5.10 Contours of likelihood cross-validation criterion (2.47) for dendrite data, relative risk of 'thin' type against 'other' types. Symbol $\oplus$ indicates optimal symmetric bandwidth h; symbol $*$ indicates optimal joint bandwidths (h_1, h_2). 105
5.11 Estimated relative risk of "thin" type against other types for the dendritic spine data. Line width proportional to relative risk. Bandwidth 83.5 microns, selected by our modified cross-validation method. 106

6.1 Perspective plots. *Top*: Plot of traffic volume intensity on the Geelong roads. *Middle*: Semi-parametric estimate of relative intensity $r(u)$ adjusted by traffic volume using method 1. *Bottom*: Semi-parametric estimate of relative intensity $r(u)$ adjusted by traffic volume using method 2. 110
6.2 Thomas process on the Geelong data linear network. 114
6.3 *Top*: Linear network K–function proposed in Ang *et al*. *Bottom*: Heat Kernel K–function. 115
6.4 Spiders data from Ang *et al*. 116
6.5 *Top*: Linear network K–function proposed in Ang *et al* with envelope. *Bottom*: Heat Kernel K–function with envelope 117

Chapter 1

Introduction and overview

In this thesis we develop new methods for analysing spatial point patterns of events on a linear network. For example, Figure 1.1 shows the locations of high-severity road accidents on the major roads in the Australian regional city of Geelong during the years 2009 to 2011. The lines represent the roads and the circles represent the accident locations.

The traditional method for analysing such information is the crash-frequency approach in which the road is divided into segments and we count the number of accidents that occurred on each segment and then apply statistical techniques for the analysis of count data.

The crash-frequency approach is open to critique because it involves the aggregation of data into discrete road segments. This is effectively equivalent to assuming accident risk is constant along each chosen road segment.

In this thesis, we explore an alternative approach which retains the exact spatial coordinates of each accident. The accident locations are treated as a spatial point pattern on the road network, and statistical methods for spatial point processes are applied. This approach allows for the accident risk to continuously vary along the road network.

This approach involves numerous challenges because it is not a straight-forward matter to adapt existing statistical methodology for two-dimensional spatial point patterns to a linear network. These challenges are the subject of this thesis.

The analysis of traffic accident data as a spatial point process on a linear network, and more specifically the Geelong data, is the motivating example data for this thesis.

1.1 Fitting point process models

Chapter 3 demonstrates a new statistical methodology and computational algorithm for the spatial analysis of point pattern data on a linear network by adapting the Berman–Turner [20] device.

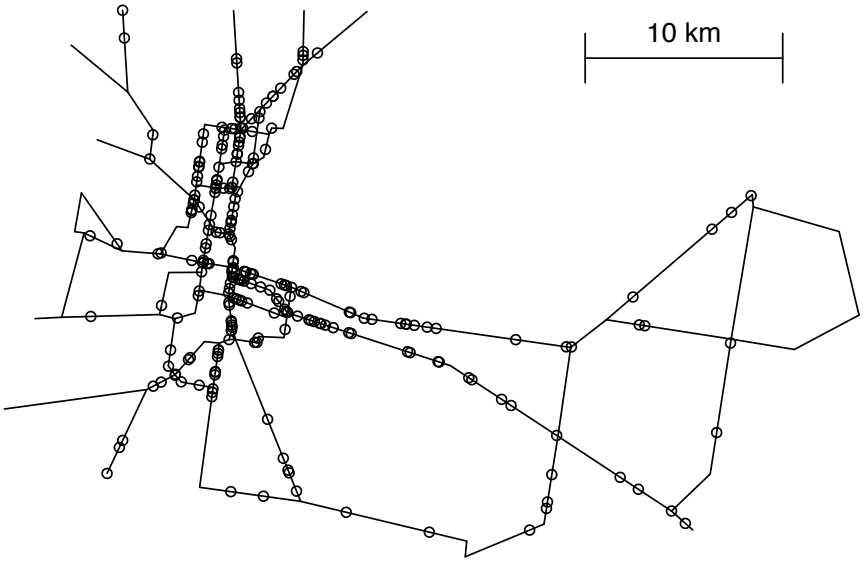

Figure 1.1: High-severity traffic accidents on state declared roads in Geelong between 2009 and 2011.

Figure 1.2 shows the result of a fitted model for the accident rate, on the Geelong roads from Figure 1.1, using the model-fitting methods for spatial point processes from Chapter 3. The model used is an inhomogeneous Poisson point process. The covariates used for this particular model were: the distance to the nearest intersection, the east-west spatial coordinate, and the traffic volume. Statistical methods for modelling point patterns in two-dimensional space are well-developed [51, 65, 73]. However, these methods require modification for use on a linear network.

The spatial point pattern data set shown in Figure 1.1 is analysed as the outcome of a random spatial point process. A point process is a stochastic mechanism which generates a spatial pattern of point events; it determines both the (random) number of events and their (random) spatial distribution.

The intensity (accident rate per unit length) is modelled using an inhomogeneous Poisson point process model restricted to a linear network. The model parameters are estimated using maximum likelihood and the parameter estimates are computed using the Berman–Turner device. The resulting computational algorithm enables the log-likelihood to be maximised using existing software designed to fit generalised linear models. The methodology is demonstrated by fitting several example models to the Geelong traffic accident data.

In Chapter 3 we consider only the simplest kind of point process model, the (inhomogeneous) *Poisson point process*. If road accidents follow a Poisson process, then individual accidents are

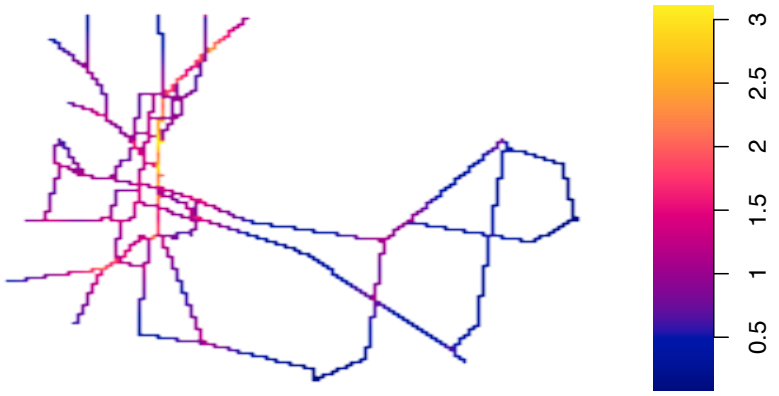

Figure 1.2: Fitted intensity (accident rate per unit length) of Poisson point process model with intensity values represented by a colour gradient

stochastically independent of one another. The term 'inhomogeneous' indicates that the rate of accidents per metre of road, called the intensity, varies with respect to location on the road network.

A Poisson process model is completely specified by its *intensity function*, which gives the intensity as a function of spatial location. The modelling process begins by proposing that the intensity function depends on a set of chosen covariates, according to a proposed functional form, involving parameters which must be estimated from the data. The most common model is a *loglinear* intensity model which assumes that the logarithm of the intensity is a linear function of the covariates, with coefficients that must be estimated from the data.

To our knowledge, the first published research to fit Poisson point process models to road accident data was [78]. However, the method used in [78] to estimate the model parameters is not statistically optimal and indeed its performance is not well understood.

In Chapter 3 we advocate using the method of *maximum likelihood* to estimate the model parameters. Maximum likelihood has theoretically optimal performance [60], and its behaviour is well-understood, allowing us to calculate confidence intervals for the model parameters, and to perform hypothesis tests about the influence of the covariates.

In general, the maximum likelihood estimates of the parameters cannot be expressed in closed form, and must be computed using iterative numerical procedures. We develop an efficient computational algorithm for the case of a loglinear intensity model, using the Berman–Turner [20] device.

1.2 Kernel estimation using diffusion

Chapter 3 deploys point process models to analyse point patterns on a linear network. However, a natural first step for analysing a point pattern, such as the data shown in Figure 1.1, would be to use kernel smoothing [121] to estimate the intensity. For the Geelong data, this would be an estimate of the spatially-varying accident rate. This approach investigates the underlying point process that produced the point pattern without assuming a particular model.

The challenge is that this is not straightforward on a linear network. Before the publication of our paper [100], reported in Chapter 4, there was no general agreement on how to perform kernel smoothing on a linear network and a rigorous foundation was lacking. Okabe *et al.* [110, p. 183] noted that "there appears to be no systematic method available for finding a continuous kernel density function on a network, and so we commonly employ heuristics". The two most popular heuristic techniques at that time, proposed by the Okabe group, ([109, 124]; [110, Chap. 9]) gave plausible results in applications, but their properties were not well understood, statistical insight was lacking, computational cost was high, and cost increased exponentially with the bandwidth. Automatic bandwidth selection was computationally prohibitive.

In Chapter 4 we develop a statistically principled approach to kernel density estimation on a linear network. Figure 1.3 shows a kernel estimate of the intensity, of the spatial point pattern shown in Figure 1.1, using the diffusion kernel developed in Chapter 4.

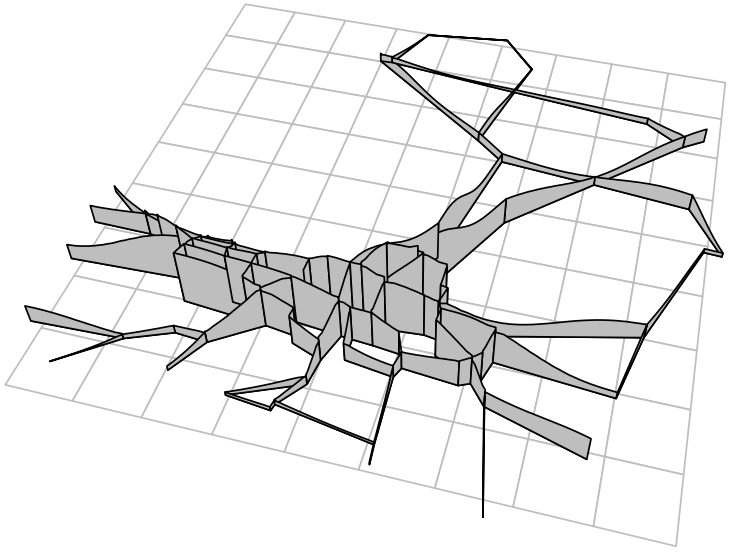

Figure 1.3: Kernel estimate of the Geelong data accident rate using the kernel diffusion method, bandwidth $\sigma = 2250$ metres. Perspective plot with vertical height proportional to intensity.

The Gaussian kernel is a standard solution for applications in one and two-dimensional cases. The correct analogue of the Gaussian kernel on a linear network is the 'heat kernel', the occupation density of Brownian motion on the network. The corresponding kernel estimator satisfies the classical time-dependent heat equation on the network,

$$\frac{\partial f}{\partial t} = \beta \frac{\partial^2 f}{\partial x^2}. \tag{1.1}$$

This 'diffusion estimator' has good statistical properties, which follow from the heat equation. This thesis proposes the use of a kernel intensity estimation method on a linear network based on the heat kernel on a linear network. It is mathematically similar to an existing heuristic technique [109, 124], in that both can be expressed as sums over paths in the network. However, the diffusion estimate is an infinite sum, which cannot be evaluated using existing algorithms.

This estimator is formed by summing the heat kernels associated with each data point, and so satisfies the heat equation. This enables us to develop a very fast algorithm for intensity estimation, based on numerical solution of the heat equation. This iterative algorithm yields intensity estimates for a sequence of bandwidths up to and including a specified maximum bandwidth: this makes it feasible to perform automatic bandwidth selection.

In Chapter 4 the diffusion estimate with automatically-selected bandwidth is demonstrated on road accident data.

1.3 Relative risk on a linear network

In Chapter 5 we aim to estimate the *relative risk*, the spatially-varying ratio of the intensities of two different types of events on a linear network.

The spatial point pattern locations in Figure 1.1 are not the only information available for the Geelong data traffic accidents. Accidents can be categorised into types of accidents: day or night; single or multi-vehicle, for example. Figure 1.4 shows the data separated into day and night accidents. Figure 1.5 shows the relative risk of night accidents versus day accidents estimated in Chapter 5.

Estimation of relative risk is different from estimation of the absolute accident rate, particularly with regard to the choice of smoothing bandwidth. For example, if the bandwidth is chosen to be *infinite*, the resulting smoothed function is constant with respect to spatial location; a constant accident rate is implausible, but a constant relative risk between two types of accidents is a reasonable null hypothesis in many applications.

Relative risk estimates are also less susceptible to Simpson's Paradox [136]. For example, the traffic accident rate is influenced by the weather, but if we assume that weather has the

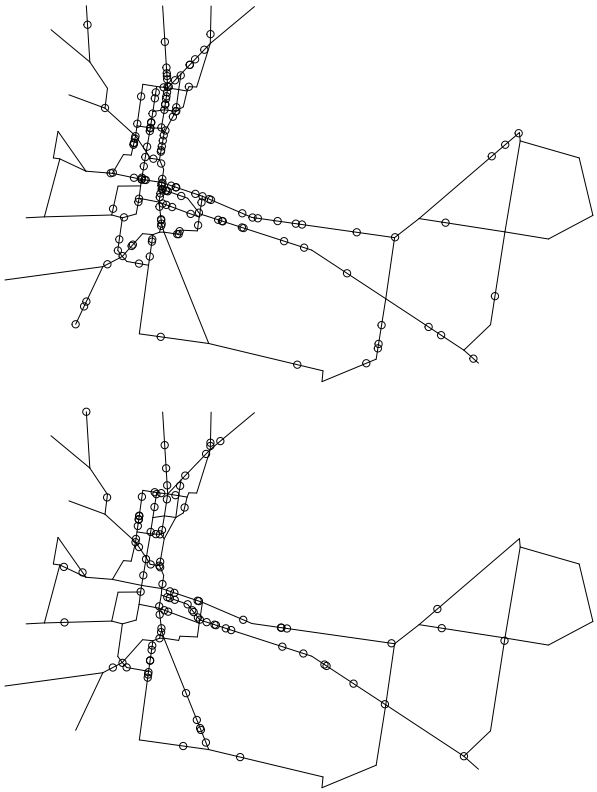

Figure 1.4: Geelong data split into daytime (*Top*) and night-time (*Bottom*) accidents.

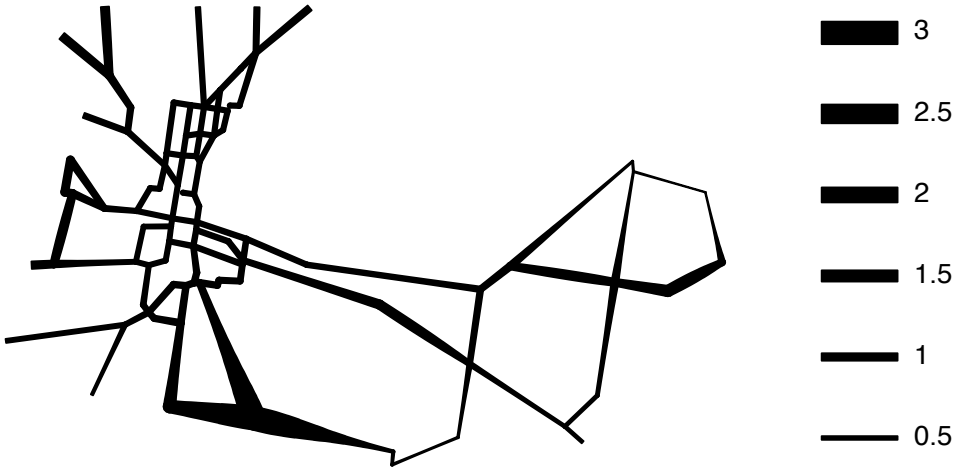

Figure 1.5: Relative risk of night versus day accidents using bandwidth 5 km. A line width plot, line width proportional to relative risk.

same multiplicative effect on day and night accident rates, then the relative risk of day and night accidents can be estimated without needing to adjust for weather.

Chapter 5 extends and adapts relative risk estimation techniques for spatial point patterns in two-dimensional space to point patterns on a linear network. The main problem is to choose the smoothing bandwidth(s) for kernel estimation. Estimation on a linear network presents new challenges and exigencies. In Chapter 5, several standard methods for bandwidth selection in two dimensions are adapted and extended to linear networks, and their performance is evaluated in simulation experiments. Kelsall and Diggle [79, 80] reported that their density-ratio cross-validation method suffered sporadic "breakdowns" in which the selected bandwidths and resulting risk estimates were very unsatisfactory. The adaptation of the method to linear networks exhibits breakdowns even more frequently. Chapter 5 details a theoretical explanation for the breakdown, in either context, and proposes a modification of the Kelsall-Diggle method to improve its performance. The smoothing kernel used is the diffusion kernel developed in the previous Chapter 4.

Chapter 2 provides background material required for the following chapters and a literature review of existing work. Chapter 6 contains discussions, conclusions, and possible further research.

Chapter 2

Preliminaries and literature review

This chapter introduces some basic definitions for linear networks, reviews existing literature on linear networks, and surveys relevant background from two-dimensional point process methods which we seek to adapt to linear networks.

The chapter is generally laid out as follows. Section 2.1 reviews the mathematical definitions required for a point pattern on a linear network. Section 2.2 describes the definitions required to study a Poisson point process on a linear network. Section 2.3 explains the existing statistical methods used to analyse traffic accident data on a road network and explains the way of conceiving the data as a spatial point pattern on a linear network, and how analysing it as the realisation of a point process is different from and better than the existing methods. Section 2.4 provides background to kernel intensity estimation methods, looks at why it is problematic to directly apply existing methods to point pattern data on a linear network, and reviews the existing heuristic methods used for kernel smoothing of point pattern data on a linear network. Section 2.5 reviews bandwidth selection methods for kernel intensity estimation methods for point pattern data in $\mathbb{R}$ and $\mathbb{R}^2$. Section 2.6 describes the existing work on relative risk analysis and bandwidth selection for relative risk for point pattern data in $\mathbb{R}$ and $\mathbb{R}^2$.

2.1 Point patterns on a linear network

A spatial point pattern data set represents the spatial locations where certain events occurred. The events are often inherently constrained to occur only in a subset of the space. In some applications, the events are constrained to occur on a linear network, that is, a subset that can be idealized as a network of lines.

Point patterns on a network of lines are found in many applications. The 'lines' that form the network may be representations of roads, rivers, rail lines, electrical wires, nerve fibres, airline routes, a length of a yarn or soil cracks. The 'points' may represent traffic accidents,

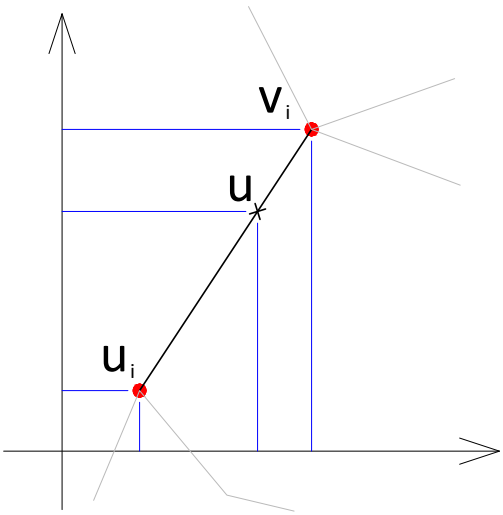

Figure 2.1: Line segment ℓ_i

vehicle thefts or street crimes [135, 96, 133, 8]; roadside trees or invasive species [123, 49]; retail stores, roadside kiosks or urban parks [107, 108, 111, 38]; insect nests [131, 8], neuroanatomical features [134, 74, 12], slubs (abnormally thick places) on a length of yarn [39] or sample points along a stream [130, 129, 122].

2.1.1 Linear network definitions

Here we adopt the notation established by Ang *et al.* [8].

The *line segment* in the plane with endpoints u and v is denoted by $[u, v] = \{tv + (1-t)u : 0 \leq t \leq 1\}$. A *linear network* L is the union $L = \bigcup_{i=1}^{N} \ell_i$ of line segments $\ell_1, \ldots, \ell_N$ in the plane, where $1 \leq N < \infty$. The total length of all line segments in L is denoted by $|L|$.

We also view L as a graph embedded in the plane, defined by a finite set of vertices $v \in V$ and edges $e \in E$, where $e = [v, v']$ (unordered), for $v, v' \in V$, such that the intersection of two different edges contains at most one point, and any such intersection point is a vertex. The *degree* $deg(v)$ of a vertex v is the number of line segments $e = [v, v']$ with an endpoint at v. A vertex is *terminal* if it has degree 1.

The representation of a network as a graph is not unique: for example, we may add a new vertex at any location along an existing segment. A statistical technique should yield identical results for different representations of the same network, and this is true for all the techniques discussed in this thesis.

The assumption that the network is embedded in two-dimensional space is made only for

simplicity. Most of the results generalise easily to a network of curves embedded in a higher-dimensional space. We note that a recent paper by Anderes *et al.* [6] discusses to use of curves for networks but we have not considered that in this thesis.

Throughout this thesis, integrals of a function on a linear network are taken with respect to one-dimensional arc length along the network. Consider a real-valued function $f : L \to \mathbb{R}$ on the linear network L. For each segment $\ell_i = [u_i, v_i]$, with length $|\ell_i| = s_i$, define $f_i : [0, 1] \to \mathbb{R}$ by $f_i(t) = f(tv_i + (1 - t)u_i)$, $0 \leq t \leq 1$. Distance along the segment is measured by $s = ts_i$. Define

$$\int_L f(u)\mathrm{d}u = \sum_{i=1}^{N} \int_0^{s_i} f_i(\frac{s}{s_i})\,\mathrm{d}s = \sum_{i=1}^{N} s_i \int_0^1 f_i(t)\,\mathrm{d}t, \tag{2.1}$$

if all terms on the right hand side exist and are finite [118].

A *path*, between two points y_1 and y_2 along L is a sequence $\pi = (y_1, v_1, v_2, \ldots, v_P, y_2)$, where $v_1, \ldots, v_P$ are vertices, such that $e_j = [v_j, v_{j+1}]$ is an edge of L, for each $j = 1, \ldots, P - 1$, while y_1 and v_1 lie on a common edge e_0 of L, and y_2 and v_P lie on a common edge e_P of L. The path length is $\ell(\pi) = \|v_1 - y_1\| + \sum_{j=1}^{P-1} \|v_{j+1} - v_i\| + \|y_2 - v_P\|$, where $\|\cdot\|$ denotes Euclidean distance. The *shortest path distance* $d_L(u, v)$ between u and v in L is the minimum of the lengths of all paths from u to v. If there are no paths from u to v (implying that the network is not connected) then we define $d_L(u, v) = \infty$.

Integrals on L can be transformed using an analogue of the conversion to polar coordinates [8, eqn. (13)]:

$$\int_L f(u)\,\mathrm{d}u = \int_0^{\infty} \sum_{u \in L : d_L(v,u) = t} f(u)\,\mathrm{d}t, \tag{2.2}$$

for any fixed $v \in L$, and any integrable function f on the network. We use this transformation for integrals of functions involving the shortest path distance in Chapter 4.

A *point pattern* $\mathbf{x}$ on a linear network L is a finite unordered set $\mathbf{x} = \{x_1, \ldots, x_n\}$ of distinct points $x_i \in L$, where $n \geqslant 0$ is not fixed in advance. Thus, an empty pattern is possible, but multiple coincident points are not allowed.

2.1.2 Visualisation of a function on a linear network

A function defined on a linear network can be displayed in at least three different ways as shown in Figure 2.2. A *colour gradient plot* (top panel) encodes the function values as colours, and draws the network segments using these colours. A *line width plot* (middle panel) in the style of Xie & Yan [133] draws the network segments with varying thickness proportional to the function value. A *perspective plot* in the style of Okabe & Sugihara [110] builds a three-dimensional scene composed of vertical walls rising above the network segments, with variable heights proportional to the function value, and displays a perspective view of this scene.

Baddeley and Turner's [17] spatial point pattern data analysis software package `spatstat` within the R statistical software environment [114] has been used extensively throughout this thesis. Existing `spatstat` functions have been used where they exist and new functions have been created when required, such as the functions for implementing the Chapter 4 diffusion kernel.

These three visualisation techniques have been implemented in `spatstat` (with contributions from the author of this thesis). We will use them interchangeably throughout the thesis.

2.1.3 Example datasets

The primary data set used throughout the thesis is the Geelong Data shown in Figure 1.1. The roads are located in the Australian regional city of Geelong, in the state of Victoria. The roads are "state-declared roads", these are the roads in the state of Victoria that are the responsibility of *VicRoads*, the state transport department. The state-declared roads are the larger roads in the road network. When we created the Geelong data, some roads were arbitrarily truncated at the boundary of the area of interest. The accidents used for the Geelong data were filtered by the accident classification "high-severity" accidents and the accidents are from three consecutive years 2009 to 2011. The source of the information for the Geelong data shown in was *CrashStats* [1], which is owned by *VicRoads*

The information about each accident we have from the data is: location, date and time, speed zone, nearby traffic control (stop-go lights for example), light conditions (day, night, street lighting), "urbanisation" (city, rural), road dry or wet, "atmosphere" (weather), total vehicles involved, numbers of people killed or injured, and a short description of what occurred.

The data from CrashStats was downloaded as a pdf map of roads with circled accidents locations for each year with a separate list of those accidents, including the information for each accident listed above. The location of the accident provided on the list was used to identify that accident on the map. The pdf map was imported into AutoCad and scaled to the correct size. An arbitrary origin was chosen and the coordinates of every vertex (road intersection or bend) was recorded, plus the coordinates of every accident. A matrix of vertices connected by roads was created by hand. The coordinates of the vertices and the connection matrix was entered into `spatstat` to create a digital linear network object. The accident coordinates were entered into `spatstat` to create a spatial point pattern on that linear network. Note that throughout this thesis we will use the terms accidents and crashes interchangeably.

The AADT (Annual Average Daily Traffic) is the number of vehicles passing a location during a year divided by 365. This information was recorded from the *VicRoads* publicly available online traffic data spreadsheets. The speed limit locations were found by interrogating the *CrashStats* data base to find enough points where we know the speed limit to create a

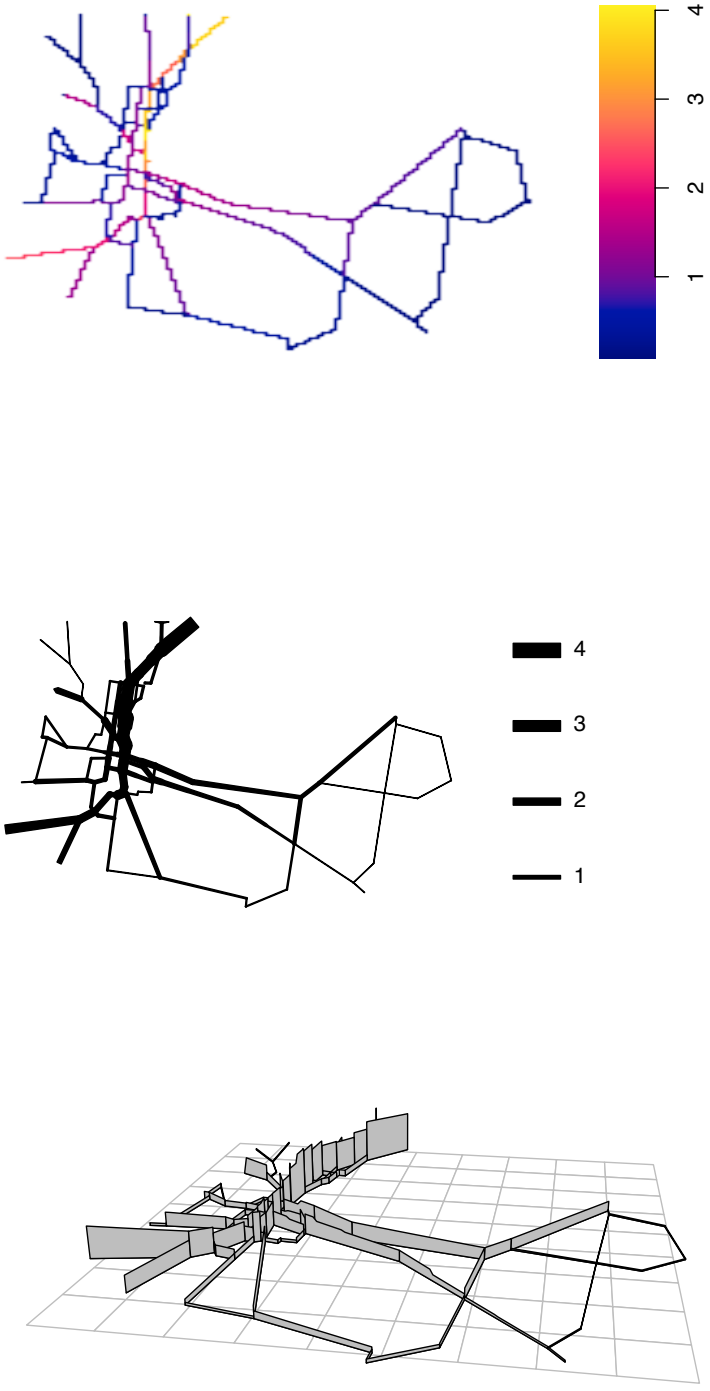

Figure 2.2: An example fitted model predicting accident intensity for the Geelong data. *Top:* Colour gradient plot *Middle:* Line width plot in the style of Xie & Yan [133] *Bottom:* Perspective view plot in the style of Okabe & Sugihara [110]

reasonable map of speed limit information for every location. The traffic volume data (ADDT) and speed limit data were entered into `spatstat` as functions on the linear network, such that there is a value for that covariate at every location in the network as shown in Figure 2.3.

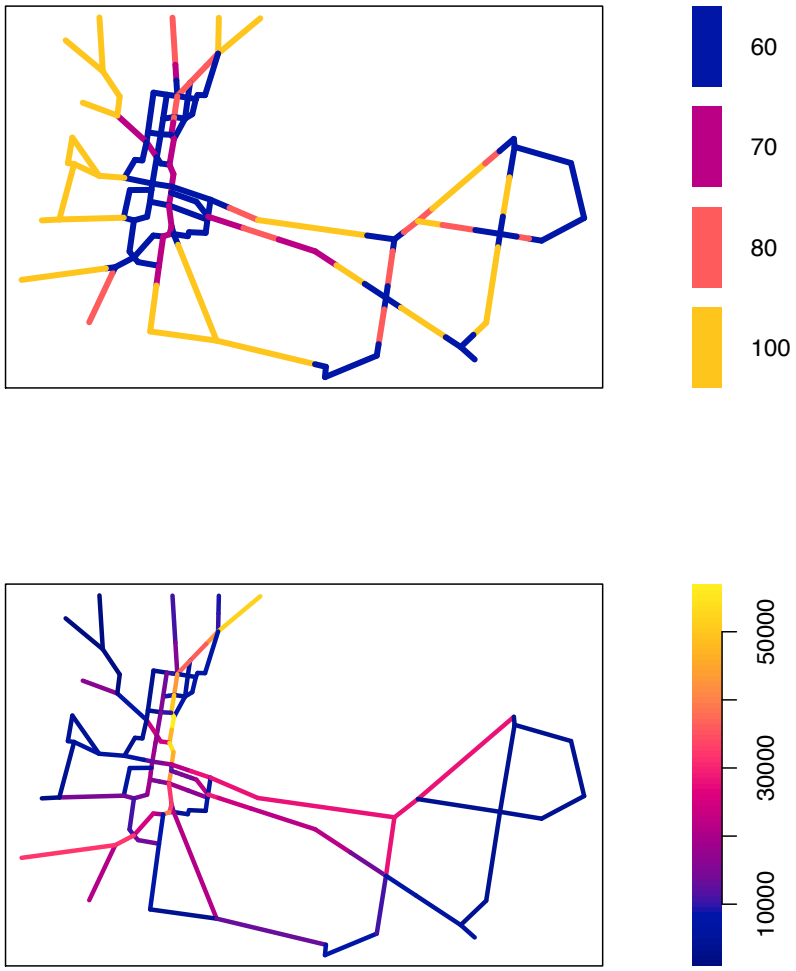

Figure 2.3: Colour gradient plots of functions on a linear network for Geelong data. *Top:* Speed limit zones in km/hour. *Bottom:* AADT in vehicles per day (averaged over a year)

The other data set used in this thesis in Chapter 5 is the dendritic spines data studied in Jammalamadaka *et al.* [74] and Baddeley *et al.* [12], some of which is shown in Figure 2.4. The network represents one branch of the dendritic tree of a neuron. The points are the locations of small protrusions called spines, which are classified into three types: mushroom, stubby, and thin. We use this as an example data set to study relative risk on a linear network.

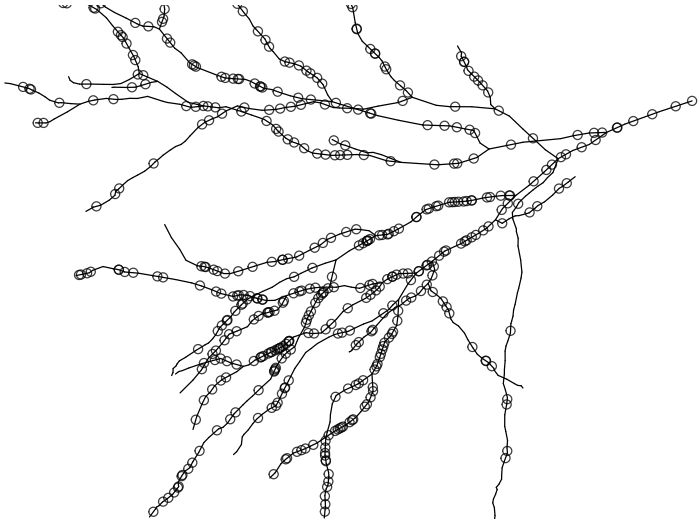

Figure 2.4: Dendritic spine data. One branch of the dendritic tree of a neuron, showing the positions of dendritic spines, of "stubby" or "mushroom" type.

2.2 Point processes on a network

In this section we describe several stochastic models for point patterns on a linear network, and discuss their potential applicability to road accident data. Models for point processes in $\mathbb{R}^n$, such as the Poisson point process, have been easily adapted to a linear network.

2.2.1 Point processes on L

A finite *point process* $\mathbf{X}$ on a linear network L is a random mechanism whose outcome is a point pattern, such that the number $N_{\mathbf{X}}(B)$ of points falling in any segment $B \subset L$ is a random variable.

The *intensity function* $\lambda(u)$ of a point process $\mathbf{X}$ is the expected number of random points per unit length, in the vicinity of location $u \in L$. Thus, for an infinitesimal segment $[u, u + \mathrm{d}u] \subset L$ of length $\mathrm{d}u$, the expected number of random points falling in the segment is $\lambda(u)\,\mathrm{d}u$. Integrating over any $B \subset L$, the expected number of random points of $\mathbf{X}$ falling in B is

$$\mu(B) = \mathbb{E}[N_{\mathbf{X}}(B)] = \int_B \lambda(u)\mathrm{d}u. \tag{2.3}$$

To exclude technical problems, we assume that the expected total number of random points in the entire point process is finite:

$$\mathbb{E}[N_{\mathbf{X}}(L)] = \int_L \lambda(u)\mathrm{d}u < \infty. \tag{2.4}$$

A version of *Campbell's formula* [43, section 13.1, p. 269] holds for linear networks:

$$\mathbb{E}\Big[\sum_{x_i \in \mathbf{X}} h(x_i)\Big] = \int_L h(u)\lambda(u)\,\mathrm{d}u, \tag{2.5}$$

for any measurable real function h on L for which the right side is absolutely integrable.

2.2.2 Homogeneous Poisson process

A point process $\mathbf{X}$ on a linear network is a *homogeneous Poisson process*, with intensity $\lambda > 0$, if it has the following properties:

(P1) for any subset $B \subseteq L$, the number $N_{\mathbf{X}}(B)$ of random points of $\mathbf{X}$ falling in B has a Poisson distribution with mean $\mu(B) = \lambda\,|B|$;

(P2) for disjoint subsets $B_1, \ldots, B_m$ the numbers of points $N_{\mathbf{X}}(B_1), \ldots, N_{\mathbf{X}}(B_m)$ are independent random variables.

These properties imply:

(P3) for any subset $B \subseteq L$, given $N_{\mathbf{X}}(B) = n$, the points of $\mathbf{X}$ in B are independent random points, uniformly distributed in B.

If a point process fails to have any one property (P1), (P2), or (P3) then it is not a homogeneous Poisson process.

These properties make it straightforward to generate simulated realisations of a homogeneous Poisson process. By property (P2), we can divide the network into its road segments and generate events on each segment independently of other road segments. For each road segment, by property (P1) we simply have to compute the length of the road segment B, calculate $\mu(B)$, and generate a realisation of a Poisson random number with mean $\mu(B)$; then by property (P3) we simply place the N points at independent, uniformly distributed locations along the road segment.

Figure 2.5 shows a simulated realisation of a homogeneous Poisson point process, on the Geelong major road network, with intensity equal to the average accident rate over three years in the Geelong accident data.

2.2.3 Inhomogeneous Poisson process

A point process $\mathbf{X}$ on a linear network is an *inhomogeneous Poisson process*, with intensity function $\lambda(u) \geq 0$, if it has property (P2) and (P1'):

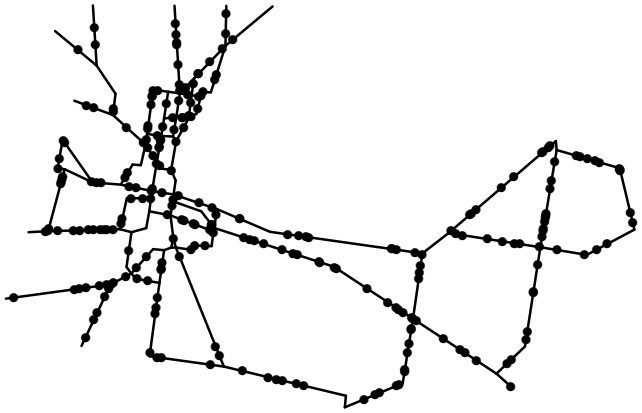

Figure 2.5: Simulated realisation of a homogeneous Poisson point process on the Geelong major road network with the same average intensity as the Geelong accident data.

(P1') for any subset $B \subset L$, the number $N_{\mathbf{X}}(B)$ of random points of $\mathbf{X}$ falling in B has a Poisson distribution with mean

$$\mu(B) = \int_B \lambda(u)\, du;$$

The properties (P2) and (P1') imply:

(P3') for any subset $B \subset L$ with $\mu(B) > 0$, if we are given that $N_{\mathbf{X}}(B) = n$, then the points of $\mathbf{X}$ in B are independent random points in B, with probability density $f(u) = \lambda(u)/\mu(B)$ for $u \in B$ and $f(u) = 0$ for $u \notin B$.

Notice that if the intensity function is constant, say $\lambda(u) \equiv \lambda$, then the stochastic model defined by properties (P1') and (P2) reduces to a homogeneous Poisson process with intensity λ, and (P3') reduces to (P3).

An important property of the Poisson process is its behaviour under *thinning*. Suppose that a homogeneous or inhomogeneous Poisson process $\mathbf{X}$, with intensity function $\lambda_{\mathbf{X}}(u)$, is *randomly thinned* by randomly deleting or retaining each point of $\mathbf{X}$. Assume that the fate of each point x in $\mathbf{X}$ is independent of the fate of other points of $\mathbf{X}$; and that the probability that a point x is retained is equal to $p(x)$, where $p(x)$ is a function with values between 0 and 1. Let $\mathbf{Y}$ be the thinned point process, consisting of all the points of $\mathbf{X}$ that are retained. Then $\mathbf{Y}$ is also a Poisson process, with intensity function $\lambda_{\mathbf{Y}}(u) = p(u)\lambda_{\mathbf{X}}(u)$.

One practical application of the thinning property is an algorithm due to Lewis and Shedler [88] for simulating an inhomogeneous Poisson process with any given intensity function $\lambda(u)$. First we find the maximum value M of the intensity over the spatial domain. We generate a

homogeneous Poisson process $\mathbf{X}$ with intensity M, using one of the techniques described above. Finally we apply random thinning to $\mathbf{X}$ with retention probability $p(u) = \lambda(u)/M$. The result is a Poisson process with intensity $p(u)M = \lambda(u)$, as required.

For the inhomogeneous Poisson process, the variance of the random sum on the RHS of equation (2.5) is given by [42], page 188

$$\mathrm{Var}\Big[\sum_{x_i \in \mathbf{X}} h(x_i)\Big] = \int_L h^2(u)\lambda(u)\,\mathrm{d}u. \tag{2.6}$$

2.2.4 Density and intensity

Suppose that events were observed to occur at the locations $x_1, \ldots, x_n$ on a linear network L. In this thesis we treat $x_1, \ldots, x_n$ as a realisation of a point process, and estimate its intensity function $\lambda(u)$, $u \in L$. Alternatively, one could treat $x_1, \ldots, x_n$ as a sample of independent random points with common probability density $f(u)$ to be estimated. These two problems are closely related, at least for Poisson processes, because of property (P3') above. The extensive literature [121, 52, 79] on density estimation can be reinterpreted for the purpose of estimating intensity on L.

2.3 Approaches to data analysis for events on L

There are different approaches to analysing point patterns on a linear network. In this thesis we use a point process approach. However, traditionally, accidents on a road network have been studied using the count data approach, and we review this methodology and contrast it with our point process approach in the next section.

2.3.1 Count regression

Statistical methods for analysing road accident risk have been the subject of intensive research for decades [97, 34, 106, 69, 112, 103, 94, 128]. Researchers in this area traditionally use crash-frequency data for their statistical analysis of road traffic accidents [94].

Lord and Mannering [94] describe crash-frequency data as, "the number of crashes occurring in some geographical space (usually a roadway segment or intersection) over some specified time period." The full data set is a list of traffic accident counts corresponding to a set of discrete road segments plus a list of covariate values also corresponding to the discrete road segments. The typical count regression approach can be summarised as follows [94]:

- The road network is broken up into convenient, optimal segments. These can be convenient for data collection purposes. They are optimal when the covariates of interest can be

considered constant over the length of the segment, thus a segment that is homogeneous as much as possible with respect to the covariates of interest.

- Choose a fixed time period over which the accident count occurs. This is often one year, or multiple years, since the accident rate may vary during different times within a year.
- Choose the range for the severity of an accident. Accidents may vary from very minor accidents to fatal accidents.
- Let the response variable be N_i , the number of accidents occurring during the fixed time period in segment i, where $i = 1, 2, \ldots, I$, and I is the total number of segments.
- Let the covariate information vector be, $\mathbf{Z}_i = (Z_{i1}, Z_{i2}, \ldots, Z_{ip})$ for covariates $(Z_{i1}, \ldots, Z_{ip})$, $i = 1, 2, \ldots, I$.
- Postulate a model by assuming that N_i is a sample drawn from a certain probability distribution, for example: Poisson, negative binomial, zero-inflated Poisson, or gamma [94]. Typically Generalised Linear Models (GLMs) are used of the form

  $$\mu = \mathrm{E}[N_i] = f(\boldsymbol{\theta}^\top \mathbf{Z}_i)$$

  is used, with $g(\mathrm{E}[N_i]) = \boldsymbol{\theta}^\top \mathbf{Z}_i$, where $g = f^{-1}$ is the link function. For example for Poisson loglinear regression, where N_i is assumed to be Poisson and

  $$\mu = \mathrm{E}[N_i] = \exp(\boldsymbol{\theta}^\top \mathbf{Z}_i). \tag{2.7}$$

- The model is fitted using Maximum Likelihood Estimation (or sometimes other methods) using standard software. (For example, the `glm` package in R.) The fitted model can be used to identify important covariates and predict accident frequency under different scenarios.

Cafiso study example

Cafiso *et al.* [34] studies injury accidents on Italian two lane rural highways. Crash-frequency was used with a negative binomial model. The main explanatory variables chosen were: average width of the road (W), the Average Annual Daily Traffic (AADT), curvature change rate (CCR), which is the sum of deflection angles per km averaged over a road segment, a summary statistic for roadside hazards (RSH), and the number of driveways per kilometre (DD).

The highway is split into segments used for the crash-frequency counts. The segments are chosen such that the covariates are approximately constant within the segment, without making the segments too short. Cafiso *et al.* [34] notes, " The final definition of a homogeneous road

section is a section where all the above-mentioned parameters (CCR, W, AADT, mean RSH) are constant." Further covariates are used, but they are derived from the basic parameters in order that they can also be considered constant on the road segments.

The model taken in the form of (2.7)

$$\mu = \exp(\boldsymbol{\theta}^\top \mathbf{Z}_i + a_1 A), \tag{2.8}$$

where μ is the expected number of injury accidents on the road segment i in a year, $\boldsymbol{\theta}^\top$ and a_1 are the model parameters, $\mathbf{Z}_i$ are covariate values for the road segment, and $A = \log(L_{seg}.\text{ADDT})$, is an accident exposure variable, where L_{seg} is the length of the road segment and ADDT is the Average Annual Daily Traffic on the segment.

The covariates chosen to include in the final three models of Cafiso *et al.* [34] are based on eliminating covariates which are too closely correlated with each other, and on the goodness of fit of the models. For example, their Model 19 includes the covariates, (L_{seg}), AADT, DD, RSH, curvature ratio (CR), which is the sum of the length of horizontal curves divided by the length of the road segment (this is different from CCR mentioned above), and operating speed differentials density (ΔV_{10}) (uses the road geometry to predict the operating speed of vehicles and then calculates how often per kilometre that speed will change by more than 10km/hr). Their fitted model 19 is

$$\mu = \exp(-7.812 + 0.67\text{DD} - 1.948\text{CR} + 0.0872\Delta V_{10} + 0.185\text{RSH} + 0.753(\log(L_{seg}.\text{ADDT}))). \tag{2.9}$$

In this study, the mean length of the road segments is 1.57 km with a range of 0.5 km to 4.29 km.

2.3.2 Critique of count regression

The response variable and the covariates for crash-frequency are tied to the division of the road into segments, and the covariates used for a road segment tend to be summary statistics of continuous variables. Cafiso *et al's* [34] paper uses and refers to many examples: the sum of the change rate of horizontal curvature, the sum of the change rate of vertical curvature, the maximum horizontal curvature rate, average sight distance, driveway density, standard deviation of operating speed profile, and operating speed differentials density.

Crash-frequency methods effectively assume that the accident risk is constant along each road segment. The analysis is required to manipulate the data and road segment choice to comply with this assumption, an assumption that may not be valid. Also, the possible explanatory covariates for crash-frequency counts must each have a constant value over each corresponding road segment.

While there may be some covariates that are constant over a road segment, most covariates of interest will naturally vary along a road segment. Sight distance is a good example. At any

location, the sight distance is the length of road that is visible to a driver at that location. If there is a true relationship between accident rate and site distance, then using average sight distance causes a loss of information and a model misspecification. The loss of information could occur, for example, if a one kilometre road segment has very good (long) sight distance, but contains 50 metres with a very short sight distance; the information about this critical 50 metres is lost amongst the other 950 metres. The misspecification bias is analysed in Section 3.5 in Chapter 3 by using Jensen's inequality.

Aggregation of covariates

The aggregation of explanatory covariates into single values for each road segment in the crash-frequency approach means that there will be covariates that cannot be used because it may be difficult to reduce them to a single aggregated value. As noted above, the aggregation of sight distance over a road segment is problematic. Consider also the distance to the nearest intersection. It is clearly not amenable to aggregation and not usable for crash-frequency data methods. However, we use this variable in some of our point process modelling examples in this thesis.

The road segments used for a crash-frequency analysis must be carefully chosen so that the compromises due to the aggregation of data are minimised. This is typically done by trying to find road segments that are close to being internally homogeneous [34]. However, it may be the road locations where the covariates have the highest variability that are the most interesting sites. In crash-frequency data, these high-variability areas will be either only at the start and end of road segments. Anastasopoulos and Mannering [5] note that these sites, "may be associated with accident clustering".

Baddeley et al. [10] consider a similar problem of aggregation in two dimensions used amongst geologists. The geological event data is aggregated by dividing an area into equal sized "pixels". They note"...counting points inside pixels is a special case of aggregating spatial data into discrete geographical areas." They note that "...it may be impossible to reconcile two spatial logistic regression models that were fitted to the same spatial point pattern using different pixel grids". An analogous argument can be used in one dimension, replacing pixel grids with discrete road segments.

Problems associated with aggregation of data have historically been described as "the ecological fallacy". Freedman states, "The ecological fallacy consists in thinking that relationships observed for groups necessarily hold for individuals" [61]. In the crash-frequency case, the individuals are accidents at point locations and the groups are the accidents on a road segment.

Spatial aggregation of data is analogous to the aggregation of data over time. The problems with aggregation over time are noted in the crash-frequency literature [94]. Lord and Mannering

[94] discuss the possible problem of aggregating precipitation data over time, "The distribution of precipitation over the month (by hour or even minute) is likely to be highly influential in generating crashes, but generally the analyst only has precipitation data that is much more aggregated and thus important information is lost by using discrete time intervals — with larger intervals resulting in more information loss. This can introduce error in model estimation as a result of unobserved heterogeneity". Spatial aggregation of data will be subject to the same issues.

Model types available to each approach

An expert in the use of crash-frequency data models may be tempted to make the argument that crash-frequency models use a large array of statistical extensions such as negative binomial models, multivariate models, random parameter models, etc. However, the bias due to aggregation is a fundamental problem, which will not be corrected by modifying the dependence structure of the model.

In Section 3.5 we also discuss problems due to the aggregation of data bias in the crash-frequency method.

2.3.3 Point process approach

Statistical methods for analysing spatial point patterns are not new. They have been applied to data in biology, epidemiology, geography, biomedicine, cosmology, archaeology, geology, and seismology [40, 51, 11]. These applications are usually for data in two dimensions. Early analysis of spatial point pattern data in ecology and forestry concentrated on area counts of events. For example, researchers would divide a forest into squares, known as quadrats, and count the number of a certain tree species inside each quadrat. However, from the late 1970's, methods of analysis have developed beyond these aggregated count methods to mathematically well-defined point process models such as likelihood-based models that use the exact location of each event (or tree in the example) [40, 51, 11]. In Chapter 3 we modify and adapt the existing two-dimensional spatial point process methods for use on a road network space.

The spatial point process method does not break up the road into discrete segments, rather it treats the road as a continuous linear space. Each data observation is the exact spatial location of a single traffic accident. There are no aggregate accident counts in spatial point process data. The data set is the *spatial point pattern* created by the locations of traffic accidents on the linear network. For example, Figure 1.1 shows the spatial point pattern of all high-severity traffic accidents on state declared roads in Geelong (Victoria, Australia) between 2009 and 2011. The roads are represented as a line network and the accidents are represented by filled dots.

The spatial point process approach allows the accident risk to continuously vary with respect to location on the road network. Thus, returning to the Cafiso *et al.* [34] study, any of the curvature summary statistic covariates can be replaced by the actual road curvature at a location, or the instantaneous rate of change of road curvature at that location. Similarly, for predicted operating speed, the covariate can be the actual predicted operating speed at a location, or the instantaneous rate of change of the predicted operating speed at that location.

The spatial point process approach allows the explanatory covariates to vary with respect to location over the whole of the road network. The practical advantage of this is brought into focus when considering the development of the Interactive Highway Safety Design Model and the use of road geometry design parameters [82]. Road geometry parameters are measured as continuously varying on existing roads. Then this data is aggregated over road segments for use in crash-frequency models. The results of the crash-frequency models are used for the prediction of accidents on proposed roads that are in the design stage. The road being designed has continuously varying geometry parameters. However, these parameters are then aggregated before being used to predict accidents by the Interactive Highway Safety Design Model. The aggregation on both sides of this exercise is unnecessary if spatial point process methods are used. Using spatial point process methods the road is analysed directly using the same covariates that the road designer uses.

The advantages of the spatial point process method compared to the crash-frequency method are discussed further in Chapter 3.

2.4 Kernel intensity estimation: basics

Kernel estimation of intensity is an important tool for investigating spatial point pattern data and is well-developed for two-dimensional data [50, 22]. It would be highly desirable to have similar methods available for data on a linear network. Prior to our paper [100], described in Chapter 4, there had been several attempts to develop kernel intensity/density estimation on a linear on a linear network ([24, 25, 26, 55, 56, 57, 133, 109, 124]; [110, Chap. 9]). These prior methods were developed ad hoc, were somewhat unsatisfactory, and there was no consensus about the best method. Okabe & Sugihara [110, p. 183] noted that "there appears to be no systematic method available for finding a continuous kernel density function on a network, and so we commonly employ heuristics". These methods are reviewed in this section. We start by recalling fundamentals of kernel density estimation in one dimension, then formulate the problem of intensity estimation on a linear network.

2.4.1 Kernel smoothing on a line

Kernel density estimation methods on the real line are well known [121, 132]. Consider data points on the real line, for example the resting heart rates of n patients. The solid circles in Figure 2.6 could represent those data points, x_i. The task is to estimate the probability density $f(x)$ of resting heart rate in the population, assuming the n observations are an independent random sample from the population.

Consider a kernel function $k(\cdot)$ on $\mathbb{R}$, which satisfies $k(u) \geq 0$, for all u, and $\int k(u)\,\mathrm{d}u = 1$. That is, k is a probability density function on $\mathbb{R}$. For ease of discussion, we will choose the kernel function to be the Gaussian probability density, with mean zero and standard deviation σ.

To form a kernel density estimate, a copy of the kernel is shifted to each data location x_i, as sketched in Figure 2.6. At any query location u, the average of the contributions from all data points is the density estimate

$$\widehat{f}(u) = \frac{1}{n}\sum_{i=1}^{n} k(u - x_i). \tag{2.10}$$

Alternatively, the points x_i, $i = 1, 2, \ldots, n$, can be considered as a realisation of a point process with intensity $\lambda(.)$ and, using the kernel, the intensity can be estimated as

$$\widehat{\lambda}(u) = \sum_{i=1}^{n} k(u - x_i), \tag{2.11}$$

as indicated by the dashed line in Figure 2.6.

Note that the estimated intensity depends on the choice of kernel and the bandwidth σ, the standard deviation of the kernel density. However, it is well known that the choice of bandwidth is far more important to the final result than the choice of kernel type [121, 132].

2.4.2 Kernel intensity problem statement

A good kernel intensity estimation method should have certain desirable properties. It should be statistically principled, computationally fast, and there should be an associated bandwidth selection procedure. We will now list the required statistical properties of an intensity estimator as first discussed by Okabe & Sugihara [110]

Let us consider kernel density estimators of the general form

$$\widehat{f}(u) = \frac{1}{n}\sum_{i=1}^{n} K(u \mid x_i) \tag{2.12}$$

and, equivalently, kernel intensity estimator

$$\widehat{\lambda}(u) = \sum_{i=1}^{n} K(u \mid x_i), \tag{2.13}$$

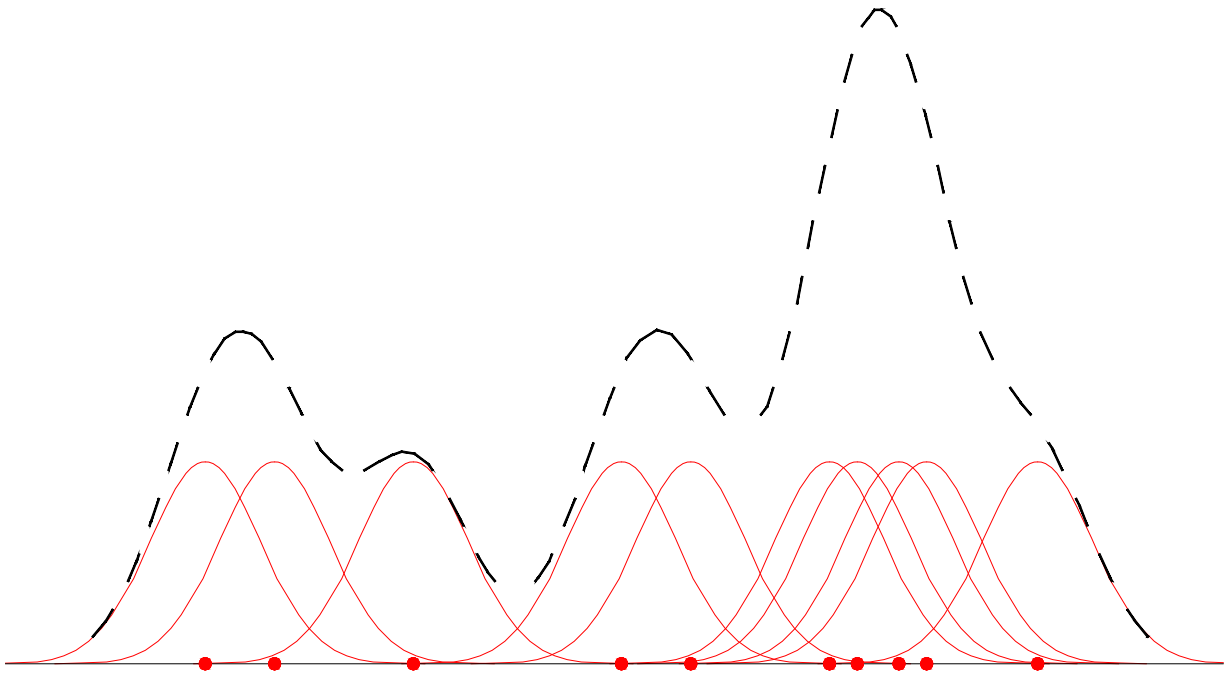

Figure 2.6: Illustration of kernel density estimate of $\lambda(\cdot)$ Solid circles denote the locations of the point pattern; the solid curves are the Gaussian kernel shifted to each data point location; and the dashed line is the kernel intensity estimate.

where $K(\cdot \mid x)$ is a kernel (*yet to be defined*) on the linear network for a source point at location x, and $x_1, \ldots, x_n$ are the data points. As before, the basic requirement is that $K(\cdot \mid x)$ should be a probability density on L, for any $x \in L$. That is, $K(\cdot \mid \cdot) \geq 0$, and

$$\int_L K(u \mid x)\, \mathrm{d}u = 1, \qquad \text{for all } x \in L. \tag{2.14}$$

If (2.14) holds, then the intensity estimator (2.13) satisfies

$$\int_L \widehat{\lambda}(u)\, \mathrm{d}u = n.$$

Lemma 2.1. *Let* $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$ *be a realisation of a point process* $\mathbf{X}$ *in* L *with intensity* $\lambda(\cdot)$*, and*

$$\widehat{\lambda}(u) = \sum_{i=1}^{n} K(u \mid x_i). \tag{2.15}$$

Then, the following holds:

(a)

$$\mathbb{E}\widehat{\lambda}(u) = \int_L K(u \mid x)\lambda(x)\, \mathrm{d}x. \tag{2.16}$$

(b) In particular, for uniform intensity, $\lambda(x) \equiv \lambda$,

$$\mathbb{E}\widehat{\lambda}(u) = \lambda \int_L K(u \mid x)\, \mathrm{d}x. \tag{2.17}$$

(c) For uniform intensity, $\widehat{\lambda}(u)$ is unbiased if and only if

$$\int_L K(u \mid x)\, \mathrm{d}x = 1, \text{ for all } u \text{ in } L. \tag{2.18}$$

(d) A sufficient condition, but not necessary condition for (2.18) *is symmetry:*

$$K(x \mid u) = K(u \mid x). \tag{2.19}$$

Proof. Proof of (a) follows by Campbell's formula (2.5) with $h(x_i) = K(u \mid x_i)$, and (b) follows by substituting λ for $\lambda(x)$ into (2.16). From (2.17) it follows that $\mathbb{E}[\widehat{\lambda}(u)] = \lambda$ iff $\int_L K(u \mid x)\, \mathrm{d}x = 1$, which proves (c).

If (2.19) holds, then the LHS of (2.18) $= \int_L K(x \mid u)\, \mathrm{d}x = 1$, by (2.14), which completes the proof of the Lemma.

□

2.4.3 Naive approach intensity estimators

One natural first impulse for choosing how to do kernel intensity estimation on a linear network would be to take the well known one-dimensional line methods and apply them directly to a linear network.

However, this approach is incorrect as it leads to fallacious results. Recall that, on a line, the standard kernel density estimator is

$$\widehat{f}(u) = \frac{1}{n} \sum_{i=1}^{n} k(u - x_i), \quad u \in \mathbb{R}, \tag{2.20}$$

where k is a smoothing kernel (probability density) on $\mathbb{R}$, and $x_1, \ldots, x_n \in \mathbb{R}$ are the observations.

Xie and Yan [133] proposed to extend (2.20) directly to a linear network L in the form

$$\widehat{f}(u) = \frac{1}{n} \sum_{i=1}^{n} k(d_L(u, x_i)), \quad u \in L, \tag{2.21}$$

where $d_L(u, x_i)$ is the shortest-path distance in the network from u to x_i, and k is a smoothing kernel *on the real line* located at x_i, and measured at u.

This assumes implicitly that the kernel density estimation procedures will remain valid if they are transplanted from the real line to a linear network. However, this is a fallacy. Okabe

et al ([109, 124]; [110, p. 180]) showed that (2.21) violates the conservation of mass in the sense of (2.14). Thus (2.21) is not a valid probability density.

To see this, note that (2.21) is of the general form (2.12) with

$$K(u \mid x) = k(d_L(u, x)). \tag{2.22}$$

The total mass of the kernel K is

$$\int_L K(u \mid x)\, \mathrm{d}u = \int_L k(d_L(u, x))\, \mathrm{d}u, \tag{2.23}$$

which may be substantially different from 1.

Figure 2.7: Naive estimate of intensity. *Left:* a data point (•) on a network containing a fork. *Right:* the naive kernel estimate (2.21). The tail of the kernel k is effectively duplicated onto each of the outgoing segments of the fork, increasing the total mass of $\hat{f}$. Mass is also lost due to truncation at terminal endpoints.

An example is shown in Figure 2.7. The network has three segments meeting at a vertex, and there is a single data point x. The density estimate is $\widehat{f}(u) = k(d_L(u, x))$. Figure 2.7 depicts $\widehat{f}(u)$ in the style of Okabe and Sugihara [110] as a perspective view, using height to represent the function value. The effect of (2.21) is that each line segment of the network carries a copy of the graph of the kernel k restricted to an interval $[a, b]$ of the real line. The segment containing x corresponds to the interval $[-A, B]$, where A, B are the distances from x to each end of the segment. The other two segments correspond to intervals of the form $[B, C]$, where $B, C > 0$ are the shortest-path distances from x to each endpoint of the segment. Part of the kernel has been duplicated on two line segments, increasing the total mass of the kernel. It is in this sense that the total mass "typically" exceeds 1. By a transformation of coordinates as in (2.2),

$$\int_L k(d_L(u, x_i))\, \mathrm{d}_1 u = \int_0^\infty k(t) m(x_i, t)\, \mathrm{d}t, \tag{2.24}$$

where $m(x, t)$ is the number of locations $u \in L$ such that $d_L(x, u) = t$. Typically $m(x, t) \geq 1$, hence the integral in (2.24) is typically greater than 1. However, the total mass of the kernel in

Figure 2.7 can also be *decreased* by truncation at a terminal vertex. An example is visible at the left hand side of Figure 2.7, where the left tail of the kernel is truncated. Truncation occurs if $m(x,t) = 0$, for some $t < h$, or equivalently, if the support of the kernel extends further than the distance from x to a terminal vertex.

Several modifications of the naive estimator have been proposed in order to reduce its severe bias ([25, 26]; [109, Sec. 3]). We shall look at the most popular of these in the next section.

2.4.4 The Okabe Group approach

In a systematic study, Okabe and Sugihara [109] consider several corrected versions of (2.21) designed to satisfy all the desiderata listed in Section 2.4.2. Their initial estimator attempts can be recast in the form of (2.12) with kernel

$$K(u \mid x) = c(x,u)\, k(d_L(x,u)), \tag{2.25}$$

where k is a smoothing kernel on the real line, and $c(x,u)$ is a piece-wise constant function of each argument, with jumps occurring only at vertices. The "base kernel" $k(x)$ is assumed [110, p. 180 with corrections] to be a continuous probability density on the real line, symmetric about zero ($k(x) = k(-x)$), non-increasing for $x > 0$, and with bounded support: $k(x) = 0$, when $|x| \geq h$, where $h > 0$ is the *half-width*. Note that h is termed the "bandwidth" in [109], [124] and [110], but we shall follow the convention in which the kernel's bandwidth is its standard deviation.

The challenge is to choose the coefficient function $c(u,x)$ so that the linear network kernel (2.25) has unit mass; i.e., it must satisfy (2.14). It is also *desirable* that the linear network kernel should also satisfy (2.18), so that the estimator (2.13) is unbiased when the density is uniform.

2.4.5 Equal-split discontinuous estimator

Okabe *et al.* [109, sec. 4] introduce the "equal-split discontinuous" kernel density estimator as an algorithm for assigning kernel values to each part of the network in such a way as to satisfy (2.14). The algorithm is detailed in [110, Algorithms 9.1–9.2, Sec. 9.3.2] and an equivalent version is stated as Algorithm 2.1.

Algorithm 2.1 (Equal-split Discontinuous Estimate)**.** Given a linear network L, a point pattern $\mathbf{x}$ on L, and a symmetric monotone kernel k on the real line with half-width h,

1. Create an empty list Q called the "queue".
2. For each data point x which is not a vertex of the network,
   (a) Identify the network segment s_x containing x.
   (b) Assign the kernel value for all $u \in s_x$ to be $K^{\mathrm{D}}(u \mid x) = k(d_L(u, x))$. See the left panel of Figure 2.8.
   (c) Place the entries (x, v_1, d_1, w_1) and (x, v_2, d_2, w_2) at the end of the queue, where v_1, v_2 are the endpoints of the segment, $d_j = d_L(x, v_j)$ are the distances to these endpoints, and $w_1 = w_2 = 1$ are weights.
3. For each data point which is a vertex v,
   (a) Determine the vertex degree $m = deg(v)$.
   (b) For each segment $[v, v']$ emanating from v, place the entry (v, v', d, w) at the end of the queue, where $d = d_L(v, v')$ and $w = 2/m$.
4. While the queue Q is not empty,
   (a) Remove the first element (z, v, d, w) from the queue.
   (b) Determine the vertex degree $m = deg(v)$ of the vertex v.
   (c) Find all segments $s = [v, v']$ which meet at v and which do not contain the previously-visited vertex z, that is $z \notin [v, v']$.
   (d) Split the kernel tail equally over these $m - 1$ extension segments (see the right panel of Figure 2.8): for each extension segment s,
      i. Assign the kernel value for all $u \in s$ to be $K^{\mathrm{D}}(u \mid x) = (w/(m-1))k(d_L(u, v) + d)$.
      ii. Update $w' = w/(m-1)$ and $d' = d + d(v, v')$.
      iii. If $d' < h$, place the element (v, v', d', w') at the end of the queue.

Effectively, Algorithm 2.1 enumerates all paths in the network starting at x, of length less than or equal to h, which do not reflect at a vertex. Viewing the collection of all such paths as a tree, the algorithm enumerates the tree "breadth-first". An alternative would be "depth-first" enumeration, which would be obtained if the list Q were a stack (first-in-last-out) rather than

a queue (first-in-first-out).

In heuristic terms, the algorithm propagates a unit mass from the starting point x along the network L. For locations on the same line segment as the starting point x, the new kernel is equivalent to the plug-in kernel $k(d_L(x, \cdot))$. At each fork in the network, the kernel's remaining tail mass is equally divided over the outgoing line segments, as shown in Figure 2.8, so that the total remaining mass is preserved. Essentially the same rationale was proposed independently by Ver Hoef and Peterson [129] for non-parametric regression on a network of rivers or streams.

2.4.6 Properties of the equal-split discontinuous estimator

The value of the equal-split discontinuous kernel $K^{\mathrm{D}}(u \mid x)$, at location u, due to a data point at x can be represented as follows. For simplicity, assume the network has no cycles, so that there is a unique shortest path between any given pair of points. If x is not a vertex,

$$K^{\mathrm{D}}(u \mid x) = \frac{k(d_L(x,u))}{(m_1 - 1)(m_2 - 1)\cdots(m_P - 1)}, \tag{2.26}$$

where $d_L(x,u)$ is the shortest path distance from x to u, and $m_1, m_2, \ldots, m_P$ are the degrees of each vertex (excluding u and x) along the shortest path from x to u. If x is a vertex,

$$K^{\mathrm{D}}(u \mid x) = \frac{2k(d_L(x,u))}{m_0(m_1 - 1)(m_2 - 1)\cdots(m_P - 1)}, \tag{2.27}$$

where m_0 is the degree of the vertex x, and $m_1, \ldots, m_P$ are the degrees of the subsequent vertices along the shortest path. It is sufficient to consider only Equation (2.27), since any point y on L, which is not a vertex of L, can be made into a vertex by breaking the line segment that contains y. The new vertex at y has degree $m_0 = 2$, so that (2.27) reduces to (2.26). This also shows that the equal-split discontinuous estimator depends only on the geometry of the network and not on the particular choice of graph representation of the network. Since the kernel has bounded support, of width $2h$, it is sufficient to assume the network has no cycles of length shorter than $2h$, since this implies there is a unique shortest path between any two given points u and x with $d_L(u,x) < h$. Referring to (2.26), we notice that the equal-split discontinuous kernel is symmetric, $K^{\mathrm{D}}(x \mid y) = K^{\mathrm{D}}(y \mid x)$.

In the general case, where the network has cycles, Sugihara *et al.* [124] show that the kernel can be represented as a sum over all possible paths.

Theorem 2.1 ([124])**.** Assuming x is not a vertex,

$$K^{\mathrm{D}}(u \mid x) = \sum_{\pi}^{*} k(\ell(\pi)) a^{D}(\pi), \tag{2.28}$$

where the asterisk indicates that the sum is over all paths $\pi = (x, v_1, \ldots, v_{P-1}, u)$ of length less than or equal to h, from x to u **that are non-reflecting** (i.e., $e_{i+1} \neq e_i$, where, e_i is the edge

containing v_{i-1} and v_i), and $\ell(\pi)$ is the length of the path, and $a^D(\pi) = 1/((m_1 - 1)(m_2 - 1)\dots(m_{P-1} - 1))$, where m_i is the degree of v_i.

Note that the right hand side of (2.28) is the sum of a finite number of terms, because of the constraint that path length be shorter than h.

Also note that equation 2.28 can cover both the vertex and non-vertex cases by using $a^D(\pi) = 2/(m_0(m_1 - 1)(m_2 - 1)\dots(m_{P-1} - 1))$, where m_0 is the degree of the vertex if x is a vertex and $m_0 = 2$ if x is not a vertex, since non-vertex points can be notionally considered as a vertex point of degree two as explained above.

2.4.7 Computation of equal-split discontinuous estimator

Sugihara *et al.* [124, p. 830] state that computation time for the equal-split discontinuous estimator does not depend greatly on bandwidth, for a given network. We suspect that this holds only for relatively short bandwidths. The algorithm effectively traverses every path in the network, of length at most h, starting from each data point x_i, excluding paths which back-track at a vertex. At each vertex of degree $m > 1$, an incoming path will be replaced by $m - 1$ outgoing paths. As h increases, the number of paths and the computation time should increase roughly exponentially.

We implemented Algorithm 2.1 in the R language [114]. Table 2.1 in Section 2.4.9 shows computation times for the Geelong accident data using the equal-split discontinuous method.

2.4.8 Equal-split continuous kernel estimator

Okabe *et al.* [109, sec. 5] and [110, sec. 9.2.3] introduce the "equal-split continuous" kernel estimator $K^{\mathrm{C}}(u \mid x)$, at location u, due to a data point at x as a modification of the previous algorithm that is designed to produce a continuous function on the network and to improve statistical performance.

The modified algorithm traverses all paths of length less than h, now including paths that reflect at a vertex. When a path reaches a vertex of degree m, there are $m-1$ outgoing branches and one incoming branch which the path has just traversed. Suppose that each outgoing branch receives a weighted copy of the tail with equal weight a, and the incoming branch receives a weighted copy of the tail with weight b. In order that mass be conserved, $(m - 1)a + b = 1$. The kernel is continuous if $a = 1 + b$. The unique solution is $a = 2/m$ and $b = 2/m - 1 < 0$. Figure 2.9 sketches this modification. The resulting function has non-negative values because we assumed k is monotone.

The algorithm is formally described in [110, Algorithm 9.3, Sec. 9.3.3] and an equivalent version is given in Algorithm 2.2.

Algorithm 2.2 (Equal-split Continuous Estimate)**.** Execute Algorithm 2.1 with the following modification: Steps 4c–4d are replaced by

4. (c) Find all segments $s = [v, v']$ which meet at v.

   (d) For each extension segment s,

   i. Update $w' = w(2/m - \delta)$, where $\delta = 1$ if the segment s contains the previously-visited vertex z, and otherwise $\delta = 0$.

   ii. Increment the kernel value $K^{\mathrm{C}}(u \mid x)$ for all $u \in s$ by $w'\, f(d_L(u, v) + d)$. See the right panel of Figure 2.9.

   iii. Update $d' = d + d_L(v, v')$. If $d' < h$, push the element (v, v', d', w') onto the stack.

The result of this algorithm has the following representation in terms of paths, which is effectively, but not explicitly, given in Okabe *et al.* [109] and [110].

Theorem 2.2. Assuming x is not a vertex,

$$K^{\mathrm{C}}(u \mid x) = \sum_{\pi} k(\ell(\pi)) a^{C}(\pi), \tag{2.29}$$

where the sum is now over *all* paths from x to u, and $a^C(\pi) = c_1 \cdots c_{P-1}$, where $c_j = \frac{2}{deg(v_j)} - \delta_j$, in which $\delta_1 = \delta_P = 0$ and $\delta_j = \mathbf{1}\{e_j = e_{j-1}\}$ is the indicator that equals 1, if the path reverses at step j, and 0 otherwise.

This kernel does have unit mass (2.14) and satisfies (2.18) so that the estimator is unbiased for the uniform density. A direct, constructive proof is given in Okabe *et al.* [109].

2.4.9 Computation of equal-split continuous estimate

Computation of the equal-split continuous estimate is very slow when h is not small. At each vertex of degree $m > 1$, each incoming path will be replaced by m outgoing paths. If $m > 2$, then a tail of the density k, of total probability mass p, will be replaced by $(m-1)$ paths each with tail mass $2p/m$ and one path with negative tail mass $(2/m-1)p$, with total signed mass p, but total absolute mass $(2(m-1)/m+1-2/m)p = (3-4/m)p$. Thus, the length and absolute mass of the stack can grow exponentially, even in simple networks.

The fourth and fifth rows of Table 2.1 show average computation times for smoothing the Geelong accident data using the equal-split continuous method with, respectively, an Epanechnikov kernel and a Gaussian kernel, again using R code. The computational cost increases exponentially with bandwidth, at a much faster rate than for the equal-split discontinuous

method because we now have the reflected paths to also compute. Okabe and Sugihara [110, pp. 182,193] recommend the discontinuous method if speed is important.

Table 2.1 below shows computation times for the Geelong accident data using the equal-split discontinuous method (Algorithm 2.1, p28) and the equal-split continuous method (Algorithm 2.2, p31) with, respectively, the Epanechnikov kernel $k(x) = (3/(4h)) \max(0, 1 - x/h)^2$ with bandwidth $\sigma = h/\sqrt{5}$, and the Gaussian kernel truncated at $\pm 5\sigma$ to satisfy the requirement of compact support. Regardless of the efficiency of the implementation, Table 2.1 suggests that the computational complexity increases exponentially with bandwidth. A missing value indicates that the algorithm did not terminate within a working day.

	σ (metres)									
Method; Kernel	50	100	250	500	750	1000	1250	1500	1750	2000
Discontinuous, Epanechnikov	0.4	0.5	0.9	1.6	2.6	4.3	6.4	9.6	14	21
Discontinuous, Gaussian	0.5	0.8	1.6	4.2	8.5	16.7	32	63	120	230
Continuous, Epanechnikov	0.6	0.9	2.4	8.6	28	101	505	3150	–	–
Continuous, Gaussian	0.6	1.0	2.7	9.6	31	107	520	3220	–	–

Table 2.1: Computation times (in seconds) for Okabe group kernel estimates of accident intensity in Geelong.

2.4.10 Critique of the equal-split discontinuous estimator

Okabe *et al.*, Okabe and Sugihara, and Sugihara *et al.* [109], [124, p. 829] and [110, p. 182] assert that the equal-split discontinuous kernel estimator (2.26)–(2.27) is unbiased when the density is uniform. However, this is not quite true:

Lemma 2.2. *Assume L has no cycles of length less than $2h$. Let $V_1 = \{v \in V : deg(v) = 1\}$ be the set of terminal vertices. Then the equal-split discontinuous estimator* (2.26)–(2.27) *satisfies* (2.14) *if and only if $x \in L_{\ominus h}$, where $L_{\ominus h} = \{u \in L : d_L(u, v) > h \text{ for all } v \in V_1\}$. That is, the kernel has mass 1 if and only if the centre of the kernel lies more than h units away from any terminal vertex. Similarly, the unbiasedness property* (2.18) *holds if and only if $u \in L_{\ominus h}$.*

The proof is another application of the 'polar' transformation (2.2).

Proof. Let $x \in L$ be a fixed point, which is not a vertex.

Using the transform (2.2) we write

$$\int_L K^{\mathrm{D}}(u \mid x)\,\mathrm{d}u = \int_0^\infty \sum_{u\in L:d_L(u,x)=t} K^{\mathrm{D}}(u \mid x)\,\mathrm{d}t \tag{2.30}$$

In the representation (2.28) of $K^{\mathrm{D}}(u \mid x)$ we note that, since $K(t) = 0$ for $t \geqslant h$, the sum in (2.28) can be restricted to paths π which are of length $\ell(\pi) < h$.

However, since L is assumed to have no cycles of length less than $2h$, any non-reflected path from x to u of length less than h is unique.

Substituting in (2.30),

$$\int_L K^{\mathrm{D}}(u \mid x)\,\mathrm{d}u = \int_0^h \sum_{u\in L:d_L(u,x)=t} \frac{k(t)}{(m_1-1)(m_2-1)\cdots(m_P-1)}\,\mathrm{d}t \tag{2.31}$$

where $m_1, m_2, \ldots m_P$ are the degrees of the vertices $v_1, v_2, \ldots v_P$ along the unique non-reflecting path x to u.

This reduces to

$$\int_0^h k(t) \sum_{u\in L:d_L(u,x)=t} \frac{1}{(m_1-1)(m_2-1)\cdots(m_P-1)}\,\mathrm{d}t. \tag{2.32}$$

A simple tree argument shows that the sum is equal to 1, yielding the result. □

Intuitively, as Figure 2.8 illustrates, at any terminal vertex the kernel will be truncated and mass will be lost. Thus, the equal-split discontinuous kernel estimator (2.26)–(2.27) is unbiased (for the uniform intensity) only when $u \in L_{\ominus h}$. This could be considered as an "edge effect".

2.5 Bandwidth selection for density and intensity estimation

In kernel estimation applications, an important task is to choose an appropriate bandwidth for the kernel. In this section we will discuss how this is done for data in $\mathbb{R}$ and $\mathbb{R}^2$.

Techniques for selecting the smoothing bandwidth h for real-valued data are surveyed in [121, 132, 76, 93] and density estimation in d dimensions is the focus of the book by Scott [120]. Additional techniques for bandwidth selection for density/intensity estimation for spatial point patterns for data in $\mathbb{R}^2$ were developed by Berman and Diggle [21].

The goal of kernel density or intensity estimation is to choose the best estimate of the density or intensity of the process that produced the data that is being smoothed. Thus, it follows that the goal of bandwidth selection is to choose the bandwidth that will produce the best estimator. As stated previously, it is well known [121, 132] that the choice of bandwidth is far more important to the final result than the choice of kernel type.

For the case of a density estimator, the mean integrated square error (MISE), first proposed by Rosenblatt [119], is generally used as the measure of the global accuracy of a kernel estimator [121]:

$$\text{MISE} = E \int \{\widehat{f}_h(x) - f(x)\}^2 \mathrm{d}x, \tag{2.33}$$

where $\widehat{f}_h(x)$ is the kernel estimator of $f(x)$ using the bandwidth h.

The MISE can be used to produce an expression for a theoretical optimal bandwidth selection. However, this expression will depend on $f(x)$, which is unknown [121].

There are a few methods to work around this difficulty, including rules of thumb and cross-validation methods. Rules of thumb only depend on summary statistics rather than the data in detail. Cross-validation methods work around the problem of not knowing the true density by using the data itself to find a bandwidth that minimises an estimate of the integrated square error (ISE):

$$\text{ISE} = \int \{\widehat{f}(x) - f(x)\}^2 \mathrm{d}x.$$

Bandwidth selection methods are easily adaptable between data in $\mathbb{R}$ and $\mathbb{R}^2$ [121]. However, adapting these methods to data on a linear network is not so straightforward. This will be explained in Chapters 4 and 5.

2.5.1 Rules of thumb

For kernel estimation in one dimension, the MISE given in (2.33) can be expressed as the sum of the integrated squared bias and the integrated variance of the estimate. Working with densities:

$$\text{MISE} = \int \{E\,[\widehat{f}(x)] - f(x)\}^2\,\mathrm{d}x + \int \operatorname{var}\widehat{f}(x)\,\mathrm{d}x.$$

Silverman [121, Equations 3.17 and 3.19] presented approximations using Taylor expansions for the above terms as

$$\int \text{bias}_h(x)^2\mathrm{d}x = \int \{E\,[\widehat{f}(x)] - f(x)\}^2\,\mathrm{d}x \approx \frac{1}{4}h^4k_2^2\int f''(x)^2\,\mathrm{d}x;$$

$$\int \operatorname{var}\widehat{f}(x)\,\mathrm{d}x \approx n^{-1}h^{-1}\int K(t)^2\,\mathrm{d}t,$$

where K is a symmetric function with $\int K(t)\,\mathrm{d}t = 1$, with mean zero, and variance k_2, where h is the kernel bandwidth (the standard deviation for a Gaussian kernel).

Parzen [113, Lemma 4A] has shown using simple calculus that the optimal h that minimises asymptotic MISE is

$$h_{\text{opt}} = k_2^{-2/5}\Big\{\int K(t)^2\mathrm{d}t\Big\}^{1/5}\Big\{\int f''(x)^2\,\mathrm{d}x\Big\}^{-1/5}n^{-1/5}.$$

If $f(x)$ is a normal density, then $\int f''(x)^2\,\mathrm{d}x = \frac{3}{8}\pi^{-1/2}\sigma^{-5}$. Under the assumption that $f(x)$ is a normal density and that the kernel K is Gaussian, Silverman [121] showed that

$$h_{\text{opt}} = (4\pi)^{-1/10}\,(\frac{3}{8}\pi^{-1/2})^{-1/5}\,\sigma n^{-1/5} \tag{2.34}$$

$$= (\frac{4}{3})^{1/5}\,\sigma n^{-1/5}. \tag{2.35}$$

However, σ needs to be estimated from the data to obtain h_{opt}. For d-dimensional data, following the above method, Silverman derived an optimal bandwidth $h_{\text{opt}}^{[1]} = (h_1^{[1]}, \ldots, h_d^{[1]})$ for d-dimensional density estimation. It is given by

$$h_i^{[1]} = (\frac{4}{d+2})^{1/(d+4)}n^{-1/(d+4)}s_i,\ \text{for } i = 1, 2, \ldots, d, \tag{2.36}$$

where s_i is the sample standard deviation of the ith coordinate values. This is often known as Silverman's rule of thumb [121, eq. (3.31), p. 48].

Taking $(\frac{4}{d+2})^{1/(d+4)} \approx 1$, in (2.36) gives the optimal bandwidth for Scott's rule of thumb [120, p. 152] as

$$h_i^{[2]} = n^{-1/(d+4)}\ s_i,\ \text{for } i = 1, 2, \ldots, d. \tag{2.37}$$

Note that (2.36) and (2.37) are equal for $d = 2$.

2.5.2 Likelihood cross-validation

Cross-validation methods test the accuracy of a model for data by dividing the data into two subsets and using one subset of the data to estimate the model, and the other subset to check how good that estimated model is, or in other words, validate the model.

Here we produce a similar explanation to Silverman [121], but with reference to spatial point pattern data and noting the following relationship between intensities and densities in spatial point pattern data. Kelsall and Diggle [79] note that, conditional on the value of n, the number of points, spatial point pattern data can be treated as a random sample from a probability distribution with density $f(\cdot)$ that is proportional to $\lambda(\cdot)$.

For bandwidth selection for kernel intensity estimation, the aim is to find the bandwidth choice h that produces an intensity estimate $\widehat{\lambda}(\cdot)$ that is close to the true intensity $\lambda(\cdot)$. However, we do not have $\lambda(\cdot)$ to use for comparison. If we did have an extra single independent observation y from $\lambda(\cdot)$, then the log-likelihood of that observation having been produced by $\lambda(\cdot)$ is $\log f(y)$, where $f(\cdot)$ is the corresponding probability density to $\lambda(\cdot)$, or we can say that $\log \lambda(y)$ is the log-likelihood in terms of intensities. Now consider intensity estimates $\widehat{\lambda}_h(\cdot)$, based on the data $\mathbf{x}$, that depend on the choice of smoothing parameter h. Then $\log \widehat{\lambda}_h(y)$ is the log-likelihood at y, in terms of intensities, of the smoothing parameter h being the optimal choice, since the data $\mathbf{x}$ is now fixed. Thus, $\log \widehat{\lambda}_h(y)$ will be at a maximum for the h that produces the highest intensity estimate at observation y.

Then consider if, in place of y, we choose a single observation x_i from the data itself, and treat it as a separate independent observation produced by $\lambda(\cdot)$. Then the log-likelihood at x_i, in terms of intensities, is $\log \widehat{\lambda}_h^{-i}(x_i)$, where $\widehat{\lambda}_h^{-i}(x_i)$ is an intensity estimate at x_i based on all the data except x_i .

We can apply this procedure to each observation from the data in turn to produce a score function for the choice of h.

Thus, *leave-one-out* cross-validation ([121]; [93, Sec. 5.3, pp. 87–95]) selects the bandwidth h^* which maximises

$$C(h) = \sum_i \log \widehat{\lambda}_h^{-i}(x_i), \tag{2.38}$$

where $\widehat{\lambda}_h^{-i}(x_i)$ is

$$\widehat{\lambda}_h^{-i}(x_i) = \widehat{\lambda}(x_i \mid \mathbf{x} \setminus \{x_i\}, h) = \sum_{j \neq i} k_t(x_i \mid x_j) = \widehat{\lambda}_h(x_i) - k_t(x_i \mid x_i). \tag{2.39}$$

Alternative cross-validation methods are discussed in [29, 35, 72, 137, 125].

2.5.3 Weaknesses of cross-validation

Weaknesses of data-based cross-validation methods are well known. Terrell [125] argues that they "have often failed to be useful" because they choose an under-estimate of the best bandwidth and consequently produce too many artefacts. Terrell [125] proposed slightly *over-smoothed* kernel estimates using his "Maximal Smoothing Principle" that is not aiming to be asymptotically optimal but conservative enough such that any large-scale features are likely to be of interest. Terrell argues that it is problematic to produce a smoothed density (or intensity) estimate that contains a peak that may be real or may be an artefact of the estimation procedure because this may lead to fallacious interpretations of the data. Terrell argues that it is preferable to use a wider bandwidth than the optimal bandwidth to oversmooth the data so that peaks or other features that survive that oversmoothing are very likely to point to something that is real.

It is unclear if Terrell's methods can be applied to linear network spatial point pattern data, since they require analytic calculation for an assumed density function.

2.6 Relative risk

Relative risk compares two spatial point patterns occurring in the same space by finding the ratio of their spatially-varying intensities. For example, in the Geelong traffic accident data it could be of interest to compare the spatial point patterns of the accidents that occur at night and those that occur during the day.

2.6.1 Relative risk function

Suppose there are two point patterns $\mathbf{x} = \{x_1, \ldots, x_m\}$ and $\mathbf{y} = \{y_1, \ldots, y_n\}$ observed in the same spatial domain W. Treating $\mathbf{x}$ and $\mathbf{y}$ as realisations of point processes $\mathbf{X}$ and $\mathbf{Y}$, respectively, our goal is to estimate the *logarithmic relative risk*

$$\rho(u) = \log \frac{\lambda_{\mathbf{X}}(u)}{\lambda_{\mathbf{Y}}(u)}, \quad u \in W, \tag{2.40}$$

where $\lambda_{\mathbf{X}}(\cdot), \lambda_{\mathbf{Y}}(\cdot)$ are the intensities of $\mathbf{X}, \mathbf{Y}$, respectively.

We could estimate $\rho(u)$ by individually estimating $\widehat{\lambda}_{\mathbf{X}}(u)$ and $\widehat{\lambda}_{\mathbf{Y}}(u)$, giving the *plug-in estimator*

$$\widehat{\rho}(u) = \widehat{\rho}_{h_1,h_2}(u) = \log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}, \quad u \in W, \tag{2.41}$$

where $\widehat{\lambda}_{\mathbf{X}}(u) = \widehat{\lambda}(u \mid \mathbf{x}, h_1)$ and $\widehat{\lambda}_{\mathbf{Y}}(u) = \widehat{\lambda}(u \mid \mathbf{y}, h_2)$ are kernel estimates of $\lambda_{\mathbf{X}}(u)$ and $\lambda_{\mathbf{Y}}(u)$, computed from $\mathbf{x}$ and $\mathbf{y}$, using bandwidths h_1 and h_2, respectively. We note the warning by Loader [92] that plug-in estimators may perform poorly.

Relative risk in $\mathbb{R}^2$ was first investigated by Bithell [22]. Bithell studied data on childhood leukaemia, in the vicinity of the Sellafield nuclear reprocessing plant in Cumbria, U.K, using relative risk to explore whether leukaemia cases increased near the nuclear processing plant. The two types of events were the home address locations of the leukaemia cases in the area and all home addresses in the area. He used two-dimensional kernel smoothing to produce intensity estimates of the two event types. He investigated whether the intensities only differed by a constant or whether the differences could be explained by proximity to the nuclear reprocessing plant

Subsequently Kelsall and Diggle [79] and [80] studied data for the location of larynx cancer cases, in the vicinity of an incinerator in Lancashire, England, for the possibility that the incinerator had caused a cluster of larynx cancer cases. The two types of events were the home addresses of the larynx cancer cases and the home addresses of lung cancer cases. The lung cancer cases serve as a sample from the susceptible population, and are presumed to be unrelated to proximity to the incinerator. Kelsall and Diggle first paper [79] generated one-dimensional point pattern data by using the squared distance of the two types of events from the location of the incinerator. Then a second paper [80] analysed the data as two-dimensional spatial point patterns using the actual locations of the larynx and lung cancer cases.

Bithell [22] did not consider the optimal choice of bandwidth for his relative risk investigation. However, Kelsall and Diggle [79] consider this question for the one-dimensional case and then they extend this to the two-dimensional case in their second paper [80].

2.6.2 Bandwidth selection for relative risk

Three possible methods for selecting the bandwidths h_1, h_2 in (2.41), discussed in Kelsall and Diggle [79], are (**M1**) *separate* selection of h_1 based only on $\mathbf{x}$ and of h_2 based only on $\mathbf{y}$; (**M2**) *joint* selection of the pair (h_1, h_2) based on $\mathbf{x}$ and $\mathbf{y}$; (**M3**) *symmetric* selection of a common bandwidth $h = h_1 = h_2$ based on $\mathbf{x}$ and $\mathbf{y}$. Method M1 could use any of the criteria described in Section 2.5, while methods M2 and M3 require the introduction of new techniques. Bandwidth selection for relative risk is different in principle from bandwidth selection for intensity. For example, if the true intensities are proportional, say $\lambda_{\mathbf{X}}(u) = c\lambda_{\mathbf{Y}}(u)$, for some constant c, then $\rho(u) = \log c$ is constant, and an infinite bandwidth $h = \infty$ is typically optimal for estimating ρ, while estimation of $\lambda_{\mathbf{X}}(u)$ requires smaller bandwidths. Consequently, method M1 is unlikely to perform well when the relative risk is almost constant.

Assuming $\mathbf{X}$ and $\mathbf{Y}$ are independent Poisson processes, the estimator (2.41) is asymptotically normal with asymptotic pointwise bias and variance of the same form as given in [79, Sec. 2]

for the two-dimensional case:

$$\mathbb{E}[\widehat{\rho}_{h_1,h_2}(u) - \rho(u)] \sim \frac{1}{2}h_1^2\frac{\lambda''_{\mathbf{X}}(u)}{\lambda_{\mathbf{X}}(u)} - \frac{1}{2}h_2^2\frac{\lambda''_{\mathbf{Y}}(u)}{\lambda_{\mathbf{Y}}(u)}; \tag{2.42}$$

$$\mathsf{var}[\widehat{\rho}_{h_1,h_2}(u)] \sim \frac{1}{2\sqrt{\pi}}\left(\frac{1}{h_1\lambda_{\mathbf{X}}(u)} + \frac{1}{h_2\lambda_{\mathbf{Y}}(u)}\right), \tag{2.43}$$

so that the asymptotic mean integrated squared error is

$$\begin{aligned}\mathsf{MISE}[\widehat{\rho}_{h_1,h_2}(u)] \sim{}& \frac{1}{2\sqrt{\pi}}\left(\frac{A_1}{h_1} + \frac{A_2}{h_2}\right) \\ &+\frac{1}{4}\left(h_1^4 B_{11} - 2h_1^2h_2^2B_{12} + h_2^4B_{22}\right),\end{aligned} \tag{2.44}$$

where $A_1 = A(\lambda_{\mathbf{X}})$ and $A_2 = A(\lambda_{\mathbf{Y}})$ are defined by $A(f) = \int_L f(u)^{-1}\mathrm{d}u$, and $B_{11} = B(\lambda_{\mathbf{X}}, \lambda_{\mathbf{X}})$, $B_{12} = B(\lambda_{\mathbf{X}}, \lambda_{\mathbf{Y}})$, $B_{22} = B(\lambda_{\mathbf{Y}}, \lambda_{\mathbf{Y}})$ are defined by $B(f, g) = \int_L (f''(u)/f(u)(g''(u)/g(u))\mathrm{d}u$. These approximations determine the asymptotically optimal bandwidths for methods M1, M2 and M3, and these have the same form as those given by [79, Sec. 2] for the two-dimensional case.

The literature is not unanimous on the relative merits of the three methods, but generally favours method M3, which constrains the two bandwidths to be equal. Kelsall and Diggle [80, p. 10] conclude that method M3 has some theoretical justification when $\lambda_{\mathbf{X}} \propto \lambda_{\mathbf{Y}}$, and report simulation experiments in which method M3 achieved the best performance. Davies *et al.* [47] demonstrate that halo-like artefacts can occur when relative risk is estimated using different smoothing bandwidths for the numerator and denominator. Support for method M3 is also given in [48] and [53, §9.3, p. 179ff.] with examples and technique described in [46, 45]. In Chapter 5 we explore using either M2 or M3.

2.6.3 Kelsall–Diggle density-ratio cross-validation in $\mathbb{R}$ and $\mathbb{R}^2$

For two-dimensional point pattern data, Kelsall and Diggle [79] proposed a method for selecting the bandwidths (h_1, h_2) in (2.41), with or without the constraint $h_1 = h_2$, by cross-validation based on integrated squared error.

$$\mathrm{ISE}[\widehat{\rho}_{\mathrm{h_1,h_2}}(\mathrm{u})] = \int_W [\widehat{\rho}_{h_1,h_2}(u) - \rho_{h_1,h_2}(u)]^2\,\mathrm{d}u.$$

$$\begin{aligned}\widetilde{C}_{\mathrm{KD}}(h_1, h_2) = -\int_W \Bigg[\widehat{\rho}_{h_1,h_2}(u)]^2\mathrm{d}u - 2\sum_{i=1}^{m}\frac{1}{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)}\log\frac{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)}{\widehat{\lambda}_{\mathbf{Y}}(x_i)} \\ - 2\sum_{j=1}^{n}\frac{1}{\widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)}\log\frac{\widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)}{\widehat{\lambda}_{\mathbf{X}}(y_j)},\end{aligned} \tag{2.45}$$

where $\widehat{\lambda}_{\mathbf{X}}(u) = \widehat{\lambda}(u \mid \mathbf{x}, h_1)$ is an estimate of $\lambda_{\mathbf{X}}(u)$ using bandwidth h_1, while $\widehat{\lambda}_{\mathbf{Y}}(u) = \widehat{\lambda}(u \mid \mathbf{y}, h_2)$ is an estimate of $\lambda_{\mathbf{Y}}(u)$ using bandwidth h_2, and where $\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i) = \widehat{\lambda}(x_i \mid \mathbf{x} \setminus \{x_i\}, h_1)$ and $\widehat{\lambda}_{\mathbf{Y}}^{-i}(y_i) = \widehat{\lambda}(y_i \mid \mathbf{x} \setminus \{y_i\}, h_2)$ are the corresponding leave-one-out estimates (see equation (2.39)).

2.6.4 Likelihood and least-squares cross-validation

A later paper by Kelsall and Diggle [81] proposed a different approach to relative risk, using a connection with binary regression, which they argued is more flexible than the density-ratio approach. If $\mathbf{X}$ and $\mathbf{Y}$ are independent Poisson processes in $\mathbb{R}^2$ with intensity functions $\lambda_{\mathbf{X}}(u), \lambda_{\mathbf{Y}}(u)$, respectively, then the superimposition $\mathbf{Z} = \mathbf{X} \cup \mathbf{Y}$ is Poisson with intensity $\lambda_{\mathbf{Z}}(u) = \lambda_{\mathbf{X}}(u) + \lambda_{\mathbf{Y}}(u)$, and a random point of $\mathbf{Z}$ at location u has probability $p(u) = \lambda_{\mathbf{X}}(u)/\lambda_{\mathbf{Z}}(u)$ of having originated from the process $\mathbf{X}$ rather than $\mathbf{Y}$.

Given data patterns $\mathbf{x}$, $\mathbf{y}$, define for each point $x_i \in \mathbf{x}$, for $i = 1, \ldots, m$,

$$\hat{p}_i = \frac{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)}{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i) + \widehat{\lambda}_{\mathbf{Y}}(x_i)}, \tag{2.46}$$

to be the estimated probability (estimated from all data other than x_i) that a point of $\mathbf{X} \cup \mathbf{Y}$ at location x_i would belong to $\mathbf{X}$ rather than $\mathbf{Y}$. Similarly, for all points $y_j \in \mathbf{y}$, $j = 1, \ldots, n$, define

$$\hat{q}_j = \frac{\widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)}{\widehat{\lambda}_{\mathbf{X}}(y_j) + \widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)},$$

to be the estimated probability that a point of $\mathbf{X} \cup \mathbf{Y}$ at y_j would belong to $\mathbf{Y}$ rather than $\mathbf{X}$.

Then the *likelihood cross-validation* criterion [81] is the negative log-likelihood

$$\widetilde{C}_{\text{LIK}}(h_1, h_2) = -\left[\sum_{i=1}^{m} \log(\hat{p}_i) + \sum_{j=1}^{n} \log(\hat{q}_j)\right]. \tag{2.47}$$

Minimisation of (2.47) has also been suggested in [9]. The *least-squares* cross-validation criterion to be minimised is

$$\widetilde{C}_{\text{LSQ}}(h_1, h_2) = \sum_{i=1}^{m} (1 - \hat{p}_i)^2 + \sum_{j=1}^{n} (1 - \hat{q}_j)^2. \tag{2.48}$$

This is the loss criterion for least squares prediction of the status of each point given the locations of all points. It is known to work well in many contexts [66]. We shall use all of the cross-validation criteria listed above in our experiments in Chapter 5.

Figure 2.8: Equal-split kernel estimate. *Left:* the algorithm begins by making a copy of the kernel function for each data point x_i, confined to the line segment containing x_i. *Right:* At each fork, the remaining tail of the kernel is split equally between the outgoing segments. If there are n outgoing segments, each outgoing segment receives a copy of the kernel tail weighted by $1/n$.

Figure 2.9: Equal-split continuous method. At each fork, with $m-1$ outgoing segments, each outgoing segment receives a copy of the kernel weighted by $2/m$, while the incoming segment receives a copy with the negative weight $2/m - 1$.

Chapter 3

Fitting point process models

This chapter demonstrates that point pattern data on a linear network can be analysed using spatial point process methods. We establish the basic elements of a point process modelling approach, develop statistical theory and algorithms for model-fitting, and apply them to the Geelong accident data. The general theory of spatial point processes does not presuppose any particular stochastic mechanism, and allows the investigation of any spatial pattern of events [42, 65]. The data represented in Figure 1.1 will be called a *point pattern*, while the hypothetical random process that generated the data is called a *point process*. Section 3.1 gives the statistical theory for estimation of the model parameters from data. Section 3.2 gives examples of specific models for point process intensity. Section 3.3 describes our computational technique for estimating the model parameters from data by adapting the Berman-Turner device for use on a linear network. Section 3.4 exhibits several applications. Section 3.5 is analysis of bias caused by the aggregation of data used by the traditional crash-frequency approach to traffic accident data. Section 3.6 discusses the applicability of using Poisson models for this data.

3.1 Statistical theory

Here we develop statistical theory for the inhomogeneous Poisson point process model [42] on a linear network. We derive expressions for the joint probability, the likelihood for the process intensity function given an observed point pattern, and the score function of that likelihood. These will be used in sections below to find the maximum likelihood estimates of our model parameters. We also derive expressions for the Fisher information, standard errors, and confidence intervals for the model parameter estimates.

3.1.1 Likelihood

Our statistical methodology will be based on the likelihood function of the point process model. This is equivalent to the probability density of the model, considered as a function of the model parameters, as we explain below.

In the simplest case of a homogeneous Poisson point process of intensity $\lambda > 0$ on a linear network L, the joint probability density is

$$f(\mathbf{x}) = f(\mathbf{x}; \lambda) = \lambda^{n(\mathbf{x})} \exp\big(-(\lambda - 1)|L|\big), \tag{3.1}$$

for each possible outcome $\mathbf{x} = \{x_1, ..., x_n\}$, where again $n(\mathbf{x})$ denotes the number of points in $\mathbf{x}$ and $|L|$ is the total length of L. Note that $n(\mathbf{x})$ is not fixed (and may even be zero). Therefore equation (3.1) is a joint probability of a variable number of random variables. The technical details required to define a probability density for a "list with variable length" [8] need not concern us here.

If we now suppose that a point pattern $\mathbf{x}$ has been recorded, we may treat $\mathbf{x}$ as fixed, and consider $f(\mathbf{x}; \lambda)$ as a function of the parameter λ only, defining the *likelihood* $L(\lambda) = f(\mathbf{x}; \lambda)$. Then we define the *maximum likelihood estimate (MLE)* of λ from the data $\mathbf{x}$ to be the value $\hat{\lambda}$ that maximises the likelihood $L(\lambda)$. Using elementary calculus, the MLE for model [3.1] is $\hat{\lambda} = n(\mathbf{x})/|L|$, the simple average intensity of points per unit length.

For an inhomogeneous Poisson point process with intensity function $\lambda(u)$ for $u \in L$, the joint probability density is

$$f(x_1, ..., x_n) = \Big[\prod_i \lambda(x_i)\Big] \exp\Big(\int_L (1 - \lambda(u))\, \mathrm{d}u\Big). \tag{3.2}$$

It is easy to check that (3.2) reduces to (3.1) in the case where the intensity function $\lambda(u)$ is constant.

Our general model assumes that the observed point pattern $\mathbf{x}$ is a realisation of an inhomogeneous Poisson point process $\mathbf{X}$ with intensity function $\lambda(u)$ which depends on a parameter (or vector of parameters) $\boldsymbol{\theta}$ through $\lambda(u) = \lambda_{\boldsymbol{\theta}}(u)$. This is a very general class of models, in the sense that we make no assumptions about the algebraic form of $\lambda_{\boldsymbol{\theta}}(u)$, other than the technical requirement that the total intensity $\int_L \lambda_{\boldsymbol{\theta}}(u)\, \mathrm{d}u$ is finite for all $\boldsymbol{\theta}$. The function $\lambda_{\boldsymbol{\theta}}(u)$ typically involves spatial covariates. Examples of such models are presented in Section 3.2.

The likelihood for $\boldsymbol{\theta}$, given observation of the point pattern $\mathbf{x}$, is

$$L(\boldsymbol{\theta}) = \Big[\prod_i \lambda_{\boldsymbol{\theta}}(x_i)\Big] \exp\Big(\int_L (1 - \lambda_{\boldsymbol{\theta}}(u))\, \mathrm{d}u\Big). \tag{3.3}$$

3.1.2 Score function

Assuming the likelihood $L(\boldsymbol{\theta})$ is positive and finite, and differentiable with respect to $\boldsymbol{\theta}$, and that there are no constraints on $\boldsymbol{\theta}$, the maximum likelihood is attained at a zero of the derivative. (See [71] for a standard reference on likelihood theory). The *score* function is defined as the derivative of the logarithm of the likelihood,

$$\boldsymbol{U}(\boldsymbol{\theta}) = \frac{\partial}{\partial \boldsymbol{\theta}} \log L(\boldsymbol{\theta}),$$

so that the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$ (if it exists) satisfies the score equation $\boldsymbol{U}(\hat{\boldsymbol{\theta}}) = 0$. In general $\boldsymbol{\theta}$ is a p-dimensional vector $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_p]$, so that the score is a p-dimensional vector-valued function

$$\boldsymbol{U}(\boldsymbol{\theta}) = [U_1(\boldsymbol{\theta}), \ldots, U_p(\boldsymbol{\theta})]^\top,$$

where $U_i(\boldsymbol{\theta}) = \frac{\partial}{\partial \theta_i} \log L(\boldsymbol{\theta})$, is the derivative with respect to the ith parameter.

For the Poisson point process likelihood (3.3), the score is

$$\boldsymbol{U}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \boldsymbol{\kappa}_{\boldsymbol{\theta}}(x_i) - \int_L \boldsymbol{\kappa}_{\boldsymbol{\theta}}(u) \lambda_{\boldsymbol{\theta}}(u) \, \mathrm{d}u, \tag{3.4}$$

where $\boldsymbol{\kappa}_{\boldsymbol{\theta}}(u) = (\partial/\partial \boldsymbol{\theta}) \log \lambda_{\boldsymbol{\theta}}(u)$ is the vector of partial derivatives of the log intensity.

3.1.3 Fisher information

A measure of the accuracy of the parameter estimator $\hat{\boldsymbol{\theta}}$ can be obtained using the Fisher information matrix. This is defined as the variance-covariance matrix of the score, that is, the $p \times p$ matrix $\boldsymbol{I} = \boldsymbol{I}(\boldsymbol{\theta})$, whose entries are the variances and covariances of the components $U_i(\boldsymbol{\theta})$. A useful equivalent definition under regularity conditions is

$$\boldsymbol{I}(\boldsymbol{\theta}) = -\mathbb{E}\left[\frac{\partial}{\partial \boldsymbol{\theta}} U(\boldsymbol{\theta})\right],$$

the negative expected (mean) value of the derivative matrix of the score, that is, the negative expected Hessian matrix (matrix of second partial derivatives) of the log-likelihood.

For an inhomogeneous Poisson point process with intensity $\lambda_{\boldsymbol{\theta}}(u)$, assuming that $\lambda_{\boldsymbol{\theta}}(u)$ is twice differentiable with respect to $\boldsymbol{\theta}$, we have

$$\frac{\partial}{\partial \boldsymbol{\theta}} \boldsymbol{U}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \boldsymbol{\zeta}_{\boldsymbol{\theta}}(x_i) - \int_L \boldsymbol{\zeta}_{\boldsymbol{\theta}}(u) \, \lambda_{\boldsymbol{\theta}}(u) \, \mathrm{d}u - \int_L \boldsymbol{\kappa}_{\boldsymbol{\theta}}(u) \boldsymbol{\kappa}_{\boldsymbol{\theta}}(u)^\top \lambda_{\boldsymbol{\theta}}(u) \, \mathrm{d}u,$$

where $\boldsymbol{\zeta}_{\boldsymbol{\theta}}(u) = (\partial/\partial \boldsymbol{\theta}) \kappa_{\boldsymbol{\theta}}(u)$ is the matrix of second derivatives of $\log \lambda_{\boldsymbol{\theta}}(u)$. By Campbell's formula for point processes [42, p. 163], the expectations of the terms involving $\boldsymbol{\zeta}_{\boldsymbol{\theta}}$ cancel, so the Fisher information is

$$\boldsymbol{I}(\boldsymbol{\theta}) = \int_L \boldsymbol{\kappa}_{\boldsymbol{\theta}}(u) \boldsymbol{\kappa}_{\boldsymbol{\theta}}^\top(u) \, \lambda_{\boldsymbol{\theta}}(u) \, \mathrm{d}u. \tag{3.5}$$

Under an appropriate large scale asymptotic regime (see [86] for further details) the parameter estimator $\hat{\boldsymbol{\theta}}$ is asymptotically multivariate normally distributed, with mean equal to the true parameter value $\boldsymbol{\theta}$ and variance-covariance matrix equal to $\boldsymbol{I}^{-1}(\boldsymbol{\theta})$.

3.1.4 Standard errors and confidence intervals

The variance-covariance matrix for the parameter estimator $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \ldots, \hat{\theta}_p)$ is the $p \times p$ matrix $\mathrm{var}(\hat{\boldsymbol{\theta}})$ whose ith diagonal entry is the variance of the parameter estimator $\hat{\theta}_i$, and whose off-diagonal (i, j) entry is the covariance of $\hat{\theta}_i$ with $\hat{\theta}_j$.

Asymptotic theory [86] indicates that $\mathrm{var}(\hat{\boldsymbol{\theta}})$ is asymptotically equal to $\boldsymbol{I}(\boldsymbol{\theta})^{-1}$. We estimate $\boldsymbol{I}^{-1}(\boldsymbol{\theta})$ by plugging in the estimated value of $\boldsymbol{\theta}$, that is by evaluating $\boldsymbol{I}^{-1}(\hat{\boldsymbol{\theta}})$, and thus we are able to obtain standard errors and confidence intervals for each component of $\boldsymbol{\theta}$.

For θ_j, the jth component of $\boldsymbol{\theta}$, the estimated standard error of $\hat{\theta}_j$ is $\sqrt{\hat{v}_{jj}}$, the square root of the (j, j) diagonal entry of $\boldsymbol{I}^{-1}(\hat{\boldsymbol{\theta}})$. The asymptotic confidence interval for the true value of θ_j, with confidence level $100(1 - \alpha)\%$, is

$$\hat{\theta}_j \pm z_{\alpha/2} \sqrt{\hat{v}_{jj}}\,, \tag{3.6}$$

where $z_{\alpha/2}$ is the upper $\alpha/2$ critical point of the standard normal distribution.

Similarly, we may obtain standard errors and confidence intervals for contrasts or linear combinations of the parameters. For a linear combination $\eta = b^\top \boldsymbol{\theta}$, where b is a vector of length p, the estimator $\hat{\eta} = b^\top \hat{\boldsymbol{\theta}}$ has estimated variance $b\boldsymbol{I}^{-1}(\hat{\boldsymbol{\theta}})b^\top$.

We can also construct confidence intervals for the value of the intensity $\lambda(u)$ at a given location u. Asymptotic theory supports the first-order Taylor approximation

$$\log \lambda_{\hat{\boldsymbol{\theta}}}(u) \approx \log \lambda_{\boldsymbol{\theta}}(u) + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})\boldsymbol{\kappa}_{\boldsymbol{\theta}}(u),$$

so that $\log \lambda_{\hat{\boldsymbol{\theta}}}(u)$ is approximately normally distributed with mean equal to the correct value $\log \lambda_{\boldsymbol{\theta}}(u)$ and variance $\boldsymbol{\kappa}_{\boldsymbol{\theta}}(u)^\top \boldsymbol{I}(\boldsymbol{\theta})^{-1} \boldsymbol{\kappa}_{\boldsymbol{\theta}}(u)$. Thus a confidence interval for the true intensity $\log \lambda_{\boldsymbol{\theta}}(u)$, with confidence $100(1 - \alpha)\%$ approximately, is

$$\log \lambda_{\hat{\boldsymbol{\theta}}}(u) \pm z_{\alpha/2} \sqrt{\hat{v}}\,, \tag{3.7}$$

where

$$\hat{v} = \boldsymbol{\kappa}_{\hat{\boldsymbol{\theta}}}(u)^\top \boldsymbol{I}(\hat{\boldsymbol{\theta}})^{-1} \boldsymbol{\kappa}_{\hat{\boldsymbol{\theta}}}(u). \tag{3.8}$$

Exponentiating gives a confidence interval for $\lambda_{\boldsymbol{\theta}}(u)$. The interval is not symmetric about the estimated value $\lambda_{\hat{\boldsymbol{\theta}}}(u)$, but has the advantage of avoiding negative intensity values.

We can also construct confidence intervals for $\mu(B)$, the expected number of random points of $\mathbf{X}$ falling in a subset $B \subseteq L$. The estimator is given by $\hat{\mu}(B) = \int_B \lambda_{\hat{\boldsymbol{\theta}}}(u) du$, where $\widehat{\boldsymbol{\theta}}$ is

asymptotically normal with mean vector $\boldsymbol{\theta}$ and variance-covariance matrix $I(\boldsymbol{\theta})^{-1}$. We derive the variance of the estimator from first principles to get

$$\operatorname{var} \widehat{\mu}(B) = \int_B \int_B \operatorname{cov}(\lambda_{\widehat{\boldsymbol{\theta}}}(u), \lambda_{\widehat{\boldsymbol{\theta}}}(v)) \mathrm{d}u \mathrm{d}v. \tag{3.9}$$

Then using the Taylor expansion, $\lambda_{\widehat{\boldsymbol{\theta}}}(u) - \lambda_{\boldsymbol{\theta}}(u) \approx (\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta})\kappa_{\boldsymbol{\theta}}(u), \lambda_{\boldsymbol{\theta}}(u)$ we get the result

$$\operatorname{cov}(\lambda_{\widehat{\boldsymbol{\theta}}}(u),\ \lambda_{\widehat{\boldsymbol{\theta}}}(v)) \approx \kappa_{\boldsymbol{\theta}}(u)^\top I(\boldsymbol{\theta})^{-1} \kappa_{\boldsymbol{\theta}}(v) \lambda_{\boldsymbol{\theta}}(u) \lambda_{\boldsymbol{\theta}}(v).$$

Substituting this result back into equation (3.9) and integrating over B we get $\operatorname{var} \widehat{\mu}(B) \approx K_{\boldsymbol{\theta}}(B)^\top I(\boldsymbol{\theta})^{-1} K_{\boldsymbol{\theta}}(B)$, where $K_{\boldsymbol{\theta}}(B) = \int_B \kappa_{\boldsymbol{\theta}}(u) \lambda_{\boldsymbol{\theta}}(u) \mathrm{d}u$.

The plug-in estimator of the variance,

$$\widehat{\operatorname{var}} \widehat{\mu}(B) = K_{\widehat{\boldsymbol{\theta}}}(B)^\top I(\widehat{\boldsymbol{\theta}})^{-1} K_{\widehat{\boldsymbol{\theta}}}(B), \tag{3.10}$$

can be used as the basis for a confidence interval for $\mu(B)$.

3.1.5 Model selection

We often wish to decide whether a particular explanatory variable should be included in the model. For example, it may be required to assess whether a traffic management strategy had any effect on accident risk. This effectively requires us to decide between two competing models, which are identical except that one of the models includes an additional explanatory variable related to the traffic management strategy. This is an instance of *model selection*.

In the example above, the null model (that the traffic management strategy had no effect) is a special case of the alternative model (that the traffic management strategy had an effect, possibly zero). In this case the models are "nested" and an appropriate decision rule is the *likelihood ratio test*. Let L_0 and L_1 be the maximum values of the likelihood under the null and alternative models respectively. Define the test statistic

$$T = 2 \log(L_1/L_0) = 2(\log L_1 - \log L_0).$$

The likelihood ratio test rejects the null hypothesis in favour of the alternative, if $T > \chi^2_{\nu,\alpha}$, where $\chi^2_{\nu,\alpha}$ is the upper α probability critical point ($1-\alpha$ quantile) of the chi-squared distribution with ν degrees of freedom, where ν is the number of extra parameters added in the alternative model.

If we need to compare models which are not nested, a possible strategy is to choose the model which minimises the Akaike Information Criterion (AIC)

$$\mathsf{AIC} = -2 \log L + 2p,$$

where, for any particular model, L is the maximum value of the likelihood, and p is the number of parameters.

3.2 Intensity models

In this section we study a series of models for point process intensity and analyse them using the techniques of Section 3.1.

3.2.1 Constant intensity

The simplest model is one in which the intensity is constant, $\lambda(u) \equiv \beta$, corresponding to a homogeneous Poisson process (Section 2.2.2). The score (3.4) reduces to $U(\beta) = n/\beta - |L|$. The maximum likelihood estimate of the parameter β is the solution of the normal equation $U(\beta) = 0$. Thus the parameter β is maximized at $\hat{\beta} = n/|L|$. The Fisher information (3.5) is $\boldsymbol{I}(\beta) = |L|/\beta$. Given an observed point pattern, we estimate β by $\hat{\beta} = n/|L|$, and estimate the Fisher information by the "plug-in" estimator $\boldsymbol{I}(\hat{\beta}) = |L|^2/n$. The asymptotic estimate of the variance $v = \mathrm{var}(\hat{\beta})$ is $\hat{v} = (\boldsymbol{I}(\hat{\beta}))^{-1} = n/|L|^2$. Accordingly the estimated standard error of $\hat{\beta}$ is $\mathrm{SE}(\hat{\beta}) = \sqrt{n}/|L|$ and an asymptotic 95 % confidence interval has endpoints $\hat{\beta} \pm 1.96\sqrt{n}/|L|$.

3.2.2 Intensity proportional to a baseline

A common model is the proportional relationship

$$\lambda(u) = \beta.W(u), \tag{3.11}$$

where $W(u)$ is a known function serving as a baseline or reference. For example, if $W(u)$ is a measure of traffic intensity at location u, then the model postulates that accidents occur in proportion to the traffic intensity, and the parameter β is the accident risk per unit of traffic intensity.

The maximum likelihood estimator for this model can be solved exactly. The score and Fisher information are $U(\beta) = n(\mathbf{x})/\beta - W_L$ and $I(\beta) = W_L/\beta$, where $W_L = \int_L W(u)\,\mathrm{d}u$. The MLE is $\hat{\beta} = n(\mathbf{x})/W_L$, the ratio of total event count to total baseline intensity. The asymptotic variance of $\hat{\beta}$ is $\mathrm{var}(\hat{\beta}) = \beta/W_L$, which can be estimated by $\hat{v} = n(\mathbf{x})/{W_L}^2$.

3.2.3 The loglinear intensity model

An important and flexible model is the loglinear intensity model

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(\boldsymbol{\theta}\mathbf{Z}(u) + A(u)), \tag{3.12}$$

where $\mathbf{Z}(u)$ and $A(u)$, $u \in L$ are known functions. Here $\mathbf{Z}(u)$ is a p-dimensional vector-valued function, consisting of p spatial covariates, each associated with one component of the p-dimensional parameter vector $\boldsymbol{\theta}$. There is a wide choice of possible covariates $\mathbf{Z}(u)$ as discussed in the Introduction.

The real-valued function $A(u)$ is not associated with a parameter, and is known as an *offset* term. It is effectively equivalent to introducing a baseline or reference intensity $W(u) = \exp A(u)$. For example, if we wish to investigate the effect of spatial covariate Z on accident risk, after adjusting for the traffic intensity given by $M(u)$, this is accomplished by a model of the form (3.12) with offset $A(u) = \log M(u)$.

Additive terms inside the exponent in (3.12) represent multiplicative factors contributing to the intensity. In particular, negative terms in the exponent represent factors which reduce the intensity and which can be interpreted as probabilities $p(u)$ in a thinning model (Section 2.2.3). In applications to traffic accidents , an appealing interpretation is that $p(u)$ is the probability that a near-accident occurring at location u will progress to become an accident.

The score function (3.4) for the loglinear model (3.12) is

$$U(\boldsymbol{\theta}) = \sum_{i=1}^{n} \mathbf{Z}(x_i) - \int_L \mathbf{Z}(u)\lambda_{\boldsymbol{\theta}}(u)\,\mathrm{d}u \tag{3.13}$$

and the Fisher information (3.5) is

$$\boldsymbol{I}(\boldsymbol{\theta}) = \int_L \mathbf{Z}(u)\mathbf{Z}(u)^\top \lambda_{\boldsymbol{\theta}}(u)\,\mathrm{d}u. \tag{3.14}$$

There is no explicit, closed-form solution for the maximum likelihood estimate for this model. Numerical iterative algorithms are required. These are discussed in Section 3.3.

In practice, the Fisher information could be estimated either by the plug-in estimator

$$\hat{\boldsymbol{I}}_1 = \boldsymbol{I}(\hat{\beta}) = \int_L \mathbf{Z}(u)\mathbf{Z}(u)^\top \lambda_{\hat{\boldsymbol{\theta}}}(u)\,\mathrm{d}u \tag{3.15}$$

or by

$$\hat{\boldsymbol{I}}_2 = \sum_i \mathbf{Z}(x_i)\mathbf{Z}(x_i)^\top, \tag{3.16}$$

which is an unbiased estimator of $\boldsymbol{I}(\boldsymbol{\theta})$ by Campbell's formula.

The covariates $\mathbf{Z}(u)$ appearing in (3.12) are termed the "canonical" covariates. These do not need to be the same as the explanatory variables provided in the original data, which we shall call the "raw" covariates. Typically the canonical covariates $\mathbf{Z}(u)$ are obtained by transforming or combining the raw covariates. Individual covariates in the raw data can be transformed, for example, by a power or logarithmic transformation, and through some investigation the best choice of transformation can be chosen. Several raw covariates can be combined to take into account the interaction effects between them. Categorical covariates can be created either from a natural choice of categories such as the speed limit of the road, judgement based categories such as good, medium and poor road condition, or a simplification of a continuous variable by partitioning into categories.

3.3 Model-fitting algorithm

In general, no explicit closed-form solution is available for the maximum likelihood estimator of the parameters of an inhomogeneous Poisson spatial point process. The parameter estimate must be found using numerical methods to maximise (3.3).

3.3.1 Discretisation methods

Numerical maximisation of the point process likelihood typically involves discretisation of the integral appearing in (3.3), so that the integral is replaced by a finite sum. If the discretisation is chosen carefully, the functional form of the discretised likelihood will be equivalent to the likelihood of a familiar statistical model, such as Poisson loglinear regression or binary logistic regression [127, 87, 30, 32, 31]. This has the important advantage that we can maximise the likelihood using existing statistical software for generalised linear models [89, 90, 91]. This allows us to harness the strengths of existing production code, including high reliability, good numerical performance, and a flexible interface making it easy to specify different models [20, 15].

One example of this approach that is familiar in GIS applications is the use of logistic regression applied to pixel presence-absence data [127, 4, 23]. This is equivalent to fitting a Poisson point process with a loglinear intensity model [11].

3.3.2 Berman–Turner device

Instead of logistic regression we have used a modification of the technique developed by Berman and Turner [20] and Baddeley and Turner [15] for fitting spatial Poisson point process models to two-dimensional spatial point pattern data. An implementation of this technique for two-dimensional point patterns is available in the package **spatstat** [16] in the **R** language [114] (see Section 3.3.3).

Here we develop a modification of the Berman–Turner device making it possible to fit Poisson point process models on a linear network. We assume a loglinear intensity model (3.12). The log-likelihood is (up to a constant)

$$\log L(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \lambda_{\boldsymbol{\theta}}(x_i) - \int_L \lambda_{\boldsymbol{\theta}}(u) du. \tag{3.17}$$

The first step in applying the Berman–Turner device is to approximate the integral in (3.17) by a finite sum. A *quadrature approximation* of the integral of any function f is an

approximation

$$\int f(u)\,\mathrm{d}u \approx \sum_{j=1}^{m} w_j\, f(u_j), \tag{3.18}$$

defined by a set of points $u_j (j = 1, \ldots, m)$ in L, and associated weights $w_j (j = 1, \ldots, m)$, called the *quadrature scheme*. To ensure that (3.18) is a reasonably good approximation, the quadrature points u_j should be spread reasonably evenly over the linear network, and the total weight $\sum_j w_j$ should equal the length of the network.

Given a point pattern dataset $\mathbf{x}$, to determine the quadrature scheme, we first generate a set of "dummy" or "sample" points $s_1, \ldots, s_k$ spread reasonably uniformly across the linear network. In the Berman–Turner approach, the dummy points are *combined with the data points* to form the set of quadrature points. That is, the $m = n + k$ quadrature points consist of the n data points $x_1, \ldots, x_n$ together with the k dummy points $s_1, \ldots, s_k$. The weights w_j are then determined by some appropriate rule. Several different weighting rules in $\mathbb{R}^2$ are canvassed in [15].

Our quadrature scheme rule for networks is built as follows. Each line segment joining two vertices of the network is treated in isolation from the rest of the network. A line segment of length ℓ is broken into subintervals of equal length Δ, a fixed and predetermined spacing, with two shorter subintervals ('rumps') at each end of the segment. The rumps have equal length $r/2$ where $r = \ell - [\ell/\Delta]\Delta$. To avoid numerical problems, when r is very small we reduce the number of subintervals of length Δ by one, so that $r/2$ is increased by $\Delta/2$.

A dummy point is placed at the centre of each subinterval. Each subinterval now contains one dummy point and may also contain one or more data points. The quadrature points u_j are the dummy points together with the data points. The weight w_j of each quadrature point is equal to the length of the subinterval it falls into, divided by the total number of quadrature points that fall into the subinterval. This ensures that the total weight associated with each subinterval is equal to the length of that subinterval. See Figure 3.1.

Applying the quadrature approximation (3.18) to the integral in (3.17) gives

$$\int_L \lambda_{\boldsymbol{\theta}}(u)\mathrm{d}u \approx \sum_{j=1}^{m} w_j \lambda_{\boldsymbol{\theta}}(u_j), \tag{3.19}$$

where $\lambda_{\boldsymbol{\theta}}(u_j)$ is the intensity at location u_j. Substituting in (3.17) gives the Berman–Turner approximation for the log-likelihood

$$\log L(\boldsymbol{\theta}) \approx \log L_{BT}(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \lambda_{\boldsymbol{\theta}}(x_i) - \sum_{j=1}^{m} w_j \lambda_{\boldsymbol{\theta}}(u_j). \tag{3.20}$$

Since we arranged that every data point is also a quadrature point, we can also rewrite

$$\sum_{i=1}^{n} \log \lambda_{\boldsymbol{\theta}}(x_i) = \sum_{j=1}^{m} y_j \log \lambda_{\boldsymbol{\theta}}(u_j),$$

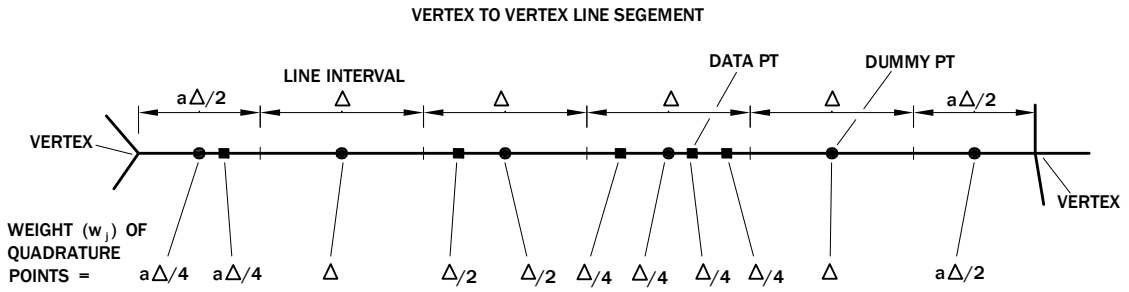

Figure 3.1: Scheme for assigning weights to quadrature points. Each line interval contains one dummy point (circles) and may contain one or more data points (squares). The weight of each quadrature point is equal to its line interval length divided by the total number of quadrature points contained in that line interval.

where y_j is the indicator variable that equals 1, if u_j is a data point, and equals 0, if u_j is a dummy point. Writing $\lambda_j = \lambda_{\boldsymbol{\theta}}(u_j)$ for the intensity at u_j, we find that the approximate log-likelihood collapses to

$$\log L_{BT}(\boldsymbol{\theta}) = \sum_{j=1}^{m} (y_j \log \lambda_j - \lambda_j w_j). \tag{3.21}$$

Since the intensity model was the loglinear model (3.12), we are assuming $\lambda_j = \exp(\boldsymbol{\theta} z_j + a_j)$, where $z_j = \mathbf{Z}(u_j)$ is the covariate value at u_j and $a_j = A(u_j)$ is the value of the offset. Taking $\lambda_j^* = \lambda_j w_j = \exp(\boldsymbol{\theta} z_j + a_j + \log w_j)$ we get

$$\log L_{BT}(\boldsymbol{\theta}) = \sum_{j=1}^{m} (y_j \log \lambda_j^* - \lambda_j^*) - b, \tag{3.22}$$

where $b = \sum_j y_j \log w_j$ does not depend on $\boldsymbol{\theta}$. Ignoring the constant b, this expression is equivalent to the log-likelihood of a Poisson loglinear regression model [98] with m observations, with responses y_j, mean responses λ_j and offsets $a_j + \log w_j$. This can be maximised using existing software for fitting generalized linear models [98]. This is a common device in fitting stochastic process models [89, 90, 91].

In applications relating to traffic accidents, a peculiar feature is that accidents frequently occur in or close to a road intersection. Accidents occurring exactly at an intersection can be accommodated by introducing additional quadrature points u_j located exactly at the intersections, with quadrature weight $w_j = 1$. The pseudo-response y_j for each intersection is equal to the number of accidents at the intersection. The derivation above still holds.

3.3.3 Software implementation

The algorithm described in Section 3.3.2 was implemented in the **R** language [114] using some infrastructure in the package `spatstat` [16]. This implementation has been included in the

spatstat package.

The implementation can be summarised as follows: Given a network L and point pattern $\mathbf{X}$, the quadrature scheme described in Section 3.3.2 is constructed. Spatial covariates are evaluated at the quadrature points. The resulting values of spatial coordinates, covariate values, and the data/dummy point indicator variable y_i are passed to existing internal code of the **spatstat** package which fits the associated GLM and extracts the fitted intensity, fitted coefficients, and Fisher information. The results are repackaged for correct use on a linear network.

3.4 Applications

In this section we use the Geelong data (Figure 1.1) to demonstrate the application of the methodology developed above. We start by fitting the constant intensity model (Section 3.4.1) and progress through a series of loglinear models of increasing complexity. Some of these models are not realistic for the Geelong data because they are too simple, using only one or two covariates. We present them here to demonstrate the methodology. Nevertheless, comparisons between these simple models can be used to assess the evidence for statistical association between accident risk and covariates. Some covariates are constructed from the network geometry, while others are external covariates such as speed limit and traffic volume.

3.4.1 Constant intensity

The constant intensity model $\lambda(u) \equiv \beta$ postulates that accidents follow a homogeneous Poisson process (Section 2.2.2) with a rate of β accidents per kilometre across the network. As explained above, this model may be unrealistic. A glance at the pattern of road accidents in Figure 1.1 indicates that the accident intensity is far from homogeneous. However, the constant intensity model is a useful benchmark for the evaluation of more suitable models.

Recall that the maximum likelihood estimate can be computed exactly in this case without the need for numerical approximation (Section 3.2.1). We can use this as a test case to cross-check the validity of the approximate solution obtained using the numerical approximation.

To apply the results of Section 3.2.1 to the Geelong data, the required information is the network length $|L| = 286.6$ kilometres and the total number of accidents $n(\mathbf{x}) = 242$ over three years. This gives the exact maximum likelihood estimate of the intensity $\hat{\beta} = n(\mathbf{x})/|L| = 242/286.6 = 0.844$ accidents per kilometre over three years, or 0.281 accidents per kilometre per year.

The plug-in estimate of the Fisher information is $\boldsymbol{I}(\hat{\beta}) = |L|^2/n(\mathbf{x}) = 286.6^2/242 = 339.4$. The estimate of the variance of $\hat{\beta}$ is $\hat{v} = \boldsymbol{I}(\hat{\beta})^{-1} = 1/339.4 = 0.0029$. The standard error for $\hat{\beta}$ is $\sqrt{\hat{v}} = 0.0543$. The 95% confidence interval for the parameter β has limits $\hat{\beta} \pm 1.96\sqrt{\hat{v}}$, that

is, $[0.738, 0.951]$.

Since $\lambda(u) \equiv \beta$, the expected intensity, according to this model, is estimated to lie between 0.738 and 0.951 accidents per kilometre over three years based on a 95% confidence interval. This result can be used to obtain an estimated *average* accident count confidence interval for a length of road over a 3-year period. Thus multiplying the per kilometre confidence intervals by 10, the *average* accident count in 10-kilometre stretches of road over a 3-year period is estimated to lie between 7.4 and 9.5 accidents. Assuming additionally that the accident rate is constant over time,the results can be extrapolated to different time periods by rescaling. For example, in a single-year period the average accident count is estimated to lie between 2.5 and 3.2 accidents in a 10-kilometre stretch.

The confidence interval for the expected accident count for a length of road over a 3-year period should not be confused with the matching prediction interval for the observed accident count for a length of road over a 3-year period. The matching prediction intervals for the accident count will be wider.

For the Berman–Turner device we used a dummy point spacing of $\Delta = |L|/1000 = 285$ metres, agreeing with the exact result. The approximate MLE using the Berman–Turner device is 0.844 accidents per kilometre per 3 years. The `spatstat` result for the variance of the parameter β leads to a 95 % confidence interval for β of $[0.745, 0.958]$, which is in close agreement with the result calculated above.

3.4.2 Linear model: Accident intensity proportional to traffic volume model

The Geelong data includes a measure of traffic volume using the Annual Average Daily Traffic (AADT) at each location u. The AADT at location u is the total number of vehicles travelling past that location, in both directions, over a year, divided by the number of days in that year. In practice the number of vehicles travelling past the location over a year is estimated by extrapolating from sample vehicle counts. The AADT at location u will be given by $M(u)$. When using AADT as an offset or covariate in loglinear models it is convenient to define $A(u) = \log M(u)$. See Figure 3.2 for a plot of Annual Average Daily Traffic (AADT) for the Geelong data.

Traffic volume as measured by AADT is used in a proportional model as described in Section 3.2.2. The model postulates that accident intensity, $\lambda(u) = \beta M(u)$, at a location u is proportional to the AADT at u. The model is not intended as a realistic model for accident intensity but, like the constant intensity model, it is a benchmark for the evaluation of more suitable models.

The parameter estimate $\hat{\beta}$ can be computed directly from the explicit MLE formula from

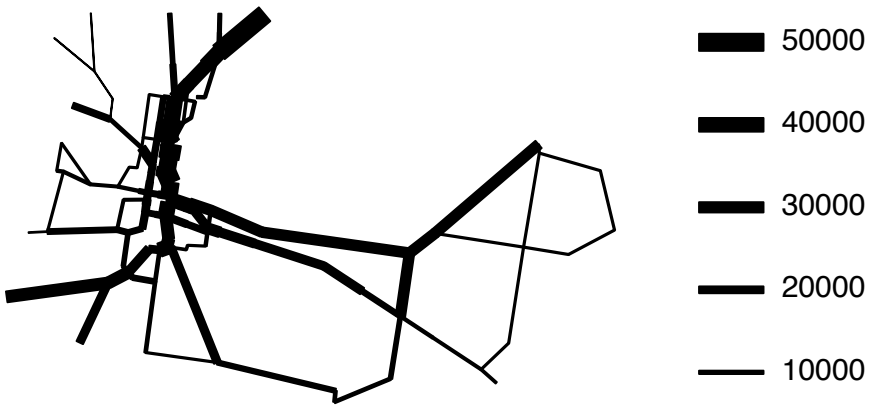

Figure 3.2: Annual Average Daily Traffic (AADT) for Geelong major roads, portrayed as a line width plot.

section 5.2, $\hat{\beta} = n(\mathbf{x})/M_L$, where $M_L = \int_L M(u)\,\mathrm{d}u$. The value of the integral is estimated from the data using numerical integration by dividing up the network into many small intervals as $\int_L M(u)\,\mathrm{d}u \approx 4.333 \times 10^6$. The maximum likelihood estimate of β can also be computed by using an offset term in a loglinear model $\lambda_{\boldsymbol{\theta}}(u) = \exp(\log\beta + \log M(u))$. Both methods produce a similar result, $\hat{\beta} = 5.60 \times 10^{-5}$. The fitted model is $\lambda_{\boldsymbol{\theta}}(u) = 5.60 \times 10^{-5} M(u)$. A plot of the fitted intensity (not shown) only differs from Figure 3.2 in proportion.

Expressing the result analogously to the constant intensity, the model average accident count for a single vehicle travelling one kilometre is estimated to be $\hat{\beta}/(3 \times 365) = 5.12 \times 10^{-8}$. Or, inverting, the average number of vehicles, each one kilometre, to produce one accident is estimated to be 1.95×10^7.

The plug-in estimate of the Fisher information is $\boldsymbol{I}(\hat{\beta}) = M_L/\hat{\beta} = 7.74 \times 10^{10}$. The estimate of the variance of $\hat{\beta}$ is $\hat{v} = \boldsymbol{I}(\hat{\beta})^{-1} = 1.29 \times 10^{-11}$. The standard error for $\hat{\beta}$ is $\sqrt{\hat{v}} = 3.59 \times 10^{-6}$. The 95% confidence interval for the parameter β has limits $\hat{\beta} \pm 1.96\sqrt{\hat{v}}$, that is, $[4.90 \times 10^{-5}, 6.30 \times 10^{-5}]$.

These results can be scaled for varying traffic volumes to give the baseline average accident estimate for different traffic volumes. For example, along a 10 km stretch of road that has a constant AADT of 20,000 vehicles, the 95% confidence interval for the expected number of accidents over 3 years along that 10 km is computed to be $[10 \times 20000 \times 4.9 \times 10^{-5}, 10 \times 20000 \times 6.3 \times 10^{-5}]$ or 9.8 to 12.6 accidents over 3 years.

3.4.3 Loglinear model: Cartesian coordinates with traffic volume offset

Our next example investigates evidence for a spatial trend in the accident rate. The proposed model is

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(\theta_0 + \theta_1 Z_1(u) + \theta_2 Z_2(u) + A(u)), \tag{3.23}$$

where in this case $Z_1(u)$ and $Z_2(u)$ are the x and y coordinates, respectively, of any location u on the linear network. Additionally $A(u) = \log M(u)$ as defined in Section 3.4.2 is an offset term that introduces a baseline intensity $M(u)$ as discussed in Section 3.2.2. This is an instance of the loglinear model, see equation (3.12); the parameter vector is $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2)$ and the vector-valued covariate function is $\mathbf{Z}(u) = (1, Z_1(u), Z_2(u))$. It is instructive to spell out the score equation for this model, to convince ourselves that the MLE is not available in closed form. The score is

$$\boldsymbol{U}(\boldsymbol{\theta}) = \begin{bmatrix} n - \int_L \lambda_{\boldsymbol{\theta}}(u)\,\mathrm{d}u \\ \sum_{i=1}^{n} Z_1(x_i) - \int_L Z_1(u)\,\lambda_{\boldsymbol{\theta}}(u)\,\mathrm{d}u \\ \sum_{i=1}^{n} Z_2(x_i) - \int_L Z_2(u)\,\lambda_{\boldsymbol{\theta}}(u)\,\mathrm{d}u \end{bmatrix}. \tag{3.24}$$

The maximum likelihood estimate (MLE) of the parameter vector $\boldsymbol{\theta}$ is the solution of the score equation $\boldsymbol{U}(\boldsymbol{\theta}) = \mathbf{0}$. Clearly the solution of the score equation is intractable.

Applying the Berman–Turner device with the same dummy spacing, the parameter estimates are $\hat{\boldsymbol{\theta}} = (-9.6273, -0.0332, 0.0244)$ so that the fitted model is

$$\lambda_{\boldsymbol{\theta}}(u) = M(u)\exp(-9.6273 - 0.0332 Z_1(u) + 0.0244 Z_2(u)). \tag{3.25}$$

The model predicts that given that there is equal traffic volume, the accident intensity decreases from left to right (west to east). Additionally, there is a smaller effect with intensity increasing from south to north, for equal traffic volumes. Figure 3.3 shows a spatial plot of the fitted intensity function.

The units in this case are accidents per kilometre of road over three years. For example, at a point on the network with coordinates $(12, 15)$ and a traffic volume of 15000 AADT the expected accident intensity is approximately 0.96 accidents per kilometre of road per three years. Moving to the west of the network to coordinate $(38, 15)$, with the same traffic volume, the expected accident intensity drops to about 0.40 accidents per kilometre of road per three years.

The Fisher Information for this intensity model is

$$\boldsymbol{I}(\boldsymbol{\theta}) = \int_L \begin{bmatrix} 1 & Z_1(u) & Z_2(u) \\ Z_1(u) & Z_1(u)^2 & Z_1(u)Z_2(u) \\ Z_2(u) & Z_1(u)Z_2(u) & Z_2(u)^2 \end{bmatrix} \lambda_{\boldsymbol{\theta}}(u)\,\mathrm{d}u. \tag{3.26}$$

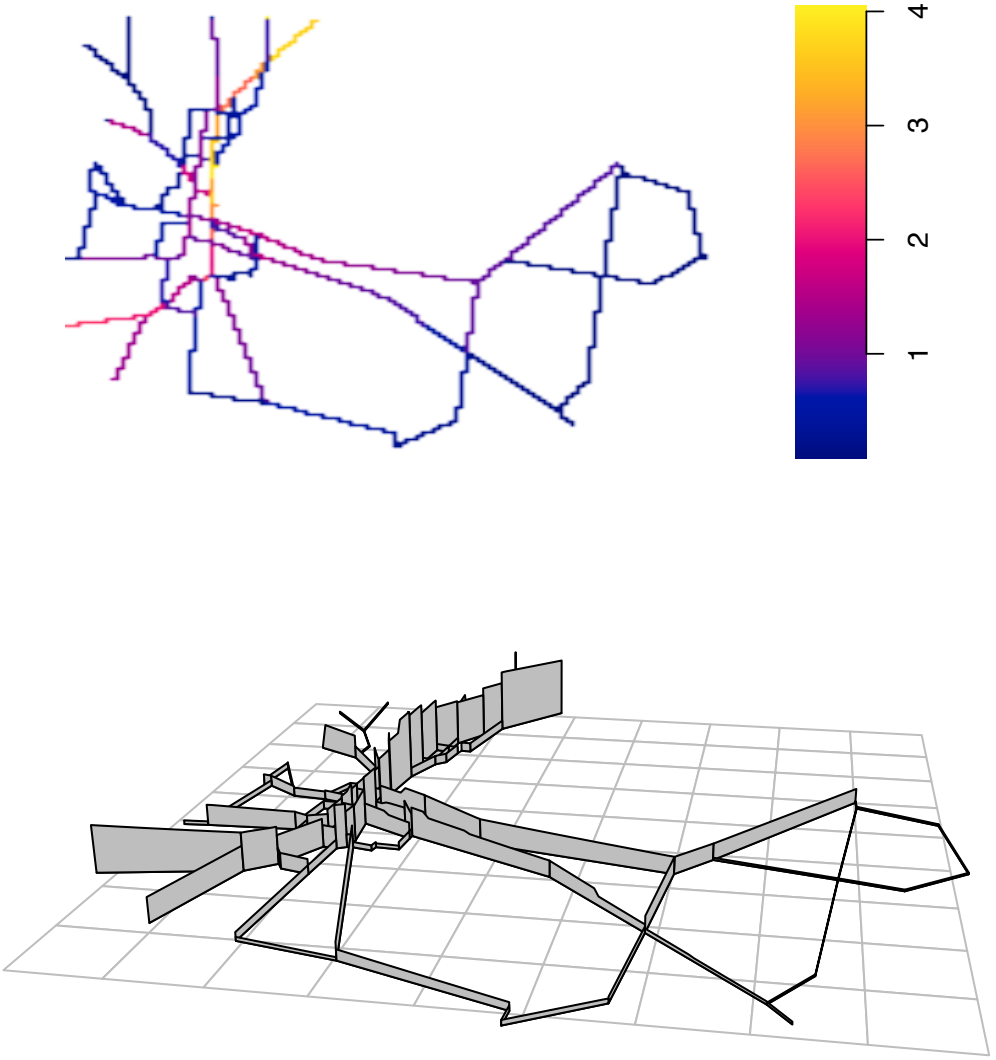

Figure 3.3: Fitted intensity of loglinear model for Geelong data using x and y coordinates with traffic volume offset. portrayed as a colour gradient (top) and a perspective plot with height proportional to fitted intensity (bottom).

To calculate a 95% confidence interval for the intensity $\lambda(u)$ at the location with coordinates $(12, 15)$ and a AADT of 15000 cars, we use equations (3.7)–(3.8) with $\boldsymbol{\kappa}_{\hat{\boldsymbol{\theta}}}(u) = \mathbf{Z}(u) = (1 \quad 12 \quad 15)$, obtaining $\log \lambda_{\hat{\boldsymbol{\theta}}}(u) = \hat{\boldsymbol{\theta}}\mathbf{Z}(u) + \log M(u) = -0.0453$ and $\hat{v} = \mathbf{Z}(u)^\top \boldsymbol{I}(\hat{\boldsymbol{\theta}})^{-1}\mathbf{Z}(u) = 0.00460$, so that the confidence interval for $\log \lambda(u)$ has limits $-0.0453 \pm 1.96 \times \sqrt{0.00460}$ or $[-0.178, 0.088]$; exponentiating this, the confidence interval for $\lambda(u)$ is $[0.84, 1.09]$. The analysis of deviance (or equivalently the likelihood ratio test) indicates very significant evidence that

θ_1 is nonzero, and significant evidence that θ_2 is nonzero, suggesting that a model depending on both the x coordinate and the y coordinate is better than a model using just one of the coordinates.

3.4.4 Loglinear model: Distance to nearest intersection with traffic volume offset

Consider the loglinear model

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(\theta_0 + \theta_1 d(u) + A(u)), \tag{3.27}$$

where $d(u)$ is the distance from location u to the nearest road intersection, measured by the shortest path in the network. Accident rates per kilometre of road are known to be generally higher in the vicinity of intersections compared to other road sections [77]. Thus this model is particularly relevant to traffic accident data. A road intersection is defined as the meeting of at least three roads. The term $A(u)$ is an offset term as defined in Section 3.4.2. The model has parameter vector $\boldsymbol{\theta} = (\theta_0, \theta_1)$ and vector covariate $\mathbf{Z}(u) = (1, d(u))$. The analysis is similar to the previous model. The score equation is analytically intractable, and can only be solved by numerical approximation. Using the Berman–Turner device with the same dummy points as above, the fitted model is

$$\lambda(u) = M(u) \exp(-9.381 - 0.264 d(u)), \tag{3.28}$$

where $A(u) = \log M(u)$ as defined in Section 3.4.2. Since the coefficient $\hat{\theta}_1 = -0.264$ is negative, the model predicts that accident intensity is higher at locations closer to an intersection. For example, substitution into the fitted model intensity (3.28) predicts that the accident intensity at 4 kilometres distance from an intersection and at a location with an AADT volume of 10,000 vehicles per day (0.29 accidents per kilometre per three years) is less than half the accident intensity within 400 metres of an intersection with the same traffic volume (0.76 accidents per kilometre per 3 years). The analysis of deviance (likelihood ratio test) indicates very significant evidence that θ_1 is nonzero.

3.4.5 Extension of nearest-intersection model

The model (3.27) can be further refined by taking into account the characteristics of the road intersections. Extending the model investigates the possibility that proximity to different types of intersections will affect the accident rate in different ways. Fitting a model that allows for these different effects will help assess whether these differences exist or not.

As an example, we can classify intersections into three-way, four-way, and more-than-four-way intersections. For any location u on the road network, let $r(u)$ be the number of roads

which meet at the intersection nearest to u. Thus $r(u) = 4$ means that the intersection nearest to u is a four-way intersection. Define $f(u) = \mathbf{1}\{r(u) = 4\}$, the indicator function that equals 1 if the intersection nearest to u is a four-way intersection, and equals 0 otherwise. Similarly define $m(u) = \mathbf{1}\{r(u) \geq 5\}$. Consider the intensity model

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(\theta_0 + \theta_1\, d(u) + \theta_2 f(u) + \theta_3 m(u) + A(u)). \tag{3.29}$$

This is an extension of the previous model (3.27) in which the accident risk predicted by (3.27) applies only when the nearest intersection is a three-way intersection. The three-way intersection case serves as a baseline for the intersection types. The extra terms involving θ_2 and θ_3 indicate that, for a given location u, the accident risk is increased or decreased by a factor $\exp(\theta_2)$ relative to the baseline if the nearest intersection is a four-way intersection, or a factor $\exp(\theta_3)$ if the nearest intersection is a meeting of five or more roads.

Thus the predicted intensity at location u is equal to the AADT term $M(u)$, and multiplied by the intercept term $\exp(\theta_0)$, multiplied by the distance to nearest intersection term $\exp(\theta_1\, d(u))$, multiplied by $\exp(\theta_2)$ for a four-way intersection, or multiplied by $\exp(\theta_3)$ for a five-way or more intersection.

The fitted model for the Geelong data is

$$\lambda(u) = M(u) \exp(-9.515 - 0.258 d(u) + 0.246 f(u) + 0.627 m(u)). \tag{3.30}$$

The fitted model indicates that the accident intensity decreases with distance from the nearest intersection, and increases with the complexity of the intersection. Thus a five-way intersection has an increased risk by a factor of $e^{0.627} = 1.872$ relative to three-way intersections. In the analysis of deviance the distance from the intersection is highly significant. The difference between three-way and four-way intersections is not significant. However the difference between three-way and more-than-four-way intersections is significant.

For example, if we again consider a location 400 metres from the nearest intersection with an Annual Average Daily Traffic volume of 10,000 vehicles per day, the fitted model predicts a accident intensity of $10000 \times \exp(-9.515 - 0.258 \times 0.4) = 0.661$ accidents per kilometre per three years at 400 metres if the nearest intersection is of degree three. For intersections of degree 4 and 5 the predictions are respectively $10000 \times \exp(-9.515 - 0.258 \times 0.4 + 0.246) = 0.851$ and $10000 \times \exp(-9.515 - 0.258 \times 0.4 + 0.627) = 1.245$ accidents per kilometre per three years. Figure 3.4 shows a spatial plot of the fitted intensity function. This model could be used to predict changes in accident rate if the road configuration is changed.

3.4.6 Loglinear model: Speed limit with traffic volume offset

Figure 3.5 shows the road speed limits for major roads in the study area. There are four speed limits: 60, 70, 80, and 100 km/hr. The 70 km/hr speed limit applies mainly along the highway

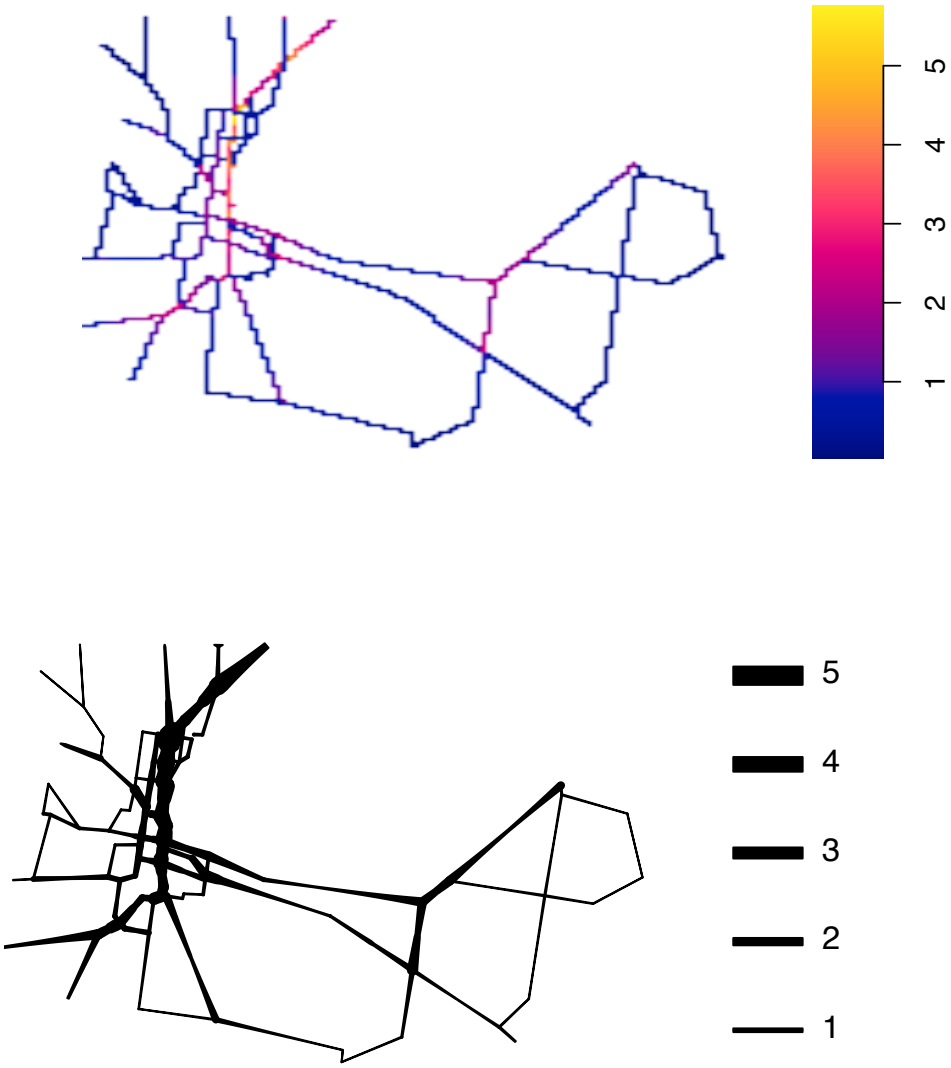

Figure 3.4: Fitted intensity of the loglinear model for Geelong data using distance to nearest intersection and intersection type with traffic volume offset, as a colour gradient plot (top) and line width plot (bottom).

route through the city of Geelong.

The speed limit could be treated either as a numerical covariate or as a factor. If it is taken to be a numerical covariate, then the simplest model would postulate that accident risk is a loglinear function of speed limit. This would imply that risk is a monotone function of speed limit. This is unlikely to hold for the Geelong data because the 70 km/h speed limit is associated with a different category of traffic.

Accordingly, we treat the speed limit as a factor (categorical variable) with four levels, similar to the treatment of intersection types in Section 3.4.4. The 60 km/hr level is used as

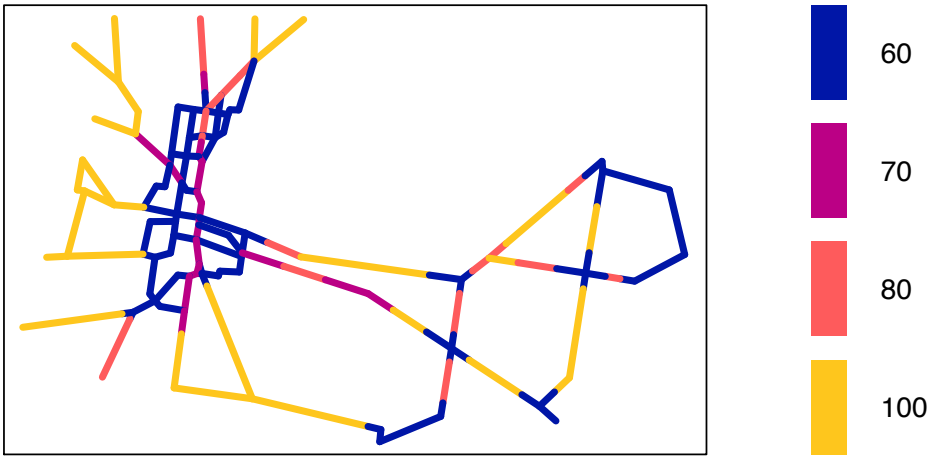

Figure 3.5: Speed limits for Geelong major roads, portrayed as a colour gradient plot.

the standard level to which the other three levels are compared. Let $sp(u)$ be the speed limit in km/hr at any location u. Define $f_{70}(u) = \mathbf{1}\{sp(u) = 70\}$, an indicator function that equals 1 where the speed limit equals 70 km/hr, and equals 0 otherwise. Similarly $f_{80}(u) = \mathbf{1}\{sp(u) = 80\}$ and $f_{100}(u) = \mathbf{1}\{sp(u) = 100\}$.

The proposed model is

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(\theta_0 + \theta_1 f_{70}(u) + \theta_2 f_{80}(u) + \theta_3 f_{100}(u) + A(u)), \tag{3.31}$$

where $A(u) = \log M(u)$ is an offset term for traffic volume as defined in Section 3.4.2. The parameter vector is $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3)$. The vector-valued covariate function is
$\mathbf{Z}(u) = (1, f_{70}(u), f_{80}(u), f_{100}(u))$. Applying the Berman–Turner device, the parameter estimates are $\hat{\boldsymbol{\theta}} = (-9.3020, -0.1751, -0.9837, -1.2974)$.
So that the fitted model is

$$\lambda_{\boldsymbol{\theta}}(u) = M(u) \exp(-9.3020 - 0.1751 f_{70}(u) - 0.9837 f_{80}(u) - 1.2974 f_{100}(u)).$$

For example, a location with a traffic volume of 10000 AADT and a speed limit of 60km/hr has, according to the model, an expected accident rate of $10000 \times \exp(-9.3020) = 0.91$ accidents per kilometre of road per three years. A location with a traffic volume of 10000 AADT and a speed limit of 100 km/hr has, according to the model, an expected accident rate of $10000 \times \exp(-9.3020 - 1.2974) = 0.25$ accidents per kilometre of road per three years. Thus the model

predicts that traffic accidents decrease in the locations with higher speed limits given equivalent traffic volumes. This result does not, of course, establish a causal connection between speed limit and accident rate, nor does it suggest that increasing the speed limit would lower the accident rate. A more plausible interpretation is that high speed limits are assigned to roads considered to be relatively safe, based on road characteristics, past accident history and traffic volumes. The result itself is based on model assumptions about the dependence of accident rate on covariates, and ignores other possible covariates.

The fitted intensities are plotted spatially in Figure 3.6. Significant differences in risk were found between the 60 and 80 km/h limits, and between the 60 and 100 km/h limits, using the likelihood ratio test.

Statistical software packages allow the option of treating a factor as *ordered*. This would be meaningful for the speed limit covariate, because the speeds are numerically ordered. However the default procedure for handling ordered factors is effectively to convert the levels to integers 1 to m and to fit models where the log intensity is a polynomial function of these integers. This would be inappropriate in the present context.

3.4.7 Loglinear model: Accident rate as a power of traffic volume

In this model the canonical covariate is the log of traffic intensity, $\log M(u)$ as defined in Section 3.4.2. This is not presented as a realistic model but the intention is to check the proposition that in general accident rates are approximately proportional to the square root of the traffic volume. Jurewicz and Bennett [77] note "that casualty accident frequency is strongly traffic volume dependent. Literature suggests that the number of accidents per km is related to the square root of the AADT (Austroads in press). Analysis of the Australian data shows the power exponent varies between 0.5 and 0.7 for most road types, but some fail to show this trend."

The proposed model is

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(\theta_0 + \theta_1 \log M(u)) = e^{\theta_0} M(u)^{\theta_1}. \tag{3.32}$$

The model predicts that the number of traffic accidents are proportional to the traffic volume with a power exponent of 0.44, since $\hat{\theta}_1 = 0.44$. The fitted model for the Geelong Data is $\lambda_{\boldsymbol{\theta}}(u) = 0.014 M(u)^{0.44}$. Thus the proposition that the accident rate is approximately proportional to the square root of the traffic volume is supported by the Geelong data.

3.4.8 Loglinear model: Using the most significant covariates

Finally we fit a model combining some of the most significant covariates. For any location u on the linear network, the model uses the x coordinate of u, its distance from u to the nearest

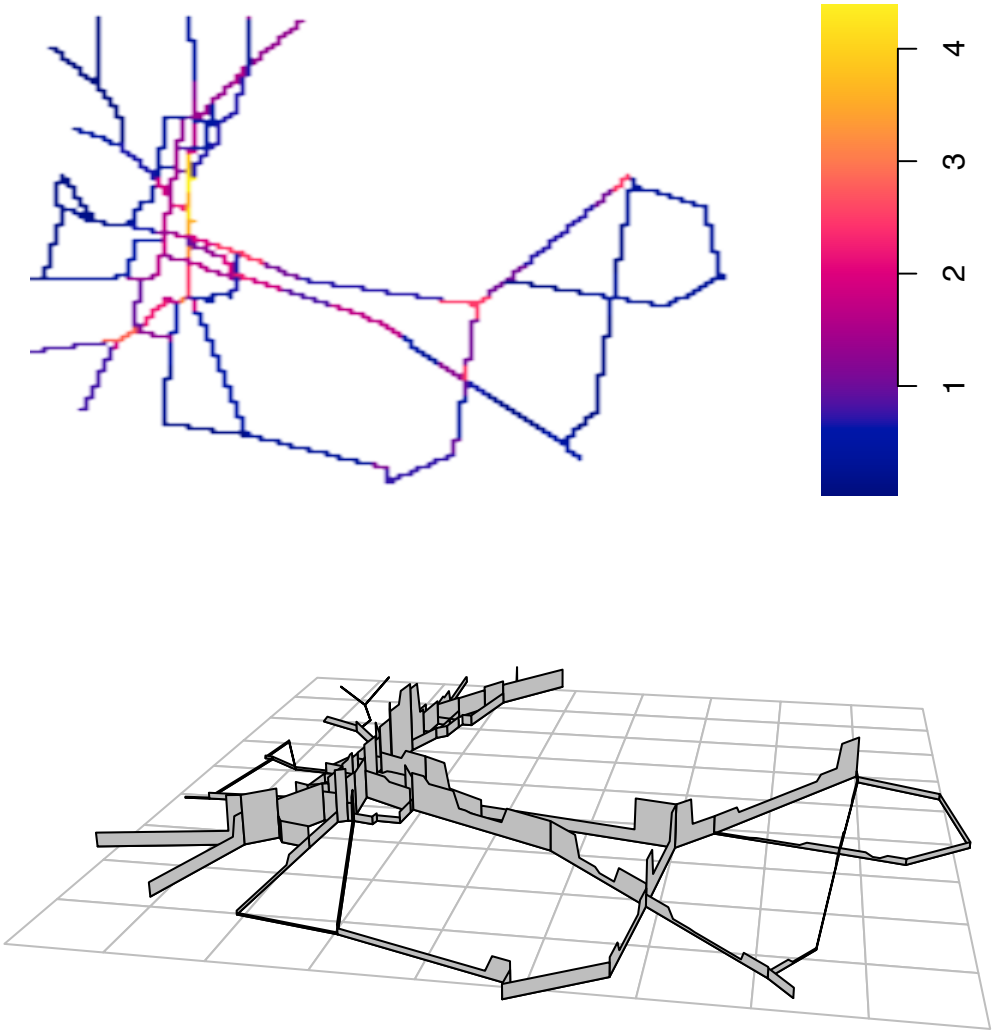

Figure 3.6: Fitted intensity (accident rate) of model depending on speed limit with a traffic volume offset, with intensity values represented by a colour gradient plot (top) and perspective plot (bottom).

road intersection measured by the shortest path in the network, $d(u)$, and traffic volume at u is included this time as a covariate with its own coefficient to be estimated by the model-fitting.

The proposed model is

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(\theta_0 + \theta_1 Z_1(u) + \theta_2 d(u) + \theta_3 A(u)) \tag{3.33}$$

The fitted model is

$$\lambda_{\boldsymbol{\theta}}(u) = \exp(-3.361 - 0.0241 Z_1(u) - 0.234 d(u) + 0.418 A(u)) \tag{3.34}$$

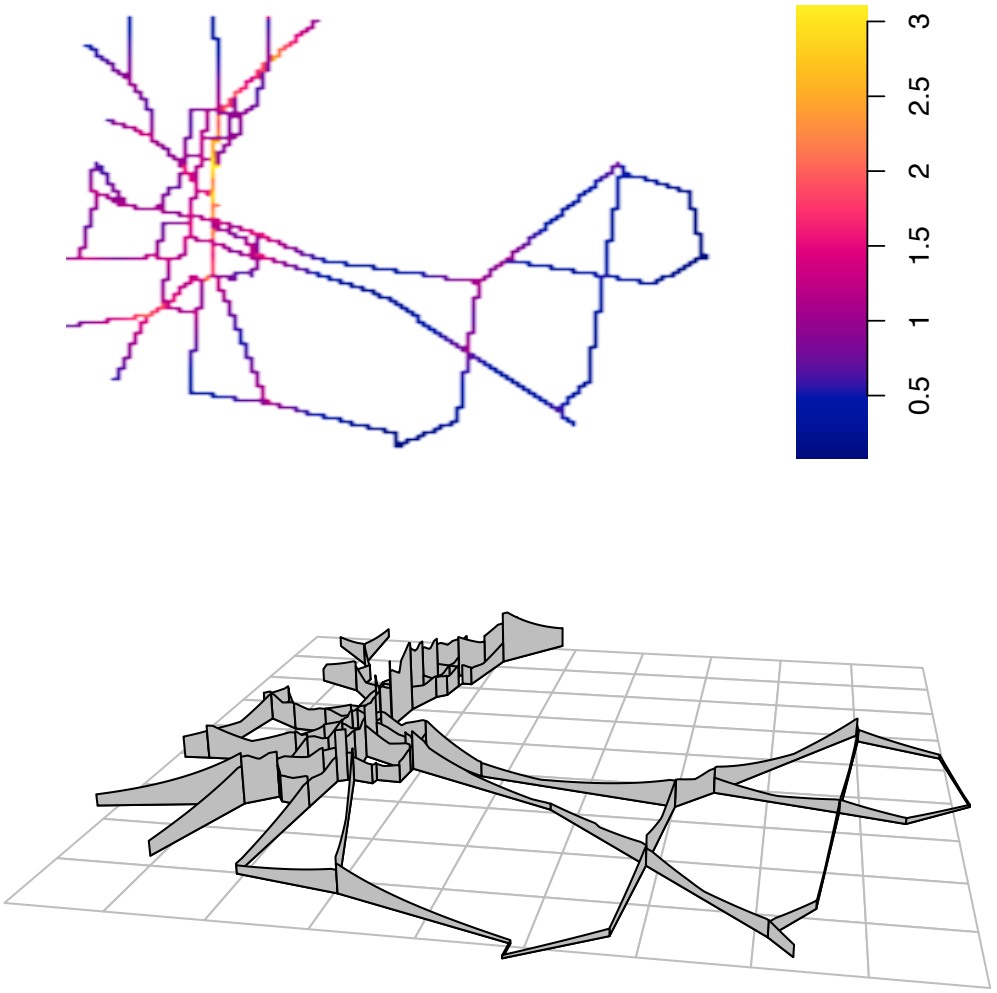

Figure 3.7: Fitted intensity (accident rate) of model depending only on most significant covariates, with intensity values represented by a colour gradient plot (top) and a perspective plot (bottom).

which can also be expressed as $\lambda_{\boldsymbol{\theta}}(u) = M(u)^{0.418} \exp(-3.361 - 0.0241 Z_1(u) - 0.234 d(u))$.

The estimate of the coefficient of the x coordinate is negative, implying that accident risk reduces from west to east. The estimate of the coefficient of $d(u)$ is negative, implying that accident risk reduces with distance from an intersection. The coefficient of the log of traffic volume parameter implies that accident risk increases with traffic volume and that it increases proportional to the 0.418 power of traffic volume, similar to the result in Section 3.4.7.

A plot of the fitted intensity is shown in Figure 3.7. This model could also be used to predict changes in accident rates if traffic volumes were changed.

3.5 Aggregation bias

Consider a road network with traffic accidents over a fixed period of time idealised as a linear network L with point events. Further consider that the distribution of traffic accidents is a known inhomogeneous Poisson process. Further, the true intensity, $\lambda(u)$, of the inhomogeneous Poisson process depends only on a covariate $Z(u)$ that varies with respect to location u over the network. Thus the true expected number of accidents on any subsection of the network $B \subset L$, is $\mathbb{E}[N(B)] = \int_B \exp(\theta Z(u))\,\mathrm{d}u$.

We wish to estimate the parameter θ that links the covariate $Z(u)$ to the intensity $\lambda(u)$. We use the crash-frequency method and fit a loglinear Poisson count regression. Thus the network is broken into road segments, and the covariate $Z(u)$ used is the mean value over each road segment; $\overline{Z(u)} = \frac{1}{|B|}\int_B Z(u)\,\mathrm{d}u$, where B represents the road segment. The expected number of accidents on a road segment B using the loglinear Poisson count regression is $|B|\exp(\theta\overline{Z(u)})$, or $\mathbb{E}[N(B)] = |B|\exp\left(\frac{1}{|B|}\int_B \theta Z(u)\,\mathrm{d}u\right)$. By Jensen's inequality

$$|B|\exp\left(\frac{1}{|B|}\int_B \theta Z(u)\,\mathrm{d}u\right) \leqslant \int_B \exp(\theta Z(u))\,\mathrm{d}u. \tag{3.35}$$

Thus, unless the covariate $Z(u)$ is truly constant along each segment, the loglinear Poisson count regression is a misspecification because the true parameter θ used with the aggregated data will always predict a smaller expected number of accidents on a road segment than the true expected number of accidents. Thus when the loglinear Poisson count regression is fitted the parameter estimate will tend to be larger than the true parameter θ. This means that if the loglinear Poisson count regression is now used to make predictions for a new set of data where the covariate $Z(u)$ is now constant, then the predicted expected number of accidents on a road segment will always be too high. If the new set of data has a varying $Z(u)$ and uses an aggregated mean of $Z(u)$ as the covariate then the results will be unpredictable since the aggregation of the varying $Z(u)$ will occur differently from the data that created the model.

The probability of a zero count of road accidents on a road segment B is $P(N(B) = 0) = \exp(-\mathbb{E}[N(B)])$. Thus if the prediction of the expected number of events on a road segment is problematic so will the predictions for zero accident counts.

We do not have a general proof but it does appear that for under-saturated aggregated crash-frequency models applied to the situation described above the resulting crash-frequency model typically will predict fewer zero counts than the true model. See below for a simple example.

It is well documented in crash-frequency literature that the crash-frequency model approach often fails to predict enough zero counts on road segments [95]. It may not be possible to determine whether this zero-inflation is caused by an unknown factor or whether the extra zero counts are merely an artifact of the data aggregation.

Furthermore, if a model has an inaccurate mean, then the estimation of the amount of dispersion will be problematic. For example, finding that a Poisson model fitted using aggregated data has too much dispersion will not necessarily indicate that Poisson was the incorrect model choice. If the crash-frequency method systematically produces an inaccurate mean, is it reasonable to continue fine-tuning further developments of that model?

3.6 Applicability of Poisson models

3.6.1 Dependence

It may be reasonable to assume that individual road accidents are stochastically independent in most cases. However, this assumption could be violated at very short distances and short time lags, for example, by complex multiple-vehicle accidents; by situations where the risk of an accident is increased by the occurrence of a previous accident (e.g., obstructions on road, spillage, reduced visibility); and by situations where the risk of an accident is decreased after a previous accident (e.g. increased vigilance, lowered speed limit, Police presence).

There is substantial literature on analysis of accident counts (the number of accidents on a particular road segment or in a particular category of accident); see [112, 69, 103, 94, 2]. Much of the literature on count data seems to favour *negative-binomial* models for count data instead of *Poisson* models for counts. A Poisson point process implies that accident counts will be Poisson-distributed, by property (P1) or (P1'). It also implies that, if we condition on the total number of accidents in the database, then the accident counts in a particular road or category will be binomially distributed, by property (P3) or (P3'). Hence the negative binomial distribution for aggregate counts is not consistent with a Poisson point process. The negative binomial distribution is "overdispersed" relative to the Poisson (and relative to the binomial) meaning that its variance to mean ratio is greater than 1.

The negative binomial distribution arises as a mixture of Poisson distributions: if Z is a random variable with a gamma distribution, and given $Z = z$, the random variable N is Poisson with mean z, then the marginal distribution of N is negative binomial. Thus, a simple scenario that supports a negative binomial distribution for accident counts, is a *Cox process*: a Poisson process where the intensity function is random with gamma-distributed values. This could be realistic in applications if we believe that some of the factors affecting accident risk are stochastic, for example, weather conditions. We will explore this in a future paper.

3.6.2 Accidents at an intersection

Many accidents occur at an intersection. For such accidents, the recorded locations are so accurate, viewed at the scale of a typical spatial analysis, that we may as well assume these accidents occur exactly at the middle of the intersection. This violates the model assumptions because, for any fixed location s, the model implies that there is zero probability that a random point of $\mathbf{X}$ will occur exactly at s.

It may therefore be appropriate to modify the point process model to allow a nonzero probability of placing a point exactly at the middle of an intersection. This means that the process no longer has an intensity function, or at least, the intensity function must be augmented by nonzero probabilities of occurrence at each intersection; and multiple accidents could occur at the same intersection over a time period.

This can be handled by an extension of the model presented above. Effectively we can think of the model as a sum of two random processes, the accidents along roads ($\mathbf{X}$) and the accidents at intersections ($\mathbf{Y}$). We model $\mathbf{X}$ as a Poisson point process, as described above. We model $\mathbf{Y}$ as a collection of random variables N_j for $j = 1, \ldots, J$, where N_j is the number of accidents recorded at road intersection j. We assume that N_j is a Poisson random variable with mean ν_j, and that the variables N_j are independent of each other and also independent of $\mathbf{X}$. This is a promising topic for future research, but is not covered in this thesis.

Chapter 4

Kernel estimation using diffusion

As stated in Chapter 2, there is currently no general agreement on how to perform kernel smoothing on a Linear Network. Previous attempts at kernel density estimation on a linear network have either been a form of the naive approach (Section 2.4.3) or heuristic approaches that are slow to compute (Section 2.4.4)

In this chapter we develop a firmer foundation for kernel density estimation on a linear network by exploiting its connection to diffusion on the network. Recall that, on the one-dimensional line, the Gaussian kernel is intimately related to Brownian motion. The connection between kernel smoothing and diffusion has previously been exploited to improve performance, edge correction and bandwidth selection for kernel density estimation of real-valued data [36, 27]. On a network the appropriate counterpart of the Gaussian kernel is the heat kernel, the occupation density of Brownian diffusion on the network. The connection with diffusion provides a sound statistical rationale, and helps establish many good theoretical properties. The heat kernel is a solution of the classical time-dependent heat equation.

In stochastic process theory there are results which express the heat kernel on a network as an infinite sum, over all paths in the network, of weighted contributions involving the Gaussian probability density [83, 84]. Many of the existing heuristic kernel estimators can also be represented as *finite* sums, over paths shorter than a given maximum length, of the values of the one-dimensional kernel [109, 110, 124]. This permits us to compare the different estimators, and show that the diffusion estimator is mathematically equivalent to an infinite-series extension of Okabe et al.'s "equal-split continuous" method ([109], sec. 5; [110], sec. 9.2.3).

A diffusion estimator is proposed in Section 4.1. Theoretical properties of the diffusion estimator are proved in Section 4.2. A fast numerical algorithm for computing this estimator is described in Section 4.3 and its numerical stability is established. Section 4.4 applies the method to the Geelong accident data, and Section 4.5 reports on computation times. Bandwidth selection techniques are proposed and demonstrated in Section 4.6. Edge corrections are discussed in Section 4.7.

4.1 Kernel smoothing and diffusion

On the real line, Chaudhuri and Marron [36] drew attention to the relation between kernel density estimation and diffusion, and Botev *et al.* [27] exploited this to develop improved density estimators which intrinsically include edge corrections and have faster convergence rates. In two dimensions, Barry and McIntyre [18] developed estimators for point process intensity based on two-dimensional diffusion. Here we develop kernel density estimation on a linear network using the counterpart of the Gaussian kernel.

4.1.1 Diffusion on the real line

Kernel smoothing of a set of data points $x_1, \ldots, x_n$ on the real line by a kernel function κ is often explained as the result of giving each data point x_i a random displacement, yielding $y_i = x_i + v_i$, where v_i is a random variable with probability density κ. The displaced point y_i then has probability density $g(x) = \kappa(x - x_i)$ so the random pattern of displaced points $y_1, \ldots, y_n$ has intensity function $\lambda(x) = \sum_i \kappa(x - x_i)$.

Vector displacement is not well-defined on a linear network. Instead, we may suppose that the data points are subjected to random motion (diffusion) on the network.

On an infinite straight line, *Brownian motion* is a random process $\{X(t),\, t \geq 0\}$ such that, for any time points $0 \leq t_1 < t_2 < \ldots < t_k$, the increments $X(t_2) - X(t_1), \ldots, X(t_k) - X(t_{k-1})$ are independent Gaussian random variables with mean 0 and variances $t_2 - t_1, \ldots, t_k - t_{k-1}$, respectively. Let $\varphi_t(x) = (1/\sqrt{2\pi t}) \exp(-x^2/(2t))$, for $-\infty < x < \infty$, denote the Gaussian density with mean 0 and variance $\sigma^2 = t$. For a Brownian motion $\{X(t),\, t \geq 0\}$, started at time $t = 0$ at a position x_0, the probability density of $X(t)$ at a later time t is $f_t(x) = \varphi_t(x - x_0)$, the Gaussian density with mean x_0 and variance $\sigma^2 = t$. For a Brownian motion started at time $t = 0$ at a *random* position X_0 with probability density $p(x)$, the probability density of X_t is

$$f_t(x) = \int_{-\infty}^{\infty} p(u)\varphi_t(x - u)\, \mathrm{d}u. \tag{4.1}$$

The function $f_t(x)$ is also the solution of the classical time-dependent *heat equation*

$$\frac{\partial f}{\partial t} = \beta \frac{\partial^2 f}{\partial x^2}, \tag{4.2}$$

with initial condition $f_0(x) = p(x)$ and with thermal diffusivity constant $\beta = 1/2$. The representation (4.1) of the solution of the heat equation is a kernel operator with kernel φ_t, so the *"heat kernel"* on the real line is φ_t. Thus, the usual Gaussian kernel estimator of a set of data points $x_1, \ldots, x_n$ on the real line is the sum of the heat kernel values $\varphi_t(x - x_i)$.

4.1.2 Diffusion on a linear network

Brownian motion on a linear network is a special case of a diffusion on a graph [62, 99]. It is a continuous-time Markov process $\{X_t,\, t \geq 0\}$, which is equivalent to one-dimensional Brownian motion on each segment of the network. At any instant where the process reaches a vertex of the network, of degree m say, it is equally likely to continue along any of the m edges incident at the vertex (including the edge it has just come from). In particular, it is instantaneously reflected from a terminal endpoint.

The probability density of Brownian motion on a linear network at time t, $f_t(x), x \in L$, satisfies the classical heat equation on the network [62, 99]. That is, at any location $x \in L$, other than a vertex, the analogue of (4.2) holds. Note that the second spatial derivative $\partial^2 f/\partial x^2$ is well defined, regardless of the choice of local coordinates on the line segment. Additionally, at any vertex v, the density f_t is continuous, and satisfies a property corresponding to the conservation of heat flow:

$$\sum_{v' \sim v} \left. \frac{\partial f}{\partial x_{[v,v']}} \right|_v = 0, \tag{4.3}$$

where the sum is over all edges $[v, v']$ that are incident at vertex v, and

$$\left. \frac{\partial f}{\partial x_{[v,v']}} \right|_v = \lim_{h \downarrow 0} \frac{f(v + h(v' - v))}{h\|v - v'\|}$$

is the first spatial derivative of f at v in the direction towards v'.

Any solution of the heat equation (4.2) and the heat flow conservation condition (4.3) on the linear network, with initial condition $f_0(x) = p(x)$, has a representation as a kernel operator

$$f_t(x) = \int_L p(u)\kappa_t(x \mid u)\, \mathrm{d}u, \tag{4.4}$$

where $\kappa_t(x \mid u)$ is the *heat kernel* on the network. Intuitively, $\kappa_t(x \mid u)\, \mathrm{d}x$ is the probability that a Brownian motion on the network, started at location $u \in L$ at time 0, will fall in the infinitesimal interval of length $\mathrm{d}x$ around the point x at time t. We can also think of $\kappa_t(x \mid u)$ as the transfer function from temperature at location u at time 0 to temperature at location x at time t.

Note that condition (4.3) implies that the first spatial derivative of f must be zero at any terminal endpoint. In physical terms this corresponds to assuming that the network is insulated so that no heat escapes from it, including from terminal endpoints.

Definition 4.1. Let $x_1, \ldots, x_n$ be a point point pattern on a linear network L. The *diffusion estimator* of intensity $\lambda(u)$, with bandwidth $\sigma = \sqrt{t}$, is

$$\widehat{\lambda}(u) = \sum_{i=1}^{n} \kappa_{\sigma^2}(u \mid x_i), \qquad u \in L. \tag{4.5}$$

4.1.3 Explicit representation of heat kernel

The heat kernel on a linear network has been studied from the viewpoint of stochastic process theory and potential theory in [83, 84, 85, 63, 64]. Its explicit form is given by the following result

Theorem 4.1 ([83, Corollary 3.4])**.** The heat kernel on a linear network is

$$\kappa_t(u \mid x) = \sum_{\pi} a(\pi)\varphi_t(\ell(\pi)), \tag{4.6}$$

where φ_t is the Gaussian density with variance t, the sum is over *all* paths from x to u, and

$$a(\pi) = \prod_{v_i \in \pi} \left(\frac{2}{deg(v_i)} - \delta_i \right), \tag{4.7}$$

where δ_i is the indicator defined for $0 < i < P$ by $\delta_i = \mathbf{1}\{e_i = e_{i-1}\}$, equal to 1, if the path reverses back on itself at vertex v_i, while $\delta_0 = \delta_P = 0$.

Details of the result are in [83, eq. (2.24), (3.11), (3.12)]. Heuristically $\varphi_t(\ell(\pi))$ is the probability density for the diffusion to have travelled a total displacement $\ell(\pi)$ at time t, the summation accounts for different possible paths through the network, and the coefficient $a(\pi)$ is a combinatorial weight for the path. Note that $a(\pi)$ may be positive or negative.

The sum in (4.6) has an infinite number of terms, since it includes all paths from x to u without restriction. Theorem 4.1 implies that this infinite sum converges. The sum includes contributions from paths which reflect at a vertex. If a path π reflects at a *terminal* vertex v_i, the factor $2/deg(v_i) - \delta_i$ is equal to 1, so this reflection does not contribute to the path coefficient $a(\pi)$. Paths which reflect at a vertex of degree 2 have path coefficient $a(\pi) = 0$, and are effectively ignored.

Comparing Theorem 4.1 and Theorem 2.2 we find that the path coefficients are identical, $a^C(\pi) = a(\pi)$. If we take $k = \varphi_t$ in (2.29), and replace the finite sum in (2.29) by the infinite sum over all possible paths from x to u, the result is equivalent to (4.6). Thus, we can regard (4.6) as an extension of (2.29) to a kernel with unbounded support.

Note however that Algorithm 2.2 cannot be used to evaluate (4.6). Algorithm 2.2 assumes the kernel has finite halfwidth $h < \infty$, and effectively enumerates all paths of length shorter than h. If Algorithm 2.2 were to be applied in practice to a kernel with unbounded support, it would not terminate in finite time, because there would be infinitely many paths to enumerate. However, this can be contemplated as a mathematical procedure. Hence we get the following result.

Lemma 4.1. *If the "equal-split continuous" algorithm (Algorithm 2.2) were applied to the Gaussian kernel and allowed to run for an infinite time, the result would converge to the heat kernel.*

This result partially reconciles the diffusion approach with the other techniques. It is also useful in practice when the bandwidth $\sqrt{t}$ is very small, since in this case the "equal-split continuous" algorithm will converge rapidly, within numerical tolerance.

4.2 Properties of the diffusion estimator

Many mathematical properties of the diffusion estimator (4.5) can be derived from its connection to the heat equation and the diffusion.

4.2.1 Basic properties

Lemma 4.2. *Suppose* $\mathbf{X}$ *is a point process on the linear network* L *with true intensity function* $\lambda(u)$, $u \in L$. *Let* $\widehat{\lambda}(u)$ *be the diffusion estimator* (4.5). *Then* $\mathbb{E}[\widehat{\lambda}(u)]$ *is the solution, at time* $t = \sigma^2$, *of the heat equation on* L *with initial condition* $\lambda(\cdot)$.

Proof. The expectation of $\widehat{\lambda}(u)$ is, by Campbell's formula (2.5),

$$\mathbb{E}[\widehat{\lambda}(u)] = \int_L \kappa_t(u \mid x)\, \lambda(x)\, \mathrm{d}_1 x. \tag{4.8}$$

Comparing this with equation (4.4) yields the result. □

This result may be useful in calculations when the true intensity is known, and in model validation. In practical terms, the heat equation solver takes over the role of the Fast Fourier Transform for smoothing on a linear network.

Lemma 4.3. *If the intensity is constant,* $\lambda(u) \equiv \lambda$, *then the diffusion estimator is unbiased.*

Okabe and Sugihara [110] gave a constructive proof of the corresponding statement, in Lemma 4.3, for the equal-split continuous estimator. In our case, the connection to the heat equation provides a much simpler route.

Proof. If $\lambda(u) \equiv \lambda$, we have $\partial^2\lambda(u)/\partial u^2 = 0$. The solution of the time-dependent heat equation (4.2), with initial condition $\lambda(\cdot) \equiv \lambda$, has time derivative identically equal to zero, so that the solution is identically equal to λ for all t. By (4.8), $\mathbb{E}[\widehat{\lambda}(u)] \equiv \lambda(u)$. □

Using the arguments in Section 2.4.2, it follows immediately from Lemma 4.3 that

Lemma 4.4. *The heat kernel* $\kappa_t(u \mid x)$ *satisfies both* (2.14) *and* (2.18).

Note that the kernel is symmetric in the sense of (2.19).

Lemma 4.5. *If L is connected, then the heat kernel converges uniformly to a constant as $t \to \infty$:*

$$\kappa_t(u \mid x) \to \frac{1}{|L|}. \tag{4.9}$$

Proof. Since L is connected, the diffusion is irreducible and recurrent, and has a unique equilibrium distribution, to which it converges in distribution from any initial state [27]. An equilibrium state is a probability density $f(x)$ satisfying $\partial^2 f/\partial x^2 = 0$ and the conservatism condition (4.3). The only solution is $f(x) = 1/|L|$. By compactness of L, convergence is uniform. □

Lemma 4.6. *The heat kernel has the semigroup property*

$$\int_L \kappa_s(y \mid u)\kappa_t(u \mid x)\,\mathrm{d}_1 u = \kappa_{t+s}(y \mid x), \tag{4.10}$$

for any $s, t > 0$.

Lemma 4.7. *The diffusion estimator* (4.5) *is an unbiased estimator of intensity* ***if and only if*** *the intensity is constant on each connected component of L.*

Proof. Suppose unbiasedness holds for a particular bandwidth $\sigma = \sqrt{t}$. By (4.8),

$$\int_L \kappa_t(u \mid x)\,\lambda(x)\,\mathrm{d}x \equiv \lambda(u).$$

By induction using (4.10), this equation also holds when t is replaced by any integer multiple kt. But, as $kt \to \infty$, the left side converges to a constant on each connected component of L, by Lemma 4.5. Hence $\lambda(u)$ must be constant on each connected component. □

Additionally, suppose that X is a Poisson point process. Using a well known identity from Daley and Vere-Jones [41, p. 188], equation (2.6), the variance of the diffusion estimator is

$$\mathsf{var}[\widehat{\lambda}(u)] = \int_L \kappa_t(u \mid x)^2 \lambda(x)\,\mathrm{d}_1 x. \tag{4.11}$$

An unbiased estimator of (4.11) is $\widehat{v}(u) = \sum_i \kappa_t(u \mid x_i)^2$. Unlike the case for kernel smoothing on the real line, $\widehat{v}(u)$ is not a simple modification of the kernel estimator (4.5). Instead one would have to compute the heat kernel $\kappa_t(\cdot \mid x_i)$ separately for each data point x_i.

4.2.2 Asymptotics

Standard asymptotic results for kernel density estimation on the real line also hold for linear networks, because the connectivity of the network can be ignored for very small bandwidths [100, Section 7.2]. Suppose there are N i.i.d. observations from the probability density $f(u)$, assumed to be twice continuously differentiable. Let $N \to \infty$ and consider the heat kernel

density estimator $\widehat{f}$ with bandwidth $h = h_N \to 0$ such that $Nh_N \to \infty$. Adapting Botev *et al* [27, Theorem 1], for any location u that is not a vertex, the behavior of $\widehat{\lambda}(u)$ is asymptotically equivalent to that of the Gaussian kernel density estimator on the infinite real line, so that $\widehat{\lambda}(u)$ is asymptotically normal with asymptotic bias, see equation (4.12), and variance, see equation (4.13).

$$\mathbb{E}[\widehat{f}(u) - f(u)] = \frac{h^2}{2}\frac{\partial^2 f(u)}{\partial u^2} + O(h^4), \tag{4.12}$$

$$\mathsf{var}[\widehat{f}(u)] = \frac{f(u)}{2\sqrt{\pi}Nh} + o(1). \tag{4.13}$$

If $h = O(N^{-1/5})$, the mean square error is of order $O(N^{-4/5})$ and the estimator is pointwise consistent.

4.3 Numerical solution of the heat equation

An efficient strategy for computing the estimator (4.5) is to solve the time-dependent heat equation numerically up to time t using a finite difference algorithm.

This is clearly much faster than the path-tracing algorithms 2.1 and 2.2, described on pages 28 and 31 respectively. Computation time will be a quadratic, rather than exponential, function of the bandwidth, with an initialization cost that is linear in the number of points.

4.3.1 Discretization of a network

First, every line segment in the network is divided into equal sub-segments of length at most Δx, which we shall call *line elements*, for example see Figure 4.1. Each line segment in the original network, with length $\ell = m\Delta x + s$, say, where m is an integer and $0 \le s < \Delta x$, is divided into $m + 1$ line elements of equal length $r = \ell/(m + 1)$, satisfying $(m/(m + 1))\Delta x < r \le \Delta x$. Henceforth the line elements are treated as if they did have length Δx, which is to say that the linear network is approximated by a slightly different network in which the length of each segment has been rounded upwards to the nearest multiple of Δx.

The endpoints of the line elements are called *nodes* z_j, $j = 1, \ldots, J$. Each state of the algorithm will be a vector of real values λ_j, for each node z_j, representing the current value of the kernel estimator $\widehat{\lambda}(z_j)$.

The initial state of the algorithm is determined from the point pattern dataset $\mathbf{x}$ as follows. All values λ_j are initialized to zero. For each data point x_i, the line element $[z_j, z_{j'}]$ containing x_i is identified; then x_i is represented as a convex combination $x_i = pz_j + (1 - p)z_{j'}$ of the endpoints, where $p = \|z_{j'} - x_i\|/\Delta x$; the values λ_j and $\lambda_{j'}$ are incremented by the weights p

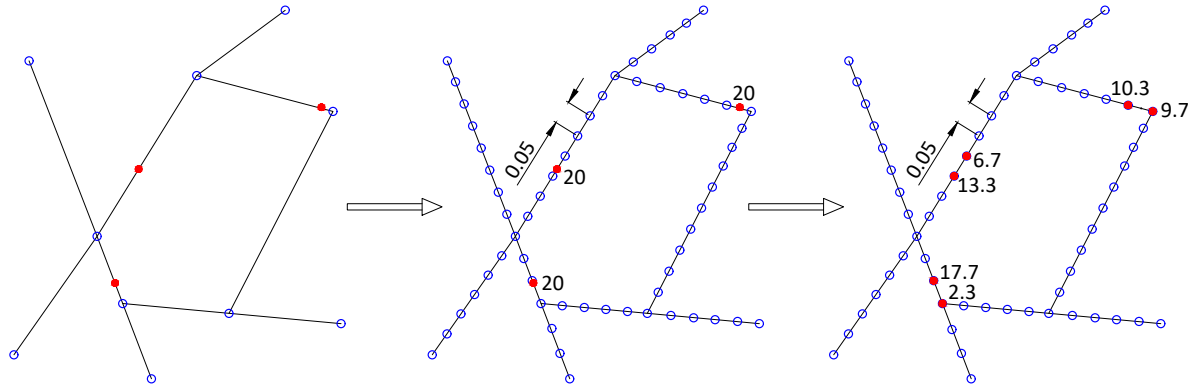

Figure 4.1: Discretization of a point pattern on a linear network. *Left panel*: Original point pattern data, consisting of 3 events on a network with 10 vertices. Each event has a mass of 1 unit. *Middle panel*: Extra vertices ("nodes") are placed evenly along each line segment at spacing $\Delta = 0.05$ units, dividing the network into line elements. the mass of each data point is spread evenly over the line element which contains it. *Right panel*: The mass associated with each event is shared between its two adjacent nodes. This is the initial condition for the propagation of the finite difference method.

and $1-p$, respectively. This is identical to the procedure typically used to discretize numerical values on the real line when computing a kernel estimator using the Fast Fourier Transform.

4.3.2 Finite difference approximation

Let $f_t(u)$ denote the solution of the time-dependent heat equation (4.2) on the linear network. We shall compute estimates of $f_t(u)$ at times t, which are multiples of a time step Δt. Let $f_j^{(k)}$ be the estimate of $f_{k\Delta t}(z_j)$, for $k = 0, 1, \ldots, M$. The time derivative will be replaced by the forward Euler finite difference

$$\frac{\partial f}{\partial t}(z_j) \approx \frac{f_j^{(k+1)} - f_j^{(k)}}{\Delta t} \tag{4.14}$$

and the space derivative by the central Euler finite difference

$$\frac{\partial^2 f}{\partial x^2}(z_j) \approx \frac{1}{(\Delta x)^2}\left[\sum_{j' \sim j}(f_{j'}^{(k)} - f_j^{(k)})\right] = \frac{1}{(\Delta x)^2}\left[\left(\sum_{j' \sim j} f_{j'}^{(k)}\right) - deg(z_j) f_j^{(k)}\right], \tag{4.15}$$

where $j' \sim j$ means that nodes z_j and $z_{j'}$ are joined by a line element, and $deg(z_j)$ is the degree of node z_j in the discretized network. The finite difference update corresponding to the heat equation (4.2)–(4.3) is

$$f_j^{(k+1)} = \alpha \sum_{j' \sim j} f_{j'}^{(k)} + (1 - \alpha\, deg(z_j)) f_j^{(k)}, \tag{4.16}$$

where $\alpha = \beta\,\Delta t/(\Delta x)^2$ with $\beta = 1/2$. While it is intuitively clear why (4.16) is a counterpart of the PDE (4.2), the connection with (4.3) may need explanation. At a node of degree not equal to 2, the update (4.16) adjusts f_j so that the discrete analogue of (4.3) would be satisfied, effectively applying a time-delayed version of (4.3). In particular, at a terminal node, (4.16) repeatedly enforces the boundary condition of insulated ends. In matrix form, the update (4.16) is

$$\mathbf{f}^{(k+1)} = \mathbf{A}\,\mathbf{f}^{(k)}, \tag{4.17}$$

where $\mathbf{f}^{(k)} = (f_1^{(k)}, \ldots, f_J^{(k)})^\top$ is the column vector of values at iteration k, and

$$\mathbf{A} = \mathbf{I} + \alpha\mathbf{M}, \tag{4.18}$$

where $\mathbf{I}$ is the identity matrix, and $\mathbf{M}$ is the centred incidence matrix with off-diagonal entries $m_{j,j'} = 1$ if $j \sim j'$ and $m_{j,j'} = 0$ otherwise, and diagonal entries $m_{j,j} = -\sum_{j'} m_{j,j'} = deg(z_j)$.

The matrix $\mathbf{A}$ is extremely sparse: most nodes have degree 2 so that most rows of the matrix have only 3 non-zero entries. Accordingly the update (4.17) can be performed efficiently using sparse matrix software.

For numerical stability, it is required that all eigenvalues of $\mathbf{A}$ should lie between 0 and 1. Eigenvalues of $\mathbf{A}$ are of the form $\zeta = 1 - \alpha\mu$, where μ is an eigenvalue of $\mathbf{L} = -\mathbf{M}$. The matrix $\mathbf{L}$ is the "Laplacian" of the graph of the discretized network L; the eigenvalues of $\mathbf{L}$ are all non-negative, and by a theorem of Anderson and Morley [7], the largest eigenvalue μ_1 satisfies

$$\mu_1 \le B = \max\{deg(z_j) + deg(z_{j'}) :\ j \sim j'\}. \tag{4.19}$$

A sufficient condition for all eigenvalues of $\mathbf{A}$ to lie between 0 and 1 is that $\alpha \le 1/B$, or equivalently

$$\Delta t \le \frac{(\Delta x)^2}{\beta B} = 2\frac{(\Delta x)^2}{B}. \tag{4.20}$$

Computation of the bound B for a linear network is straightforward.

The approximation of the tail of the kernel may be poor. At each iterative step, information propagates by a distance Δx, so the numerical approximation to the kernel has a half-width of $H = N\Delta x$, where $N = \sigma^2/\Delta t$ is the number of iterations executed. As a rule of thumb, we stipulate $H \ge 3\sigma$ to ensure adequate approximation of the tails, implying

$$\Delta t \le \frac{\sigma\,\Delta x}{3}. \tag{4.21}$$

In summary the algorithm is as follows:

Algorithm 4.3 (Diffusion Estimator, Finite Difference Algorithm)**.** Given a linear network L, a point pattern $\mathbf{x} = \{x_1, \ldots, x_n\}$ on L, and a smoothing bandwidth $\sigma > 0$,

1. Choose a spatial resolution Δx and discretize the linear network L as described in Section 4.3.1. Construct the associated matrix A in (4.18).
2. Correspondingly discretize the data points x_i. That is, initialize $\mathbf{f}^{(0)} = (f_1^{(0)}, \ldots, f_J^{(0)})$ by setting each $f_j^{(0)}$ equal to the sum of fractional masses at the node z_j induced by all data points x_i, as described in Section 4.3.1.
3. Choose the time step Δt to satisfy (4.20) and (4.21), where B is the Anderson-Morley bound (4.19) for the discretized network.
4. Recursively apply the update (4.17) for $N = \sigma^2/\Delta t$ steps.
5. Return $\mathbf{f}^{(N)}$ as the estimate $\widehat{\lambda}(\cdot)$.

We have implemented Algorithm 4.3 in the R language. The original network is specified by its vertices (given as x, y coordinate pairs) and edges (given as pairs of indices of vertices). The linear network geometry and accident locations are handled using functionality from the R package `spatstat` [17, 14]. The matrix $\mathbf{M}$ described below equation (4.18) is constructed as a sparse matrix using the `Matrix` package [19]. The matrix $\mathbf{A}$ defined in (4.18) is then evaluated, and the iterative update (4.17) is applied, using sparse matrix arithmetic. Code was byte-compiled [126] for additional speed in the examples.

4.4 Application to the Geelong accident data

Figure 4.2 shows the result of applying the heat kernel Algorithm 4.3 to the Geelong accident data of Figure 1.1 for the arbitrarily-chosen bandwidth of $\sigma = 1000$ metres.

The road network shown in Figure 1.1 has a total length of $286,585$ metres and a diameter (maximum distance between any two points by the shortest path) of $50,554$ metres. The maximum vertex degree is 5, and the Anderson-Morley bound is $B = 9$. We fixed $\Delta x = 50$ metres. The maximum permissible value of Δt according to (4.20) is $50^2/(9/2) = 554$ while the propagation constraint (4.21) imposes $\Delta t \leq \sigma\, \Delta x/3 = 16.7\sigma$.

4.5 Computation time comparisons

Table 4.1 shows computation times for the Geelong accident data using the equal-split discontinuous method (Algorithm 2.1, p28) and the equal-split continuous method (Algorithm 2.2,

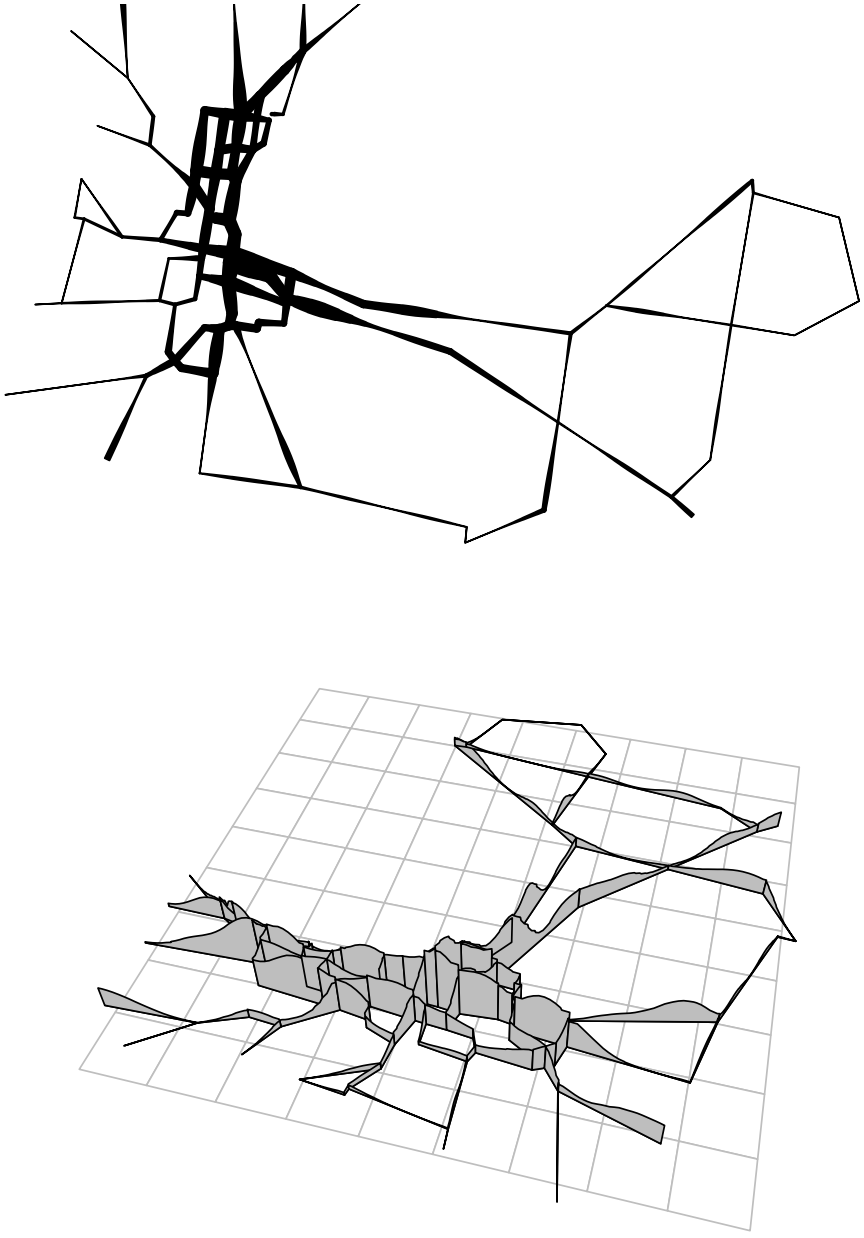

Figure 4.2: Diffusion estimate of intensity for Geelong road accident data with bandwidth $\sigma = 1000$ metres. *Top:* line width plot, line width proportional to intensity, in the style of Xie & Yan [133]; *Bottom:* perspective plot in the style of Okabe & Sugihara [110] with vertical height proportional to intensity.

	σ (metres)									
Method; Kernel	50	100	250	500	750	1000	1250	1500	1750	2000
Discontinuous, Epanechnikov	0.4	0.5	0.9	1.6	2.6	4.3	6.4	9.6	14	21
Discontinuous, Gaussian	0.5	0.8	1.6	4.2	8.5	16.7	32	63	120	230
Continuous, Epanechnikov	0.6	0.9	2.4	8.6	28	101	505	3150	–	–
Continuous, Gaussian	0.6	1.0	2.7	9.6	31	107	520	3220	–	–
Diffusion	**0.1**	**0.1**	**0.1**	**0.2**	**0.3**	**0.5**	**0.8**	**1.2**	**1.4**	**1.8**

Table 4.1: Computation times (in seconds) for diffusion kernel and Okabe group kernel estimates of accident intensity in Geelong.

p31). (See also Section 2.4.7 and Section 2.4.9)

The last row of Table 4.1 shows computation times for the diffusion estimator applied to the Geelong data. Despite the use of a finer spatial resolution ($\Delta x = 50$ metres instead of 200 metres) the finite difference algorithm is substantially faster than the path-tracing algorithms.

4.6 Bandwidth selection

Techniques for selecting the smoothing bandwidth σ for real-valued data [121, 132, 76, 93] can be adapted to linear networks. By the arguments in Section 4.2.2, statistical performance should be roughly the same on a linear network as on the real line. However, computational complexity and cost could be prohibitive for some techniques.

Using (4.12)–(4.13), the asymptotically optimal bandwidth (minimizing asymptotic integrated MSE) is

$$\sigma^*_A = \big(2\sqrt{\pi} N\, I(f)\big)^{-1/5}, \tag{4.22}$$

where $I(f) = \int_L (\partial^2 f/\partial u^2)^2 \,\mathrm{d}u$. With a suitable pilot estimate of f, it could be feasible to estimate $I(f)$ and calculate σ^*_A. The partial derivative $\partial^2 f/\partial u^2$ can be approximated using finite differences as in (4.15), or alternatively by taking the difference of two successive iterates of (4.16) and using (4.2) and (4.14). In our experience, this method is highly sensitive to the choice of pilot estimate.

Data-based bandwidth selection is made possible by the finite difference method, Algorithm 4.3, which computes the kernel estimator (4.5) for a sequence of intermediate values of σ^2 at no extra cost.

"Leave-one-out" cross-validation ([121]; [93, Sec. 5.3, pp. 87–95]) selects the bandwidth σ^*_- which maximises

$$C(\sigma) = \sum_i \log \widehat{\lambda}_{-i}(x_i), \tag{4.23}$$

where $\widehat{\lambda}_{-i}(x_i)$ is the estimate of $\lambda(x_i)$ based on all the data except x_i:

$$\widehat{\lambda}_{-i}(x_i) = \sum_{j \neq i} \kappa_t(x_i \mid x_j) = \widehat{\lambda}(x_i) - \kappa_t(x_i \mid x_i).$$

While this is feasible on a linear network, it is much more computationally intensive than on the real line, where $\widehat{\lambda}_{-i}(x_i) = \widehat{\lambda}(x_i) - k(0)$ can be calculated easily. Unless there is a simple approximation to $\kappa_t(x \mid x)$, calculation of (4.23) on a network effectively requires us to run the finite difference algorithm separately for each data point. This is computationally expensive, but does also allow estimation of the variance, as described under equation (4.11). Similar comments apply to the effective degrees of freedom $\text{dof} = \sum_i \left[\log \widehat{\lambda}(x_i) - \log \widehat{\lambda}_{-i}(x_i)\right]$.

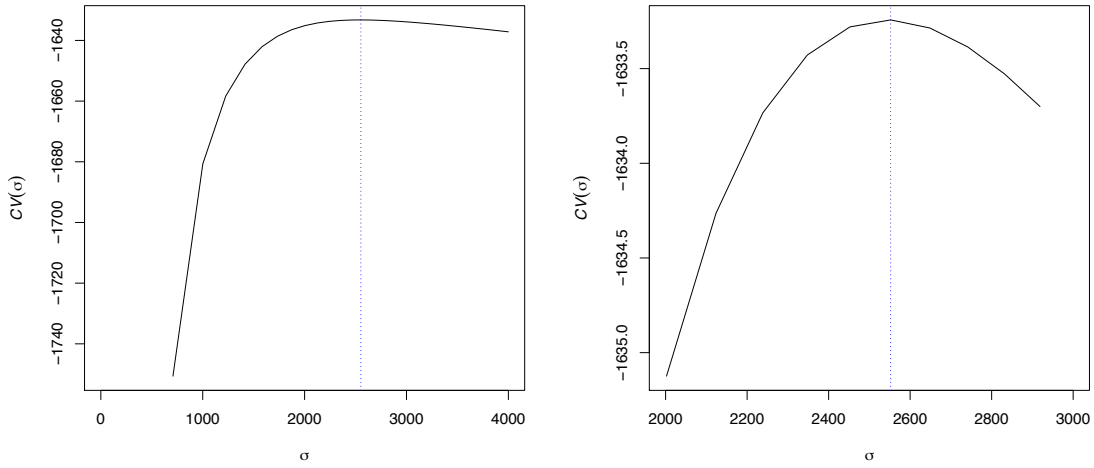

Figure 4.3: Automatic bandwidth selection for Geelong accident data. Leave-one-out cross-validation criterion $C(\sigma)$ against bandwidth σ. Right panel is close-up of left panel around optimal value.

Figure 4.3 shows the leave-one-out cross validation criterion for the Geelong data, giving an optimized bandwidth of $\sigma^*_- = 2550$ metres. Figure 4.4 shows the resulting kernel estimate of intensity. Computation time was 8 minutes to calculate the leave-one-out cross validation value for the range of bandwidths as shown in Figure 4.3 and 2.5 seconds to calculate the diffusion kernel values as shown in Figure 4.4 using the optimized bandwidth.

In *two-fold* cross-validation the data $\mathbf{x}$ are divided randomly into two subsets $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}$. Intensity estimates $\widehat{\lambda}^{(1)}(\cdot), \widehat{\lambda}^{(2)}(\cdot)$ are computed from each subset, for a sequence of values of σ.

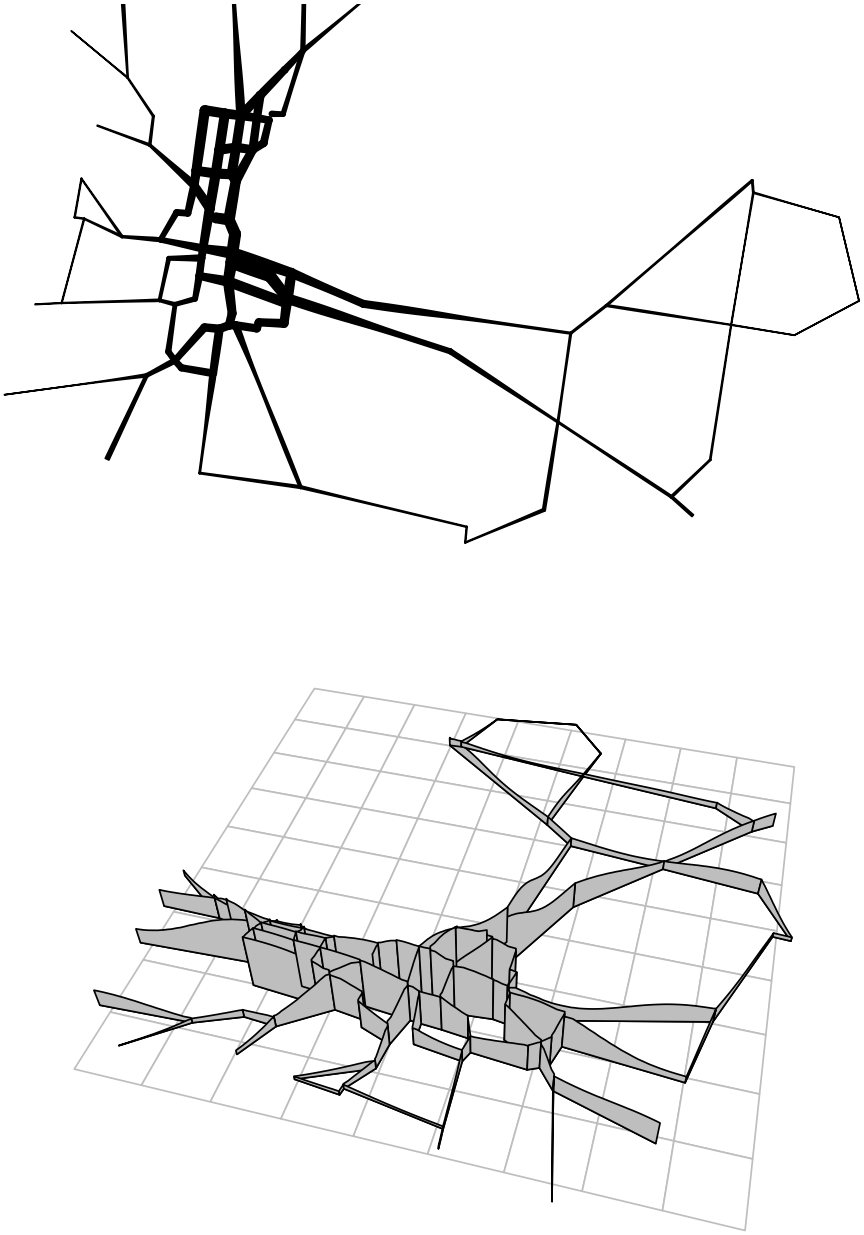

Figure 4.4: Counterpart of Figure 4.2 using the bandwidth $\sigma = 2250$ metres selected by leave-one-out cross-validation.

For each value of σ we calculate the cross-validation criterion

$$D(\sigma) = \sum_{x_i \in \mathbf{x}^{(1)}} \log \widehat{\lambda}^{(2)}(x_i) + \sum_{x_i \in \mathbf{x}^{(2)}} \log \widehat{\lambda}^{(1)}(x_i). \tag{4.24}$$

The bandwidth σ_X^* which maximizes $D(\sigma)$ is optimal for a dataset of size $N/2$. Using the asymptotic rule $\sigma \sim N^{-1/5}$ derived above, σ_X^* should be adjusted by a factor $2^{-1/5} = 0.87$ for smoothing a dataset of size N. In the Geelong example this produced results comparable to the leave-one-out method. Computation time for σ_X^* was 3.5 seconds. Two-fold cross-validation produces bandwidth estimates quickly, at the expense of sampling variability.

4.6.1 Rules of thumb

An alternative to cross-validation would be to adapt one of the popular rules of thumb for bandwidth selection (described in section 2.5) to the context of linear networks.

One way of adapting these rules is to ignore the network and treat the data as a two-dimensional point pattern. In this case the Scott rule (2.37) and Silverman rule (2.36) are equivalent. For the Geelong data, this rule of thumb gives bandwidths of 2.56 and 2.24 km in the east-west and north-south directions, respectively. These are quite close to the cross-validated choice of 2.55 km.

An alternative would be to apply Scott's (2.37) or Silverman's (2.36) rule for *one-dimensional* coordinate data to an orthogonal projection of the spatial points onto a one-dimensional axis chosen to maximise the sample standard deviation of the projected coordinates. The maximised standard deviation is $s = \sqrt{a}$, where a is the largest eigenvalue of the sample variance-covariance matrix of the spatial coordinates. Scott's rule would give

$$h = n^{-1/5}\sqrt{a}. \tag{4.25}$$

For the Geelong data this yields $h = 2.36$ km. Silverman's rule is inflated by the factor $(4/3)^{1/5} = 1.059$ and in the case of the Geelong data gives $h = 2.50$ km.

4.7 Edge effects and corrections

An observed network may be merely a subset of a larger, unobserved network. This may give rise to edge effects associated with the creation of artificial dead ends. For example, in the Geelong data, some terminal vertices were artificially created where the road map was truncated at the edge of the survey area, or where a road's administrative classification was downgraded from major to minor road at a certain location. These are different from true dead ends.

We emphasise that "edge corrections" are not necessary for the diffusion estimator presented here. In one- and two-dimensional density estimation from observations inside a restricted sampling region, the fixed-bandwidth kernel estimator (2.20) requires bias correction, because kernel mass is lost at the edges of the sampling region. Such edge effects do not occur in the diffusion estimator, which is mass-preserving. Indeed this is a major advantage of diffusion kernels [27].

Nevertheless, it could be useful to treat artificial endpoints differently from true dead ends, by allowing kernel mass to be lost at artificial endpoints, effectively treating them as absorbing rather than reflecting states of the diffusion. This yields upper and lower bounds for the "correct" density estimate. Given a network L which is a subset of a larger network L^+, write κ and κ^+ for the heat kernels on L and L^+. An artificial endpoint is an endpoint of L which is not an endpoint of L^+. Write κ^A for the kernel on L corresponding to the diffusion in which each artificial endpoint is an absorbing state: this can be obtained by deleting all terms in the sum (4.6) associated with paths which reach an artificial endpoint. Then for any $x, y \in L$, we have

$$\kappa_t^A(y \mid x) \leq \kappa_t^+(y \mid x) \leq \kappa_t(y \mid x). \tag{4.26}$$

4.8 Subsequent papers

This chapter was published in 2016 in McSwiggan *et al.* [100]. Subsequent papers have proposed other methods kernel smoothing on a linear network.

Rakshit *et al.* [117] (this author is also a co-author) developed a method for large data sets using a two-dimensional kernel. The method is very fast but ignores the connectivity of the network.

Two similar methods are proposed in Rakshit *et al.* [117]. They are motivated by two-dimensional kernel methods used for two-dimensional spatial point pattern kernel smoothing that each have a different edge correction method. The first is the "uniform" edge correction

$$\widehat{\lambda}^U(u) = \frac{1}{c_W(u)} \sum_{i=1}^{n} \kappa_t(u - x_i), \quad u \in W, \tag{4.27}$$

where κ is a two-dimensional kernel, and

$$c_W(u) = \int_W \kappa(u - v)\mathrm{d}v, \quad u \in W, \tag{4.28}$$

is the mass of the kernel centred at u that falls inside the window W. In [117] we adapt this to a linear network L, replacing W by L and replacing κ by a two-dimensional smoothing

kernel. Then the correction factor is the one-dimensional "section" area cut through the two-dimensional kernel by the linear network,

$$c_L(u) = \int_L \kappa(v-u)\mathrm{d}v, \quad u \in L. \tag{4.29}$$

The division by this correction factor reduces the two-dimensional kernel to a normalised one-dimensional kernel on the linear network, giving the Uniform correction intensity estimator

$$\widehat{\lambda}^U(u) = \frac{1}{c_L(u)} \sum_{i=1}^{n} \kappa(u-x_i), \quad u \in L. \tag{4.30}$$

The second method uses the correction $c_L(u)$ in a different way, motivated by Jones' [75] two-dimensional edge correction method. It is called the Jones-Diggle correction intensity estimator,

$$\widehat{\lambda}^{JD}(u) = \sum_{i=1}^{n} \frac{\kappa(u-x_i)}{c_L(x_i)}, \quad u \in L. \tag{4.31}$$

Computation for the two-dimensional kernel using the Fast Fourier Transform is quick compared to the diffusion kernel, particularly as bandwidth increases. Cross-validation bandwidth selection is also faster than for the diffusion kernel. The paper [117] derives the statistical properties of the estimators, and also explores adaptive bandwidth smoothing. The method works very well for large data sets where computation time may become prohibitive for the diffusion kernel method. However, since the connectivity of the network is ignored, the statistical properties are not as sound.

Moradi *et al.* [105] correct the naive kernel estimator described in Section 2.4.3 by using a geometric correction factor similar to that developed by Ang *et al.* [8] based on their *circumference* quantity for linear networks, which in turn was based on an adaptation of Diggle's [50] edge-corrected method. Ang *et al.* [8] use a correction factor, $m(x_i, d_L(x_i, x_j))$ within the formulation of their "geometrically corrected K–function". This quantity is the number of points that are the same distance from x_i as is x_j, where distance is the shortest path distance measured along the network. Moradi *et al* [105] use this correction to normalise the naive estimator that, now using intensity instead of density, would transform (2.21) to

$$\widehat{\lambda}^D(u) = \sum_{i=1}^{n} \frac{k(d_L(u,x_i))}{\int_0^\infty k(r)m(x_i,u)\mathrm{d}r}, \quad \mathrm{u} \in L, \tag{4.32}$$

where $r = d_L(u, x_i)$

Moradi *et al.* [104] develop a method using one-dimensional Voronoi tessellations on the linear network generated by the point pattern. Then the *Voronoi estimator* is an intensity estimate which is based on the inverse of the tessellation size. They then correct for the

anticipated over-smoothed and under-smoothed areas by using an automatic random thinning process on the point pattern prior to the generation of the Voronoi tessellations.

Unlike the diffusion method described in this thesis none of the above methods satisfies both the conservation of mass (2.14) and unbiasedness (2.18) properties.

Chapter 5

Relative risk on a network

5.1 Introduction

Chapter 4 covered kernel intensity estimation on a linear network and we proposed the use of our diffusion kernel as the method to best achieve this. A logical next step is to move onto the estimation of the relative risk, the ratio of rates of occurrence of different types of events occurring on a network of lines. For example, returning to the study of traffic accidents on a road network, Figure 5.1 shows the Geelong data split into two types of events; accidents that occur during the day and accidents that occur at night.

Relative risk estimation for spatial point patterns in two-dimensional space is well developed [79, 80, 81, 58, 59, 37, 54, 70, 47]. In this chapter we adapt and extend the two-dimensional techniques to point patterns on a linear network. Kernel estimates of relative risk can be obtained simply by taking the ratio of kernel estimates of the intensity functions (accident rates) of the two types of event. The main problem is to choose the smoothing bandwidth for kernel estimation, and to decide whether the numerator and denominator should be estimated using the same bandwidth (a "symmetric regimen", [47]) or whether different bandwidths may be permitted [79, 80]. Estimation on a linear network presents new challenges and exigencies: computation is much slower than in Euclidean space because the Fast Fourier Transform cannot be used; the leave-one-out kernel estimate is very costly.

In this chapter, several standard methods for bandwidth selection in two dimensions are adapted and extended to linear networks, and their performance is evaluated in simulation experiments. The methods include Scott's normal reference rule of thumb [120, p. 152], Kelsall and Diggle's density-ratio cross-validation [79, 80], and Kelsall and Diggle's binary likelihood and binary least squares cross-validation [81].

Kelsall and Diggle [79, 80] reported that their density-ratio cross-validation method suffered sporadic "breakdowns" in which the selected bandwidths and resulting risk estimates were very

unsatisfactory. Our adaptation of the method to linear networks exhibits breakdowns even more frequently. We have found a theoretical explanation for breakdown, in both contexts, and propose a modification of the Kelsall-Diggle method to improve its performance. Our simulation experiments demonstrate this improvement.

Throughout this chapter the smoothing kernel is chosen to be the classical heat kernel, the analogue for linear networks of the Gaussian kernel [100]. This is the Gaussian extension of the popular "equal-split continuous" rule ([109], [110, Chap. 9]). However, the methods of this chapter could be applied to any of the competing kernel estimators discussed in the literature.

Section 5.2 states the direct adaptations of the relative risk function and the Kelsall-Diggle cross-validation method [80] for two-dimensional data to a linear network. Section 5.3 analyses the weaknesses of these methods and proposes our modified version. Section 5.4 proposes a fast approximation to the leave-one-out estimate which is needed for practical applications. Section 5.5 reports the results of a simulation experiment to measure performance of the methods. Section 5.6 reports our analysis of two datasets: the Geelong road accidents, and the spatial pattern of protrusions on the dendritic tree of a neuron.

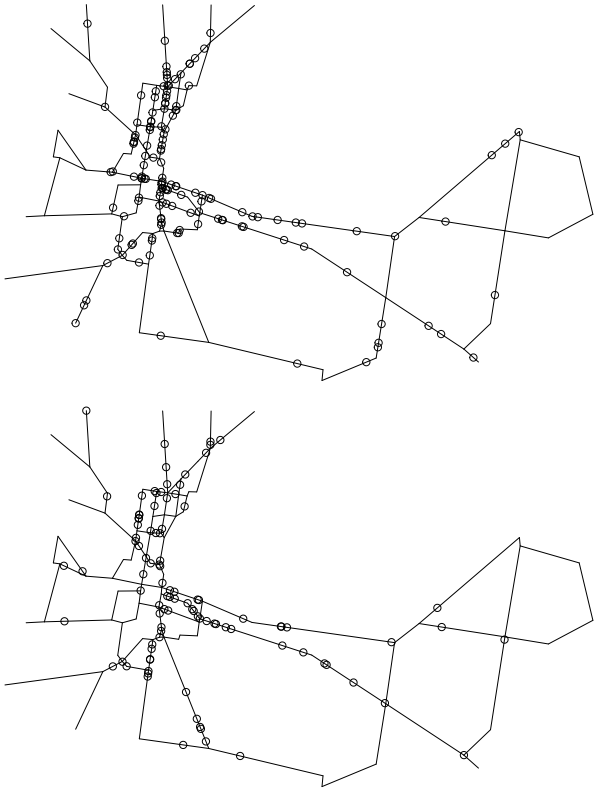

Figure 5.1: Geelong data split into daytime (*Top*) and night-time (*Bottom*) accidents.

5.2 Relative risk on L

The definition of relative risk function given in (2.40) can be adapted to linear networks. Suppose there are two point patterns $\mathbf{x} = \{x_1, \ldots, x_m\}$ and $\mathbf{y} = \{y_1, \ldots, y_n\}$ observed in the same spatial domain L. Treating $\mathbf{x}$ and $\mathbf{y}$ as realisations of point processes $\mathbf{X}$ and $\mathbf{Y}$ on L, respectively, the *logarithmic relative risk* function is defined by

$$\rho(u) = \log \frac{\lambda_{\mathbf{X}}(u)}{\lambda_{\mathbf{Y}}(u)}, \quad u \in L, \tag{5.1}$$

where $\lambda_{\mathbf{X}}(\cdot), \lambda_{\mathbf{Y}}(\cdot)$ are the intensities of $\mathbf{X}, \mathbf{Y}$, respectively.

Similarly, the *plug-in estimator* of relative risk given in (2.41) takes the form

$$\widehat{\rho}(u) = \widehat{\rho}_{h_1,h_2}(u) = \log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}, \quad u \in L, \tag{5.2}$$

where $\widehat{\lambda}_{\mathbf{X}}(u) = \widehat{\lambda}(u \mid \mathbf{x}, h_1)$ and $\widehat{\lambda}_{\mathbf{Y}}(u) = \widehat{\lambda}(u \mid \mathbf{y}, h_2)$ are kernel estimators of $\lambda_{\mathbf{X}}(u)$ and $\lambda_{\mathbf{Y}}(u)$, computed from $\mathbf{x}$ and $\mathbf{y}$, using bandwidths h_1 and h_2, respectively.

5.2.1 Kelsall-Diggle cross-validation on linear networks

As stated in Section 2.6.3 for two-dimensional point pattern data, Kelsall and Diggle [80] proposed a method for selecting bandwidths for the relative risk estimator (2.41) using cross-validation based on integrated squared error. Adapting Kelsall and Diggle's equation (2.45) to a linear network to choose the bandwidths (h_1, h_2) in (5.1), with or without the constraint $h_1 = h_2$, and re-expressed in terms of the intensity rather than the probability density, the criterion becomes

$$\begin{aligned} \widetilde{C}_{\mathrm{KD}}(h_1, h_2) = &- \int_L \left[\log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \right]^2 \mathrm{d}u - 2 \sum_{i=1}^{m} \frac{1}{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)} \log \frac{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)}{\widehat{\lambda}_{\mathbf{Y}}(x_i)} \\ &- 2 \sum_{j=1}^{n} \frac{1}{\widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)} \log \frac{\widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)}{\widehat{\lambda}_{\mathbf{X}}(y_j)}, \end{aligned} \tag{5.3}$$

where $\widehat{\lambda}_{\mathbf{X}}(u) = \widehat{\lambda}(u \mid \mathbf{x}, h_1)$ is the estimator (4.5) of $\lambda_{\mathbf{X}}(u)$ using bandwidth h_1, while $\widehat{\lambda}_{\mathbf{Y}}(u) = \widehat{\lambda}(u \mid \mathbf{y}, h_2)$ is the estimator of $\lambda_{\mathbf{Y}}(u)$ using bandwidth h_2, and $\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i), \widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)$ denote the corresponding leave-one-out estimators. The bandwidth pair (h_1, h_2) or (h, h) should be chosen to minimise the criterion (5.3).

5.3 Improved cross-validation method

Kelsall and Diggle [79, 80] reported that, in two dimensions, their cross-validation criterion (5.3) suffered occasional "breakdowns" in which the selected bandwidth values were extreme and the

resulting estimates very unsatisfactory. Similar breakdowns occurred in our experiments with the analogue of the Kelsall-Diggle criterion on a linear network (reported in Section 5.5, (with more in the Supplementary material of McSwiggan *et al.* [101]).

This motivates us to re-visit the original derivation of the Kelsall-Diggle method. Define the integrated squared error of estimation of ρ

$$\mathrm{ISE}(\widehat{\rho}) = \int_L \left(\widehat{\rho}(u) - \rho(u)\right)^2 \mathrm{d}u. \tag{5.4}$$

We now discuss approximation of the ISE from data, along the same lines as Kelsall and Diggle [80], but expressed in terms of point process intensity rather than probability density. We use a slightly more general formulation so that we may revisit it.

5.3.1 General derivation

If the plug-in estimator (5.2) is used, expanding the square in (5.4) gives

$$\begin{aligned}\mathrm{ISE}(\widehat{\rho}) = &\int_L \left[\log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\right]^2 \mathrm{d}u - 2\int_L \log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \log \frac{\lambda_{\mathbf{X}}(u)}{\lambda_{\mathbf{Y}}(u)} \mathrm{d}u \\ &+ \int_L \left[\log \frac{\lambda_{\mathbf{X}}(u)}{\lambda_{\mathbf{Y}}(u)}\right]^2 \mathrm{d}u.\end{aligned} \tag{5.5}$$

The last term on the right hand side of (5.5) is constant in any given application, and may be omitted for optimisation purposes. The middle term on the right hand side involves the unknown true intensities. Following the approach of Kelsall and Diggle [79, 80] we would replace the true intensities by approximations, based on a Taylor expansion of the logarithm:

$$\begin{aligned}\log \lambda_{\mathbf{X}}(u) &\approx \log \lambda^0_{\mathbf{X}}(u) + \frac{1}{\lambda^0_{\mathbf{X}}(u)}\left[\lambda_{\mathbf{X}}(u) - \lambda^0_{\mathbf{X}}(u)\right] \\ &= \log \lambda^0_{\mathbf{X}}(u) + \frac{\lambda_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{X}}(u)} - 1 \qquad (5.6) \\ \log \lambda_{\mathbf{Y}}(u) &\approx \log \lambda^0_{\mathbf{Y}}(u) + \frac{\lambda_{\mathbf{Y}}(u)}{\lambda^0_{\mathbf{Y}}(u)} - 1, \qquad (5.7)\end{aligned}$$

where $\lambda^0_{\mathbf{X}}(u), \lambda^0_{\mathbf{Y}}(u)$ are some chosen "reference estimates" to be discussed below. Note that the Taylor expansions are performed about the reference estimates. The approximations (5.6)–(5.7) give

$$\log \frac{\lambda_{\mathbf{X}}(u)}{\lambda_{\mathbf{Y}}(u)} \approx \log \frac{\lambda^0_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{Y}}(u)} + \frac{\lambda_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{X}}(u)} - \frac{\lambda_{\mathbf{Y}}(u)}{\lambda^0_{\mathbf{Y}}(u)} \tag{5.8}$$

and hence

$$\begin{aligned}\log\frac{\lambda_{\mathbf{X}}(u)}{\lambda_{\mathbf{Y}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} &\approx \log\frac{\lambda^0_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{Y}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\\ &\quad+\frac{\lambda_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{X}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\\ &\quad+\frac{\lambda_{\mathbf{Y}}(u)}{\lambda^0_{\mathbf{Y}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}.\end{aligned} \tag{5.9}$$

Collecting terms we obtain the cross-validation criterion

$$\begin{aligned}C(h_1,h_2) &= \int_L\left[\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\right]^2\mathrm{d}u - 2\int_L\log\frac{\lambda^0_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{Y}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\mathrm{d}u\\ &\quad-2\int_L\frac{\lambda_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{X}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\mathrm{d}u + 2\int_L\frac{\lambda_{\mathbf{Y}}(u)}{\lambda^0_{\mathbf{Y}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\mathrm{d}u.\end{aligned} \tag{5.10}$$

The last two terms on the right hand side of (5.10) still contain the unknown true intensity functions $\lambda_{\mathbf{X}}, \lambda_{\mathbf{Y}}$. Following [79, 80] these terms can be estimated from data by "leave-one-out averaging" [67], or equivalently the Campbell-Mecke formula [102]:

$$\int_L\frac{\lambda_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{X}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\mathrm{d}u \approx \mathbb{E}\left[\sum_{i=1}^m\frac{1}{\widehat{\lambda}^0_{\mathbf{X}}(x_i)}\log\frac{\widehat{\lambda}^{-i}_{\mathbf{X}}(x_i)}{\widehat{\lambda}_{\mathbf{Y}}(x_i)}\right], \tag{5.11}$$

where

$$\begin{aligned}\widehat{\lambda}^{-i}_{\mathbf{X}}(x_i) &= \widehat{\lambda}(x_i \mid \mathbf{x}\setminus\{x_i\}, h) = \sum_{j\neq i}\kappa_{h_1}(x_j \mid x_i)\\ &= \widehat{\lambda}_{\mathbf{X}}(x_i) - \kappa_{h_1}(x_i \mid x_i)\end{aligned} \tag{5.12}$$

is the leave-one-out estimate of intensity of $\mathbf{X}$ based on all points of $\mathbf{x}$ except the query point x_i. The approximation (5.11) would be exact, by the Campbell-Mecke formula, if the leave-one-out estimator was non-random, so heuristically we expect the right-hand side of (5.11) to be a consistent estimator of the left-hand side. Similarly we approximate

$$\int_L\frac{\lambda_{\mathbf{Y}}(u)}{\lambda^0_{\mathbf{Y}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\mathrm{d}u \approx \mathbb{E}\left[\sum_{j=1}^n\frac{1}{\lambda^0_{\mathbf{Y}}(y_j)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(y_j)}{\widehat{\lambda}^{-j}_{\mathbf{Y}}(y_j)}\right], \tag{5.13}$$

yielding the empirical cross-validation criterion

$$\begin{aligned}\widetilde{C}(h_1,h_2) &= \int_L\left[\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\right]^2\mathrm{d}u - 2\int_L\log\frac{\lambda^0_{\mathbf{X}}(u)}{\lambda^0_{\mathbf{Y}}(u)}\log\frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\mathrm{d}u\\ &\quad-2\sum_{i=1}^m\frac{1}{\lambda^0_{\mathbf{X}}(x_i)}\log\frac{\widehat{\lambda}^{-i}_{\mathbf{X}}(x_i)}{\widehat{\lambda}_{\mathbf{Y}}(x_i)} - 2\sum_{j=1}^n\frac{1}{\lambda^0_{\mathbf{Y}}(y_j)}\log\frac{\widehat{\lambda}^{-j}_{\mathbf{Y}}(y_j)}{\widehat{\lambda}_{\mathbf{X}}(y_j)}.\end{aligned} \tag{5.14}$$

5.3.2 Derivation of Kelsall–Diggle criterion

Kelsall and Diggle [80] choose the reference estimators $\lambda^0_{\mathbf{X}}(u)$ and $\lambda^0_{\mathbf{Y}}(u)$ in (5.10) and (5.14) to be the *current* estimators $\widehat{\lambda}_{\mathbf{X}}(u) = \widehat{\lambda}(u \mid \mathbf{x}, h_1)$ and $\widehat{\lambda}_{\mathbf{Y}}(u) = \widehat{\lambda}(u \mid \mathbf{y}, h_2)$, respectively. This allows some algebraic simplification of (5.10) yielding

$$\begin{aligned} C_{\text{KD}}(h_1, h_2) = &- \int_L \left[\log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \right]^2 \mathrm{d}u - 2 \int_L \frac{\lambda_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{X}}(u)} \log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \mathrm{d}u \\ &+ 2 \int_L \frac{\lambda_{\mathbf{Y}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \mathrm{d}u. \end{aligned} \tag{5.15}$$

Similarly taking the reference intensities at the data points to be the current empirical estimators, $\lambda^0_{\mathbf{X}}(x_i) = \widehat{\lambda}^{-i}_{\mathbf{X}}(x_i)$ and $\lambda^0_{\mathbf{Y}}(y_j) = \widehat{\lambda}^{-j}_{\mathbf{Y}}(y_j)$, the empirical cross-validation criterion (5.14) becomes

$$\begin{aligned} \widetilde{C}_{\text{KD}}(h_1, h_2) = &- \int_L \left[\log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \right]^2 \mathrm{d}u - 2 \sum_{i=1}^{m} \frac{1}{\widehat{\lambda}^{-i}_{\mathbf{X}}(x_i)} \log \frac{\widehat{\lambda}^{-i}_{\mathbf{X}}(x_i)}{\widehat{\lambda}_{\mathbf{Y}}(x_i)} \\ &- 2 \sum_{j=1}^{n} \frac{1}{\widehat{\lambda}^{-j}_{\mathbf{Y}}(y_j)} \log \frac{\widehat{\lambda}^{-j}_{\mathbf{Y}}(y_j)}{\widehat{\lambda}_{\mathbf{X}}(y_j)}. \end{aligned} \tag{5.16}$$

A possible explanation for the breakdown of $\widetilde{C}_{\text{KD}}$ is now clear. The general form of the cross-validation criterion (5.10) is derived by replacing the true intensities $\lambda_{\mathbf{X}}(u)$ and $\lambda_{\mathbf{Y}}(u)$ by Taylor approximations (5.6) and (5.7) about the "reference" estimators $\lambda^0_{\mathbf{X}}(u)$ and $\lambda^0_{\mathbf{Y}}(u)$, respectively. In the case of the Kelsall-Diggle cross-validation criterion (5.16) the reference estimates are taken to be the current kernel estimators $\widehat{\lambda}_{\mathbf{X},h_1}(u)$ and $\widehat{\lambda}_{\mathbf{Y},h_2}(u)$. For small bandwidths, these estimators could be highly biased because of undersmoothing, and Taylor expansions about these estimators could yield poor approximations to the true intensities.

5.3.3 Our proposed alternative

We propose taking the "reference" estimators $\lambda^0_{\mathbf{X}}(u), \lambda^0_{\mathbf{Y}}(u)$ in (5.10) and (5.14) to be *over-smoothed* kernel estimators obtained by setting h_1, h_2 to the maximum values under consideration, say H_1, H_2. The Kelsall-Diggle argument then leads to our proposed "modified Kelsall-

Diggle" cross-validation criterion,

$$\begin{aligned}\widetilde{C}_{\text{OVER}}(h_1, h_2) = &\int_L \left[\log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\right]^2 \mathrm{d}u \\ &- 2\int_L \log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \log \frac{\widehat{\lambda}(u \mid \mathbf{x}, H_1)}{\widehat{\lambda}(u \mid \mathbf{y}, H_2)} \mathrm{d}u \\ &- 2\sum_{i=1}^{m} \frac{1}{\widehat{\lambda}^{-i}(x_i \mid \mathbf{x}, H_1)} \log \frac{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)}{\widehat{\lambda}_{\mathbf{Y}}(x_i)} \\ &- 2\sum_{j=1}^{n} \frac{1}{\widehat{\lambda}^{-j}(y_j \mid \mathbf{y}, H_2)} \log \frac{\widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)}{\widehat{\lambda}_{\mathbf{X}}(y_j)}.\end{aligned} \tag{5.17}$$

This has marginally greater computational cost than the Kelsall-Diggle criterion (5.16) due to the addition of the second term on the right hand side of (5.17).

Our proposal, to use an over-smoothed estimator as the reference for the Taylor expansion, could be compared to the use of "pre-smoothed" estimators by Hall *et al.* [68].

An even simpler alternative could be to take the reference intensities to be constant, $\lambda^0_{\mathbf{X}}(u) = m/|L|$ and $\lambda^0_{\mathbf{Y}}(u) = n/|L|$, where $m = n(\mathbf{x})$ and $n = n(\mathbf{y})$ are the observed numbers of points in the patterns $\mathbf{x}$ and $\mathbf{y}$. This would yield the cross-validation criterion

$$\begin{aligned}\widetilde{C}_{\text{UNIF}}(h_1, h_2) = &\int_L \left[\log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)}\right]^2 \mathrm{d}u \\ &- 2(\log \frac{m}{n}) \int_L \log \frac{\widehat{\lambda}_{\mathbf{X}}(u)}{\widehat{\lambda}_{\mathbf{Y}}(u)} \mathrm{d}u \\ &- 2\frac{|L|}{m} \sum_{i=1}^{m} \log \frac{\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)}{\widehat{\lambda}_{\mathbf{Y}}(x_i)} \\ &- 2\frac{|L|}{n} \sum_{j=1}^{n} \log \frac{\widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)}{\widehat{\lambda}_{\mathbf{X}}(y_j)}.\end{aligned} \tag{5.18}$$

This criterion is computationally cheaper than the other cross-validation criteria, but may lead to suboptimal choices.

Other strategies include numerically stabilising the cross-validation by adding a small constant value to the reference intensities [70, 28]. Instead of constraining $h_1 = h_2$, it would be possible to use the constraint $h_1/h_2 = (n_1/n_2)^{-1/5}$, or to allow $h_1 \neq h_2$ and introduce a penalty for discrepancy between them e.g. $(h_1 - h_2)^2$, or simply to constrain the bandwidths to be greater than a certain realistic minimum value. The latter option is discussed in Section 5.6.1.

5.4 Approximation to leave-one-out estimator

As noted in Section 4.6, computation of the leave-one-out estimators of intensity $\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i), \widehat{\lambda}_{\mathbf{Y}}^{-j}(y_j)$ is more complicated on a linear network than in two-dimensional space. Exact calculation of $\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)$ (say) would require us to run the heat equation solver for the point pattern $\mathbf{x}^{-i} = \mathbf{x} \setminus \{x_i\}$. The solver would have to be executed n times to obtain all the values $\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i)$, for $i = 1, \ldots, n$.

An alternative is to use the relation $\widehat{\lambda}_{\mathbf{X}}^{-i}(x_i) = \widehat{\lambda}(x_i) - \kappa(x_i \mid x_i)$ from (2.39), and to find an approximation for $\kappa(x_i \mid x_i)$. Invoking McSwiggan *et al.* [100, equ. (23)] or Krostrykin *et al.* [83, Corollary 3.4] we can write $\kappa(u \mid u)$ as an infinite sum,

$$\kappa_t(u \mid u) = \sum_{\Pi} a(\Pi) \phi_{\sqrt{t}}(\ell(\Pi)), \tag{5.19}$$

over all possible cycles $\Pi = (v_0, \ldots, v_{m+1})$ in the network, with $m \geq 0$, where $v_0 = u$, $v_{m+1} = u$ and $v_1, \ldots, v_m$ are vertices. Here $\ell(\Pi)$ is the total length of the path Π, and $a(\Pi)$ is a combinatorial coefficient, while ϕ_σ is the Gaussian density with mean 0 and standard deviation σ.

A simple approximation is obtained by truncating the sum in (5.19), retaining only the terms with $m = 0$ or $m = 1$ steps. This could be portrayed as the analogue of a first order Taylor approximation. If u lies on a segment of length $s = s(u)$ and is a distance $x = x(u)$ from the left endpoint, and if the left and right endpoints have degree d and d', respectively, the proposed approximation to $\kappa(u \mid u)$ is

$$\begin{aligned} \kappa_t(u \mid u) \;\approx\; \kappa^*_\sigma(u) &= \phi_\sigma(0) + \left(\frac{2}{d} - 1\right) \phi_\sigma(2x(u)) \\ &\quad + \left(\frac{2}{d'} - 1\right) \phi_\sigma(2(s(u) - x(u))), \end{aligned} \tag{5.20}$$

where $\sigma = \sqrt{t}$. This approximation has the advantage that it can be computed rapidly from the spatial coordinates and network geometry.

The approximation (5.20) is likely to be very accurate when $\sigma \ll s(u)$, and is likely to become progressively less accurate as σ increases. To improve the performance for large t, we constrain the approximation (5.20) to be greater than or equal to $1/|L|$, which is the limiting value of $\kappa_t(u \mid u)$ as $t \to \infty$.

Results in the Supplementary material of McSwiggan *et al.* [101] demonstrate that the approximation (5.20) is highly satisfactory for this purpose.

5.5 Simulation experiments

Kelsall and Diggle [80] compared the performance of their proposed bandwidth selection method with that of other methods, using a suite of simulations on the one-dimensional unit interval. We have run analogous experiments on a linear network, using all of the bandwidth selection criteria mentioned above.

5.5.1 Description of experiments

Kelsall and Diggle [80] considered nine different scenarios by combining three possible choices for the relative risk function $r(u) = \lambda_{\mathbf{X}}(u)/\lambda_{\mathbf{Y}}(u)$ with three possible choices for the denominator intensity $d(u) = \lambda_{\mathbf{Y}}(u)$. The numerator intensity is then $\lambda_{\mathbf{X}}(u) = r(u)d(u)$. The three possible risk functions $r(u)$ were a constant function and two Gaussian densities. The three possible denominator intensities $d(u)$ were a constant function and two linear transformations of the sine function.

Figure 5.2 shows the linear network used in our experiments. It has a total length of 4.1 units and a diameter of 1.25 units (sc. the maximum path distance between any two points) and is inscribed in the unit square.

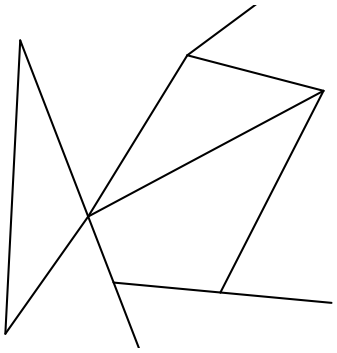

Figure 5.2: Linear network used in the experiments.

Three risk functions $r(u)$ were used in our experiments. Risk function 1 is constant; risk functions 2 and 3 have a single peak, obtained by evaluating the heat kernel (bandwidths 0.4 and 0.12, respectively) for a single data point placed at the centre of the network. Similarly we used three functions for the denominator intensity $d(u)$, namely $d_1(x, y) \equiv 1$, $d_2(x, y) = 1 + (1/2)\sin(2\pi x)$ and $d_3(x, y) = 1 + (3/4)\sin(4\pi x)$, where (x, y) are the Cartesian coordinates. These six functions are plotted in the Supplementary material of McSwiggan *et al.* [101]).

5.5.2 Representative results

Here we present detailed results for one case, with risk function 2 and denominator function 3, shown in Figure 5.3. Simulated realisations were generated with fixed numbers of points, $n(\mathbf{x}) = 50$ and $n(\mathbf{y}) = 200$.

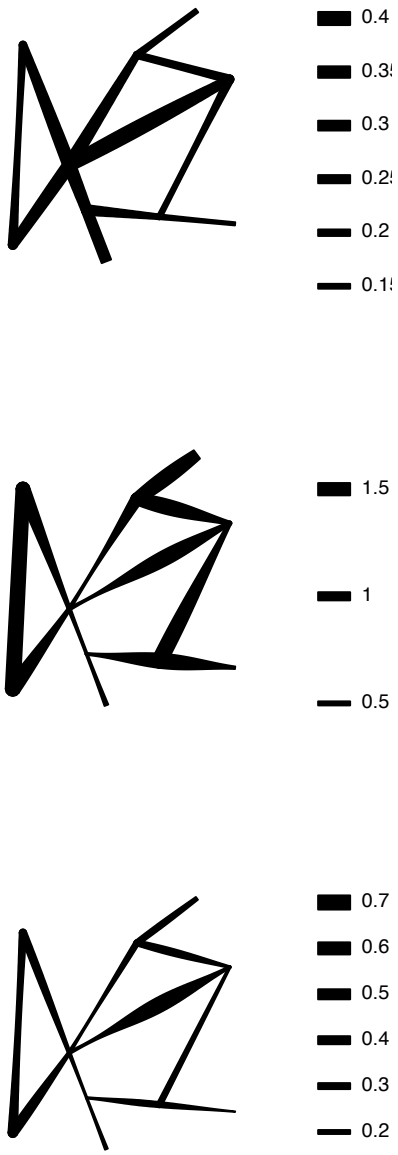

Figure 5.3: Simulation experiment reported in this section. *Top:* relative risk $r(u)$. *Middle:* denominator intensity $\lambda_{\mathbf{Y}}(u) = d(u)$. *Bottom:* numerator intensity $\lambda_{\mathbf{X}}(u) = r(u)d(u)$. Line width plots in the style of [133], with line width proportional to function value.

Figure 5.4 shows boxplots of the ISE values attained by each of the bandwidth selection

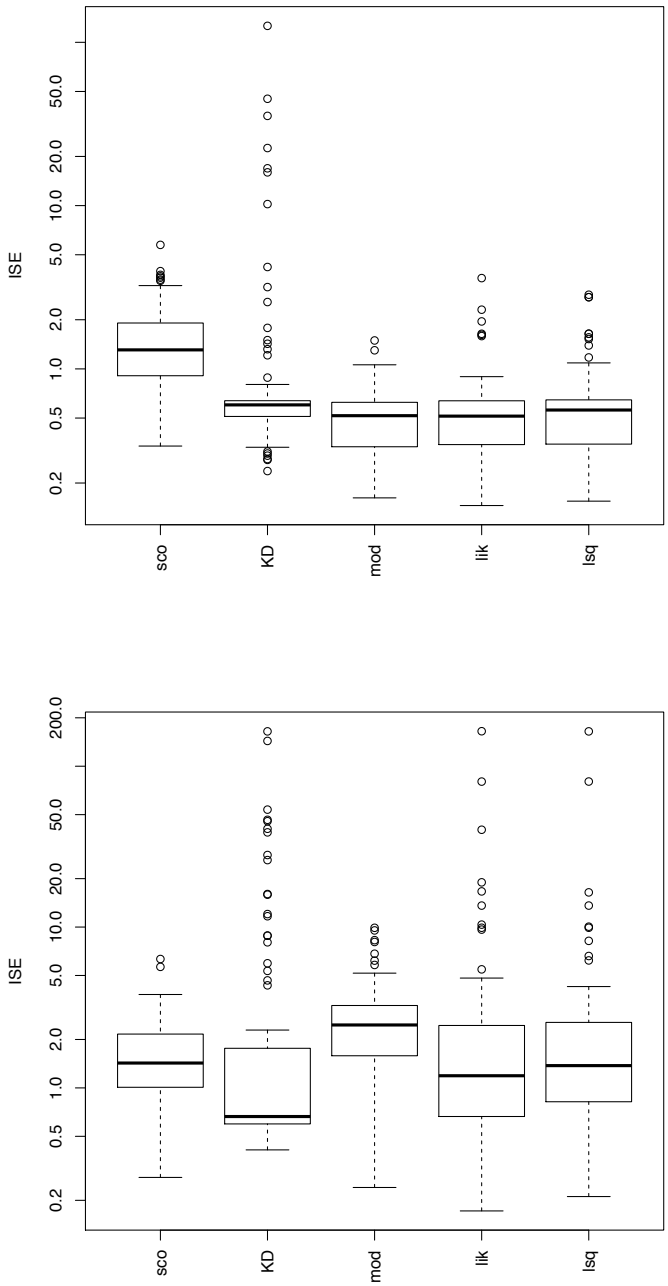

Figure 5.4: Boxplots of ISE values achieved by different methods for bandwidth selection in a simulation experiment (case $i = 2, j = 3$). *Top*: bandwidths h_1, h_2 are constrained to be equal (method M3). *Bottom:* bandwidths unconstrained (method M2). Note logarithmic scale for ISE.

methods. Here `sco` indicates Scott's rule of thumb as adapted in (4.25); `KD` is Kelsall–Diggle cross-validation (5.3); `mod` is our modification (5.17); `lik` is likelihood cross-validation (2.47); and `lsq` is least squares cross-validation (2.48). Each bandwidth selection method was applied to the same set of 100 simulated realisations. For each simulated dataset the bandwidth, or pair of bandwidths, selected by each method was used to smooth the data, yielding an estimate of ρ, and the ISE for this estimate was computed from (5.4) using the true value of $\rho(u)$ which is known exactly in the simulation experiment. The upper panel of Figure 5.4 shows boxplots of the ISE values obtained when the bandwidths are constrained to be equal according to method M3, and the lower panel when they are not constrained (method M2).

	Minimal			Maximal		
METHOD	h_1	h_2	h	h_1	h_2	h
`sco`	0	0.00	0	0.00	0.00	0.00
`KD`	0.01	0.10	0	0.25	0.83	0.41
`mod`	0	0.80	0	0.02	0.08	0.24
`lik`	0	0.27	0	0.02	0.48	0.23
`lsq`	0	0.13	0	0.02	0.58	0.23

Table 5.1: Fraction of outcomes of each bandwidth-selection method in which the selected bandwidth is equal to the minimum permitted bandwidth (*Minimal*) or the maximum permitted bandwidth (*Maximal*). Here h_1, h_2 are the bandwidths selected jointly without any constraint (method M2), and h is the symmetric bandwidth (method M3). Simulation experiment case $i = 2, j = 3$.

Table 5.1 reports the fraction of outcomes in which the bandwidth selected by each method is equal to the minimum or maximum bandwidth value considered. For the Kelsall-Diggle method in the symmetric case, 41% of the selected bandwidths equal the maximum permitted bandwidth. While large bandwidths may be quite satisfactory in some cases, selecting the minimum available bandwidth will almost always produce a poor estimate of relative risk, and this happens frequently in the asymmetric case ($h_1 \neq h_2$).

Figure 5.5 shows a scatterplot matrix for the values of bandwidth h obtained by each of the methods in the constrained case $h_1 = h_2 = h$, method M3. Interestingly, our modified version of the Kelsall-Diggle method yields bandwidths which are highly correlated with the likelihood cross-validation and least squares cross-validation methods, and are only weakly correlated with the original Kelsall-Diggle method. Scott's rule of thumb has low correlation with all other methods, suggesting that it would be unwise to use the rule-of-thumb bandwidth estimate as an initial guess at the cross-validated bandwidth estimate.

Analogous figures for the bandwidths h_1 and h_2 respectively, in the case where the band-

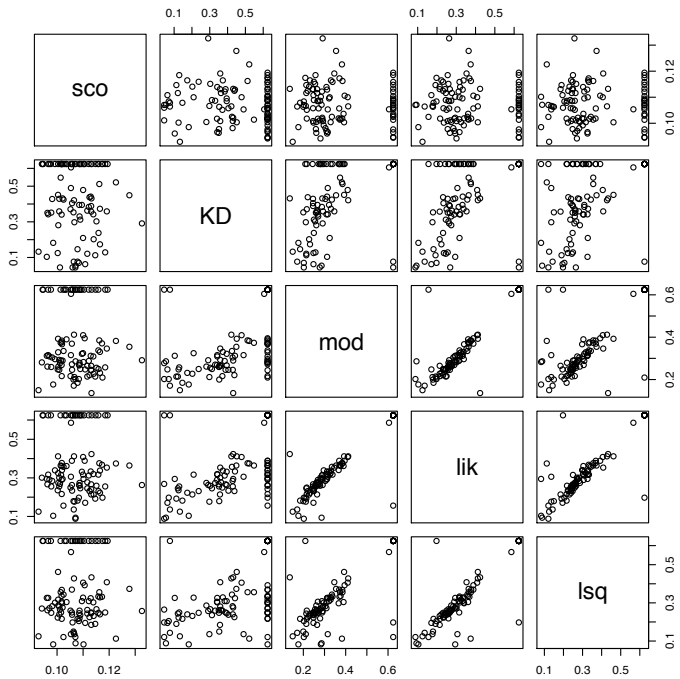

Figure 5.5: Scatterplot matrix for the bandwidth values selected by each method under the constraint $h_1 = h_2 = h$, method M3. Simulation experiment case $i = 2, j = 3$.

widths are permitted to be different, are given in the supplementary material of McSwiggan *et al.*. They show that the bandwidth h_2, which serves to smooth the denominator, is frequently chosen to be an extremely small or extremely large value. This appears to be the main cause of breakdown in bandwidth selection when the bandwidths are not constrained to be equal.

5.5.3 Summary of performance

The Supplementary material of McSwiggan *et al.* [101] gives detailed results from the suite of nine simulation experiments. Following is a summary of the main findings.

Bandwidth values selected using the fast approximation (5.20) to the leave-one-out estimate agreed very closely with those selected using the exact leave-one-out estimate, giving us confidence that the approximation is reliable.

The most significant finding is that estimates of relative risk were often much less accurate if we allowed $h_1 \neq h_2$ than when we constrained $h_1 = h_2$. This applied to all of the cross-validation methods. This may appear paradoxical unless we remember that the cross-validation criterion is only a data-based estimate of true performance, so that unconstrained minimisation of the cross-validation criterion is not guaranteed to produce better true performance than constrained minimisation. In our experiments, method M3 consistently outperformed method M2.

Investigation showed that when $h_1 \neq h_2$, the selected value of h_1 was usually appropriate, but that the selected value of h_2 was frequently much too small. Plots of the cross-validation criteria in individual examples often showed a steep decline in $C(h_1, h_2)$ as $h_2 \to 0$. Since the expressions for the cross-validation criteria are symmetric in $\mathbf{x}$ and $\mathbf{y}$, this one-sided behaviour is probably attributable to the different numbers of points in the two patterns.

Each method exhibits occasional "breakdown" in which the estimate is quite poor. Cross-validation methods have better performance than Scott's rule-of-thumb overall. However, the Scott rule of thumb is computationally cheaper, and is less susceptible to breakdown – it is "consistently mediocre".

The Kelsall-Diggle method often has higher ISE and higher frequency of breakdown than other cross-validation methods. In some cases the K-D method was unusable, with an infinite median ISE. The K-D method and our modified method often gave quite different results, lending support to the argument about the Taylor expansion.

Somewhat surprisingly, our modified method, the likelihood cross-validation and the least square cross validation method often selected quite similar bandwidths and gave similar results. Our modified method typically has the lowest frequency of breakdowns and the lowest median ISE, although its performance is mediocre in some cases.

5.5.4 General comments on experiments

Statistical performance will depend on the maximum bandwidth specified when running the bandwidth selection algorithm, because several of the methods have a high probability of selecting the maximum bandwidth.

In their experiments, Kelsall and Diggle [79, 80] measured the performance of estimators by the median ISE. Our figures suggest that summaries such as the median and mean of ISE could be hard to interpret because of the very different shapes of the distributions of ISE values obtained from each method.

The theoretical analysis presented by Kelsall and Diggle [79, 80] assumed that $\lambda_{\mathbf{X}}, \lambda_{\mathbf{Y}}$ are bounded away from zero. Their (and our) experiments include scenarios where the minimum density is very small, so this could explain the poor performance.

5.6 Examples

Two real data examples are studied here. Evidence for spatially-varying relative risk is weak in the first example, and very strong in the second example.

5.6.1 Geelong road accidents

An important question for road safety management is whether some specific locations have high accident risk at night, after allowing for the inherently greater baseline risk of nighttime driving. For the Geelong data, classified into day and night accidents in Figure 5.1, we considered estimation of the relative rate of night versus day accidents. The modified Scott rule of thumb gave bandwidths of about 2.7 km for each pattern. We computed the Kelsall-Diggle (5.16), modified Kelsall-Diggle (5.17), likelihood (2.47) and least squares (2.48) cross-validation criteria for relative risk. We nominated a maximum bandwidth of $h_{\max} = 5$ km, and searched over a grid of $N = 400$ candidate values of bandwidth $h = (k/N)^{1/2} h_{\max}$ for $k = 1, \ldots, N$. The maximum bandwidth was also used to compute the reference intensities for our modified criterion. The selected bandwidths are shown in Table 5.2. Total time to compute all four criteria was about 3 minutes if leave-one-out estimates were calculated exactly, and about 5 seconds if the fast approximation (5.20) was used.

Optimal bandwidths, selected using the Kelsall-Diggle criterion (5.16), and our modification (5.17) are acceptable values. The likelihood and least squares criteria produce acceptable bandwidths in the symmetric case, but in the asymmetric case the bandwidth h_2 for the denominator is too small: the corresponding estimates of relative risk have values as high as 10^{15}. Figure 5.6 shows the estimate of the ratio of night to day intensities using symmetric bandwidth 5 km. For reference, the overall ratio of night to day accidents is $98/144 = 0.68$. The figure

	Symmetric	Asymmetric	
Method	h	h_1	h_2
`Scott`	2.76	2.76	2.68
`KD`	5.00 (∞)	5.00	5.00
`mod`	5.00 (∞)	5.00	3.42
`lik`	5.00 (∞)	2.92	0.25
`lsq`	5.00 (∞)	2.60	0.25

Table 5.2: Automatically-selected bandwidths for the Geelong accidents separated into night and day accidents. Symmetric bandwidths selected by method M3; asymmetric bandwidth pairs by method M2; exact calculation. The symbol ∞ indicates that infinite bandwidth achieved a better cross-validation score than the selected bandwidth, $C(\infty, \infty) < C(h, h)$.

suggests that the relative risk of night time to day time accidents is up to 4 times higher on some of the more remote roads. This is plausible for reasons including the higher speed limits and the absence of street lighting along the remote roads. However, the calculation does not include adjustment for diurnal differences in traffic volume.

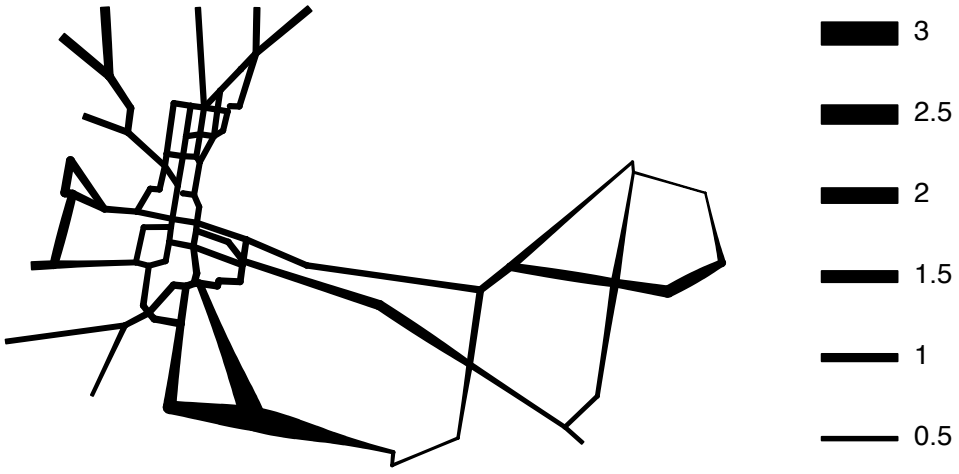

Figure 5.6: Relative risk of night versus day accidents using bandwidth 5 km. Line width proportional to relative risk.

Figure 5.7 shows contours of the cross-validation criteria (5.17) and (2.47) as functions of (h_1, h_2), for the Geelong data split into day and night accidents. Our modified criterion (5.17) has convex contours and a clearly defined minimum in this case. The likelihood criterion (2.47) is quite regular for large bandwidth values, but has a steep slope when h_2 is small, which explains the incorrect choice of h_2 in the unconstrained case. Contour plots for the other cross-validation criteria are given in an online supplement.

Figure 5.7 suggests that a practical remedy for the selection of incorrect bandwidths might be simply to restrict attention to bandwidths larger than a data-dependent threshold. Consideration of (2.46) suggests using the mean nearest-neighbour distance between each type of accident. For the Geelong data, the mean distance from a daytime accident location to the nearest nighttime accident location is 0.98 km.

The Geelong data also include information on the number of vehicles involved in the accident. Single-vehicle accidents include accidents occurring when a driver loses control of the vehicle, and accidents involving a pedestrian. There were 100 single-vehicle accidents, 115 two-vehicle accidents, 21 three-vehicle and 6 four-vehicle accidents. Figure 5.8 shows the estimated ratio of accident rates of single- and multiple-vehicle accidents, again using the bandwidth 5 km selected by the symmetric method M3. For reference, the ratio of numbers of single-vehicle to multiple-vehicle accidents is $100/142 = 0.70$.

5.6.2 Dendritic spines data

Figure 5.9 shows the dendritic spines data studied in Jammalamadaka *et al.* and Baddeley *et al.* [74, 12]. The network represents one branch of the dendritic tree of a neuron. The points are the locations of small protrusions called spines, which are classified into three types: mushroom, stubby and thin. Key research questions concern the spatial distribution of spines on the network, and differences in spatial distribution between different types of spines [74]. Analysis in Baddeley *et al.* [12] suggested that the mushroom and stubby types are uniformly distributed while the thin types are found more frequently near the ends of the dendritic tree, at the left side of Figure 5.9.

Bandwidth selection was performed using the fast approximation (5.20) to the leave-one-out estimates. Mean and median nearest neighbour distances were less than 7 microns and the Scott rule gave bandwidths of 12 to 18 microns. Bandwidths from 15 to 300 microns were considered, incurring a total computation time of 81 seconds (whereas the exact computation would have taken 147 minutes). Contour plots for the cross-validation criteria are given in an online supplement. The bandwidths selected by each method are shown in Table 5.3.

Figure 5.10 shows the contours of the likelihood cross-validation criterion for the dendrite data, indicating strong support for a value of h_1 around 80 microns, but supporting a range of h_2 values. Other contour plots are given in the online supplements. Figure 5.11 shows the estimated relative risk using the bandwidths selected by our modified method. The ratio of counts of thin spines to other spines is $115/451 = 0.26$.

This dataset contains strong evidence for spatially-varying relative risk and, perhaps as a consequence, there is broad agreement between the different cross-validation methods for bandwidth selection.

	Symmetric	Asymmetric	
Method	h	h_1	h_2
`Scott`	17.6	17.6	12.5
`KD`	82.2	84.9	300
`mod`	83.5	93.7	15
`lik`	79.4	84.9	300
`lsq`	77.9	68.7	300

Table 5.3: Automatically-selected bandwidths (fast method) for the dendritic spines, relative risk of 'thin' type against 'other' types.

This chapter has adapted the existing techniques for analysing relative risk in two-dimensional spatial point patterns to spatial point patterns on a linear network, focussing on the problem of bandwidth selection, and successfully applied those methods to the Geelong data and the dendtritic spines data. Futher discussion of these techniques are given in Chapter 6.

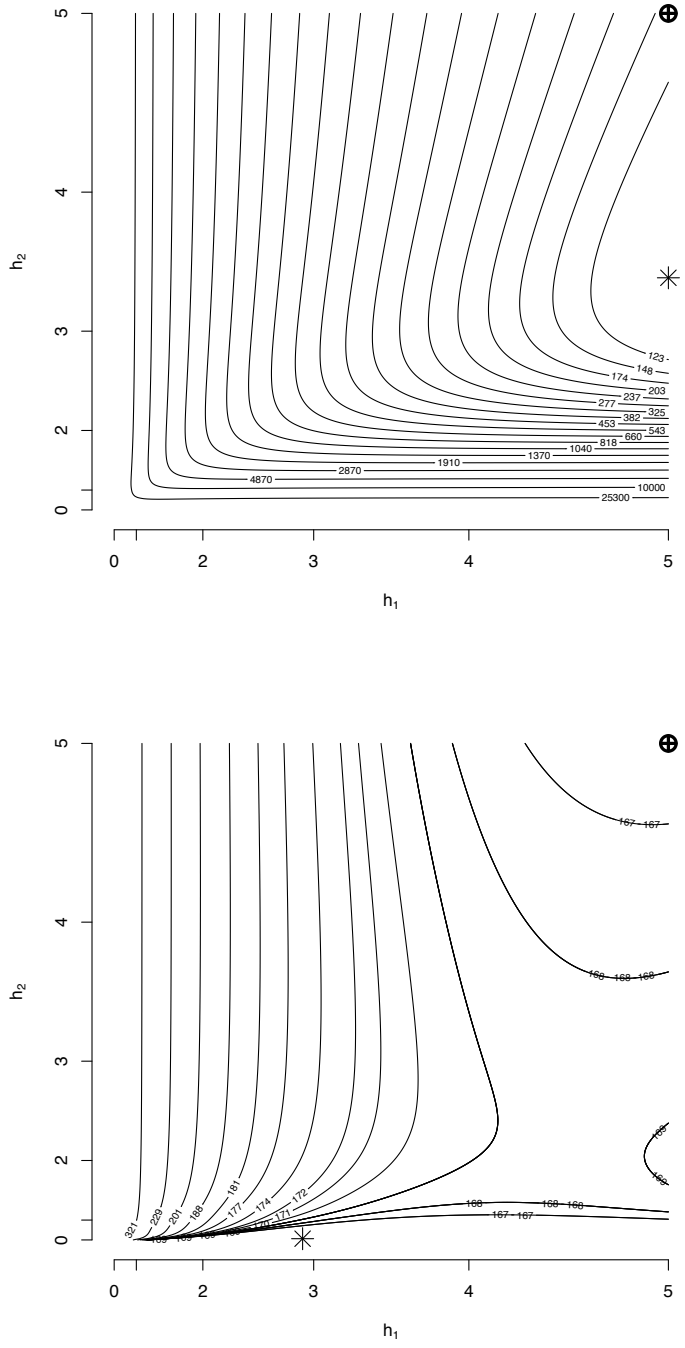

Figure 5.7: Contours of cross-validation criterion as a function of the smoothing bandwidths h_1, h_2, for the Geelong data separated into night and day accidents. *Top*: modified Kelsall-Diggle criterion (5.17). *Bottom:* negative likelihood cross-validation criterion (2.47). Geelong data, relative risk, night versus day. Symbol $\oplus$ indicates optimal symmetric bandwidth h; symbol $*$ indicates optimal joint bandwidths (h_1, h_2).

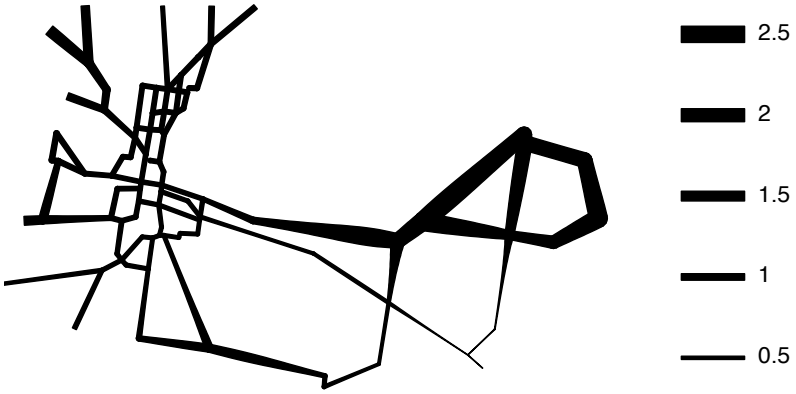

Figure 5.8: Relative risk of single-vehicle versus multiple-vehicle accidents in the Geelong data, using bandwidth 5 km. Line width proportional to relative risk.

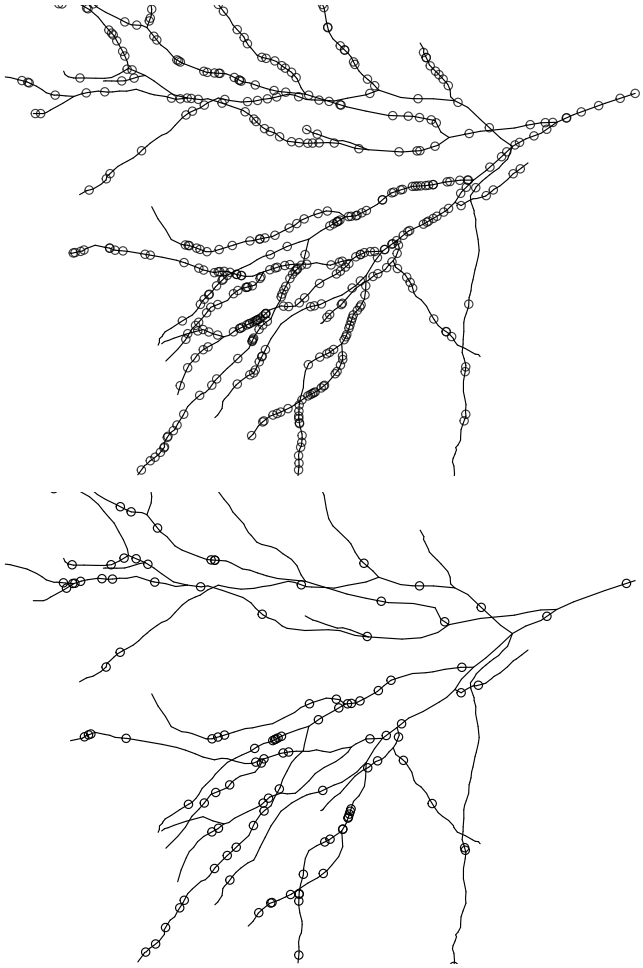

Figure 5.9: Dendritic spine data. One branch of the dendritic tree of a neuron, showing the positions of dendritic spines, of "stubby" or "mushroom" type (*Top*) and "thin" type (*Bottom*).

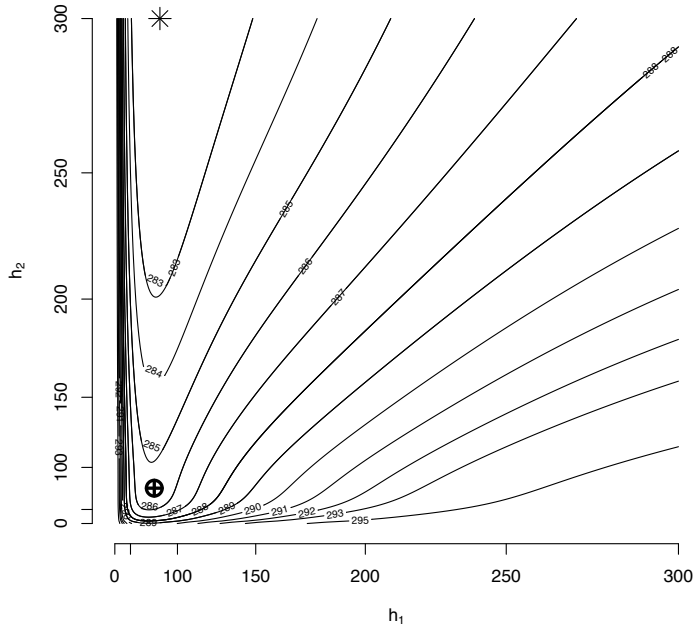

Figure 5.10: Contours of likelihood cross-validation criterion (2.47) for dendrite data, relative risk of 'thin' type against 'other' types. Symbol $\oplus$ indicates optimal symmetric bandwidth h; symbol $*$ indicates optimal joint bandwidths (h_1, h_2).

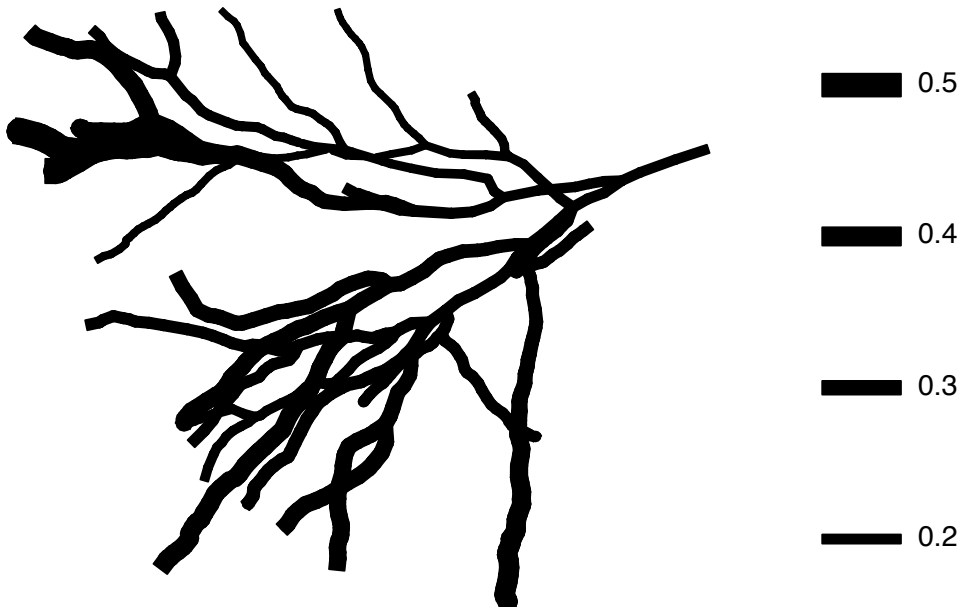

Figure 5.11: Estimated relative risk of "thin" type against other types for the dendritic spine data. Line width proportional to relative risk. Bandwidth 83.5 microns, selected by our modified cross-validation method.

Chapter 6

Discussion

This chapter provides commentary on the original contributions made in Chapters 3–5, and notes some topics which could be the subject of future research.

6.1 Parametric model-fitting

Chapter 3 described fundamentals of a statistical methodology for spatial modelling and analysis of road accident data (including supporting algorithms and computational implementation). The methodology appears to be new in the context of road accident analysis. It idealises the accident locations as the outcome of a spatial point process on a linear network.

The point process approach allows us to treat the accident intensity as spatially varying at any scale, and to model the intensity as a function of explanatory variables which may be spatially varying because of their dependence on road geometry, road condition, physical environment and other relevant causes. Accident intensity may vary smoothly along a road, or may have abrupt jumps associated with abrupt changes in conditions.

For practical purposes, we assumed the accident locations follow a Poisson point process (appropriateness of the Poisson model was discussed). The model parameters were estimated using maximum likelihood, approximated using a new version of the Berman–Turner device. The associated computational algorithm enables us to maximise the log-likelihood using existing software for fitting generalised linear models.

The methodology was demonstrated by fitting several models to the Geelong traffic accident data and interpreting the results. This demonstration elucidated some basic principles of the technique, but was intended mainly for expository purposes.

Our analysis of the Geelong data is not claimed to be definitive or conclusive. A satisfactory, realistic analysis would require a larger sample size and a wider choice of explanatory variables. In particular, road geometry (ideally the geometric road design parameters relating

to cross-section, horizontal alignment and vertical alignment could be extracted directly from the original design documentation), road management (such as the positions of traffic lights) and road condition would have been useful covariates. The scope of traffic accident intensity models is limited only by the availability of covariates, but in practice, a large sample size is required for satisfactory model-fitting. At the time Chapter 3 was written, our code was computationally inefficient and did not allow us to analyse larger datasets. We note that, recently Rakshit *et al.* [116] have developed efficient code to handle large data sets on linear networks.

The chapter did not discuss methods for testing goodness-of-fit of the point process models, or for informal validation of the models. This is a topic for future research. We foresee that goodness-of-fit and model validation methods for two-dimensional point process models could be extended to linear networks.

6.2 Kernel estimation

Chapter 4 developed a principled approach to kernel density estimation on a linear network, which is mathematically natural, enjoys good statistical properties, and is extremely fast to compute.

The new method can be partially reconciled with existing heuristic techniques: it is mathematically equivalent to an infinite-sum generalization of the 'equal-split continuous' rule [109, 110] applied to the Gaussian density. We obtained a completely different mathematical characterization of the diffusion estimator, and used a completely different algorithm to compute it.

For simplicity we considered only fixed-bandwidth smoothing. Adaptive smoothing can also be dealt with using diffusion; indeed this was a principal purpose of [27] and similar methods should be applicable here.

Our discretization procedure effectively rounds the length of each segment to the nearest multiple of Δx. While this is intuitively reasonable, a formal justification would effectively require establishing that the solution of the heat equation is continuous as a function of the network edge lengths for a given connectivity.

In addition to estimating probability density or point process intensity, kernels are also used for many other purposes, such as estimation of relative risk, smoothing of data observed at sample points, and local likelihood. Our work now enables these activities to be performed on a linear network.

Kernel smoothing is a non-parametric alternative to the parametric estimation of intensity that was the subject of Chapter 3. The kernel estimates of accident intensity for the Geelong data obtained in Chapter 4 can be compared side-by-side with the predictions (fitted intensities) of the various loglinear models fitted in Chapter 3 to the same data.

Another possibility is a semi-parametric analysis in which the accident rate is adjusted for the traffic volume by computing the kernel estimate with weights inversely proportional to traffic volume. The result is an estimate of the ratio $r(u) = \lambda(u)/A(u)$ where $\lambda(u)$ is the accident intensity and $A(u)$ is the traffic volume, implicitly assuming that $r(u)$ is smoothly varying.

There are two ways in which this could be done. We will call these, *method 1* and *method 2*.

In method 1, we smooth the point pattern first and then divide the estimated intensity by traffic volume at the corresponding location

$$\hat{r}(u) = \frac{\widehat{\lambda}(u)}{A(u)} = \frac{1}{A(u)} \sum_{i=1}^{n} \kappa(u \mid x_i). \tag{6.1}$$

In method 2, each data point is weighted by dividing it by traffic volume at that location, then we smooth the weighted point pattern

$$\hat{r}(u) = \sum_{i=1}^{n} \frac{\kappa(u \mid x_i)}{A(x_i)}. \tag{6.2}$$

Thus, now an accident occurring in a low traffic volume location has a smaller contribution than an accident in a high traffic volume location since its kernel will have less mass.

Figure 6.1 shows a plot of traffic volume on the Geelong roads and also shows the results of applying method 1 and method 2 using the diffusion kernel, discussed in Chapter 4, with a bandwidth of 2500 metres. The plots show that the two produce different results. Note that method 2 will always be continuous but method 1 will only be continuous if $A(u)$ is continuous. Other covariates could be used instead of $A(u)$ and multiple covariates and functions of covariates could also be used.

6.3 Relative risk

Chapter 5 has demonstrated that existing methodology for estimating relative risk in spatial point patterns in two dimensions can be adapted to point patterns on a linear network, with broadly similar results. However, the known weaknesses of cross-validation methods seem to be amplified on a linear network. We identified at least one explanation for these problems and proposed a solution which may also be applicable to the two-dimensional case. Additionally, the computational challenges of data analysis are much greater on a network; kernel estimates take much longer to compute, and leave-one-out estimates cannot easily be calculated. We proposed a workable approximation to the leave-one-out calculation.

Our proposed techniques could be applied to any of the kernel estimators that have been proposed in the literature on linear networks. However, the diffusion (heat) kernel estimator

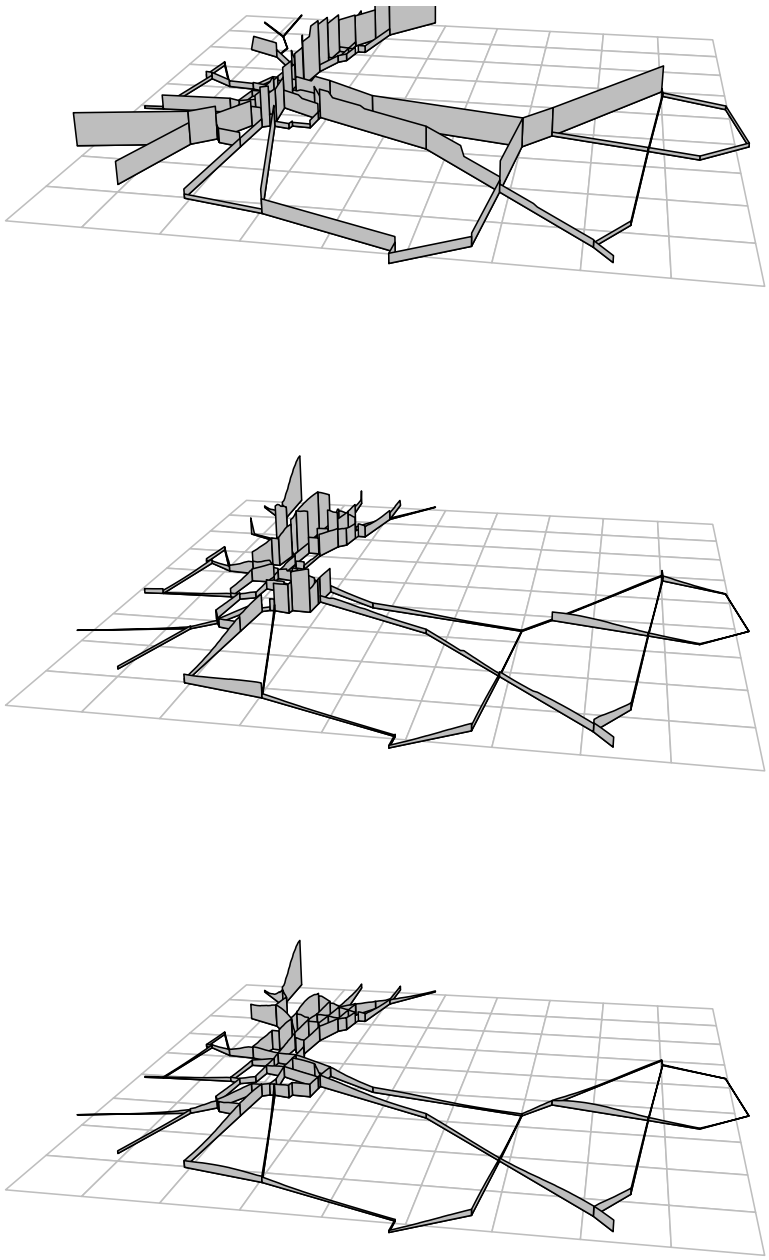

Figure 6.1: Perspective plots. *Top*: Plot of traffic volume intensity on the Geelong roads. *Middle*: Semi-parametric estimate of relative intensity $r(u)$ adjusted by traffic volume using method 1. *Bottom*: Semi-parametric estimate of relative intensity $r(u)$ adjusted by traffic volume using method 2.

has many practical advantages. The finite-element algorithm for solving the heat equation is fast, and it automatically provides kernel estimates for a sequence of intermediate values of bandwidth as well as for the desired bandwidth. Total computation time increases quadratically with the bandwidth, so it becomes important to choose a sharp upper bound on the maximum bandwidth to be considered.

In a suite of experiments, we found that suboptimal (and sometimes unusable) estimates were obtained if the smoothing bandwidths of numerator and denominator were permitted to be different. Simple rules of thumb performed reasonably well, and were the least susceptible to "breakdown". Overall best performance was achieved by our modification of the Kelsall-Diggle density ratio cross-validation method.

We recommend using the diffusion (heat) kernel estimator, and to select the bandwidth using cross-validation with a symmetric bandwidth, using our one-step approximation to the leave-one-out estimator. An *infinite* bandwidth may be valid and can easily be included in the calculations. Two-dimensional convolution smoothing [117] could be used as a first approximation.

Adaptive smoothing, in the style of [3], can be implemented using the slicing algorithm of [44]. Bandwidth selection can be performed using the same cross-validation criteria as above (applied to the global bandwidth parameter).

There are many avenues for future research. Extension to more than two types of points is straightforward. For faster computation in very large networks, convolution kernels should be considered [117]. It would be useful to extend the methods of [70], for identifying regions of (statistically) significantly elevated relative risk, to linear networks. It remains a challenge to adapt the oversmoothing principle of [125] to a linear network.

We believe our modification to the Kelsall-Diggle cross-validation criterion would also perform well for two-dimensional spatial and spatio-temporal point patterns.

6.4 Kernel estimates and the K–function

This section discusses a possible topic for future research.

Ripley's K–function [33] is a standard tool in the analysis of point patterns in two-dimensional space. Adapting this technique to point patterns on a linear network is a logical step, but this has proven to be quite complicated [110, Chap. 6], [8, 12, 13, 115] and computationally demanding [116].

For point patterns in two-dimensional space, there is a close relationship between estimates of Ripley's K–function [33] and leave-one-out kernel estimates of intensity. Namely, the empirical K–function (without edge correction) is proportional to the sum, over all data points, of the leave-one-out estimates of intensity at these points, using a uniform kernel. This relationship

is not particularly useful in the two-dimensional setting, but it may be so on a linear network.

For point patterns on a linear network we propose to replace the empirical K–function by the sum of leave-one-out kernel estimates of intensity based on the heat kernel. This has many advantages: edge correction is intrinsic, computation is relatively fast, all the paths are considered, and good theoretical properties are retained.

6.4.1 Connection between K–function and kernel estimate

In two-dimensional space, if $\mathbf{x} = \{x_1, \ldots, x_n\}$ is a point pattern observed in a study region W, the un-corrected empirical K–function is

$$\widehat{K}(r) = \frac{|W|}{n(n-1)} \sum_i \sum_{j \neq i} I(d_{i,j} \leq r), \quad r \geq 0, \tag{6.3}$$

where $|W|$ is the area of W and $d_{i,j} = \|x_i - x_j\|$ is the Euclidean distance between points x_i and x_j. This can be suggestively rewritten

$$\widehat{K}(r) = \pi r^2 \frac{|W|}{n(n-1)} \sum_i \sum_{j \neq i} \kappa_r(x_j \mid x_i) \tag{6.4}$$

where

$$\kappa_r(u \mid v) = \frac{I(\|u - v\| \leq r)}{\pi r^2} \tag{6.5}$$

is the probability density of the uniform distribution on the disc of radius r centred at v. The inner summation in (6.4) is

$$\hat{\lambda}^{-i}(x_i) = \sum_{j \neq i} \kappa_r(x_j \mid x_i), \tag{6.6}$$

which we recognise as the leave-one-out estimate of point process intensity at the data point x_i using the uniform kernel (6.5). An obvious generalisation would be to replace the uniform kernel by another kernel. Noting that the replacement kernel must have the symmetrical property $K(x \mid u) = K(u \mid x)$.

6.4.2 Heat kernel K–function

On a linear network L, suppose we have observed a point pattern $\mathbf{x} = \{x_1, \ldots, x_n\}$, where $x_i \in L$ for $i = 1, \ldots, n$. Then we may consider the analogue of (6.4),

$$\widehat{H}(\sigma) = \frac{|L|}{n(n-1)} \sum_i \sum_{j \neq i} \kappa_\sigma(x_i \mid x_j), \quad \sigma \geq 0, \tag{6.7}$$

where κ_σ now denotes the *heat kernel* (4.6) with bandwidth σ. This may be regarded as a smoothed and renormalised version of the K–function, and we draw particular attention to the fact that the distance variable r has been replaced by the bandwidth σ.

For a homogeneous Poisson process on L with intensity λ, the double sum in (6.7) has expectation

$$\mathbb{E}[\sum_i \sum_{j \neq i} \kappa_\sigma(x_i \mid x_j)] = \lambda^2 |L|$$

so that $\mathbb{E}[\widehat{H}(\sigma)] \equiv 1$.

Computation of $\widehat{H}(r)$ is feasible using the computational tools described in Chapter 4.

6.4.3 Examples

The examples below are initial experimental investigations of the above notion. We do a comparison between the linear network K–function proposed in Ang *et al* [8] and implemented in *spatstat* and the implementation of (6.7). In order to make a direct comparison, we divide the Linear network K–function proposed in Ang *et al* by r, the distance term, so that now the expected value for a homogeneous Poisson process will be 1 for both methods.

Figure 6.2 shows the first example, a randomly generated Thomas process (using the diffusion kernel from Chapter 4) on the Geelong data linear network with parent intensity of 0.5 points/km, mean number of points per cluster 20, and standard deviation 2 km. Both K–function methods are used to analyse the data and the results for both are shown in Figure 6.3.

Figure 6.4 shows the second example. This is the spiders data from *spatstat*, which was first used in the paper by Ang *et al* [8]. In this example we also show the envelope produced by the maximums and minimums of 39 simulations. Again, both K–function methods are used to analyse the data and the results for both are shown in Figure 6.5.

The comparison examples do not show a large difference between the two methods. Possibly an example on very densely connected network would show more of a difference, since considering all of the paths on such network may highlight the differences between the two methods.

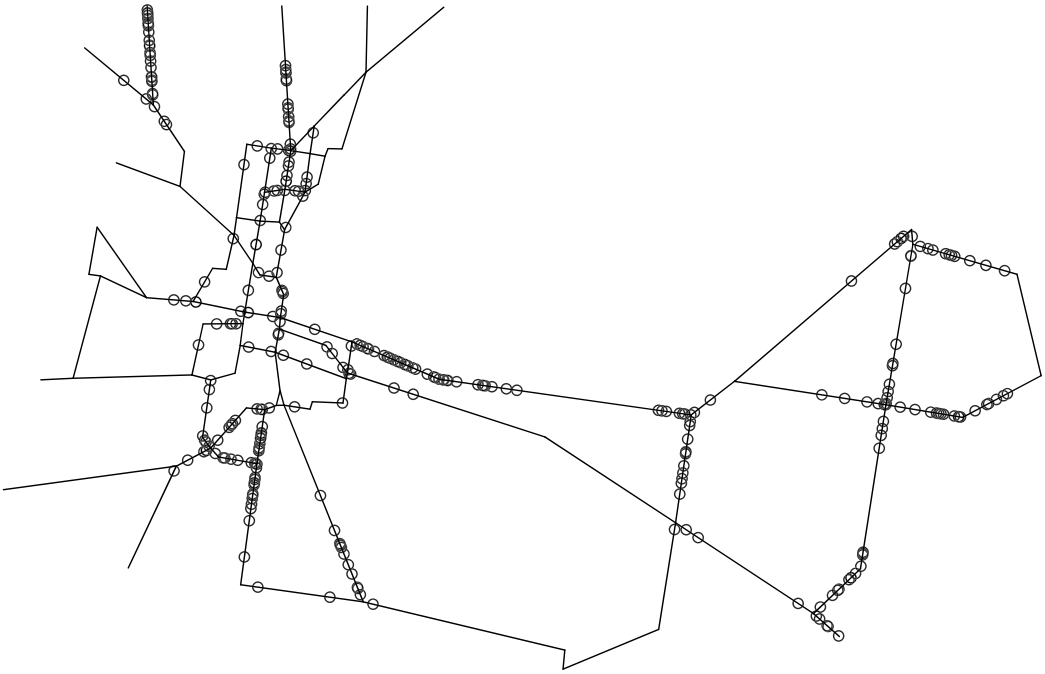

Figure 6.2: Thomas process on the Geelong data linear network.

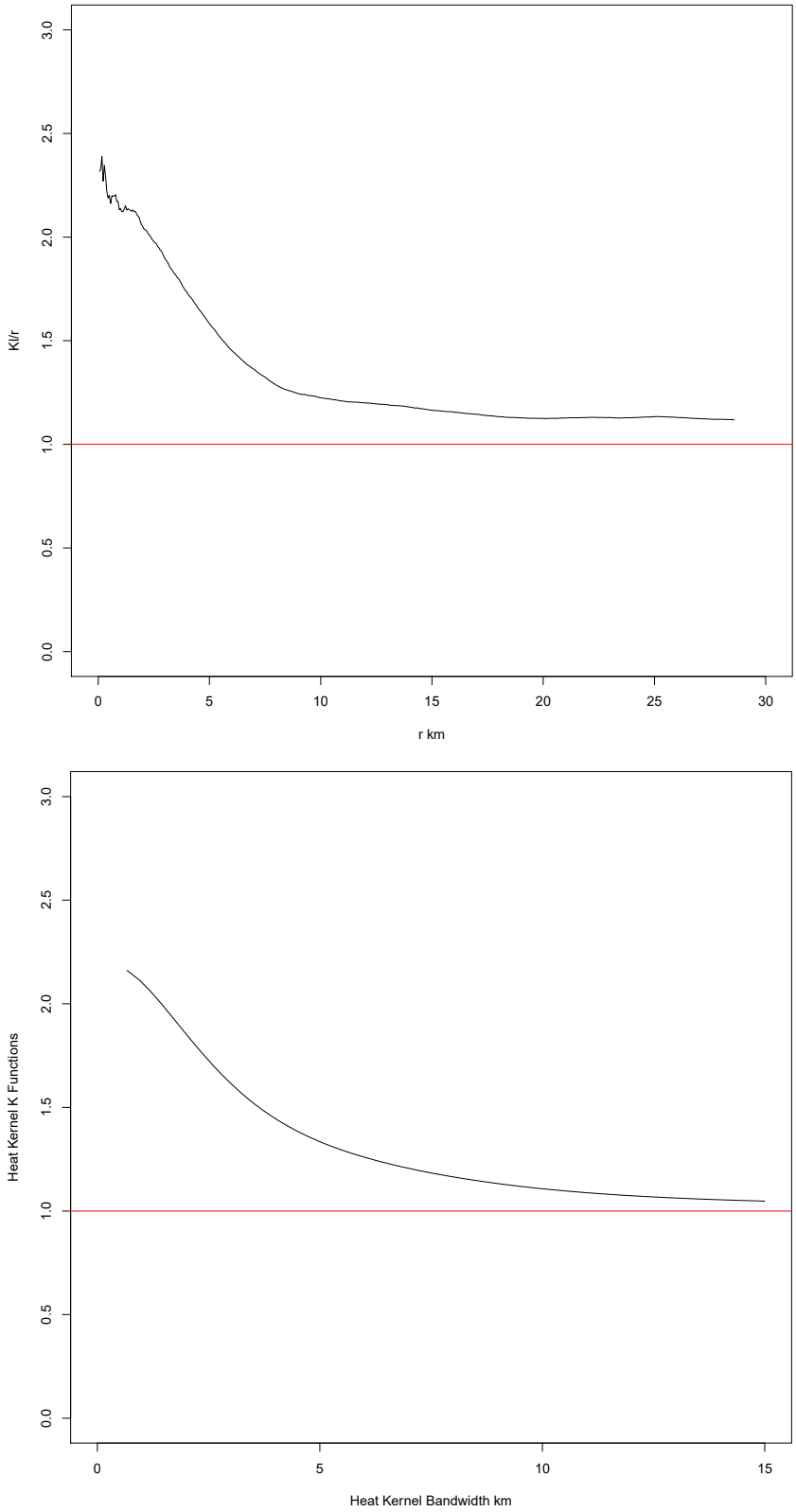

Figure 6.3: *Top*: Linear network K–function proposed in Ang *et al.* *Bottom*: Heat Kernel K–function.

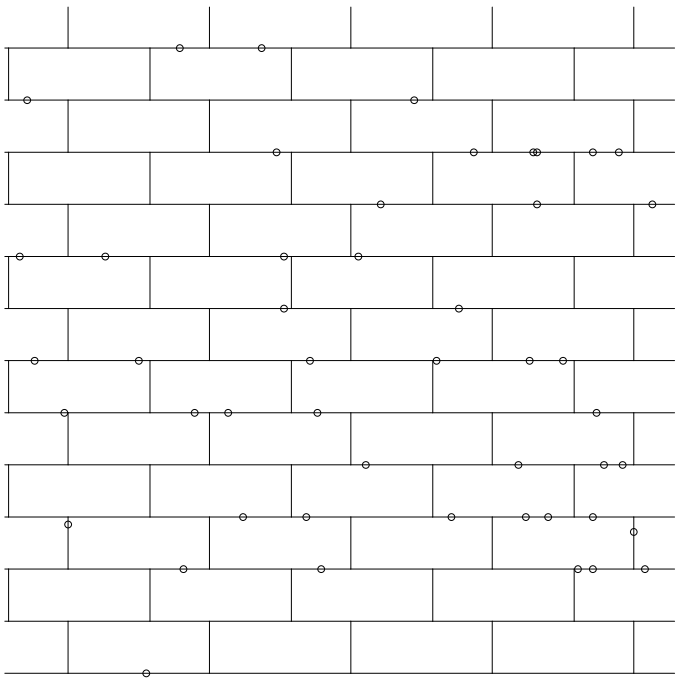

Figure 6.4: Spiders data from Ang *et al.*

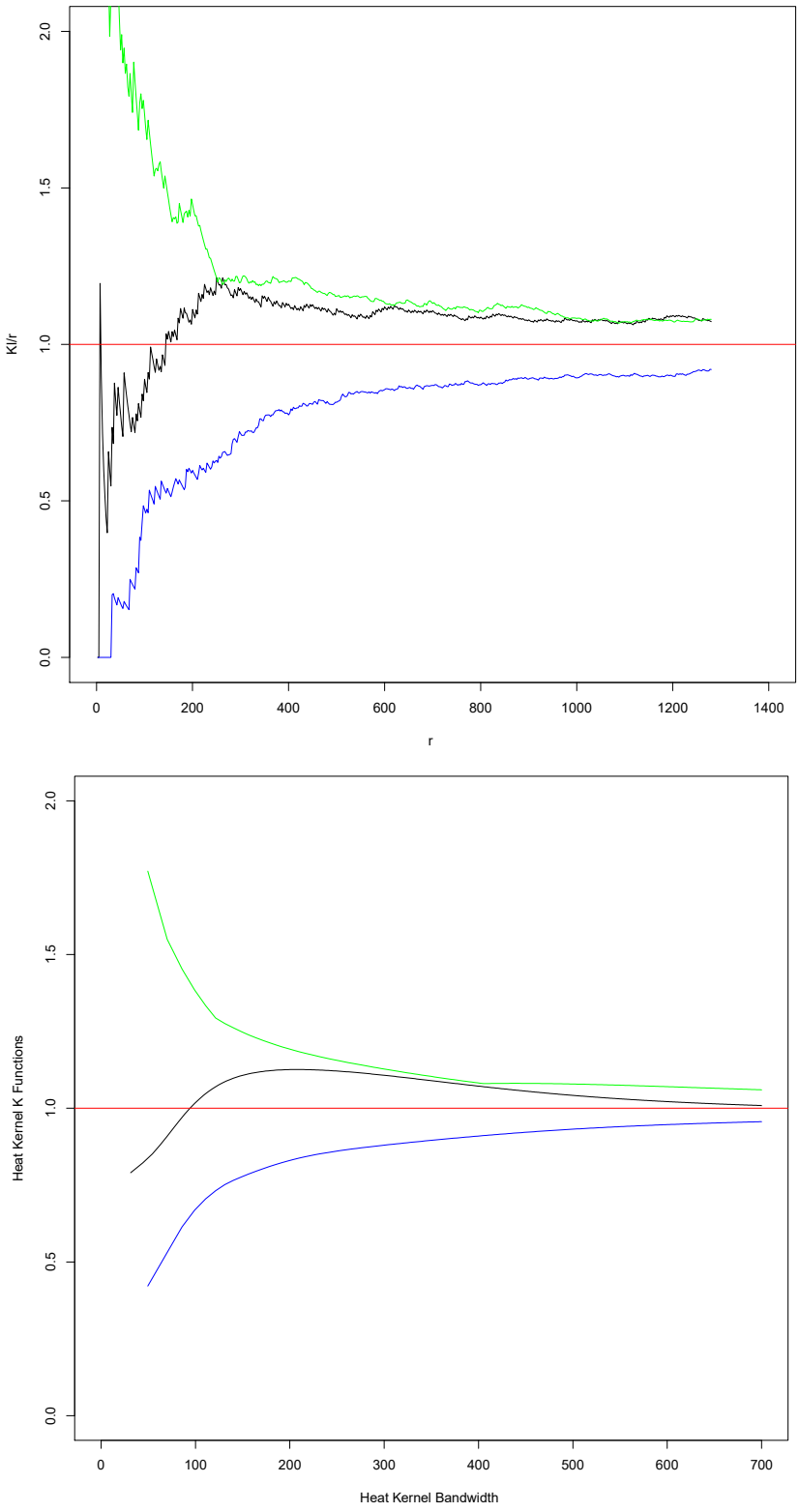

Figure 6.5: *Top*: Linear network K–function proposed in Ang *et al* with envelope. *Bottom*: Heat Kernel K–function with envelope

Bibliography

[1] *CrashStats website.* https://www.vicroads.vic.gov.au/safety-and-road-rules/safety-statistics/crash-statistics.

[2] M. Abdel-Aty and A. Radwan, 'Modeling traffic accident occurrence and involvement', *Accident Analysis and Prevention* **32** (2000), 633–642.

[3] I. Abramson, 'On bandwidth estimation in kernel estimates – a square root law', *Annals of Statistics* **10** (1982), no. 4, 1217–1223.

[4] F. Agterberg, 'Automatic contouring of geological maps to detect target areas for mineral exploration', *Journal of the International Association for Mathematical Geology* **6** (1974), 373–395.

[5] P. C. Anastasopoulos and F. L. Mannering, 'A note on modeling vehicle accident frequencies with random-parameters count models', *Accident Analysis & Prevention* **41** (2009), no. 1, 153–159.

[6] E. Anderes, J. Møller, and J. G. Rasmussen, *Isotropic covariance functions on graphs and their edges* (2019). to appear in Annals of Statistics arXiv 1710.01295.

[7] W. N. Anderson Jr and T. D. Morley, 'Eigenvalues of the Laplacian of a graph', *Linear and multilinear algebra* **18** (1985), no. 2, 141–145.

[8] Q. Ang, A. Baddeley, and G. Nair, 'Geometrically corrected second order analysis of events on a linear network, with applications to ecology and criminology', *Scandinavian Journal of Statistics* **39** (2012), 591–617.

[9] A. Azzalini, A. Bowman, and W. Haerdle, 'On the use of nonparametric regression for model checking', *Biometrika* **76** (1989), 1–11.

[10] A. Baddeley, M. Berman, N. Fisher, A. Hardegen, R. Milne, D. Schuhmacher, R. Shah, and R. Turner, 'Spatial logistic regression and change-of-support in Poisson point processes', *Electron. J. Statist.* **4** (2010), 1151–1201.

[11] A. Baddeley, M. Berman, N. Fisher, A. Hardegen, R. Milne, D. Schuhmacher, and R. Turner, 'Spatial logistic regression and change-of-support for Poisson point processes', *Electronic Journal of Statistics* **4** (2010), 1151–1201.

[12] A. Baddeley, A. Jammalamadaka, and G. Nair, 'Multitype point process analysis of spines on the dendrite network of a neuron', *Applied Statistics (Journal of the Royal Statistical Society, Series C)* **63** (2014), no. 5, 673–694.

[13] A. Baddeley, G. Nair, S. Rakshit, and G. McSwiggan, ''Stationary' point processes are uncommon on linear networks', *STAT* **6** (2017), no. 1, 68–78.

[14] A. Baddeley, E. Rubak, and R. Turner, *Spatial point patterns: Methodology and applications with R* (Chapman and Hall/CRC, London, 2015).

[15] A. Baddeley and R. Turner, 'Practical maximum pseudolikelihood for spatial point patterns (with discussion)', *Australian and New Zealand Journal of Statistics* **42** (2000), no. 3, 283–322.

[16] A. Baddeley and R. Turner, 'Spatstat: an R package for analyzing spatial point patterns', *Journal of Statistical Software* **12** (2005), no. 6, 1 – 42.

[17] A. Baddeley and R. Turner, 'Spatstat: an R package for analyzing spatial point patterns', *Journal of Statistical Software* **12** (2005), no. 6, 1–42. URL: `www.jstatsoft.org`, ISSN: 1548-7660.

[18] R. Barry and J. McIntyre, 'Estimating animal densities and home range in regions with irregular boundaries and holes: a lattice-based alternative to the kernel density esimator', *Ecological Modelling* **222** (2011), 1666–1672.

[19] D. Bates and M. Maechler, *Matrix: Sparse and dense matrix classes and methods* (2015). `http://CRAN.R-project.org/package=Matrix`. R package version 1.2-1.

[20] M. Berman and T. Turner, 'Approximating point process likelihoods with GLIM', *Applied Statistics* **41** (1992), 31–38.

[21] M. Berman and P. Diggle, 'Estimating weighted integrals of the second-order intensity of a spatial point process', *Journal of the Royal Statistical Society: Series B (Methodological)* **51** (1989), no. 1, 81–92.

[22] J. F. Bithell, 'An application of density estimation to geographical epidemiology', *Statistics in Medicine* **9** (1990), no. 6, 691–701.

[23] G. Bonham-Carter, *Geographic Information Systems for geoscientists: modelling with GIS*, in *Computer Methods in the Geosciences* **13** (Pergamon Press/ Elsevier, Kidlington, Oxford, UK, 1995).

[24] G. Borruso, 'Network density and the delimitation of urban areas', *Transactions in GIS* **7** (2003), 177–191.

[25] G. Borruso, 'Network density estimation: Analysis of point patterns over a network', in *Computational Science and its Applications — ICCSA 2005*, *Eds.* O. Gervasi, M. Gavrilova, V. Kumar, A. Laganà, H. Lee, Y. Mun, D. Taniar, and C. Tan, in *Lecture Notes in Computer Science* **3482**, pp. 126–132 (Springer, Berlin/Heidelberg, 2005).

[26] G. Borruso, 'Network density estimation: A GIS approach for analysing point patterns in a network space', *Transactions in GIS* **12** (2008), 377–402.

[27] Z. Botev, J. Grotowski, and D. Kroese, 'Kernel density estimation via diffusion', *Annals of Statistics* **38** (2010), no. 5, 2916–2957.

[28] A. W. Bowman and A. Azzalini, *Applied smoothing techniques for data analysis: the kernel approach with S-Plus illustrations* (Oxford University Press, Oxford, 1997).

[29] A. Bowman, 'An alternative method of cross-validation for the smoothing of density estimates', *Biometrika* **71** (1984), 353–360.

[30] D. Brillinger, 'Comparative aspects of the study of ordinary time series and of point processes', in *Developments in Statistics*, *Ed.* P. Krishnaiah, pp. 33–133 (Academic Press, New York, London, 1978).

[31] D. Brillinger and H. Preisler, 'Two examples of quantal data analysis: a) multivariate point process, b) pure death process in an experimental design', in *Proceedings, XIII International Biometric Conference, Seattle* (International Biometric Society, 1986), 94–113.

[32] D. Brillinger and J. Segundo, 'Empirical examination of the threshold model of neuron firing', *Biological Cybernetics* **35** (1979), 213–220.

[33] B.Ripley, 'Modelling spatial patterns (with discussion)', *Journal of the Royal Statistical Society, series B* **39** (1977), 172–212.

[34] S. Cafiso, A. D. Graziano, G. D. Silvestro, G. L. Cava, and B. Persaud, 'Development of comprehensive accident models for two-lane rural highways using exposure, geometry, consistency and context variables', *Accident Analysis & Prevention* **42** (2010), no. 4, 1072 – 1079.

[35] R. Cao, A. Cuevas, and W. G. Manteiga, 'A comparative study of several smoothing methods in density estimation', *Computational Statistics & Data Analysis* **17** (1994), no. 2, 153–176.

[36] P. Chaudhuri and J. Marron, 'Scale space view of curve estimation', *Annals of Statistics* **28** (2000), 408–428.

[37] A. B. Clark and A. B. Lawson, 'An evaluation of non-parametric relative risk estimators for disease maps', *Computational statistics & data analysis* **47** (2004), no. 1, 63–78.

[38] A. Comber, C. Brunsdon, and E. Green, 'Using a GIS-based network analysis to determine urban greenspace accessibility for different ethnic and religious groups', *Landscape and Urban Planning* **86** (2008), 103–114.

[39] D. R. Cox, 'Some statistical methods connected with series of events', *Journal of the Royal Statistical Society. Series B (Methodological)* **17** (1955), no. 2, 129–164. `http://www.jstor.org/stable/2983950`.

[40] N. A. C. Cressie, *Statistics for Spatial Data*, in *Wiley series in probability and mathematical statistics. Applied probability and statistics.* (Wiley, New York, rev. ed., c1993).

[41] D. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes* (Springer-Verlag, New York, 1988).

[42] D. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes. Volume I: Elementary Theory and Methods* (New York, second ed., 2003).

[43] D. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes. Volume II: General Theory and Structure* (Springer-Verlag, New York, second ed., 2008).

[44] T. Davies and A. Baddeley, 'Fast computation of spatially adaptive kernel estimates', *Statistics and Computing* **28** (2018), 937–956.

[45] T. M. Davies, M. L. Hazelton, J. C. Marshall, et al., 'Sparr: analyzing spatial relative risk using fixed and adaptive kernel density estimation in r', *Journal of Statistical Software* **39** (2011), no. i01.

[46] T. M. Davies, J. C. Marshall, and M. L. Hazelton, 'Tutorial on kernel estimation of continuous spatial and spatiotemporal relative risk', *Statistics in medicine* **37** (2018), no. 7, 1191–1221.

[47] T. Davies, K. Jones, and M. Hazelton, 'Symmetric adaptive smoothing regimens for estimation of the spatial relative risk function', *Computational Statistics and Data Analysis* **101** (2016), 12–28.

[48] T. Davies and A. Lawson, 'An evaluation of likelihood-based bandwidth selectors for spatial and spatiotemporal kernel estimates', *Journal of Statistical Computation and Simulation* **89** (2019), no. 7, 1131–1152.

[49] B. Deckers, K. Verheyen, M. Hermy, and B. Muys, 'Effects of landscape structure on the invasive spread of black cherry *Prunus serotina* in an agricultural landscape in Flanders, Belgium', *Ecography* **28** (2005), 99–109.

[50] P. Diggle, 'A kernel method for smoothing point process data.', *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **34** (1985), no. 2, 138.

[51] P. Diggle, *Statistical Analysis of Spatial Point Patterns* (Arnold, London, 2nd ed. ed., 2003).

[52] P. Diggle and J. S. Marron, 'Equivalence of smoothing parameter selectors in density and intensity estimation', *Journal of the American Statistical Association* **83** (1988), no. 403, 793–800.

[53] P. Diggle, *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns* (Chapman and Hall/CRC, Boca Raton, FL, third ed., 2014).

[54] P. Diggle, P. Zheng, and P. Durr, 'Non-parametric estimation of spatial segregation in a multivariate point process: bovine tuberculosis in Cornwall, UK', *Applied Statistics* **54** (2005), 645–658.

[55] J. Downs and M. Horner, 'Characterising linear point patterns', in *Proceedings of the GIScience Research UK Conference (GISRUK), Maynooth, Ireland*, *Ed.* A. Winstanley (National University of Ireland Maynooth, County Kildare, Ireland, 2007), 421–424.

[56] J. Downs and M. Horner, 'Network-based kernel density estimation for home range analysis', *Proceedings of the 9th International Conference on Geocomputation, Maynooth, Ireland* (2007).

[57] J. Downs and M. Horner, 'Spatially modelling pathways of migratory birds for nature reserve site selection', *International Journal of Geographical Information Science* **22** (2008), no. 6, 687–702.

[58] T. Duong and M. Hazelton, 'Plug-in bandwidth matrices for bivariate kernel density estimation', *Journal of Nonparametric Statistics* **15** (2003), no. 1, 17–30.

[59] T. Duong and M. Hazelton, 'Cross-validation bandwidth matrices for multivariate kernel density estimation', *Scandinavian Journal of Statistics* **32** (2005), 485–506.

[60] R. A. Fisher, 'Theory of statistical estimation', in *Mathematical Proceedings of the Cambridge Philosophical Society*, **22**, no. 05 (Cambridge Univ Press, 1925), 700–725.

[61] D. A. Freedman, 'Ecological inference and the ecological fallacy', *International Encyclopedia of the Social & Behavioral sciences* **6** (1999), 4027–4030.

[62] M. Freidlin and A. Wentzell, 'Diffusion processes on graphs and the averaging principle', *Annals of Probability* **21** (1993), 2215–2245.

[63] B. Gaveau and M. Okada, 'Differential forms and heat diffusion on one dimensional singular varieties', *Bulletin des Sciences Mathématiques, Deuxième serie* **115** (1991), 61–80.

[64] B. Gaveau, M. Okada, and T. Okada, 'Explicit heat kernels on graphs and spectral analysis', in *Several Complex Variables*, *Ed.* J. Fornaess, in *Princeton Mathematical Notes* **38**, pp. 364–388 (Princeton University Press, Princeton, NJ, 1993).

[65] A. Gelfand, P. Diggle, M. Fuentes, and P. Guttorp (eds.), *Handbook of Spatial Statistics* (CRC Press, Boca Raton, Fl., 2010).

[66] W. Haerdle, *Applied nonparametric regression* (Cambridge University Press, Cambridge, 1990).

[67] P. Hall and J. Marron, 'Local minima in cross-validation functions', *Journal of the Royal Statistical Society, Series B* **53** (1991), 245–252.

[68] P. Hall, J. Marron, and B. Park, 'Smoothed cross-validation', *Probability Theory and Related Fields* **92** (1992), 1–20.

[69] H. Hautzinger, 'Regression analysis of aggregate accident data: some methodological considerations and practical experiences', *Accident Analysis and Prevention* **18** (1986), 95–102.

[70] M. Hazelton and T. Davies, 'Inference based on kernel estimates of the relative risk function in geographical epidemiology', *Biometrical Journal* **51** (2009), 98–109.

[71] R. V. Hogg and A. T. Craig, *Introduction to Mathematical Statistics.(5th edition)* (Upper Saddle River, New Jersey: Prentice Hall, 1995).

[72] S. Hu, D. S. Poskitt, and X. Zhang, 'Bayesian adaptive bandwidth kernel density estimation of irregular multivariate distributions', *Computational Statistics and Data Analysis* **56** (2012), 732–740.

[73] J. Illian, A. Penttinen, H. Stoyan, and D. Stoyan, *Statistical analysis and modelling of spatial point patterns* (John Wiley and Sons, Chichester, 2008).

[74] A. Jammalamadaka, S. Banerjee, B. Manjunath, and K. Kosik, 'Statistical analysis of dendritic spine distributions in rat hippocampal cultures', *BMC Bioinformatics* **14** (2013), no. 287.

[75] M. Jones, 'Simple boundary corrections for kernel density estimation', *Statistics and Computing* **3** (1993), 135–146.

[76] M. Jones, J. Marron, and S. Sheather, 'A brief survey of bandwidth selection for density estimation', *Journal of the American Statistical Association* **91** (1996), no. 433, 401–407.

[77] C. Jurewicz and P. Bennett, 'Road safety engineering risk assessment part 7: Crash rates database', (2010). ISBN 978-1-921709-02-9.

[78] A. Karaganis and A. Mimis, 'A spatial point process for estimating the probability of occurrence of a traffic accident', *ERSA conference papers* **06p640** (European Regional Science Association, Vienna, 2006).

[79] J. Kelsall and P. Diggle, 'Kernel estimation of relative risk', *Bernoulli* **1** (1995), 3–16.

[80] J. Kelsall and P. Diggle, 'Non-parametric estimation of spatial variation in relative risk', *Statistics in Medicine* **14** (1995), 2335–2342.

[81] J. Kelsall and P. Diggle, 'Spatial variation in risk of disease: a nonparametric binary regression approach', *Applied Statistics* **47** (1998), 559–573.

[82] G. Koorey, 'Road data aggregation and sectioning considerations for crash analysis', *Transportation Research Record: Journal of the Transportation Research Board* (2009), no. 2103, 61–68.

[83] V. Kostrykin, J. Potthoff, and R. Schrader, 'Heat kernels on metric graphs and a trace formula', in *Adventures in Mathematical Physics*, *Eds.* F. Germinet and P. Hislop, in *Contemporary Mathematics* **447**, pp. 175–198 (American Mathematical Society, Providence, RI, 2007).

[84] V. Kostrykin, J. Potthoff, and R. Schrader, 'Brownian motions on metric graphs', *Journal of Mathematical Physics* **53** (2012), no. 095206.

[85] V. Kostrykin and R. Schrader, 'Laplacians on metric graphs: eigenvalues, resolvents and semigroups', in *Quantum Graphs and their Applications*, *Eds.* G. Berkolaiko, R. Carlson, S. Fulling, and P. Kuchment, in *Contemporary Mathematics* **415**, pp. 201–225 (American Mathematical Society, Providence, RI, 2006).

[86] Y. Kutoyants, *Statistical inference for spatial Poisson processes*, in *Lecture Notes in Statistics* **134** (Springer, New York, 1998).

[87] P. Lewis, 'Recent results in the statistical analysis of univariate point processes', in *Stochastic point processes*, *Ed.* P. Lewis, pp. 1–54 (Wiley, New York, 1972).

[88] P. Lewis and G. Shedler, 'Simulation of non-homogeneous Poisson processes by thinning', *Naval Logistics Quarterly* **26** (1979), 406–413.

[89] J. Lindsey, *The analysis of stochastic processes using GLIM* (Springer, Berlin, 1992).

[90] J. Lindsey, *Modelling frequency and count data* (Oxford University Press, 1995).

[91] J. Lindsey, *Applying generalized linear models* (Springer, 1997).

[92] C. Loader, 'Bandwidth selection: classical or plug-in?', *Annals of Statistics* **27** (1999), no. 2, 415–438.

[93] C. Loader, *Local regression and likelihood* (Springer, New York, 1999).

[94] D. Lord and F. Mannering, 'The statistical analysis of crash frequency data: a review and assessment of methodological alternatives', *Transportation Research A* **44** (2010), 291–305.

[95] D. Lord, S. P. Washington, and J. N. Ivan, 'Poisson, Poisson-gamma and zero-inflated regression models of motor vehicle crashes: balancing statistical fit and theory', *Accident Analysis & Prevention* **37** (2005), no. 1, 35 – 46.

[96] Y. Lu and X. Chen, 'On the false alarm of planar K-function when analyzing urban crime distributed along streets', *Social Science Research* **36** (2007), 611–632.

[97] F. L. Mannering and C. R. Bhat, 'Analytic methods in accident research: methodological frontier and future directions', *Analytic methods in accident research* **1** (2014), 1–22.

[98] P. McCullagh and J. Nelder, *Generalized linear models* (Chapman and Hall, second ed., 1989).

[99] P. McDonald and R. Meyers, 'Diffusions on graphs, Poisson problems and spectral geometry', *Transactions of the American Mathematical Society* **354** (2002), 5111–5136.

[100] G. McSwiggan, A. Baddeley, and G. Nair, 'Kernel density estimation on a linear network', *Scandinavian Journal of Statistics* **44** (2016), no. 2, 324–345.

[101] G. McSwiggan, A. Baddeley, and G. Nair, 'Estimation of relative risk for events on a linear network', *Statistics and Computing* (2019).

[102] J. Mecke, 'Stationäre zufällige maße auf lokalkompakten abelschen gruppen', *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* **9** (1967), 36–58.

[103] S.-P. Miaou, 'The relationship between truck accidents and geometric design of road sections: Poisson versus negative binomial regression', *Accident Analysis and Prevention* **26** (1994), 471–482.

[104] M. M. Moradi, O. Cronie, E. Rubak, R. Lachieze-Rey, J. Mateu, and A. Baddeley, 'Resample-smoothing of voronoi intensity estimators', *Statistics and Computing* **29** (2019), no. 5, 995–1010. `https://doi.org/10.1007/s11222-018-09850-0`.

[105] M. M. Moradi, F. J. Rodríguez-Cortés, and J. Mateu, 'On kernel-based intensity estimation of spatial point patterns on linear networks', *Journal of computational and graphical statistics a joint publication of American Statistical Association, Institute of Mathematical Statistics, Interface Foundation of North America.* **27** (2018-04-03), no. 2, 302,311.

[106] D. Morin, 'Application of statistical concepts to accident data', *Highway Research Record* **188** (1967), 72–79. Published by Highway Research Board of the US National Academies.

[107] A. Okabe and M. Kitamura, 'A computational method for market area analysis on a network', *Geographical Analysis* **28** (1996), 330–349.

[108] A. Okabe and K. Okunuki, 'A computational method for estimating the demand of retail stores on a street network and its implementation in GIS', *Transactions in GIS* **5** (2001), 209–220.

[109] A. Okabe, T. Satoh, and K. Sugihara, 'A kernel density estimation method for networks, its computational method and a GIS-based tool', *International Journal of Geographical Information Science* **23** (2009), 7–32.

[110] A. Okabe and K. Sugihara, *Spatial analysis along networks: Statistical and computational methods* (John Wiley & Sons, 2012).

[111] K. Okunuki and A. Okabe, 'Solving the Huff-based competitive location model on a network with link-based demand', *Annals of Operations Research* **111** (2003), 239–252.

[112] S. Oppe, 'A comparison of some statistical techniques for road accident analysis', *Accident Analysis and Prevention* **24** (1992), 397–423.

[113] E. Parzen, 'On estimation of a probability density function and mode', *The Annals of Mathematical Statistics* **33** (1962), no. 3, 1065–1076.

[114] R Development Core Team, *R: A language and environment for statistical computing* (Vienna, Austria, 2009). ISBN 3-900051-07-0.

[115] S. Rakshit, G. Nair, and A. Baddeley, 'Second-order analysis of point patterns on a network using any distance metric', *Spatial Statistics* **22** (2017), no. 1, 129–154.

[116] S. Rakshit, A. Baddeley, and G. Nair, 'Efficient code for second order analysis of events on a linear network', *Journal of Statistical Software, Articles* **90** (2019), no. 1, 1–37.

[117] S. Rakshit, T. Davies, M. M. Moradi, G. McSwiggan, G. Nair, J. Mateu, and A. Baddeley, 'Fast kernel smoothing of point patterns on a large network using two-dimensional convolution', *International Statistical Review* **0**, no. 0. `https://onlinelibrary.wiley.com/doi/abs/10.1111/insr.12327`.

[118] S. Rakshit, G. Nair, and A. Baddeley, 'Second-order analysis of point patterns on a network using any distance metric', *Spatial Statistics* **22** (2017), 129 – 154.

[119] M. Rosenblatt, 'Remarks on some nonparametric estimates of a density function', *The Annals of Mathematical Statistics* **27** (1956), no. 3, 832–837.

[120] D. Scott, *Multivariate Density Estimation. Theory, Practice and Visualization* (John Wiley and Sons, New York, 1992).

[121] B. Silverman, *Density Estimation for Statistics and Data Analysis* (Chapman and Hall, London, 1986).

[122] N. Som, P. Monestiez, J. Ver Hoef, D. Zimmerman, and E. Peterson, 'Spatial sampling on streams: principles for inference on aquatic networks', *Environmetrics* **25** (2014), no. 5, 306–323.

[123] P. Spooner, I. Lunt, A. Okabe, and S. Shiode, 'Spatial analysis of roadside Acacia populations on a road network using the network K-function', *Landscape Ecology* **19** (2004), 491–499.

[124] K. Sugihara, T. Satoh, and A. Okabe, 'Simple and unbiased kernel function for network analysis', in *ISCIT 2010 (International Symposium on Communication and Information Technologies)* (IEEE, 2010), 827–832.

[125] G. Terrell, 'The maximal smoothing principle in density estimation', *Journal of the American Statistical Association* **85** (1990), 470–476.

[126] L. Tierney, 'Code analysis and parallelizing vector operations in R', *Computational Statistics* **24** (2009), 217–223.

[127] J. Tukey, 'Discussion of paper by F.P. Agterberg and S.C. Robinson', *Bulletin of the International Statistical Institute* **44** (1972), no. 1, 596. Proceedings, 38th Congress, International Statistical Institute.

[128] G. Vandenbulcke-Plasschaert, *Spatial analysis of bicycle use and accident risks for cyclists* (Ph.D. thesis, Université Catholique de Louvain, 2011). ISBN 978-2-87558-019-1.

[129] J. Ver Hoef and E. Peterson, 'A moving average approach for spatial statistical models of stream networks', *Journal of the American Statistical Association* **105** (2010), 6–18.

[130] J. Ver Hoef, E. Peterson, and D. Theobald, 'Spatial statistical models that use flow and stream distance', *Environmental and Ecological Statistics* **13** (2006), no. 4, 449–464.

[131] S. Voss, B. Main, and I. Dadour, 'Habitat preferences of the urban wall spider *Oecobius navus* (Araneae, Oecobiidae)', *Australian Journal of Entomology* **46** (2007), 261–268.

[132] M. Wand and M. Jones, *Kernel smoothing* (Chapman and Hall, 1995).

[133] Z. Xie and J. Yan, 'Kernel density estimation of traffic accidents in a network space', *Computers, Environment and Urban Systems* **32** (2008), 396–406.

[134] A. Yadav, Y. Gao, A. Rodriguez, D. Dickstein, S. Wearne, J. Luebke, P. Hof, and C. Weaver, 'Morphologic evidence for spatially clustered spines in apical dendrites of monkey neocortical pyramidal cells', *Journal of Comparative Neurology* **520** (2012), no. 13, 2888–2902.

[135] I. Yamada and J.-C. Thill, 'Comparison of planar and network K-functions in traffic accident analysis', *Journal of Transport Geography* **12** (2004), 149–158.

[136] G. U. Yule, 'Notes on the theory of association of attributes in statistics', *Biometrika* **2** (1903), no. 2, 121–134.

[137] X. Zhang, M. L. King, and R. J. Hyndman, 'A Bayesian approach to bandwidth selection for multivariate kernel estimation', *Computational Statistics and Data Analysis* **50** (2006), 3009–3031.