Strategies for mean and modal multivariate local regression

JAMES TAYLOR

Strategies for mean and modal multivariate local regression

James Taylor

A Thesis presented for the degree of
Doctor of Philosophy

The Statistics and Probability Group
Department of Mathematical Sciences
University of Durham
England
May 2012

Strategies for mean and modal multivariate local regression

James Taylor

Submitted for the degree of Doctor of Philosophy
May 2012

Abstract

Local polynomial fitting for univariate data has been widely studied and discussed, but up until now the multivariate equivalent has often been deemed impractical, due to the so-called *curse of dimensionality.* Here, rather than discounting it completely, density is used as a threshold to determine where over a data range reliable multivariate smoothing is possible, whilst accepting that in large areas it is not. Further, the challenging issue of multivariate bandwidth selection, which is known to be affected detrimentally by sparse data which inevitably arise in higher dimensions, is considered. In an effort to alleviate this problem, two adaptations to generalized cross-validation are implemented, and a simulation study is presented to support the proposed method. It is also discussed how the density threshold and the adapted generalized cross-validation technique introduced herein work neatly together. Whilst this is the major focus of this thesis, modal regression via mean shift is discussed as an alternative multivariate regression technique. In a slightly different vein, bandwidth selection for univariate kernel density estimation is also examined, and a different technique is proposed for a density with a multimodal distribution. This is supported by a simulation study and its relevance in modal regression is also discussed.

Declaration

The work in this thesis is based on research carried out in the Statistics and Probability Group, the Department of Mathematical Sciences, Durham University, England. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Acknowledgements

I would like to take this opportunity to thank my supervisor, Jochen Einbeck. As well as offering me guidance through some challenging statistics he has also been an excellent mentor. Jochen's door has always been open and I feel immensely privileged to have had him as my supervisor.

I would also like to thank Marco Apollonio, of the Dept. of Zoology and Evolutionary Genetics, University of Sassari, and Tom Mason, of the School of Biological and Biomedical Sciences, University of Durham, for permitting the use of the chamois data.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Broadly speaking, the aim of regression is to identify the underlying trend in a data set, whilst simultaneously not representing the random variation within it. There are two main reasons for wishing to do this. Firstly, in some situations a visual trend in data can be very useful, and secondly, finding an expression which relates variables could help in the prediction at further observations if this is desirable. Nonparametric regression is a large class of such regression techniques in which, as Wand and Jones (1995) point out, the model is shaped completely from the data. This is particularly useful when a parametric model is too restrictive. Such nonparametric methods are often referred to as *smoothing* and the result of such smoothing can be seen visually in two or three dimensions as a smooth curve or surface. More specifically, as detailed in Ramsay and Silverman (2005), for a function to be smooth it should have one or more existing derivatives. There are several different nonparametric regression techniques, which can largely be split into the categories of spline-based and local methods. Smoothing splines, dating from Whittaker (1923), and P-splines (Eilers and Marx, 1996) are both examples of spline-based methods. A large part of this thesis is devoted to a local method, local polynomial regression, which in its current form dates from Stone (1977).

Univariate nonparametric regression is widely discussed and used so this will not be elaborated on extensively here. Instead this thesis focuses on

multivariate nonparametric regression methods which are not so prevalent, although several techniques do exist such as the additive models of Hastie and Tibshirani (1990) and thin plate splines, introduced by Duchon (1977). Here the multivariate case of local *linear* regression, a particular form of local polynomial regression, is mainly examined. This multivariate technique has often been deemed impractical due to the problems encountered in regions of sparse data, which become practically an unavoidable part of data in higher dimensions. This issue is often referred to as the *curse of dimensionality*. However, multivariate local regression has been implemented successfully in Cleveland and Devlin (1988), through LOESS in two and three-dimensional data and in Fowlkes (1987) for data of even higher dimensions. Fowlkes (1987) achieves this in the context of the evaluation of the fit of binary logistic models.

The main focus of this thesis is to examine the curse of dimensionality in the context of local linear regression and introduce techniques which make it avoidable, and so regression feasible, for data of any reasonable dimension. Additionally, modal regression in the multivariate setting is introduced in Chapter 4 as an alternative nonparametric regression technique and in Chapter 5 a bandwidth selection tool for univariate kernel density estimation on multimodal data is developed. These additional topics are introduced individually in the relevant chapters while this chapter serves to introduce the local linear ideas. Here the basic methodology will be explained in detail and the curse of dimensionality will be explored further. An overview of competing techniques and software is also provided.

1.1 Multivariate local linear regression

Given d-dimensional covariates $\mathbf{X}_i = (X_{i1}, ..., X_{id})^T$ with density $f(\cdot)$ and scalar response values Y_i where $i = 1, ..., n$, the task is to estimate the mean function $m(.) = E(Y|X = .)$ at a vector $\mathbf{x} = (x_1, ..., x_d)$. Assumed is that

$$Y_i = m(\mathbf{X}_i) + \epsilon_i \tag{1.1}$$

where ϵ_i are random variables with zero mean and variance σ_ϵ^2. Local linear regression uses a kernel-weighted version of least squares, in order to fit hyperplanes of the form $\beta_0 + \boldsymbol{\beta}_1^T \mathbf{x}$ *locally*, i.e., at each target point $\mathbf{x} \in \mathbb{R}^d$. Both the scalar β_0 and the vector $\boldsymbol{\beta}_1$ depend on $\mathbf{x}$, but this dependence is suppressed for notational ease.

Taylor's Theorem is crucial in constructing this least squares problem. The multivariate version of Taylor's Theorem is used frequently in this thesis, particularly in Chapter 2, and it is given, in full, in the appendix. Using this, m can be expressed around a data point $\mathbf{x}_0 = (x_{01}, ..., x_{0d})$ as

$$m(\mathbf{x}) \approx m(\mathbf{x}_0) + \frac{\partial m(\mathbf{x}_0)}{\partial x_1}(x_1 - x_{01}) + ... + \frac{\partial m(\mathbf{x}_0)}{\partial x_d}(x_d - x_{0d}) \tag{1.2}$$

$$= \beta_0 + \beta_{11}(x_1 - x_{01}) + ... + \beta_{1d}(x_d - x_{0d}). \tag{1.3}$$

In local linear regression this is used to find the regression estimate, $\hat{m}(\mathbf{x})$, by minimizing with respect to $\boldsymbol{\beta} = (\beta_0, \boldsymbol{\beta}_1^T)^T = (\beta_0, \beta_{11}, ..., \beta_{1d})^T$;

$$\sum_{i=1}^{n} \left\{ Y_i - \beta_0 - \sum_{j=1}^{d} \beta_{1j}(X_{ij} - x_j) \right\}^2 K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x}). \tag{1.4}$$

The estimator of the mean function $\hat{m}(\mathbf{x})$ is $\hat{\beta}_0$. Here K is a multivariate kernel function with $\int K(\mathbf{u})d\mathbf{u} = 1$ and

$$K_{\mathbf{H}}(\mathbf{x}) = |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2}\mathbf{x}). \tag{1.5}$$

The $d \times d$ matrix $\mathbf{H}$ is known as the *bandwidth matrix* and must be selected.

Minimization (1.4) is a weighted least squares problem. The solution to this can be expressed as

$$\hat{\beta}_0 = \hat{m}(\mathbf{x}) = \mathbf{e}_1{}^T (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{Y} \tag{1.6}$$

where

$$\mathbf{X} = \begin{bmatrix} 1 & X_{11} - x_1 & ... & X_{1d} - x_d \\ 1 & X_{21} - x_1 & ... & X_{2d} - x_d \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} - x_1 & ... & X_{nd} - x_d \end{bmatrix} \tag{1.7}$$

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix} \tag{1.8}$$

$$\mathbf{W} = \text{diag}\left\{K_{\mathbf{H}}(\mathbf{X}_1 - \mathbf{x}), ..., K_{\mathbf{H}}(\mathbf{X}_n - \mathbf{x})\right\} \tag{1.9}$$

and $\mathbf{e}_1$ is a vector with 1 as its first entry and 0 in the other d entries. To clarify, in this thesis, univariate data refers to data with one predictor variable and one response variable, bivariate data refers to data with two predictor variables and one response variable and trivariate data refers to data with three predictor variables and one response variable.

Local polynomial regression in general, and local linear regression in particular, has many advantages which makes it of interest to find a solution to the problem of the curse of dimensionality. Firstly, the idea has great intuitive appeal, as it is easily visualized and understood which data points are contributing to the estimation at a point. Furthermore, kernels are attractive from a theoretical point of view, since they allow straightforward asymptotic analysis. It has been found that the technique exhibits excellent theoretical properties. Local polynomials were shown to achieve optimal rates of convergence in Stone (1980). In the univariate case, Fan (1993) showed that local linear regression attains 100% minimax efficiency. The asymptotic bias and variance are known to have the same order of magnitude at the boundary as in the interior of the data, which is particularly useful for higher dimensional data sets (Ruppert and Wand, 1994). Work by Cleveland and Devlin (1988) and Hastie and Loader (1993a) also suggests that multivariate local polynomial regression is favourable in terms of computational speed. Other advantages, as detailed in Hastie and Loader (1993b), include that it adapts easily to different data design and also has the interesting side-effect of implicitly providing the gradient of $\hat{m}$ at $\mathbf{x}$ through the same least squares calculation. Indeed, this is given by $\hat{\boldsymbol{\beta}}_1$.

The estimated gradient at a point $\mathbf{x}_0$ is

$$\nabla\hat{m}(\mathbf{x}_0) = \begin{bmatrix} \frac{\partial\hat{m}(\mathbf{x}_0)}{\partial x_1} \\ \vdots \\ \frac{\partial\hat{m}(\mathbf{x}_0)}{\partial x_d} \end{bmatrix} = \begin{bmatrix} \hat{\beta}_{11}(\mathbf{x}_0) \\ \vdots \\ \hat{\beta}_{1d}(\mathbf{x}_0) \end{bmatrix}$$

which is the direction of maximum slope at that point. Newell and Einbeck (2007) describe the applications of this in the univariate setting, in areas such as the analysis of growth curves and change point problems. It is also useful in expressions for the bias, variance, confidence intervals and some bandwidth selection. There is also the possibility of producing derivatives of a higher order, but in order to do this it is necessary to fit a higher degree of polynomial locally. Fan and Gijbels (1996) suggest that in order to calculate the j-th derivative, the use of a local polynomial of degree $j+1$ is optimal.

It is also easy to examine *directional* derivatives, using the by-product of the regression, $\hat{\boldsymbol{\beta}}_1$. These represent the derivative of the regression estimate at a point in any chosen direction, defined as a vector $\mathbf{u}$. The directional derivative is then calculated as

$$\frac{\nabla \hat{m}.\mathbf{u}}{|\mathbf{u}|}. \tag{1.10}$$

Such quantities could be useful in the analysis of the gradient of a function over a surface, such as the variation in a climate variable in a particular direction over a region/country/continent.

1.1.1 Bias and variance

Two quantities which are referred to frequently in this thesis are the bias and variance of the estimator $\hat{m}$, and so it is useful to define them here. The bias, $E(\hat{m}) - m$, measures the difference between the true function and the regression estimate. The variance measures the amount that $\hat{m}$ depends on the one data sample used to generate it. For multivariate local linear regression, these are quantified in Ruppert and Wand (1994) as

$$\text{Bias}(\hat{m}(\mathbf{x})) = \frac{1}{2}\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\left\{\mathbf{Q}_m(\mathbf{x}) + \mathbf{R}_m(\mathbf{x})\right\} \tag{1.11}$$

where $\mathbf{R}_m(\mathbf{x})$ is a vector of Taylor series remainder terms and

$$\mathbf{Q}_m(\mathbf{x}) = \left[(\mathbf{X}_1 - \mathbf{x})^T\mathcal{H}_m(\mathbf{x})(\mathbf{X}_1 - \mathbf{x}), ..., (\mathbf{X}_n - \mathbf{x})^T\mathcal{H}_m(\mathbf{x})(\mathbf{X}_n - \mathbf{x})\right]^T$$

where $\mathcal{H}_m(\mathbf{x})$ is the $d \times d$ Hessian matrix of m. The variance can be derived immediately from (1.6) as

$$\text{Var}(\hat{m}(\mathbf{x})) = \mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{V}\mathbf{W}\mathbf{X}(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{e}_1 \tag{1.12}$$

where $\mathbf{V}$ is a diagonal matrix with the error variance, σ_ϵ^2, in each entry.

The asymptotic approximations ($\mathbf{H} \to 0$, $n\mathbf{H} \to \infty$ as $n \to \infty$) of the bias and variance will also be discussed in this thesis and they are, as given in Ruppert and Wand (1994),

$$\text{Bias}\left\{\hat{m}(\mathbf{x})\right\} = \frac{1}{2}\mu_2(K)\text{trace}\left\{\mathbf{H}\mathcal{H}_m(\mathbf{x})\right\} + o_p\left\{\text{trace}(\mathbf{H})\right\} \tag{1.13}$$

where $\mu_2(K)\mathbf{I} = \int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}$.

$$\text{Variance}\left\{\hat{m}(\mathbf{x})\right\} = \frac{1}{n}|\mathbf{H}|^{-\frac{1}{2}} \int K(\mathbf{u})^2 d\mathbf{u} \frac{\sigma_\epsilon^2}{f(\mathbf{x})} \left\{1 + o_p(\mathbf{1})\right\}. \tag{1.14}$$

The mean squared error (MSE) and the mean integrated squared error (MISE) are measures of error used throughout this thesis. They are useful since they can be expressed in terms of bias and variance.

$$MSE(\hat{m}(\mathbf{x})) = E[(\hat{m}(\mathbf{x}) - m(\mathbf{x}))^2] \tag{1.15}$$

which can also be expressed as

$$MSE(\hat{m}(\mathbf{x})) = \left[\text{Bias}\left\{\hat{m}(\mathbf{x})\right\}\right]^2 + \text{Variance}\left\{\hat{m}(\mathbf{x})\right\} \tag{1.16}$$

The MISE is the global extension of the local MSE.

$$MISE(\hat{m}(\mathbf{x})) = \int \left\{\left[\text{Bias}\left\{\hat{m}(\mathbf{x})\right\}\right]^2 + \text{Variance}\left\{\hat{m}(\mathbf{x})\right\}\right\} d\mathbf{x}. \tag{1.17}$$

Using (1.13) and (1.14), the asymptotic MISE can be expressed as

$$\begin{aligned} AMISE(\mathbf{H}) = \int \Bigg(&(\frac{1}{2}\mu_2(K)\text{trace}\left\{\mathbf{H}\mathcal{H}_m(\mathbf{x})\right\})^2 + \frac{1}{n}|\mathbf{H}|^{-\frac{1}{2}} \int K(\mathbf{u})^2 d\mathbf{u} \frac{\sigma_\epsilon^2}{f(\mathbf{x})} \\ &+ o_p\left\{n^{-1}|\mathbf{H}|^{-\frac{1}{2}} + \text{trace}^2(\mathbf{H})\right\}\Bigg) d\mathbf{x} \end{aligned} \tag{1.18}$$

1.1.2 The univariate case ($d = 1$)

The expressions (1.4) and (1.6) are provided for local *linear* regression since this is the type of polynomial regression implemented in the multivariate case in this thesis. However, at times here, local *constant* (polynomial of degree zero) as well as local linear regression (polynomial of degree one) is used in univariate examples. For this reason, the basic methodology of the general case of univariate local polynomial regression is set out below. To estimate at x_0 with n observations (X_i, Y_i), analogously to (1.4), one minimizes with respect to $\beta_0, ...\beta_p$,

$$\sum_{i=1}^{n} \{Y_i - \beta_0 - \beta_1(X_i - x_0) - ... - \beta_p(X_i - x_0)^p\}^2 \kappa\left(\frac{X_i - x_0}{h}\right)$$

where h is a univariate bandwidth, κ is a univariate kernel function, and p is the degree of the polynomial. Practically, this is carried out using the same least squares equation (1.6) and again $\hat{m}(x_0) = \hat{\beta}_0$. In the local linear case $\mathbf{X}$ is as in (1.7) with $d = 1$, and for $p = 0$, $\mathbf{X}$ is an $n \times 1$ vector with 1 in each entry. By implementing (1.6), it is trivial to show that for $p = 0$

$$\hat{m}(x_0) = \frac{\sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right) Y_i}{\sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right)}, \tag{1.19}$$

which is also known as the Nadaraya-Watson estimator. For $p = 1$,

$$\mathbf{X}^T\mathbf{W}\mathbf{X} = \begin{bmatrix} \sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right) & \sum_{i=1}^{n} (X_i - x_0)\kappa\left(\frac{X_i - x_0}{h}\right) \\ \sum_{i=1}^{n} (X_i - x_0)\kappa\left(\frac{X_i - x_0}{h}\right) & \sum_{i=1}^{n} (X_i - x_0)^2\kappa\left(\frac{X_i - x_0}{h}\right) \end{bmatrix} \tag{1.20}$$

and

$$\mathbf{X}^T\mathbf{W}\mathbf{Y} = \begin{bmatrix} \sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right) Y_i \\ \sum_{i=1}^{n} (X_i - x_0)\kappa\left(\frac{X_i - x_0}{h}\right) Y_i \end{bmatrix} \tag{1.21}$$

thus the univariate local linear estimator can be expressed as

$$\hat{m}(x_0) = \frac{\sum_{i=1}^{n} Y_i \kappa\left(\frac{X_i - x_0}{h}\right) \{S_{n,2} - (X_i - x_0)S_{n,1}\}}{\sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right) \{S_{n,2} - (X_i - x_0)S_{n,1}\}} \tag{1.22}$$

where

$$S_{n,j} = \sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right) (X_i - x_0)^j \tag{1.23}$$

(notation used here is as in Fan and Gijbels (1996)). There also exist asymptotic approximations to the bias and variance in the univariate case, which will be referred to later. The asymptotic variance for both local constant and local linear regression is

$$\text{Variance}\{\hat{m}(x)\} = \frac{\int \kappa^2(u)du\sigma_\epsilon^2}{nhf(x)} + o_p[(nh)^{-1}]. \tag{1.24}$$

For univariate local linear regression, analogously to the multivariate case (1.13)

$$\text{Bias}\{\hat{m}(x)\} = \frac{1}{2}h^2 m''(x)\mu_2(\kappa) + o_p(h^2). \tag{1.25}$$

In the local constant case,

$$\text{Bias}\{\hat{m}(x)\} = h^2\left[\frac{m'(x)f'(x)}{f(x)} + \frac{m''(x)}{2}\right]\mu_2(\kappa) + o_p(h^2). \tag{1.26}$$

Here, $\mu_2(\kappa) = \int u^2\kappa(u)du$. These univariate expressions are taken from Simonoff (1996).

1.2 Different ways of representing $\hat{m}(\mathbf{x})$ in multivariate local regression

The local linear regression estimate at $\mathbf{x}$ is expressed in (1.6) as the solution to a least squares problem. This solution takes the same shape for local constant regression, but with $\mathbf{X}$ replaced by an $n \times 1$ vector with 1 in each entry. In either case, the estimator is a *linear smoother* which means that the vector of fitted values $\hat{\mathbf{Y}}$, can also be expressed in the form

$$\hat{\mathbf{Y}} = \mathbf{S}\mathbf{Y}, \tag{1.27}$$

where $\mathbf{S}$ is known as the *smoother matrix*. This expresses the regression estimate as a weighted sum of the Y_i and this leads to an alternative way of representing the local estimator–as a quotient of summations i.e.

$$\hat{m}(\mathbf{x}) = \frac{\sum_{i=1}^n w_i Y_i}{\sum_{i=1}^n w_i}. \tag{1.28}$$

Examples of this in the univariate case are given in (1.19) and (1.22) and in the multivariate case the Nadaraya-Watson estimator is usually expressed

as

$$\hat{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij}-x_j}{h_j}\right) Y_i}{\sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij}-x_j}{h_j}\right)}. \tag{1.29}$$

Local linear regression can also be expressed in the form (1.28) in higher dimensions. The equivalent expressions for both bivariate and trivariate data are set out in the appendix (A.2).

1.2.1 Using Cramer's rule

There is a further alternative method for calculating $\hat{m}(\mathbf{x})$ which has advantages when compared with those stated above. The following is based on Cramer's rule and is not used widely in the smoothing community.

$$\hat{m}(\mathbf{x}) = \frac{\det(\mathbf{X}^T\mathbf{W}\mathbf{R})}{\det(\mathbf{X}^T\mathbf{W}\mathbf{X})} \tag{1.30}$$

where

$$\mathbf{R} = \begin{bmatrix} Y_1 & X_{11} - x_1 & \dots & X_{1d} - x_d \\ Y_2 & X_{21} - x_1 & \dots & X_{2d} - x_d \\ \vdots & \vdots & \ddots & \vdots \\ Y_n & X_{n1} - x_1 & \dots & X_{nd} - x_d \end{bmatrix}.$$

The proof of this result is provided in the appendix (A.3).

In addition, $\det(\mathbf{A}^T) = \det(\mathbf{A})$, as specified in Petersen and Pedersen (2008), which can lead to variations in the right hand side of (1.30).

When generalized, the above result is actually the same as Cramer's rule, but it was found and derived independently of this. The proof therefore generalizes to a proof of Cramer's rule. It is also the case that since Cramer's rule solves linear systems such as least squares, similar results can be used to calculate any element of $\hat{\boldsymbol{\beta}}$, providing, for example, the gradient at a point. This method is also applicable for local regression using any degree of polynomial, but makes no sense for local constant regression.

In the literature there does not appear to be any mention of Cramer's rule being used computationally for local polynomial regression, but it is used theoretically in this context in at least two papers, Delaigle and Meister (2011) and Horng (2004). This is undoubtedly because there are more

efficient ways of solving least squares computationally such as Cholesky factorization or QR decomposition. Cramer's rule is generally used as a theoretical result in wider mathematics and computationally it is often ignored due to its inefficiency in comparison with other methods. In R functions designed to perform local polynomial regression, such as `locpoly` in the **KernSmooth** package by Wand and Ripley (2010), algorithms are used which are significantly faster than calculating determinants.

However, there are merits to using this method computationally when compared with other basic ways that a novice might implement multivariate local polynomial regression. Implementation of (1.30) turns out to be more stable in practice than the *textbook* local polynomial regression method, least squares, as well as being significantly faster. For example, for data set E, described in Chapter 2, method (1.30) is seven times faster. Expressing $\hat{m}(\mathbf{x})$ as a quotient of sums (1.28) is practical only in small dimensions. This cannot be expressed for d dimensions without again including inverses and the problems involved with their calculation. So, in summary, (1.30) is attractive for use in local polynomial regression, when compared with (1.6) and (1.28), due to both its simplicity and its speed. However, it is less useful for the general linear model since this does not suffer from the issues associated with the calculation of inverses of matrices.

1.3 Choices

The mean function can be well approximated using the techniques described in section 1.1. However, before applying these some choices must be made, in particular a multivariate kernel function and bandwidth matrix must be selected. Another choice when looking at local polynomial regression more generally is the degree of polynomial to fit. Properties of multivariate local *quadratic* regression (degree of polynomial two) are provided in Ruppert and Wand (1994), and in the examples in Cleveland and Devlin (1988) the best fits are often produced using this type of regression. However, in this thesis use is restricted primarily to degree one polynomials, since, as mentioned in Fan and Gijbels (1996), these have been shown to give the best compromise

between bias and variance, in particular at the boundaries, whilst keeping computational costs reasonable. Clearly, a natural aim is to keep both the bias and variance as small as possible and the success of the local linear and constant estimators in achieving this is investigated in the simulation study below.

1.3.1 Local linear v. local constant regression

Fan (1992) and others compare in depth local linear with local constant regression. As expressed in Fan and Gijbels (1996), asymptotically, and in the interior of the data, local linear and local constant regression have variance of the same magnitude but local constant regression suffers from high bias (see results (1.24)-(1.26)). Furthermore, at the boundary local linear regression automatically adapts and so has the same levels of bias and variance as in the interior, whereas the local constant case suffers from a further increase in bias. Ruppert and Wand (1994) show that the same occurs in higher dimensions. This theory applies asymptotically but the small simulation study presented below gives an insight into what occurs in practice with a finite sample size.

Simulation study

The aim of this study was to analyse the variance and mean squared error (MSE) of local constant and local linear regression and observe how these vary over the boundary and interior of a data set. Four different data sets, of size $n = 150$, with bivariate covariates, were each simulated 200 times. For each data set the covariates were sampled from the uniform distribution between 0 and 1 and the response was simulated from the bivariate normal function with parameters (μ, σ) and error variance ϵ_i. Table 1.1 summarizes these details for each data set and Fig. 1.1 displays these underlying regression functions. The top plot shows the function used in simulations 1 and 2 and the bottom plot in simulations 3 and 4.

These functions were chosen to represent data with a variety of both curvature at the boundary (in simulations 3 and 4 the response is almost

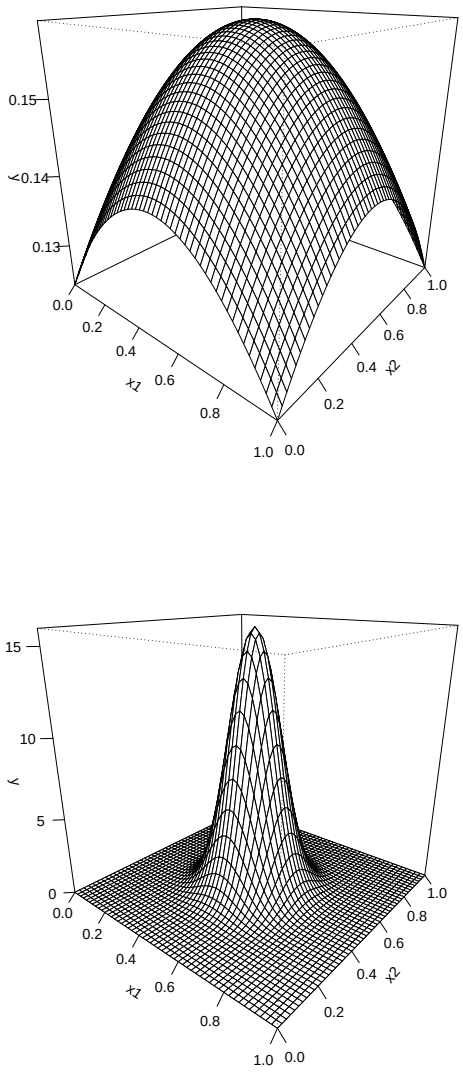

Figure 1.1: The underlying functions used to generate the data in simulations 1-4. The top plot shows the function used in simulations 1 and 2 and the bottom plot in simulations 3 and 4.

Simulation	μ	σ	$\epsilon_i \sim$
1	(0.5,0.5)	(1,1)	N(0,0.005)
2	(0.5,0.5)	(1,1)	N(0,0.001)
3	(0.5,0.5)	(0.1,0.1)	N(0,0.005)
4	(0.5,0.5)	(0.1,0.1)	N(0,0.001)

Table 1.1: The parameters and error variance used to simulate the response (simulated from the bivariate normal function with parameters (μ, σ) and error variance ϵ_i) in simulations 1-4.

constant at the boundary) and error variance. Local constant and local linear regression were carried out on each of the 800 data sets described above. The bandwidth selection method used was $AGCV$, a technique developed in Chapter 3. In each data set, each of the 150 points was then classified as either a boundary or interior point. In this study the boundary was defined as the region within one bandwidth of the edge of the data (a more detailed definition is given in Chapter 2). Then, in each of these two classes, and for each method of regression, the following two quantities were calculated,

$$\text{Average variance} = \frac{1}{n_c} \sum_{i_c=1}^{n_c} \text{Var}(\hat{m}(\mathbf{X}_{i_c})) \tag{1.31}$$

and

$$MSE = \frac{1}{n_c} \sum_{i_c=1}^{n_c} [\hat{m}(\mathbf{X}_{i_c}) - m(\mathbf{X}_{i_c})]^2, \tag{1.32}$$

where $\mathbf{X}_{i_c}$ are data in that class and n_c is the number of data points in that class. Here, $\hat{m}$ is calculated using (1.6) in the local linear case, and the equivalent least squares formulation in the local constant case. Expression (1.12) is employed in order to calculate the variance in (1.31).

When calculated, (1.31) and (1.32) nicely quantify the variance and mean squared error of the estimators in each of these regions. These quantities are displayed in box plots in Figs. 1.2-1.5. Each box plot represents these variance and MSE values for the 200 simulations of one function. In the plot labels, *In* represents the points in the interior, *Bo* the points in the boundary, *LL* local linear regression and *LC* local constant.

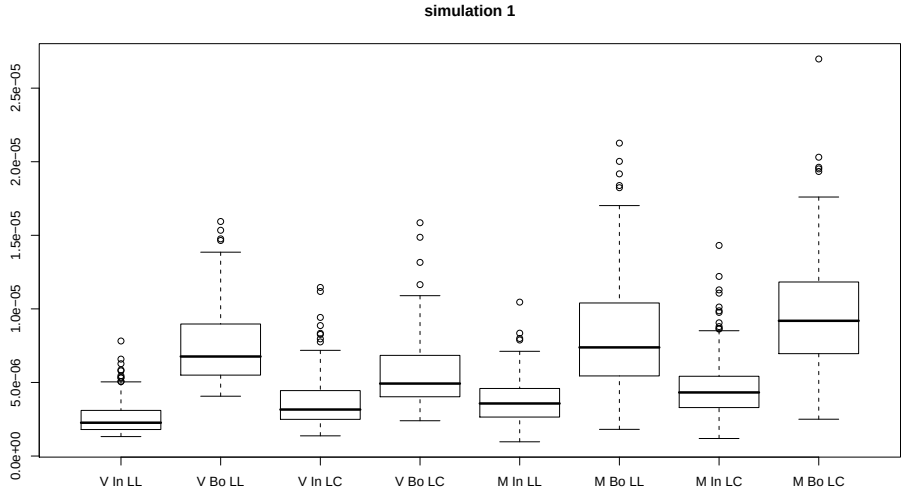

Figure 1.2: Average variance and MSE for simulation 1. V–variance, M–MSE, In–interior, Bo–boundary, LL–local linear, LC–local constant.

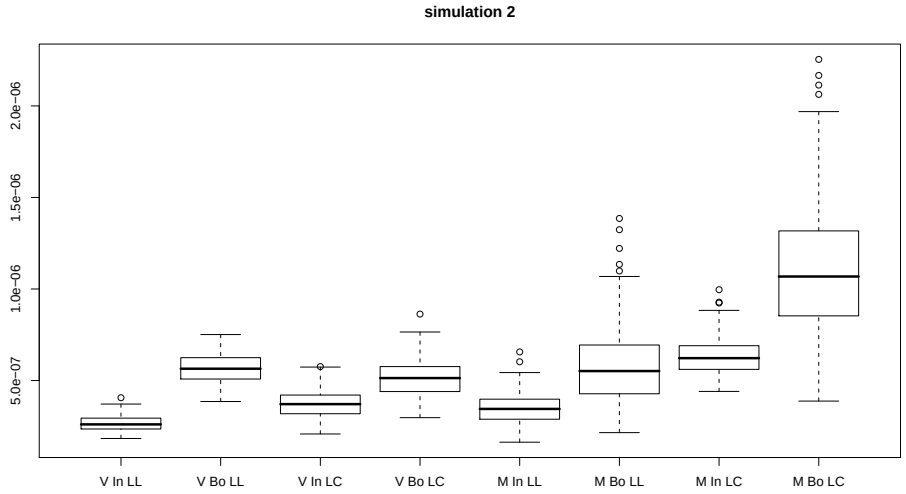

Figure 1.3: Average variance and MSE for simulation 2. V–variance, M–MSE, In–interior, Bo–boundary, LL–local linear, LC–local constant.

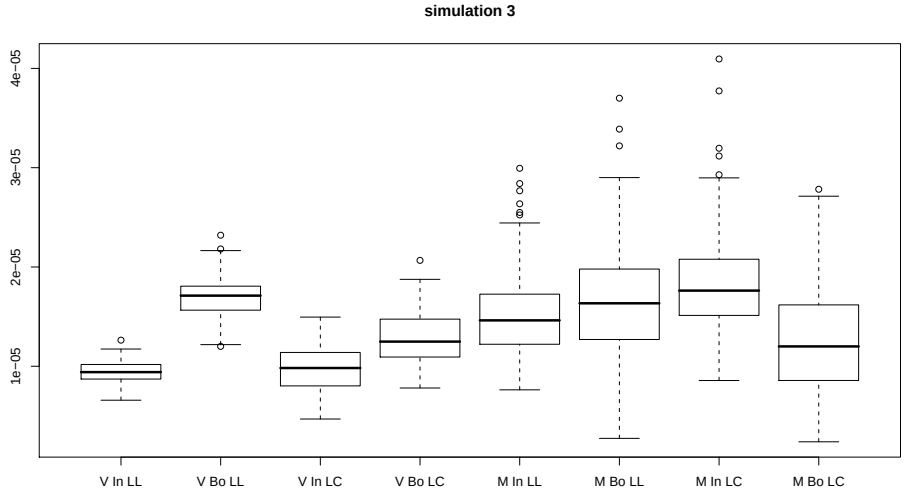

Figure 1.4: Average variance and MSE for simulation 3. V–variance, M–MSE, In–interior, Bo–boundary, LL–local linear, LC–local constant.

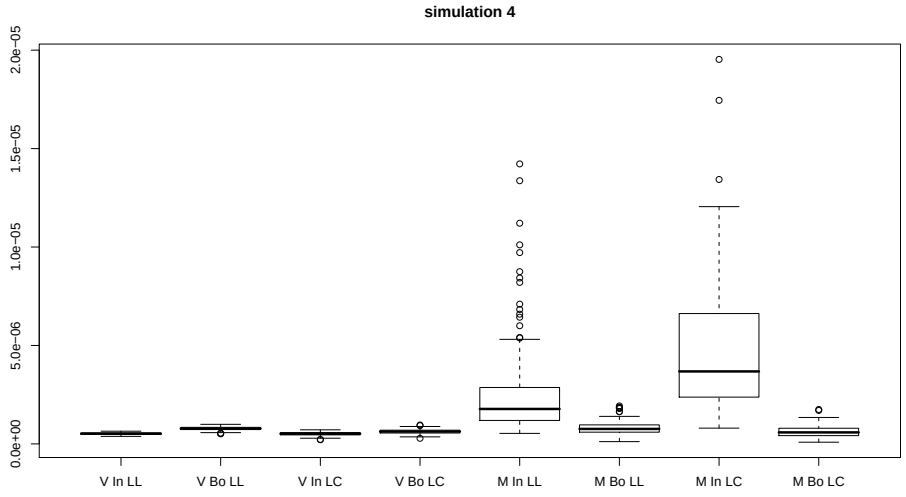

Figure 1.5: Average variance and MSE for simulation 4. V–variance, M–MSE, In–interior, Bo–boundary, LL–local linear, LC–local constant.

These plots show that throughout the simulations, in the interior, the variance and the MSE are lower for local linear regression. This agrees with the asymptotic theory (see (1.24)-(1.26) for the univariate case), and the poor performance of the local constant regression in this area will be mostly due to high bias. The results at the boundary are more interesting and reveal that the variance there is higher for local linear than local constant regression. However, on average, the MSE is still better at the boundary for local linear regression, except when the true function in that region is constant (simulations 3 and 4), in which case the advantage this gives to local constant regression leads to a smaller MSE. In the simulations in which the variance is larger and the MSE smaller for local linear regression (simulations 1 and 2), this can be attributed to this estimator adapting to the boundary and thus suffering significantly less bias than local constant regression does here.

As is also observed in this study, Fan and Gijbels (1992) note that for local linear regression the variance is higher, in practice, at the boundary than in the interior, and they attribute this to the fact that "*less observations contribute in computing the estimator.*" This is particularly likely to be the case if the bandwidths are kept relatively small, as is the case with AGCV, rather than, as some practitioners propose, using very large bandwidths as a remedy for the curse of dimensionality. Ruppert and Wand (1994) also stress that for finite samples, in certain situations, local constant regression can be "*considerately more accurate*" near the boundary. They claim this is due to the high variance and nonorthogonality of the regression parameters in local linear regression.

The fact that, in the simulations in which the response is not constant at the boundary (1 and 2) the MSE is on average lower for local linear regression, indicates that this method offers the best compromise of bias and variance and explains why it is preferred by many in the literature and in this thesis. However, there are individual simulations where the variance that this estimator gains at the boundary adds more to the error in the estimation there than any boundary bias endured by the local constant estimator. For

this reason, in contradiction to the asymptotic theory, but supported by Ruppert and Wand (1994), there may be data sets where local constant regression is more suitable at the boundary. In examples such as this it may be useful to apply data adaptive ridging (Seifert and Gasser, 2000) as a compromise between local linear and local constant regression. Chu and Marron (1991) also emphasize the importance of choosing an estimator based on the data set.

For these reasons, if the threshold developed in Chapter 2, (2.18), which, as will be shown, works by placing an upper bound indirectly on the variance, rejects local linear regression at $\mathbf{x}$, it could be that the threshold (in its local constant form), accepts local constant regression as an alternative method.

1.3.2 Kernel function

Kernel functions are crucial to the idea of local polynomial regression, and contribute by assigning weight to the data points. In general, a kernel function acting at $\mathbf{x}$ assigns more importance, and so weight, to a data point closer to $\mathbf{x}$ than to those further from it. There are several different types of kernel function, but in this thesis either the Gaussian or the Epanechnikov kernel is used. The Gaussian kernel is particularly useful in higher dimensions since it assigns weights to points further from the point of interest, and although these weights are small they reduce the chance of computational instability occurring in the estimation. This makes it more suitable for the analysis of regions of sparse data. Hastie and Tibshirani (1990) state that, at least in the univariate case, the choice of kernel is not important regarding the quality of the regression, which means that significant accuracy is not sacrificed by using a Gaussian kernel. However, in the literature, the Epanechnikov kernel is often regarded as optimal, and so this is used at times when data sparsity is not considered to be an issue. Fan and Gijbels (1996) state that this kernel is optimal in terms of minimizing the asymptotic MSE and MISE for a point in the interior, and that it is fast computationally.

In the univariate case the kernel functions are as follows. The Gaussian

kernel is

$$\kappa(t) = \frac{\exp(-t^2/2)}{\sqrt{2\pi}}, \tag{1.33}$$

and the Epanechnikov kernel is

$$\kappa(t) = \frac{3(1-t^2)}{4} \tag{1.34}$$

(for $|t| \leq 1$.)

These are then extended for use in the multivariate setting. The most common way of generating the K used in (1.5) from the univariate kernels is through a *product* kernel, and this is what is used here. This takes the form

$$K(\mathbf{x}) = \prod_{j=1}^{d} \kappa(x_j) \tag{1.35}$$

where κ is one of the univariate kernel functions (1.33) or (1.34).

Whilst the choice of degree of polynomial and kernel function do have an impact, the most important choice to make is that of bandwidth matrix, and this will be discussed in depth in the next section.

1.4 Bandwidth selection

The selection of the bandwidth is the most important choice you can make in smoothing, since it is this that effectively determines how smooth the resulting regression estimate is. In multivariate local linear regression, $\mathbf{H}$ is crucial in determining the amount and direction of smoothing since it determines the size of the *neighbourhood* in which the smoother acts at each point. The term *neighbourhood* describes the ellipsoid which encloses data points that are considered in the estimation at $\mathbf{x}$. As the elements of $\mathbf{H}$, h_{jk}, tend to 0, the regression estimate will follow the data very closely, and as the h_{jk} tend to infinity so the overall fit tends to a hyperplane of d dimensions. One searches for an estimate between these two extremes.

$\mathbf{H}$ is a symmetric, positive definite $d \times d$ matrix, and as a result there are $d(d+1)/2$ smoothing parameters to select. This can be simplified greatly by using a diagonal matrix of the form $\mathbf{H} = \text{diag}(h_1^2, ..., h_d^2)$ or made even

simpler by having just one smoothing parameter h and forming the matrix $h^2\mathbf{I}$. The latter of these options is often not useful since this results in the same amount of smoothing in each covariate direction. The diagonal matrix is an improvement since it allows this to vary. However, it does not have the flexibility of a full matrix which also allows smoothing in other directions, which, as Wand and Jones (1995) point out in the context of multivariate density estimation, can be very advantageous for some data sets. In this way there is a flexibility versus complexity trade-off when choosing the type of bandwidth matrix to employ. Clearly the more parameters one needs to choose, the more complex the selection becomes, but at the same time, the simpler bandwidth matrices perform less well, according to Chacón (2009), in terms of MISE, at least in the context of related kernel density estimation. Thus, a good compromise, supported by Yang and Tschernig (1999) and others, is the diagonal matrix. Chacón (2009) shows that this is sufficient in the majority of cases, by using relative efficiency to determine whether the gain in computational ease caused by using a simpler matrix is worth the increase in MISE for a data set. In the interest of computational simplicity a diagonal bandwidth matrix, of the form $\mathbf{H} = \text{diag}(h_1^2, ..., h_d^2)$, is used throughout this thesis.

Figs. 1.7-9 demonstrate the importance of the choice of bandwidth, using a data set of *US temperatures*. This bivariate data set is from the **SemiPar** package by Wand (2010), and consists of measurements of the average January minimum temperature in 56 US cities. The covariates are the latitude and longitude of the cities. Fig. 1.6 presents the data in the form suggested by Wand (2010). Here the higher temperatures are represented by lighter shades of grey.

Fig. 1.7 shows the local linear regression surface for this data using a bandwidth matrix of an appropriate magnitude ($(h_1, h_2) = (2.45, 3.53)$ selected using $AGCV$). Fig. 1.8 shows the undersmoothing which occurs when the bandwidth values used are too small ($(h_1, h_2) = (0.5, 1)$), and Fig. 1.9 shows the oversmoothing with large bandwidths ($(h_1, h_2) = (50, 50)$). It is clear from these figures how poor estimation can become when inappropriate

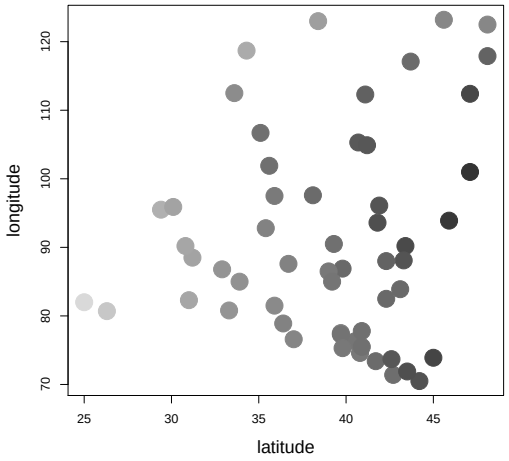

Figure 1.6: The *US temperature* data. The minimum temperatures range from 0 to 65 F. The higher temperatures are represented by the lighter shades of grey, and the lower temperatures by the darker shades.

bandwidth choices are made.

1.4.1 The bias variance trade-off

Ultimately, the treatment of bandwidth selection in this thesis concerns how one chooses the parameters $h_1^2, ..., h_d^2$. Most methods involve examining the bias and variance of $\hat{m}$, using different $\mathbf{H}$, since this provides an effective measure of the desirable features of an estimate. A *trade-off* occurs because as one of these quantities increases the other decreases. As the bandwidth parameters tend to infinity, the variance decreases, but unfortunately the bias increases, and the opposite occurs as the parameters tend to zero. This becomes apparent as one examines (1.13) and (1.14) since it is clear that by reducing the magnitude of the elements of the diagonal of $\mathbf{H}$, the bias will be reduced, but at the same time $\mathbf{H}$ also appears in the denominator in (1.14), and so a reduction in h_j will lead to an increase in variance. The issue therefore is how to choose these parameters in order to obtain a desirable

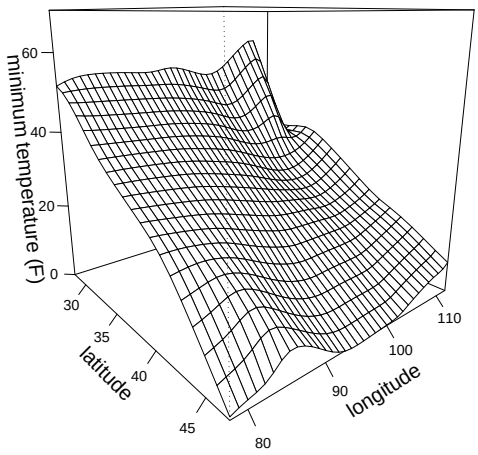

Figure 1.7: Local linear regression on *US temp.* data ($(h_1, h_2) = (2.5, 3.5)$).

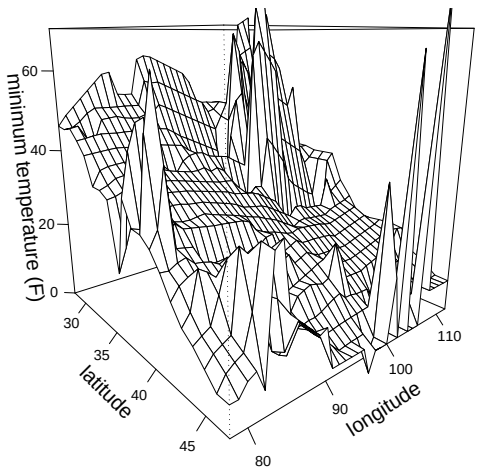

Figure 1.8: Local linear regression on *US temp.* data ($(h_1, h_2) = (0.5, 1)$).

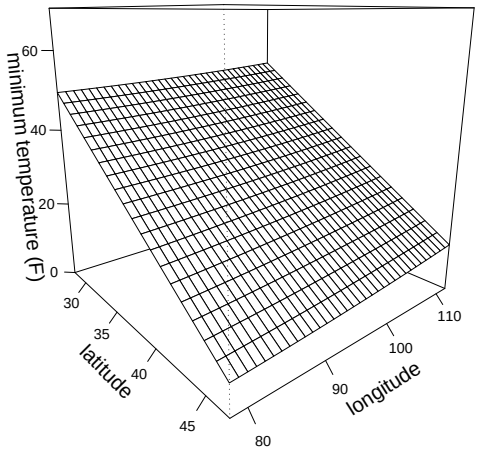

Figure 1.9: Local linear regression on *US temp.* data ($(h_1, h_2) = (50, 50)$).

balance of bias and variance, and so the best fit.

In local polynomial regression, there are two principal ways of using the bias and variance to select the bandwidth. The majority of methods use a global criterion such as the MISE, but others favour the power of graphical diagnostics. According to Cleveland and Loader (1996), by using diagnostics one is able to see how bias and variance vary throughout a data set, and so decide on where to prioritise each of them. This is the function of the M Plot described in Cleveland and Devlin (1988) which displays how the bias and variance compose the MSE for different bandwidth values. In this vein, Ramsay and Silverman (2005) highlight that in some instances trial and error is even considered as an acceptable bandwidth selection tool. Ruppert and Wand (1994) state that bias increases in areas of greater curvature and smoothing, and variance increases with a larger conditional variance of Y given $\mathbf{X} = \mathbf{x}$ and data sparsity. Therefore it might be useful for a bandwidth selection tool to select different bandwidths in different regions of the data to take these factors into account. One such method which achieves

this is the *empirical-bias bandwidth selector*, formulated in Ruppert (1997). Here, bias is estimated empirically, and an exact formula for the variance of a finite sample is employed, and as a result no asymptotics are required. Ruppert (1997) claims that a principal advantage of this method is that it automatically produces a larger bandwidth in areas of sparse data.

In nearest-neighbour smoothing, implemented in the LOESS procedure (Cleveland and Devlin, 1988) in the multivariate setting (a straight-forward extension of the univariate LOWESS of Cleveland (1979)), the bandwidth also adapts to different areas of the data. Here, a different approach is taken in which the estimation of $\hat{m}(\mathbf{x})$ uses the q nearest $\mathbf{X}_i$ values to $\mathbf{x}$. In this way the distance from $\mathbf{x}$ to the q-th nearest $\mathbf{X}_i$ is used locally as an effective bandwidth in the kernel function. The actual smoothing parameter is q, and this can be selected using an M plot. Despite the advantages of graphical diagnostics and varying bandwidths, Cleveland and Loader (1996) concede that the computational costs are often too great and so the use of an automatic selection method is often preferable. Furthermore, even Cleveland and Devlin (1988) acknowledge that minimizing a criterion is acceptable in terms of prediction. For these reasons, the efforts to find a suitable bandwidth selection tool in this thesis are confined to those which minimize the MISE.

1.4.2 Minimization of the MISE

Given the importance of bias and variance and the fact that it can be expressed in terms of these two quantities, it is sensible to attempt to minimize the MISE. Indeed this is what a large number of existing methods seek to do. The optimal choice for the bandwidth matrix would be found by differentiating (1.18), with respect to $\mathbf{H}$, and finding the minimum by equating it to zero. This is derived later, resulting in (3.17), for the case $h_1 = ... = h_d = h$. Fan and Gijbels (1996) do this for the MSE and so quantify the local optimal bandwidth. The problem with these optimal bandwidth matrices, and indeed the equivalents in univariate local polynomial regression (such as (2.22)), is that they contain unknown quantities, usually functionals of m or

f. A strategy is therefore needed to solve this. In the univariate setting there exist a large variety of such bandwidth selection strategies which broadly speaking fit into two categories; *classical* methods such as cross-validation and Mallow's Cp and *plug-in* methods, such as that in Ruppert, Sheather and Wand (1995), which aim to substitute these unknowns by some other quantity such as a local cubic. Fan and Gijbels (1996) also propose a rule of thumb which approximates m globally by a quartic polynomial. In the multivariate case there is comparatively little written, although there still exist a range of possible methods. Many classical methods can be simply extended (as is done with $AGCV$ in Chapter 3), and Yang and Tschernig (1999) use a form of local cubic regression to estimate the required unknown second derivative of m in the multivariate setting.

A further alternative approach, particularly useful for $d > 5$, is the implementation of variable selection and bandwidth selection simultaneously, which was suggested by Cleveland and Devlin (1988). More recently, Lafferty and Wasserman (2008) introduced the *rodeo* (regularization of derivative expectation operator). This initially assigns a large bandwidth in every covariate direction, before gradually decreasing those assigned to covariates which are considered relevant, until a threshold is reached. The bandwidths assigned to the irrelevant variables remain large and so effectively these variables are removed from the local regression problem. By reducing the dimension in this way the curse of dimensionality is more likely to be avoided. This technique is very useful for data with $d > 5$, in which some dimensions could be considered irrelevant, and variable selection in general should be considered as a first step with data of this type. For instance, it could be applied before $AGCV$, which, for $d > 5$, is too time-consuming. There is a large amount of literature on variable selection in nonparametric regression, in particular Vidaurre, Bielza and Larrañaga (2011) implement a lasso locally to reduce the number of variables in local regression.

A criticism of the rodeo, which is also made in Vidaurre, Bielza and Larrañaga (2011), is that it uses a strange definition of relevance of a variable – it depends on how linear the function is in that covariate direction. The

paradox is that local linear regression is to be used except when the true function is linear, and it is likely that poor results are attained sometimes as a result. Whilst suitable for data of large dimensions, due to this criticism and its *greedy* nature, other methods are likely to be more effective for data of $2 < d < 5$ of which all dimensions could be considered relevant.

1.4.3 Cross-validation

The classical methods work by selecting the $\mathbf{H}$ which minimizes an expression which is an approximation of a measure of error, such as the MISE. Classical methods could be more suitable for data of high dimensions since plug-in methods rely on asymptotics. The asymptotic assumption of bandwidths tending to zero seems to be inappropriate in order to select the relatively large bandwidths needed for multivariate local smoothing and this is the justification for focussing on cross-validation, first introduced in Wahba and Wald (1975), in this thesis. A further reason for using cross-validation is that it can be easily simplified computationally. In local linear regression, cross-validation is defined as

$$CV(\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n}\{Y_i - \hat{m}_{-i}(\mathbf{X}_i)\}^2 \tag{1.36}$$

where $\hat{m}_{-i}(\mathbf{X}_i)$ is the *leave-one-out* estimator. Härdle, Müller, Sperlich and Werwatz (2004) state that the minimization of (1.36) is the equivalent to the minimization of the MSE. A simplification of (1.36) is generalized cross-validation (GCV), developed by Craven and Wahba (1979), and this can be derived directly from (1.36). The following derivation is taken largely from Gentle, Härdle, and Mori (2004).

Expression (1.27) implies that

$$\hat{m}(\mathbf{X}_i) = \sum_{j=1}^{n} S_{ij}Y_j, \tag{1.37}$$

and

$$\hat{m}_{-i}(\mathbf{X}_i) = \sum_{j\neq i}^{n} \frac{S_{ij}Y_j}{1 - S_{ii}} \tag{1.38}$$

where S_{ij} are elements of the smoother matrix $\mathbf{S}$, and the weights $S_{ij}/1-S_{ii}$ are standardized to sum to 1. This can be re-written as

$$\hat{m}_{-i}(\mathbf{X}_i) = \sum_{j \neq i}^{n} S_{ij} Y_j + S_{ii} \hat{m}_{-i}(\mathbf{X}_i). \tag{1.39}$$

Using (1.37) and (1.39),

$$\hat{m}(\mathbf{X}_i) - \hat{m}_{-i}(\mathbf{X}_i) = S_{ii} \left\{ Y_i - \hat{m}_{-i}(\mathbf{X}_i) \right\}, \tag{1.40}$$

and thus

$$Y_i - \hat{m}_{-i}(\mathbf{X}_i) = \frac{Y_i - \hat{m}(\mathbf{X}_i)}{1 - S_{ii}}. \tag{1.41}$$

In this way, CV($\mathbf{H}$) can be rewritten as

$$CV(\mathbf{H}) = \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{Y_i - \hat{m}(\mathbf{X}_i)}{1 - S_{ii}} \right\}^2. \tag{1.42}$$

(1.42) is simply a re-arranged version of (1.36) and this becomes GCV when the S_{ii} values are replaced by their average value. The criterion then takes the form

$$GCV(\mathbf{H}) = \frac{1}{n} \sum_{i=1}^{n} \left\{ \frac{Y_i - \hat{m}(\mathbf{X}_i)}{1 - \frac{\text{trace}(\mathbf{S})}{n}} \right\}^2. \tag{1.43}$$

GCV suggests the bandwidth matrix $\mathbf{H} = \text{diag}(h_1^2, \ldots, h_d^2)$ which minimizes (1.43). Fahrmeir and Tutz (2001) highlight that since GCV is simply the averaged squared residual, corrected by a factor, $(1 - \frac{\text{trace}(\mathbf{S})}{n})^{-2}$, it is computationally less costly than CV.

$\mathbf{S}$ will be examined further, in terms of *influence*, in Chapter 2. In fact, in order to compute the GCV more easily and quickly, which should always be a consideration when extending procedures to higher dimensions, it is useful to compute the diagonal elements of $\mathbf{S}$ using the expression for the influence (2.2).

Gentle, Härdle, and Mori (2004) highlight that GCV is simply a weighted version of cross-validation, with weights $(1 - S_{ii})^2/(1 - \text{trace}(\mathbf{S})/n)^2$. It is this bandwidth matrix selector which is the focus of the efforts in Chapter 3. Here, some adaptations are made in order to combat the problems which bandwidth selection also encounters when faced with the curse of dimensionality.

1.5 The curse of dimensionality

Scott (1992) describes the curse of dimensionality as "*the apparent paradox of neighbourhoods in higher dimensions - if the neighbourhoods are 'local', then they are almost surely 'empty', whereas if a neighbourhood is not 'empty', then it is not 'local'.*" If there is not sufficient data in a neighbourhood, then the variance of the fit is too high, or with some kernel functions, such as the Epanechnikov kernel, the calculations break down completely. One solution to this is to increase the bandwidth parameters, but, as noted by Cornillon, Hengartner and Matzner-Løber (2011), this leads to an estimator with a large bias. Hastie, Tibshirani and Friedman (2001) note that another problem that occurs is that the majority of data points are closer to the boundary than to another point. This makes prediction more difficult since one must extrapolate from nearby data points rather than interpolating between points. Hastie, Tibshirani and Friedman (2001), Cleveland and Devlin (1988) and Fowlkes (1987) agree that the way to overcome these problems would be to increase n in order to capture complexities in the regression surface that might otherwise be lost through the necessary introduction of larger bandwidths. Of course, increasing n is often not a realistic option for a given data set, but, putting their statement in other words, there must be sufficient data around $\mathbf{x}$ for a reliable estimate to be made at that point. This is the attitude adopted in this thesis, and in Chapter 2 a solution is described which essentially identifies such "reliable" regions by dismissing all neighbourhoods which do not contain enough data. The actual smoothing step is then only performed over such regions in which estimation is considered reliable, where the bias and variance of $\hat{m}$ can be kept reasonably low. This is achieved through a threshold imposed on a suitable estimate of the density f. The threshold, (2.18), is developed from the asymptotic influence function and aims only to accept points at which the influence, and as a result the variance, is small.

1.5.1 Beyond the data range

In an effort to further understand the curse of dimensionality, a small study was carried out in order to identify approximately the region in which a threshold should advise against estimation outside of the centre of the data mass. Here, the behaviour of local polynomial regression in remote parts of the data range is examined.

In this study only the Gaussian kernel function is used since it returns estimates of some form, other than a computational error, at points further away from the data mass than other kernel functions. This allows a more complete understanding to be developed. Primarily, univariate local polynomial regression is examined here since it is simpler to use and evaluate in this exploratory setting. Here, local constant and local linear regression are both evaluated.

Local polynomial estimates were calculated and plotted for a grid of points which extends through the whole of the data range and then beyond it, up to a point far enough from the data that R returns *NaN* as a regression estimate. This was carried out on a number of real and simulated data sets. Among the real data sets was the *fossil* data set, illustrated in Fig. 1.10, from the **SemiPar** package of Wand (2010) on R. This contains 106 observations on the ages and ratios of strontium isotopes of fossil shells. This analysis was repeated for several different bandwidth values.

Figs. 1.11-12 illustrate the typical results of such analyses and show the possibilities that can occur beyond the data range. These both show the regression estimates for the *fossil* data set, for which the covariate data range is 91.79 to 123. Figs 1.11-12 examine estimates on a grid where age varies from 50 to 200. Fig. 1.11 shows local constant regression with $h = 1$, and Fig. 1.12 shows local linear regression with $h = 0.5$. These values of h are chosen purely to illustrate different possible eventualities of such an analysis, and are not necessarily the optimal ones.

Within the data range, between 91.79 and 123, the function is estimated reasonably, with the difference in smoothness between the two plots accounted for largely by the difference in h values. The estimates appear to

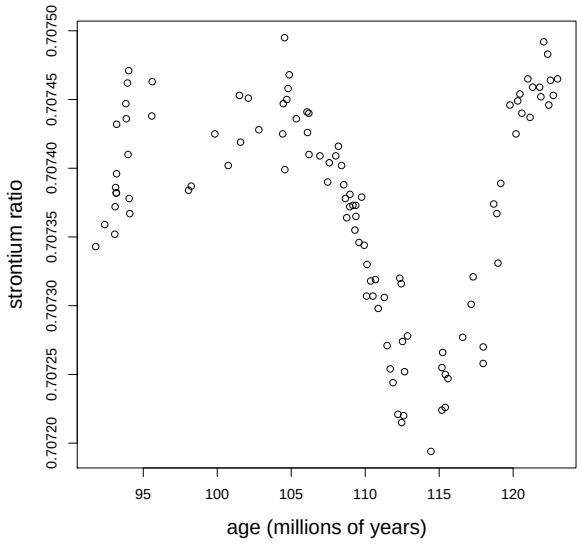

Figure 1.10: The *fossil* data set.

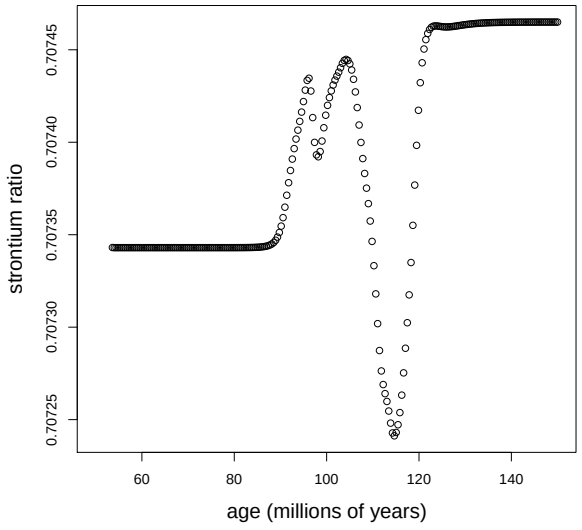

Figure 1.11: Local constant regression with $h = 1$ for the *fossil* data set.

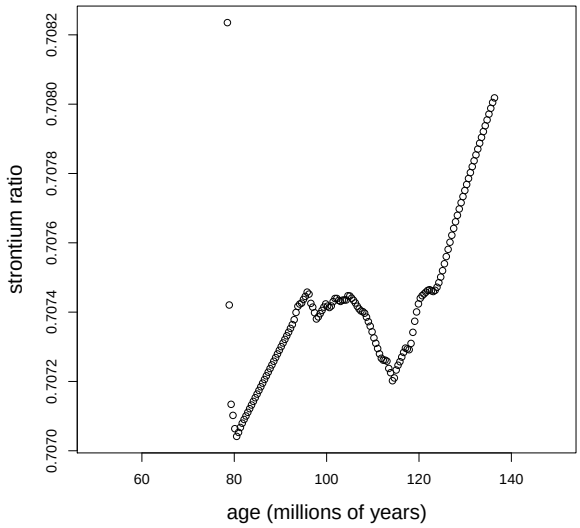

Figure 1.12: Local linear regression with $h = 0.5$ for the *fossil* data set.

be reasonable for a short distance outside of this data range in both cases. As one regresses further away from the actual data, in Fig. 1.11 the estimator eventually settles on a constant fit, before R starts to return *NaN*, which is not represented on the plot. Moving further from the data in Fig. 1.12, the fit becomes linear, before a small period of computational instability, visible at approximately age=80, before *NaN*s are also returned here.

The observations from these analyses are summarized in Fig. 1.13. This figure shows the possible different stages of behaviour that the estimate can exhibit and the order in which they can occur as you move away from the data, when reasonably sized bandwidths are used. Depending on the bandwidth and the degree of polynomial used, and the nature of the data, not all of these will occur every time.

There is always a period of *normal* estimation, in which it appears that the regression estimates are reasonable and the trend of the data from inside the data range is being continued. There is also always a point at which R starts to return *NaN*. This occurs because the numbers produced by the

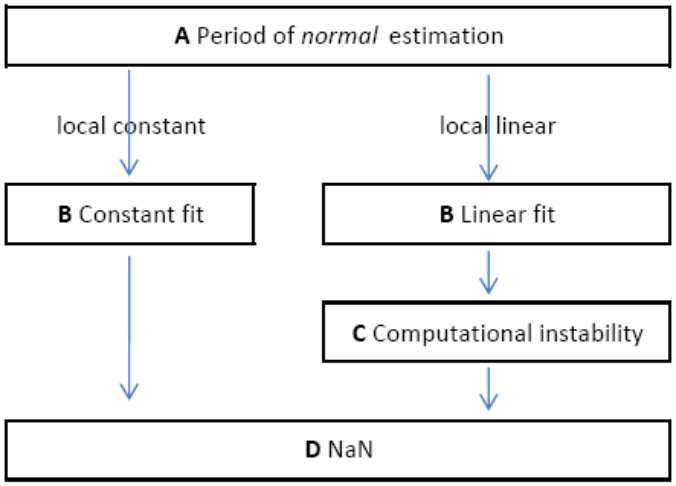

Figure 1.13: The different stages of behaviour an estimator can exhibit outside the data range. Behaviour varies from **A** to **D** as one moves further from the data.

kernel function become so small that R treats them as 0, and the division of 0 by 0 in the regression problem returns *NaN*. In the course of this study, some theory was developed on the intermediate stages, **B** and **C**. This is set out below.

The effect of the last data point(s)

In this sub-section it is assumed that the value of h used is of a reasonable or small magnitude, since not all of the following observations apply when larger bandwidths are implemented.

For these smaller bandwidths stage **B** occurs as one regresses further from the data points i.e. after the period of normal estimation, the estimates settle into a constant fit for local constant regression, or a linear fit for local linear regression. Throughout the analyses it was observed that these trends were dependent on the data points nearest to them i.e. the points on the very edge of the data. In fact, the constant estimate that is produced in the local constant fitting is usually the y-value of the nearest data point to the

closest edge of the data. Analogously in the local linear case, the gradient of the approximately linear fit observed is usually approximately equal to the gradient between the two data points nearest to that edge of the data. These data points do not have to be outlying for this to be observed. This is clearly a demonstration of an undesirably high level of variance, which is not surprising in an area of data sparsity.

This phenomenon can be explained theoretically for local constant regression, as follows. It should be highlighted that the following applies only for $Y_i>0$, and that the Gaussian kernel function, (1.33), is employed.

Define X_1 as the data point on the edge of the data range and X_2 as the second closest point to the edge. Also define x_0 as the point of interest, which is far outside of the data range. Say,

$X_1 - x_0 = ah$ where $a \in \mathbb{R}$.

$X_2 - x_0 = bh$ where $b \in \mathbb{R}$ and $b> a$.

Now, the local constant regression estimator is

$$\hat{m}(x_0) = \frac{\sum_{i=1}^{n} Y_i \kappa\left(\frac{X_i - x_0}{h}\right)}{\sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right)}$$

which, when incorporating the Gaussian kernel, $\kappa(t) = \frac{\exp(-t^2/2)}{\sqrt{2\pi}}$, and the above substitutions, can be equated to

$$\hat{m}(x_0) = \frac{Y_1 \exp(-a^2/2) + A}{\exp(-a^2/2) + B},$$

where $A< \sum_{i=2}^{n} Y_i \exp(-b^2/2)$ and $B<(n-1)\exp(-b^2/2)$. Equally,

$$\begin{aligned} \hat{m}(x_0) &= \frac{\exp(-a^2/2)(Y_1 + C)}{\exp(-a^2/2)(1 + D)} \\ &= \frac{Y_1 + C}{1 + D} \end{aligned} \tag{1.44}$$

where $C< \sum_{i=2}^{n} Y_i \exp((a^2 - b^2)/2)$ and $D<(n-1)\exp((a^2 - b^2)/2)$.

This demonstrates that if C and D are close to 0, then the regression estimate at x_0 will be equal to Y_1, the response value of the data point on the edge of the data. This is likely to happen in several different scenarios. Firstly, if X_1 is an actual outlier then $a^2 - b^2$ will be smaller (more negative)

and so C overall closer to 0. Secondly, a smaller bandwidth makes this more likely to happen since this makes a and b larger, and so $a^2 - b^2$ smaller (more negative). The final factor is the size of the sum of the Y_i. This does not have as significant an impact, but if this quantity is extremely large then C is less likely to be close to 0. C and D must be very small for this constant fit to occur, and so in the region slightly closer to the data, where these are sufficiently large to have an impact, estimates will be returned which are very close to, and tend to, Y_1.

It is likely that similar theory could be developed for the local linear case, however this is significantly more complex.

Larger bandwidths

As mentioned earlier, these effects are not obvious when larger bandwidths are used, and this is demonstrated in the theory at the end of the previous section. When a larger bandwidth is used in local constant regression, a constant fit other than Y_1 is usually observed. Larger bandwidths make it impossible for C to become very close to zero before the regression computationally breaks down. This is because the difference between a and b will not be as large. In this case, points other than the nearest data point are still making a contribution to the estimate at x_0.

Similarly, in local linear regression with a larger bandwidth, the linear trend is not observed outside the data range, and instead a curve is observed. No definitive conclusions can be made, but it is likely that this also occurs due to more data points contributing substantially to the estimate at x_0. It is possible that when a larger bandwidth is used, the regression cannot reach stage **B** before computationally breaking down.

Computational instability

Some theory is now presented regarding the stage of computational instability, **C**. This stage only occurs with local linear regression and it manifests itself in two possible ways. It can either appear in the form it does in Fig. 1.12, or the regression estimate can appear to approximately resemble a

constant fit at this stage (see Fig. 1.14 for an example of this). The second of these can be identified as a computational problem by examining the formula for local linear regression for univariate data, given earlier as (1.22).

For an x_0 situated far from the data, $S_{n,1}$ will be moderately sized, and $S_{n,2}$ will be comparatively very large. In some cases this term may be so much larger that it makes the term associated with $S_{n,1}$ insignificant in the calculation. If this is swamped then the $S_{n,2}$ terms on the numerator and denominator cancel and the remaining expression is identical to the univariate local constant estimator, (1.19). Therefore this instability appears as a local constant regression estimate at that point.

The second type of computational instability, demonstrated in Fig. 1.12 at approximately age=80, can be understood by examining the least squares form of the estimator, (1.6). It is caused by $\mathbf{X}^T\mathbf{W}\mathbf{X}$, which is inverted in this problem, being close to a singularity. The reason that local constant regression does not suffer in the same way can be explained theoretically, using the properties of condition numbers. Petersen and Pedersen (2008) define the condition number of a matrix as "*the ratio between the largest and the smallest singular value of the matrix. The condition number can be used to measure how singular a matrix is. If the condition number is large, it indicates that the matrix is nearly singular*". For local constant regression, the matrix $\mathbf{X}^T\mathbf{W}\mathbf{X}$ ($\mathbf{X}$ is the univariate local constant equivalent of (1.7)) is a scalar. Therefore, using the definition of a condition number $c(\mathbf{A}) = \|\mathbf{A}\| \, . \, \left\|\mathbf{A}^{-1}\right\|$ (Petersen and Pedersen (2008)) it is clear that $c(\mathbf{X}^T\mathbf{W}\mathbf{X})$ is always 1. Thus, the condition number is never large, and so the matrix is never close to being singular. As a result, as one estimates further from the data the estimation remains steady, becomes a constant fit, and then immediately returns *NaN*.

Idealisations

The following idealisation, although not having been observed, is interesting theoretically. It describes what might happen if R could calculate estimates further from the data without computationally breaking down and returning

NaN. This theorizes what happens when one estimates at a point, x_0, which is sufficiently far from the data that all the data points can be considered equally far from it. In this case the contributions, given through the kernel function, of each point are very small and approximately the same.

For univariate data and for the local constant case, a crude calculation indicates that the mean of the response values, $\bar{Y}$, would be estimated at such a point. For univariate data the local constant estimate is

$$\hat{m}(x_0) = \frac{\sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right) Y_i}{\sum_{i=1}^{n} \kappa\left(\frac{X_i - x_0}{h}\right)}. \tag{1.45}$$

In this idealisation, define $w = \kappa\left(\frac{X_i - x_0}{h}\right)$ for all X_i, resulting in

$$\hat{m}(x_0) = \frac{w \sum_{i=1}^{n} Y_i}{nw} = \bar{Y}. \tag{1.46}$$

In the local linear case, estimates this far from the data again resemble a linear fit, but here, crude calculations indicate that the gradient of this fit is the same as that which one would obtain performing simple linear regression on the data. The gradient of the univariate local linear estimator, (1.22), is

$$\hat{m}'(x_0) = \frac{\sum \kappa\left(\frac{X_i - x_0}{h}\right) \sum Y_i (X_i - x_0) \kappa\left(\frac{X_i - x_0}{h}\right) - \sum (X_i - x_0) \kappa\left(\frac{X_i - x_0}{h}\right) \sum \kappa\left(\frac{X_i - x_0}{h}\right) Y_i}{\sum \kappa\left(\frac{X_i - x_0}{h}\right) \left\{\sum \kappa\left(\frac{X_i - x_0}{h}\right) (X_i - x_0)^2 - (X_i - x_0) \sum \kappa\left(\frac{X_i - x_0}{h}\right) (X_i - x_0)\right\}}. \tag{1.47}$$

If w is defined as above, then (1.47) becomes

$$\begin{aligned}
\hat{m}'(x_0) &= \frac{\sum_{i=1}^{n} w \sum_{i=1}^{n} w Y_i (X_i - x_0) - \sum_{i=1}^{n} w (X_i - x_0) \sum_{i=1}^{n} w Y_i}{\sum_{i=1}^{n} w \left\{\sum_{i=1}^{n} w (X_i - x_0)^2 - (X_i - x_0) \sum_{i=1}^{n} w (X_i - x_0)\right\}} \\
&= \frac{n \sum_{i=1}^{n} Y_i (X_i - x_0) - \sum_{i=1}^{n} (X_i - x_0) \sum_{i=1}^{n} Y_i}{\sum_{i=1}^{n} \left\{\sum_{i=1}^{n} (X_i - x_0)^2 - (X_i - x_0) \sum_{i=1}^{n} (X_i - x_0)\right\}} \\
&= \frac{n \sum_{i=1}^{n} Y_i (X_i - x_0) - \sum_{i=1}^{n} (X_i - x_0) \sum_{i=1}^{n} Y_i}{n \sum_{i=1}^{n} (X_i - x_0)^2 - \left(\sum_{i=1}^{n} (X_i - x_0)\right)^2} \\
&= \frac{n \sum_{i=1}^{n} X_i Y_i - \sum_{i=1}^{n} X_i \sum_{i=1}^{n} Y_i}{n \sum_{i=1}^{n} X_i^2 - \left(\sum_{i=1}^{n} X_i\right)^2} \\
&= \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n} (X_i - \bar{X})^2}
\end{aligned} \tag{1.48}$$

which is the coefficient for the gradient in simple linear regression.

This can be generalized in the multivariate case by considering the gradient in matrices form. Take for example the derivative in the x_1 direction.

As expressed earlier, this is $\hat{\beta}_{11}$, which is obtained in the minimization (1.4). Specifically,

$$\hat{\beta}_{11} = \mathbf{e}_2{}^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{Y} \tag{1.49}$$

where $\mathbf{e}_2$ is a vector with 1 as its second entry and 0 in the other d entries. In this idealised scenario, the weight matrix would take the form

$$\mathbf{W} = \text{diag}\{w, ..., w\} \tag{1.50}$$

and (1.49) becomes

$$\hat{\beta}_{11} = \mathbf{e}_2{}^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}, \tag{1.51}$$

which is the slope coefficient in the x_1 direction in multiple linear regression. This may be more useful, since in the multivariate setting it probably makes more sense to consider all the data an equal distance from such a point, and so the assumption that all data make an equal contribution would be fairer.

With trivariate data

Since the focus of this thesis is multivariate data, it is of interest to establish whether the behaviour of local polynomial regression outside the data range, observed above, translates to the multivariate setting. The same analysis as above was carried out but on trivariate data clouds. For each data set, regression was performed along a line of points, passing through the centre of the data cloud and extending out of it, in each covariate direction. One real data set on which this was attempted was the *California Air Pollution* data, of size $n = 345$, which measures the response of ozone level to various meteorological variables in Upland, California, U.S.A., in 1976. This is included in the **SemiPar** package by Wand (2010). Fig. 1.14 shows a typical result from this analysis. This shows how the local linear regression estimate changes as the covariate *inversion base height* is varied through the centre of the data cloud. This data cloud is shown in Fig. 2.1. The data range of this covariate is from 0 to 5000, and here the regression estimates are calculated from 0 to 15,000.

From these analyses, as demonstrated in Fig. 1.14, it is apparent that behaviour similar to that in the univariate setting occurs here. Similarities

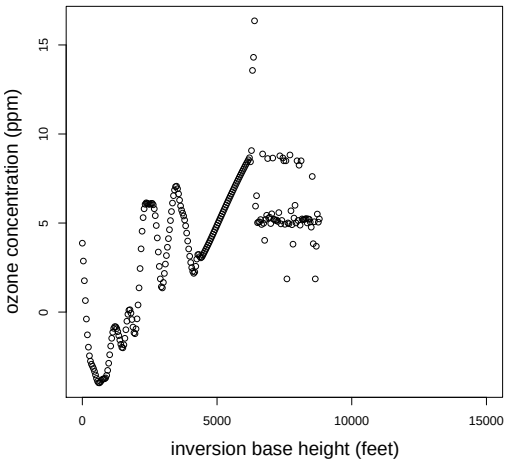

Figure 1.14: Trivariate local linear regression performed on the *California Air Pollution* data. This plot displays the estimate of ozone level v. base height (one of the covariates).

include the period where a linear fit is observed outside the data range, and the computational instability. However, this behaviour is not observed as consistently and the trends outside the data range vary, depending on from where one estimates relative to the data cloud. However this inconsistency usually occurs sufficiently far from the data cloud to not be of real interest in this study. As one estimates just outside the data cloud, and gradually estimates further away, there is always a small period of what could be considered normal estimation, followed by an approximately linear fit.

It is difficult to determine the factors which influence the slope of this period of linearity. It is likely to mimic the relationship between the nearest two points, as in the univariate case, although this is only speculation. The trivariate data sets also replicate the behaviour of the univariate examples by not exhibiting these standard patterns when larger bandwidths are employed.

Overall, this study provided some interesting results. It did not help

directly with the development of the threshold in Chapter 2, but it did help with the development of a greater understanding of the curse of dimensionality. It is clear that in the region outside the data range, where the local linear regression estimate becomes part of an approximately linear fit, determined only by the nearest two points, the variance is too high, and it is totally unreasonable to estimate here. It is desirable therefore for the density threshold to cut off estimation at some point closer to the data cloud than the point at which this linearity is reached, in the period of what could still be considered *normal* estimation. If this occurs then it is likely that at any point accepted by the threshold, with any data set, the density is of a magnitude large enough for reasonable regression, as areas of a similar magnitude of density demonstrated in this study.

It is also of interest to observe how poorly local polynomial regression behaves outside of the data range. Parametric estimators do not behave with the same instability or large variance in these areas, and this study supports the notion that local regression simply should not be attempted in these regions.

1.5.2 Competing methods

In nonparametric regression, the most common solution to the curse of dimensionality is to use an additive model. This models m as a sum of univariate functions, each estimated by a method of smoothing, with one for each covariate dimension i.e.

$$\hat{m}_A(\mathbf{x}) = \hat{\alpha} + \sum_{j=1}^{d} \hat{m}_j(x_j), \tag{1.52}$$

where $\hat{\alpha}$ is a constant and $\hat{m}_j$ are smooth univariate functions. If it can be shown, for example by a locally weighted regression analysis, that there is no interaction between the different variables, then Cleveland and Devlin (1988) suggest that additive models can be used with confidence. However, if this is not the case, Wand and Jones (1995) indicate that there is the potential for greater error in the estimation since additive models do not have the same flexibility as local polynomial regression. Variations to additive models exist,

such as projection pursuit regression, developed by Friedman and Stuetzle (1981).

A further alternative class of nonparametric regression techniques is that of *adaptive* multivariate smoothing. Based on the fact that the curse of dimensionality is not as severe for very smooth functions, Cornillon, Hengartner and Matzner-Løber (2011) build on the work of Lepski (1991) and others to develop a multivariate smoothing technique which "*adapts to the underlying smoothness of the true regression function.*" This is essentially an iterative bias reduction procedure which iterates from a very smooth pilot estimator until the prediction error is minimized. One potentially questionable feature of this method, and others, is the quality of the estimation in areas where there is no data. It is likely that an estimate at such a point, which is effectively a pilot estimate, is less informative than no estimate at all. The disadvantages of these competing methods form a large part of the motivation for the development of the techniques in Chapters 2 and 3.

1.6 R and the np package

The majority of the computational analysis covered in this thesis is carried out on R (R Development Core Team, 2010). There exists very little in terms of packages or code already available to perform multivariate local polynomial regression, and for this reason all functions were written from scratch, usually using the Cramer's rule methodology given in (1.30). The **np** package, by Hayfield and Racine (2008), does contain code for multivariate local polynomial regression, and the associated bandwidth selection, however in practice this behaves strangely in areas of sparse data. The function `npreg` works with data of any dimension, but while (1.30) returns NaN in areas of very sparse data, an adjustment is made in the **np** code which results in NaN never being returned, and instead estimates extremely close to 0 are given. This is clearly not always a sensible regression estimate, for example in data where all Y_i have a magnitude in the thousands. Additionally, at points in regions of the data which are almost as sparse as these, where the use of (1.30) still returns an estimate, this estimate differs

from that given by `npreg`. These observations mean that the adjustments made by the **np** package cause the regression estimates to diverge from the basic exact expression for local polynomial regression, (1.6), since (1.30) is identical to this. It should be noted that in this thesis, the simulated data examined in three dimensions is generally quite sparse, in an effort to mimic the impact of the curse of dimensionality in a higher number of dimensions, and in areas of higher density `npreg` returns identical regression estimates to (1.6) and (1.30).

Chapter 3 discusses in detail the effect that data sparsity has on bandwidth selection, and in particular how the *NaN*s returned in this setting mean that the inclusion of points from this area should be avoided. In the **np** package, in the function `npregbw`, this problem is not encountered and all points, as well as those with poor regression estimates mentioned above, are included. One is given the choice between the Kullback-Leibler method and least squares cross-validation (LSCV) described in Li and Racine (2004). The Kullback-Leibler method is better suited to categorical data, as seems to be a big focus of the package in general, and LSCV is to be used otherwise. Under scrutiny, LSCV was found to often give bandwidth parameters which were too large, which could be due to the inclusion of poor regression estimates. This poor performance is demonstrated in the simulation study in Chapter 3. The authors themselves concede in a vignette (Hayfield and Racine, 2008), published with the package, that their bandwidth selection methods may produce poor results, possibly due to "*outliers or the discretisation of continuous data.*"

In adapting to the computational problems which arise in areas of sparse data, the approach of the **np** package appears to be that any estimate is better than no estimate. After all, this helps computationally in bandwidth selection and other areas. However in doing this the quality and purity of regression estimates are sacrificed in these regions, and these are then included in bandwidth selection, possibly to its detriment. This contrasts with the philosophy of this thesis which is to first establish the feasibility of regression at a point, and then to decide against any regression at all at that

point if regression is considered infeasible. In this way, any regression estimation carried out produces an estimate exactly equal to (1.6). Due to this contrast in philosophies and its poor performance in bandwidth selection, the **np** package was rarely used in the regression context in the examples presented in this thesis.

1.7 Representing multivariate regression visually

Univariate regression is usually represented visually by a curve, and in the bivariate case by a surface, however representation becomes an issue for $d > 2$. Visualization is often one of the most important goals in smoothing so it is vital that there are techniques available. Fortunately there is a range of options, at least for data with covariates of up to three dimensions.

One widely-used technique is conditioning plots, which are particularly suitable if one or more of the covariates can be split into categories. A curve or surface can then be displayed in each plot, depending on the dimension of the remaining covariates within each category. Alternatively these could all be displayed on one plot, using colour to distinguish between the different categories, however analysis then becomes difficult when the different colours overlap.

Unfortunately, conditioning plots are not suitable when the covariates cannot be categorized easily. For this reason it is sensible to use colour more broadly. The problem is that it is difficult to interpret results other than at a basic level, however Fig. 1.6 and Figs. 2.1-2 in the next chapter demonstrate that it is effective at this level. In Fig. 2.2, at each grid point in the data range at which the density is considered high enough for regression, using the threshold developed in Chapter 2, a coloured point is plotted, the colour of which depends on the regression estimate at that point. Here, for the smallest regression values the points are bright green, and the largest values are bright red. Any values in between are represented on this green-red scale. At any point in the data range at which the density is not considered to be high enough, no point is plotted. In this way, it is easier to view the points of actual interest, while at the same time nicely representing the

region where estimation is reasonable as a mass of colour.

A further improvement is derived from the use of the **rgl** package, by Adler and Murdoch (2011), on R to construct this plot. The advantage here is that one can move the resulting plot around on the computer screen in order to best explore different parts of the data range. Another possibility in the **rgl** package is to represent each point by a sphere, rather than a two-dimensional cross. By varying the radius of the spheres and the fineness of the grid one can create the impression that the spheres merge into one object and so create a single coloured shape, which defines each region within the data range where smoothing could be considered reliable. However, one should be careful when adjusting the sphere radius and grid fineness, since if the spheres are merged too much the ability to analyse the data in the middle of the coloured region is lost. Therefore, they should be varied depending on the data and the aim of the smoothing.

The procedure described above and applied to create Fig. 2.2 is unique in creating a plot which represents both density, and local polynomial regression, throughout a grid, by highlighting areas of high density and then displaying the regression estimates via colour. This is only effective for trivariate data, unless implemented within conditioning plots. The three-dimensional contour plots used in Scott (1992) and Bowman and Azzalini (1997) are similar but are used only in the density estimation context. In this thesis, as the data of interest increases in dimension, the focus moves away from visualizing the regression and towards producing the best possible regression estimate at a single point at which it is considered feasible.

1.8 Density estimation

Density estimation is an important tool, used throughout this thesis. Wand and Jones (1995) suggest that *nonparametric* density estimation is particularly important in high dimensions due to the issues faced by parametric methods in this setting. In this thesis kernel density estimation is used.

The kernel density estimate, for a multivariate point $\mathbf{x}$ is;

$$\hat{f}(\mathbf{x}) = n^{-1} \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x} - \mathbf{X}_i) \tag{1.53}$$

where $K_{\mathbf{H}}$ is a multivariate kernel function as defined earlier and again a bandwidth matrix is needed. In practice, multivariate kernel density estimation has not been observed to suffer as severely from the curse of dimensionality and therefore acts as a reliable first measure of a data set in the procedure developed in Chapter 2.

Kernel density estimation is also the focus of Chapter 5, but in this case in the univariate setting. The kernel density estimator for a univariate random variable X, with standard deviation σ, is (analogously to the multivariate estimate (1.53))

$$\hat{f}(x) = (nh)^{-1} \sum_{i=1}^{n} \kappa \left(\frac{x - X_i}{h} \right), \tag{1.54}$$

where κ is a kernel function and h is the univariate bandwidth. Chapter 5 is devoted to the important issue of bandwidth selection in this context. The techniques developed there are based on the *normal reference rule* of Silverman (1986) which aims to select the asymptotically optimal bandwidth. Due to its importance, the asymptotically optimal bandwidth is derived below, again using the MISE as the starting point for its derivation. In the density estimation context,

$$\text{MISE}(f, \hat{f}) = E \int \{\hat{f}(x) - f(x)\}^2 \, dx, \tag{1.55}$$

which can also be written in this context in terms of the bias and variance of the density estimate,

$$\text{MISE}(f, \hat{f}) = \int \left\{ \left[\text{Bias} \left(\hat{f}(x) \right) \right]^2 + \text{Var} \left(\hat{f}(x) \right) \right\} dx. \tag{1.56}$$

The asymptotic approximations ($h \to 0$, $nh \to \infty$ as $n \to \infty$) of the bias and variance are derived in detail in Silverman (1986). A summary of these derivations is given below.

$$\begin{aligned} \text{Bias} \left(\hat{f}(x) \right) &= E\hat{f}(x) - f(x) \\ &= \int h^{-1} \kappa \left\{ (x - y)/h \right\} f(y) dy - f(x). \end{aligned} \tag{1.57}$$

Using the substitution $y = x - hu$ and assuming $\int \kappa(u)du = 1$,

$$\text{Bias}\left(\hat{f}(x)\right) = \int \kappa(u)\left\{f(x-hu) - f(x)\right\}du, \tag{1.58}$$

and using the Taylor series and the assumption that $\int u\kappa(u)du = 0$,

$$\text{Bias}\left(\hat{f}(x)\right) = \frac{h^2 f''(x)\int u^2\kappa(u)du}{2} + O(h^4). \tag{1.59}$$

Now, the exact variance at $\hat{f}(x)$ is given as

$$\text{Var}\left(\hat{f}(x)\right) = n^{-1}\int h^{-2}\kappa\left\{(x-y)/h\right\}^2 f(y)dy - n^{-1}\left\{f(x) + \text{Bias}\left(\hat{f}(x)\right)\right\}^2. \tag{1.60}$$

Here, again use $y = x - hu$, as well as expression (1.59) to give

$$\text{Var}\left(\hat{f}(x)\right) \approx n^{-1}h^{-1}\int f(x-hu)\kappa(u)^2du - n^{-1}\left\{f(x) + O(h^2)\right\}^2. \tag{1.61}$$

The implementation of a Taylor series then yields

$$\text{Var}\left(\hat{f}(x)\right) \approx \frac{f(x)\int \kappa^2(u)du}{nh} + O(n^{-1}). \tag{1.62}$$

Substituting these asymptotic expressions into (1.56), one obtains the AMISE,

$$\text{AMISE}(h) = \frac{h^4}{4}\int \left[f''(x)\right]^2 dx \left[\int u^2\kappa(u)du\right]^2 + \frac{1}{nh}\int \kappa^2(u)du. \tag{1.63}$$

Minimizing (1.63) w.r.t. h yields the asymptotically optimal bandwidth,

$$h_{\text{opt}} = \kappa_0\left\{\int \left[f''(x)\right]^2 dx\right\}^{-1/5} n^{-1/5}, \tag{1.64}$$

where $\kappa_0 = [\int u^2\kappa(u)\,du]^{-2/5}[\int \kappa^2(u)\,du]^{1/5}$ is a (known) constant depending only on kernel moments.

The main aim of this thesis is to provide methods of dealing with the curse of dimensionality in multivariate local polynomial regression, and Chapters 2 and 3 concentrate on this. Chapters 4 and 5 cover the slightly different, yet still related, topics of modal regression and bandwidth selection for univariate multimodal density estimation respectively.

Chapter 2

Assessing the reliability of local linear regression

Recall that the curse of dimensionality is a problem which arises in higher dimensions and results in reliable estimation using local polynomial regression not being possible in neighbourhoods where the data is too sparse. For this reason, using this technique is often discounted as an option when looking at multivariate data, and other techniques such as additive models or thin-plate splines are favoured. There is however a significant increase in flexibility when using local polynomials, particularly in comparison with additive models, and for this reason the primary aim of the research composing Chapters 2 and 3 is to find techniques which avoid the curse of dimensionality.

2.1 A solution using density

The solution presented in this chapter is one which essentially ignores all neighbourhoods which do not contain enough data, and so only performs smoothing over some region in which estimation is considered reliable, where the bias and variance of $\hat{m}$ can be kept reasonably low. In this way the curse of dimensionality is avoided. This method is not universal in the sense that it does not produce estimates over the whole data range, but it is satis-

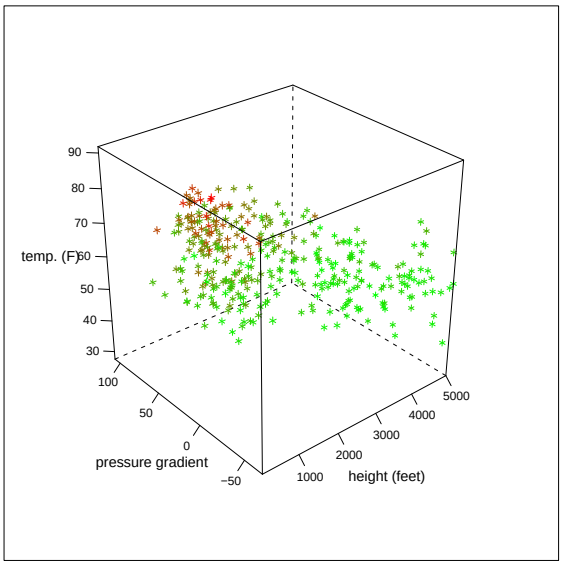

Figure 2.1: *California Air Pollution* data. Red represents the higher values of ozone concentration (the response, in ppm), and green the lower.

factory in the sense that it produces estimates, with all the advantages of local polynomial regression, in some regions of the data space. Fig. 2.2 illustrates the idea for trivariate data. For one, two and three-dimensional covariates an *envelope* can be created to display visually the region in which reliable estimation is possible. In dimensions higher than this it becomes both harder to visualize, and computationally more demanding to calculate this feasibility over a whole grid and so one concentrates on whether regression is advisable at particular points of interest over the data range. Fig. 2.1 shows the *California Air Pollution* data, introduced in Chapter 1. Here, the response is represented by colour, where red represents the higher values of ozone concentration, and green the lower. Fig. 2.2 shows an example of an envelope in which smoothing can be considered somewhat reliable and here the colour represents the smoothed regression estimate at the included points. This example features two main regions where smoothing could be considered reliable.

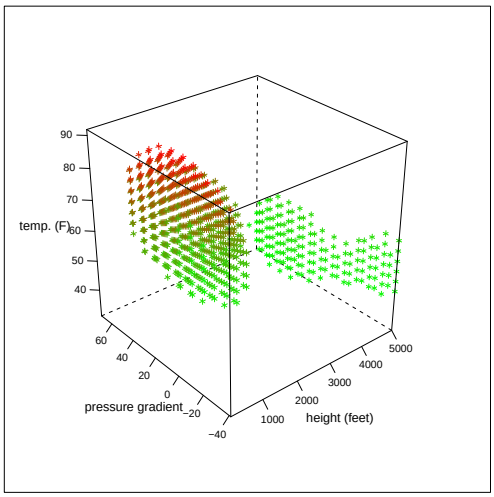

Figure 2.2: Regression estimates displayed at only those points at which regression is considered feasible for the *California Air Pollution* data. Red represents the higher estimates of ozone (ppm), and green the lower.

To find these areas, and to discover where there is sufficient data, the density f of $\mathbf{X}$ is examined, using (1.53). For reasons that will become clear later, the same $\mathbf{H}$ is used in calculating $\hat{f}$ as in the regression step. A threshold T is sought such that, if $\hat{f}(\mathbf{x})>T$ at a point $\mathbf{x}$, then an estimate using local linear regression can be considered somewhat reliable, and otherwise, care should be taken and an alternative method sought, possibly local constant fitting. Intuitively, T should depend on n and $\mathbf{H}$, as $\hat{f}$ does, as decreasing either of them will reduce the number of data points which are locally available at $\mathbf{x}$, requiring in turn a larger threshold to allow reliable estimation.

2.1.1 The influence

In the derivation of the density threshold which follows, the concept of influence is crucial. By definition, the influence at a data point $\mathbf{X}_i$, $\text{infl}(\mathbf{X}_i)$, is the diagonal element of the ith row of the smoother matrix $\mathbf{S}$. This describes

the contribution of observation $\mathbf{X}_i$ to the estimation at $\mathbf{x} = \mathbf{X}_i$.

The following theorem is taken from Loader (1999).

"*Suppose the weight function* $K(\boldsymbol{x})$ *is non-negative, symmetric and decreasing on* $[0, \infty)$. *Then*

1. *the influence function dominates the variance;*

$$\frac{1}{\sigma_\epsilon^2} Var(\hat{m}(\boldsymbol{x})) \leq infl(\boldsymbol{x})$$

2. *at the observation points* $\boldsymbol{X}_i$,

$$infl(\boldsymbol{X}_i) \leq 1 \tag{2.1}$$

and hence local regression is variance-reducing."

Hence, bounding the influence implies bounding the variance, (1.12). If an observation is very *influential* then the estimate at that point will be very sensitive to it, and so the variance higher. This occurs primarily at data points close to the boundary where the influence is close to 1. Local regression is more feasible away from the boundary and since, according to Hastie and Loader (1993b), so much of the data space in higher dimensions can be considered as the boundary region, it is this area which needs to be classified, and potentially excluded from the regression problem.

Now, the diagonal element of the ith row of $\mathbf{S}$ is

$$\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\tilde{\boldsymbol{e}}_i$$

where $\mathbf{x} = \mathbf{X}_i$ and $\tilde{\boldsymbol{e}}_i$ is a vector of length n with 1 in the ith position. This is equivalent to

$$\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1} \begin{pmatrix} K_{\mathbf{H}}(\mathbf{0}) \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

where $(K_{\mathbf{H}}(\mathbf{0}), 0, ..., 0)^T$ is a vector of length $d+1$. This is then equivalent to

$$\text{infl}(\mathbf{X}_i) = |\mathbf{H}|^{-1/2}\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{e}_1 K(\mathbf{0}).$$

where $\mathbf{x} = \mathbf{X}_i$ in $\mathbf{X}$. This formula is given in Loader (1999), and below, generalized for any $\mathbf{x}$. Here $\mathbf{x} = \mathbf{x}$ in $\mathbf{X}$.

$$\text{infl}(\mathbf{x}) = |\mathbf{H}|^{-1/2}\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{e}_1 K(\mathbf{0}). \tag{2.2}$$

Justifying infl$(\mathbf{X}_i) \leq 1$

Although stated in Loader (1999) that infl$(\mathbf{X}_i) \leq 1$, this result is not proved there.

Through a thorough examination of extremely isolated points, using very small bandwidths, it is observed that, at a data point, the influence is never greater than 1. This is proved below for the univariate local constant case.

For univariate local constant regression, where $\kappa_h(x) = \frac{1}{h}\kappa(\frac{x}{h})$, where h is the bandwidth,

$$\mathbf{X}^T\mathbf{W}\mathbf{X} = \sum_{i=1}^{n} \kappa_h\left(X_i - x\right)$$

so using (2.2)

$$\begin{aligned}
\text{infl}(X_j) &= \frac{\kappa_h(0)}{\sum_{i=1}^n \kappa_h\left(X_i - X_j\right)} \\
&= \frac{\kappa_h(0)}{\kappa_h(0) + \sum_{i\neq j} \kappa_h(X_i - X_j)} \\
&\leq 1
\end{aligned}$$

since $\kappa_h(x)$ is always non-negative.

Also, for univariate local linear regression, it is shown below that at a point which is as isolated as possible, which one would expect to have the highest possible influence, the influence is 1.

For univariate local linear regression,

$$\mathbf{X}^T\mathbf{W}\mathbf{X} = \begin{bmatrix} \sum_{i=1}^n \kappa_h(X_i - x_0) & \sum_{i=1}^n (X_i - x_0)\kappa_h(X_i - x_0) \\ \\ \sum_{i=1}^n (X_i - x_0)\kappa_h(X_i - x_0) & \sum_{i=1}^n (X_i - x_0)^2\kappa_h(X_i - x_0) \end{bmatrix}$$

so using (2.2)

$$\text{infl}(x_0) = \frac{\kappa_h(0)\sum_{i=1}^n (X_i - x_0)^2\kappa_h(X_i - x_0)}{\sum_{i=1}^n (X_i - x_0)^2\kappa_h(X_i - x_0)\sum_{i=1}^n \kappa_h(X_i - x_0) - [\sum_{i=1}^n (X_i - x_0)\kappa_h(X_i - x_0)]^2}$$

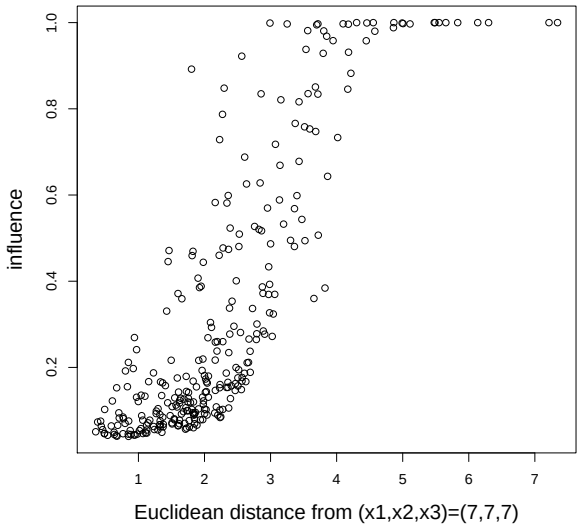

Figure 2.3: The influence plotted against the Euclidean distance of the point from the centre of the data cloud, (7,7,7), for trivariate data simulation A.

If x_0 is completely isolated so that $\kappa_h(X_i - x_0) = 0$ for all $X_i \neq x_0$ then

$$\begin{aligned}
\text{infl}(x_0) &= \frac{\kappa_h(0)}{\sum_{i=1}^n \kappa_h(X_i - x_0) - \frac{\left[\sum_{i=1}^n (X_i - x_0)\kappa_h(X_i - x_0)\right]^2}{\sum_{i=1}^n (X_i - x_0)^2 \kappa_h(X_i - x_0)}} \\
&= \frac{\kappa_h(0)}{\kappa_h(0) - 0} \\
&= 1.
\end{aligned}$$

This is shown for univariate local linear regression, but is almost certainly the case for higher dimensions too.

Fig. 2.3 plots the diagonal elements of the smoother matrix, i.e. the influence values at $\mathbf{X}_i$, for the trivariate simulated data set A (introduced later in this chapter). The plot shows the influence values versus the Euclidean distances of the points from the centre of the data cloud. This simulated data set has the form of a cloud which is denser in the middle and becomes gradually sparser moving to the extremes in each covariate direction. Corre-

sponding to this on the plot, are the high influence values, and the absence of low values at larger Euclidean distances. The influence values peak at 1, as discussed above. This plot is similar to the *self-influence plots* displayed for univariate data in Buja, Hastie and Tibshirani (1989).

All of the above suggest strongly that the influence is never greater than 1, however this does not constitute a proof.

2.1.2 Deriving a density threshold

In order to relate the influence to the density, asymptotics are used to formulate an *influence function* for any $\mathbf{x}$ over the data range, analogous to the influence formula given for actual data points, (2.2), above. The following assumptions are needed for the asymptotics: $\mathbf{x} \in \mathbb{R}^d$ is in the support of f which is continuously differentiable and $f(\mathbf{x}) > 0$. All second-order derivatives of m are continuous and the kernel is compactly supported and bounded. Also assume that each entry of $\mathbf{H}$ tends to 0 and $n^{-1}|\mathbf{H}|^{-1/2} \to 0$ as $n \to \infty$.

In notation, for sequences of real numbers, U_n and V_n, $U_n = O(V_n) \Leftrightarrow \exists_{c>0}\forall_{n\in\mathbb{N}}(|\frac{U_n}{V_n}| \leq c)$ and $U_n = o(V_n) \Leftrightarrow \forall_{c>0}\exists_N\forall_{n\geq N}(|\frac{U_n}{V_n}| < c)$, where $N \in \mathbb{N}$. Hence O(1) means that the sequence is bounded and o(1) means that it tends to 0 as $n \to \infty$. Convergence in terms of probability is expressed similarly; $U_n = O_p(V_n) \Leftrightarrow \forall_{c>0}\exists_{N,M}\forall_{n\geq N}\left\{P(|\frac{U_n}{V_n}| \geq M) < c\right\}$ and $U_n = o_p(V_n) \Leftrightarrow \forall_{c>0}P(|\frac{U_n}{V_n}| \leq c) \to 1$ as $n \to \infty$, where $N \in \mathbb{N}$ and $M \in \mathbb{R}$. In terms of probability, a sequence which is bounded is represented by $O_p(1)$ and a sequence which tends to 0 as $n \to \infty$, by $o_p(1)$. In the instances in which a matrix or vector appears within the order notation, this should be read component by component. A matrix/vector with 1 in each entry is represented by $\mathbf{1}$.

Now $\mathbf{X}^T\mathbf{W}\mathbf{X}$

$$= \begin{bmatrix} \sum_{i=1}^n K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x}) & \sum_{i=1}^n K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \\ \\ \sum_{i=1}^n K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x}) & \sum_{i=1}^n K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \end{bmatrix} \tag{2.3}$$

In approximating each of these entries, the following result derived using Chebyshev's inequality is used, where ς is a summation,

$$\varsigma = E(\varsigma) + O_p\left(\sqrt{\mathrm{Var}(\varsigma)}\right).$$

Firstly,

$$\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x}) = E\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})\right) + O_p\left(\sqrt{\mathrm{Var}\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})\right)}\right). \tag{2.4}$$

Since the $\mathbf{X}_i$ are i.i.d.

$$\begin{aligned} & E\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})\right) \\ &= nE\left(K_{\mathbf{H}}(\mathbf{X}_1 - \mathbf{x})\right) \\ &= n\int K_{\mathbf{H}}(\mathbf{t} - \mathbf{x}) f(\mathbf{t}) d\mathbf{t} \\ &= n\int |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2}(\mathbf{t} - \mathbf{x})) f(\mathbf{t}) d\mathbf{t} \end{aligned}$$

Then using the substitution $\mathbf{u} = (u_1, ..., u_d)^T = \mathbf{H}^{-1/2}(\mathbf{t} - \mathbf{x})$ one obtains

$$\begin{aligned} & n\int |\mathbf{H}|^{-1/2} K(\mathbf{u}) f(\mathbf{x} + \mathbf{H}^{1/2}\mathbf{u}) |\mathbf{H}|^{1/2} d\mathbf{u} \\ &= n\int K(\mathbf{u}) f(\mathbf{x} + \mathbf{H}^{1/2}\mathbf{u}) d\mathbf{u} \\ &= n\left(f(\mathbf{x})\int K(\mathbf{u}) d\mathbf{u} + o(1)\right) \end{aligned} \tag{2.5}$$

since according to Taylor's theorem

$$f(\mathbf{x} + \mathbf{H}^{1/2}\mathbf{u}) = f(\mathbf{x}) + f'(\mathbf{x})(\mathbf{H}^{1/2}\mathbf{u})^T + O(\mathbf{H}).$$

$$\begin{aligned} \mathrm{Var}\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})\right) &= n\left[E\left((K_{\mathbf{H}}(\mathbf{X}_1 - \mathbf{x}))^2\right) - \left(E\left(K_{\mathbf{H}}(\mathbf{X}_1 - \mathbf{x})\right)\right)^2\right] \\ &= n\left[\int |\mathbf{H}|^{-1} K^2(\mathbf{H}^{-1/2}(\mathbf{t} - \mathbf{x})) f(\mathbf{t}) d\mathbf{t} - \left(\int |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2}(\mathbf{t} - \mathbf{x})) f(\mathbf{t}) d\mathbf{t}\right)^2\right] \end{aligned}$$

Then using the same substitution as above

$$
\begin{aligned}
&\text{Var}\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})\right) \\
&= n\left[\int |\mathbf{H}|^{-1} K^2(\mathbf{u}) f(\mathbf{x} + \mathbf{H}^{1/2}\mathbf{u})|\mathbf{H}|^{1/2} d\mathbf{u} - \left(\int |\mathbf{H}|^{-1/2} K(\mathbf{u}) f(\mathbf{x} + \mathbf{H}^{1/2}\mathbf{u})|\mathbf{H}|^{1/2} d\mathbf{u}\right)^2\right] \\
&= n\left[|\mathbf{H}|^{-1/2} f(\mathbf{x})\left(\int K^2(\mathbf{u}) d\mathbf{u} + o(1)\right) - \left(f(\mathbf{x})\int K(\mathbf{u}) d\mathbf{u} + o(1)\right)^2\right] \\
&= n\left[|\mathbf{H}|^{-1/2} f(\mathbf{x})\left(\int K^2(\mathbf{u}) d\mathbf{u} + o(1)\right) - \left(f^2(\mathbf{x})\int K^2(\mathbf{u}) d\mathbf{u} + o(1)\right)\right] \\
&= n|\mathbf{H}|^{-1/2}\left[f(\mathbf{x})\int K^2(\mathbf{u}) d\mathbf{u} - |\mathbf{H}|^{1/2}\left(f^2(\mathbf{x})\int K^2(\mathbf{u}) d\mathbf{u} + o(1)\right)\right] \\
&= n|\mathbf{H}|^{-1/2}\left[f(\mathbf{x})\int K^2(\mathbf{u}) d\mathbf{u} + o(1)\right] \\
&= \frac{n^2\left[f(\mathbf{x})\int K^2(\mathbf{u}) d\mathbf{u} + o(1)\right]}{n|\mathbf{H}|^{1/2}} \\
&= n^2.O\left(\frac{1}{n|\mathbf{H}|^{1/2}}\right) \\
&= o(n^2) \qquad (2.6)
\end{aligned}
$$

Using (2.4)

$$
\begin{aligned}
\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x}) &= n\left(f(\mathbf{x})\int K(\mathbf{u}) d\mathbf{u} + o(1)\right) + O_p\left(\sqrt{o(n^2)}\right) \\
&= n\left(f(\mathbf{x})\int K(\mathbf{u}) d\mathbf{u} + o(1)\right) + n.o_p(1) \\
&= n\left(f(\mathbf{x})\int K(\mathbf{u}) d\mathbf{u} + o_p(1)\right). \qquad (2.7)
\end{aligned}
$$

Similarly

$$
\begin{aligned}
&E\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i-\mathbf{x})(\mathbf{X}_i-\mathbf{x})\right)\\
&= nE\left(K_{\mathbf{H}}(\mathbf{X}_1-\mathbf{x})(\mathbf{X}_1-\mathbf{x})\right)\\
&= n\int K_{\mathbf{H}}(\mathbf{t}-\mathbf{x})(\mathbf{t}-\mathbf{x})f(\mathbf{t})d\mathbf{t}\\
&= n\int |\mathbf{H}|^{-1/2}K(\mathbf{H}^{-1/2}(\mathbf{t}-\mathbf{x}))(\mathbf{t}-\mathbf{x})f(\mathbf{t})d\mathbf{t}\\
&= n\int |\mathbf{H}|^{-1/2}K(\mathbf{u})\mathbf{H}^{1/2}\mathbf{u}f(\mathbf{x}+\mathbf{H}^{1/2}\mathbf{u})|\mathbf{H}|^{1/2}d\mathbf{u}\\
&= n\mathbf{H}^{1/2}\int \mathbf{u}K(\mathbf{u})f(\mathbf{x}+\mathbf{H}^{1/2}\mathbf{u})d\mathbf{u}
\end{aligned}
$$

According to Taylor's theorem, there exists a ξ_u such that $f(\mathbf{x}+\mathbf{H}^{1/2}\mathbf{u}) = f(\mathbf{x}) + f'(\xi_u)(\mathbf{H}^{1/2}\mathbf{u})^T$ where ξ_u is on the line $\mathbf{x}+t\mathbf{H}^{1/2}\mathbf{u}$ with $t\in[0,1]$. Including this,

$$
\begin{aligned}
&E\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i-\mathbf{x})(\mathbf{X}_i-\mathbf{x})\right)\\
&= n\mathbf{H}^{1/2}\int \mathbf{u}K(\mathbf{u})\left(f(\mathbf{x})+(\mathbf{H}^{1/2}\mathbf{u})^T\nabla f(\xi_u)\right)d\mathbf{u}\\
&= n\mathbf{H}^{1/2}\left[f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u}+\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})\mathbf{H}^{1/2}\nabla f(\xi_u)d\mathbf{u}\right]\\
&= n\mathbf{H}^{1/2}\left[f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u}+\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})\mathbf{H}^{1/2}(\nabla f(\mathbf{x})+o(\mathbf{1}))d\mathbf{u}\right]\\
&= n\mathbf{H}^{1/2}\left[f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u}+\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)\mathbf{H}^{1/2}\nabla f(\mathbf{x})+o(\mathbf{H}^{1/2})\right]\\
&= n\mathbf{H}^{1/2}\left[f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u}+\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)\mathbf{H}^{1/2}\left(\nabla f(\mathbf{x})+o(1)\right)\right]\\
&= n\mathbf{H}^{1/2}f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u}+n\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)\mathbf{H}^{1/2}\nabla f(\mathbf{x})\left(1+o(1)\right)
\end{aligned}
\tag{2.8}
$$

And again using (2.4)

$$
\begin{aligned}
&\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i-\mathbf{x})(\mathbf{X}_i-\mathbf{x})\\
&= n\mathbf{H}^{1/2}f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u}+n\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)\mathbf{H}^{1/2}\nabla f(\mathbf{x})\left(1+o_p(1)\right)
\end{aligned}
\tag{2.9}
$$

Similarly

$$\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T$$
$$= nf(\mathbf{x}) \int \mathbf{u}^T K(\mathbf{u}) d\mathbf{u} \mathbf{H}^{1/2} + n\nabla f(\mathbf{x})^T \mathbf{H}^{1/2} \left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u}) d\mathbf{u} \right) \mathbf{H}^{1/2} (1 + o_p(1)) \tag{2.10}$$

And finally

$$\begin{aligned} & E\left(\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \right) \\ &= nE\left(K_{\mathbf{H}}(\mathbf{X}_1 - \mathbf{x})(\mathbf{X}_1 - \mathbf{x})(\mathbf{X}_1 - \mathbf{x})^T \right) \\ &= n \int K_{\mathbf{H}}(\mathbf{t} - \mathbf{x})(\mathbf{t} - \mathbf{x})(\mathbf{t} - \mathbf{x})^T f(\mathbf{t}) d\mathbf{t} \\ &= n \int |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2}(\mathbf{t} - \mathbf{x}))(\mathbf{t} - \mathbf{x})(\mathbf{t} - \mathbf{x})^T f(\mathbf{t}) d\mathbf{t} \\ &= n \int |\mathbf{H}|^{-1/2} K(\mathbf{u}) \mathbf{H}^{1/2}\mathbf{u}(\mathbf{H}^{1/2}\mathbf{u})^T f(\mathbf{x} + \mathbf{H}^{1/2}\mathbf{u}) |\mathbf{H}|^{1/2} d\mathbf{u} \\ &= n\mathbf{H}^{1/2} \int K(\mathbf{u}) \mathbf{u}\mathbf{u}^T f(\mathbf{x} + \mathbf{H}^{1/2}\mathbf{u}) d\mathbf{u} (\mathbf{H}^{1/2})^T \\ &= n\mathbf{H}^{1/2} \left[\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u}) d\mathbf{u} \right) f(\mathbf{x}) + o(\mathbf{1}) \right] \mathbf{H}^{1/2} \end{aligned} \tag{2.11}$$

And

$$\begin{aligned} & \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \\ & = n\mathbf{H}^{1/2} \left[\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u}) d\mathbf{u} \right) f(\mathbf{x}) + o_p(\mathbf{1}) \right] \mathbf{H}^{1/2} \end{aligned} \tag{2.12}$$

So $\mathbf{X}^T\mathbf{W}\mathbf{X}$ can be written as

$$\begin{bmatrix} (2.7) & (2.10) \\ \\ (2.9) & (2.12) \end{bmatrix} \tag{2.13}$$

For (2.2) one needs the top left entry of the inverse of (2.13). For a general block matrix $\mathbf{A}$, such as this one, Petersen and Pedersen (2008) state that

this is equivalent to, $(A_{11} - A_{12}A_{22}^{-1}A_{21})^{-1}$. Denoting matrix (2.13) as $\mathbf{A}$,

$$
\begin{aligned}
&(A_{11} - A_{12}A_{22}^{-1}A_{21}) \\
&= n\left(f(\mathbf{x})\int K(\mathbf{u})d\mathbf{u} + o_p(1)\right) - \\
&\left(nf(\mathbf{x})\int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\mathbf{H}^{1/2} + n\left(\nabla f(\mathbf{x})^T\mathbf{H}^{1/2}\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\mathbf{H}^{1/2}\right)(1+o_p(1))\right)\times \\
&\left(n\mathbf{H}^{1/2}\left(\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)f(\mathbf{x}) + o_p(\mathbf{1})\right)\mathbf{H}^{1/2}\right)^{-1}\times \\
&\left(n\mathbf{H}^{1/2}f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u} + n\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)\mathbf{H}^{1/2}\nabla f(\mathbf{x})\,(1+o_p(1))\right) \\
&= n\left(f(\mathbf{x})\int K(\mathbf{u})d\mathbf{u} + o_p(1)\right) - n\left(f(\mathbf{x})\int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\mathbf{H}^{1/2} + o_p(\mathbf{1}^T\mathbf{H}^{1/2})\right)\times \\
&\left(n\left(\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)f(\mathbf{x})\mathbf{H}^{1/2} + o_p(\mathbf{H})\right)\right)^{-1}\times n\left(\mathbf{H}^{1/2}f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u} + o_p(\mathbf{H}^{1/2}\mathbf{1})\right)
\end{aligned}
\tag{2.14}
$$

Within (2.14), defining a_n as a sequence $a_n = o_p(\mathbf{H})$, b_n as a sequence $b_n = o_p(\mathbf{1})$ and c_n as a sequence $c_n = O(\mathbf{H}^{-1})$ one uses the Kailath Variant from Petersen and Pedersen (2008) to re-express the inverse. The Kailath Variant states that $(\mathbf{A}+\mathbf{BC})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{I}+\mathbf{C}\mathbf{A}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{A}^{-1}$. Here, say $\mathbf{A} = \mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)f(\mathbf{x})\mathbf{H}^{1/2}$, $\mathbf{B} = a_n$ and $\mathbf{C} = \mathbf{I}$. Hence

$$
\begin{aligned}
&\left(\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)f(\mathbf{x})\mathbf{H}^{1/2} + o_p(\mathbf{H})\right)^{-1} \\
&= \left(\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)f(\mathbf{x})\mathbf{H}^{1/2}\right)^{-1} - c_n a_n(\mathbf{I}+c_n a_n)^{-1}c_n \\
&= \left(\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)f(\mathbf{x})\mathbf{H}^{1/2}\right)^{-1} - b_n c_n \\
&= \left(\mathbf{H}^{1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)f(\mathbf{x})\mathbf{H}^{1/2}\right)^{-1} + o_p(\mathbf{H}^{-1}) \\
&= \mathbf{H}^{-1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}(f(\mathbf{x}))^{-1}\mathbf{H}^{-1/2} + o_p(\mathbf{H}^{-1})
\end{aligned}
$$

Substituting this in to (2.14) one obtains

$$
\begin{aligned}
&(A_{11} - A_{12}A_{22}^{-1}A_{21}) \\
&= n\left(f(\mathbf{x})\int K(\mathbf{u})d\mathbf{u} + o_p(1)\right) - n\left(f(\mathbf{x})\int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\mathbf{H}^{1/2} + o_p(\mathbf{1}^T\mathbf{H}^{1/2})\right)\times \\
&\left(\mathbf{H}^{-1/2}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}(f(\mathbf{x}))^{-1}\mathbf{H}^{-1/2} + o_p(\mathbf{H}^{-1})\right)\times\left(\mathbf{H}^{1/2}f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u} + o_p(\mathbf{H}^{1/2}\mathbf{1})\right) \\
&= n\left(f(\mathbf{x})\int K(\mathbf{u})d\mathbf{u} + o_p(1)\right) - n\left(\int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\mathbf{H}^{-1/2} + o_p(\mathbf{1}^T\mathbf{H}^{-1/2})\right) \\
&\times\left(\mathbf{H}^{1/2}f(\mathbf{x})\int \mathbf{u}K(\mathbf{u})d\mathbf{u} + o_p(\mathbf{H}^{1/2}\mathbf{1})\right) \\
&= n\left[f(\mathbf{x})\left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right] + o_p(1)\right]
\end{aligned}
\tag{2.15}
$$

Applying the inverse as mentioned earlier, one obtains an approximation for the top left entry of the inverse of (2.13)

$$
\begin{aligned}
&(B_{11} - B_{12}B_{22}^{-1}B_{21})^{-1} \\
&= n^{-1}(f(\mathbf{x}))^{-1}\left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right]^{-1} + o_p(n^{-1})
\end{aligned}
$$

Substituting this in (2.2) gives an approximation to the influence function

$$
\text{infl}(\mathbf{x}) = \frac{K(\mathbf{0})}{nf(\mathbf{x})|\mathbf{H}|^{1/2}}\left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right]^{-1} + o_p(n^{-1}|\mathbf{H}|^{-1/2}) \tag{2.16}
$$

Loader (1999) expresses $\left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right]^{-1}$ as $\mathbf{e}_1^T\mathbf{M}_1^{-1}\mathbf{e}_1$ where $\mathbf{M}_1$ is $\left(\int K(\mathbf{u})A(\mathbf{u})A(\mathbf{u})^T d\mathbf{u}\right)$ and $A(\mathbf{u}) = (1, \mathbf{u})^T$. It can be shown that these two are equivalent again using Petersen and Pedersen (2008): In Loader (1999) $\mathbf{M}_1$ is

$$
\begin{bmatrix} \int K(\mathbf{u})d\mathbf{u} & \int K(\mathbf{u})\mathbf{u}^T d\mathbf{u} \\ \int K(\mathbf{u})\mathbf{u}d\mathbf{u} & \int K(\mathbf{u})\mathbf{u}\mathbf{u}^T d\mathbf{u} \end{bmatrix} \tag{2.17}
$$

so $(A_{11} - A_{12}A_{22}^{-1}A_{21})^{-1}$ is $\left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right]^{-1}$.

The above calculations for the asymptotic approximation to $(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}$ are more general compared to those in most other sources, notably Ruppert and Wand (1994), since here the kernel moments are not assumed to vanish. This allows for non-symmetric kernels, as well as the handling of boundary points.

Although the original definition of influence, (2.2), only applies at the observed values $\mathbf{X}_i$, the asymptotic influence given by (2.16) can be computed at every $\mathbf{x}$. It can be seen as the influence which would be expected under idealized (asymptotic) conditions for a (hypothetical) data point situated at $\mathbf{x}$. Similarly the inequality (2.1) applies only to the observed values $\mathbf{X}_i$. However, due to the implicit averaging process happening in the computation of the asymptotic influence function, any $\mathbf{x}$ which is situated in between or close to data points $\mathbf{X}_i$ is still likely to possess the property $\text{infl}(\mathbf{x}) \leq 1$. In other words, in populated regions of the predictor space, the asymptotic influence will be less than 1, while it will exceed 1 in very sparse or remote regions. Therefore, using this asymptotic approximation, a natural choice of T is straightforwardly derived by bounding the influence by 1. This dismisses local regression at observations for which $\text{infl}(\mathbf{X}_i)$ is very large;

$$\frac{K(\mathbf{0})}{nf(\mathbf{x})|\mathbf{H}|^{1/2}}\left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right]^{-1} \leq 1$$

so

$$f(\mathbf{x}) \geq \frac{K(\mathbf{0})}{n|\mathbf{H}|^{1/2}}\left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right]^{-1}$$

so

$$T = \frac{\rho K(\mathbf{0})}{n|\mathbf{H}|^{1/2}} \tag{2.18}$$

where

$$\rho = \left[\int K(\mathbf{u})d\mathbf{u} - \int \mathbf{u}^T K(\mathbf{u})d\mathbf{u}\left(\int \mathbf{u}\mathbf{u}^T K(\mathbf{u})d\mathbf{u}\right)^{-1}\int \mathbf{u}K(\mathbf{u})d\mathbf{u}\right]^{-1}. \tag{2.19}$$

The bandwidth matrix, $\mathbf{H}$, featuring in this *density* threshold stems from an expression involving the influence of the *regression*, which explains the

earlier statement that the bandwidth matrix used for the density estimation should be the same as that used in the actual regression step.

2.1.3 Selection of ρ

Of great importance are the limits used in the integrals in ρ. These can be altered, in an effort to make the asymptotic approximation to the influence closer to the true influence in the area of interest. In order to do this, one makes reference to the boundary of the data. The boundary is defined as the entire edge of the support of $f(\mathbf{x})$, in every direction in the covariate space, such that outside the boundary $f(\mathbf{x}) = 0$. A boundary point can be thought of as a point $\mathbf{x}$ with $f(\mathbf{x}) > 0$ such that, if a kernel $K_{\mathbf{H}}$ is centred at $\mathbf{x}$, parts of the within-bandwidth region of $K_{\mathbf{H}}$ would fall into a region with $f(\mathbf{x}) = 0$; Ruppert and Wand (1994) provide a rigorous definition of boundary points. In this alteration, the upper integral limit is always ∞. If one estimates at an interior point, then the lower integral limit would be $-\infty$. For a boundary point, the lower integral limit would need to be altered according to the distance to the boundary (for instance, if $\mathbf{x}$ is half a bandwidth h_j away from the boundary of the support of f in each coordinate direction, then the lower limit of each integral would be -0.5). This is of crucial importance here since the boundary region, where data becomes sparse, is the region of interest. Hence, in order to represent the true influence as accurately as possible in the area of interest, the lower integral limit is replaced by a small negative value, say a, which reflects the distance between the boundary of f and the area for which the criterion is optimized (the integrals in (2.19) are d-variate, but the same a is always used for each co-ordinate direction here). In this way, the region in which there is doubt over the validity of local polynomial regression as a suitable regression technique can be assessed reliably.

Fig. 2.4 shows how ρ varies as a changes for trivariate covariates. This relationship is completely data-independent and suggests that a value of a between -0.5 and -1 is approximately the point at which ρ stabilises as a moves away from 0, which is one way of justifying a selection here. However,

the primary method of selection of a has been to work backwards and look directly at the data by examining the absolute error of estimated points as in Fig. 2.5. This was carried out for a variety of real and simulated data sets of varying dimension.

The following is a comprehensive list of the simulations included in this chapter. The data sets vary in their sparsity but are intended to be particularly sparse in three dimensions in an effort to simulate the problems of even higher dimensions, while maintaining computational ease.

- A- 3-dimensional covariates simulated through a t-distribution with 4 degrees of freedom centred at 7. The response values were generated according to the model $m(\mathbf{X}_i) = 12\sin(X_{i1}) - 5\sin(X_{i2}) - 3\cos(X_{i3})$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- B- 3-dimensional covariates simulated through a t-distribution with 4 degrees of freedom centred at 7. The response values were generated according to the model $m(\mathbf{X}_i) = -8\log(X_{i1}) + 5\sin(5X_{i2}) + 10\log(X_{i3})$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- C- 3-dimensional covariates simulated through a t-distribution with 4 degrees of freedom centred at 7. The response values were generated according to the model $m(\mathbf{X}_i) = 12\log(X_{i1}) - 5\sin(X_{i2}) + 10\cos(X_{i3})$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- D- 3-dimensional covariates simulated through a t-distribution with 2 degrees of freedom centred at 15. The response values were generated according to the model $m(\mathbf{X}_i) = -12\cos(X_{i1}) + 5\sin(5X_{i2}) + 10\log(X_{i3}) + 17$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- E- 5-dimensional covariates simulated through a t-distribution with 2 degrees of freedom centred at 15. The response values were generated according to the model $m(\mathbf{X}_i) = -12\cos(X_{i1}) + 5\sin(5X_{i2}) + 10\log(X_{i3}) + \cos(3X_{i4}) + 7\tan(X_{i5}) + 17$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- F- 3-dimensional covariates simulated through a t-distribution with 4 degrees of freedom centred at 7. The response values were generated

according to the model $m(\mathbf{X}_i) = \log(X_{i1})X_{i2}X_{i3}$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- G- 3-dimensional covariates simulated through a t-distribution with 4 degrees of freedom centred at 7. The response values were generated according to the model $m(\mathbf{X}_i) = X_{i1}X_{i2}\sin(5X_{i3})$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- H- 3-dimensional covariates simulated through a t-distribution with 4 degrees of freedom centred at 7. The response values were generated according to the model $m(\mathbf{X}_i) = \log(X_{i1})\sin(X_{i2})\cos(X_{i3})$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

- I- 3-dimensional covariates simulated through a t-distribution with 2 degrees of freedom centred at 15. The response values were generated according to the model $m(\mathbf{X}_i) = \cos(2X_{i1})\sin(5X_{i2})\log(X_{i3}) + 17$ and $\epsilon_i \sim N(0,1), i = 1, ..., 300$.

Figs. 2.5-2.7 demonstrate typical results and show how suitable $f(\mathbf{x})$ is as a quantity on which to apply a threshold. Fig. 2.5 shows the absolute error, $|m(\mathbf{X}_i) - \hat{m}(\mathbf{X}_i)|$, against $\hat{f}(\mathbf{X}_i)$ for simulation D and the vertical line in this figure shows approximately where the threshold should cut, in order that the extreme errors, associated with lower density, are not considered. The figures show that the points at which large errors occur can always be excluded, via the threshold, by choosing a particular a, and so ρ. In Fig. 2.5 the vertical line represents T with $a = -0.85$ and consistently this value of a performed well in these analyses, regardless of dimension. It should be noted that although a remains constant, ρ varies depending on the dimension.

Figs. 2.6-2.7 examine the MSE of the points in a data set which are accepted by the threshold using different values of ρ. In Fig. 2.6 the results using simulation D are displayed and the results for simulation E are shown in Fig. 2.7. A value of $a = -0.85$ gives $\rho = 3.12$ and $\rho = 6.1$ respectively. In both of these cases the curves seem to flatten at approximately these values of ρ, again suggesting a good choice of a, and a threshold successful in eliminating large errors. Any further increase in T seems pointless.

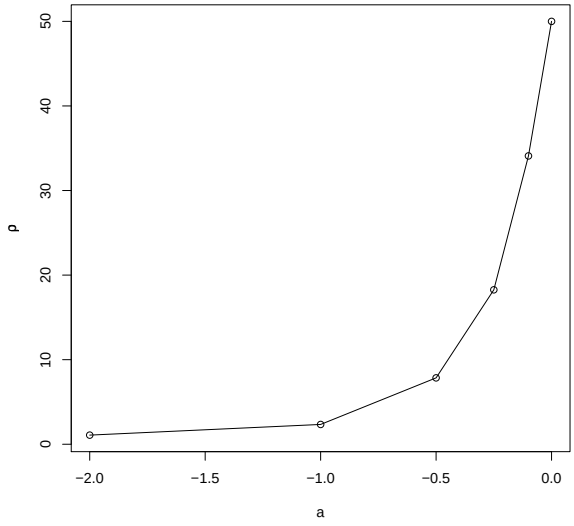

Figure 2.4: ρ v. the integral limit, a, for trivariate data (data independent).

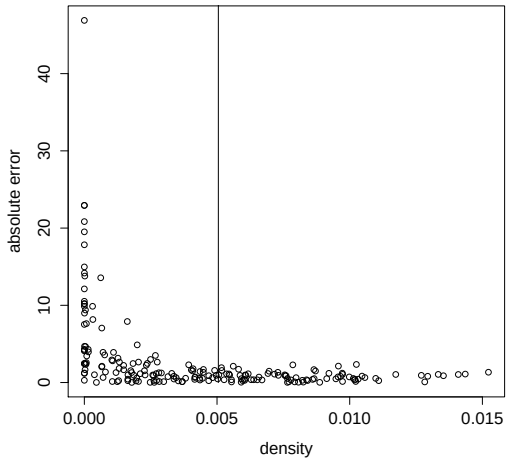

Figure 2.5: $|m(\mathbf{X}_i) - \hat{m}(\mathbf{X}_i)|$ v. $\hat{f}(\mathbf{X}_i)$ for simulation D. The vertical line represents the density at which T, with $a = -0.85$, cuts.

There is no theoretical argument which will show exactly where the threshold should cut. The most important aim of the threshold is to rule out extreme estimates. These are estimates at which either estimation breaks down computationally, or where $|m(\mathbf{X}_i) - \hat{m}(\mathbf{X}_i)|$ is very large when considering the magnitude of the response values. If this is achieved, at any point accepted by the threshold, smoothing can be performed reasonably, with only a small error. The various analyses carried out here do suggest that by making $a = -0.85$, this threshold is capable of succeeding in these ways. This value corresponds to a point situated $0.85h_j$ inside the boundary. This is quite intuitive as this is just about the region where one would assume that data sparsity becomes a problem. At all points at which R returns a computational instability or a *NaN* value as the local linear estimate, in all simulations, the density is lower than the threshold, and so rightly smoothing is considered inappropriate. The threshold also falls in the period earlier described as the "period of normal estimation" as you leave the data range. This is desirable in cutting out all the points where extreme boundary effects occur. Since, according to Hastie and Loader (1993b), in higher dimensions much of the data range suffers from boundary effects, it is reassuring that the points at which issues arise at the boundaries in these univariate examples are not considered suitable by the threshold.

2.1.4 An attempt to justify the use of asymptotics

Asymptotics play a crucial role here in relating density to a bound on reliable smoothing. To check that the use of the asymptotic approximation to the influence is justified, a small simulation study was carried out. Using asymptotics it was ascertained that

$$\text{infl}(\mathbf{x}) = |\mathbf{H}|^{-1/2}\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{e}_1 K(\mathbf{0}) \approx \frac{\rho K(\mathbf{0})}{nf(\mathbf{x})|\mathbf{H}|^{1/2}} \tag{2.20}$$

Re-arranging this, suggests that

$$\rho \approx nf(\mathbf{x})\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{e}_1 \tag{2.21}$$

Using simulated data set A, the value in the right hand side of (2.21) was calculated for a grid of $\mathbf{x}$-values over the part of the cloud where data was

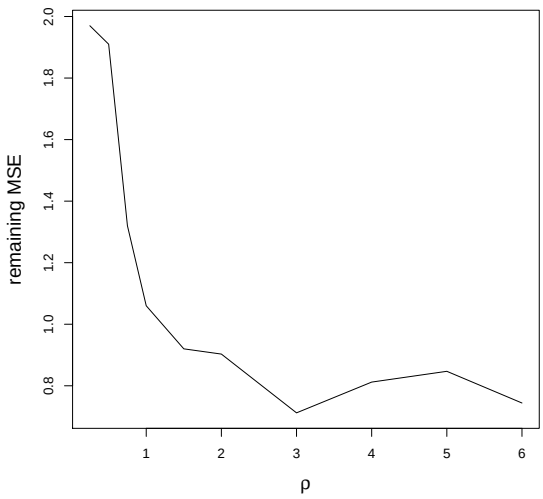

Figure 2.6: The MSE of the points in simulation D which are accepted by the threshold using different values of ρ.

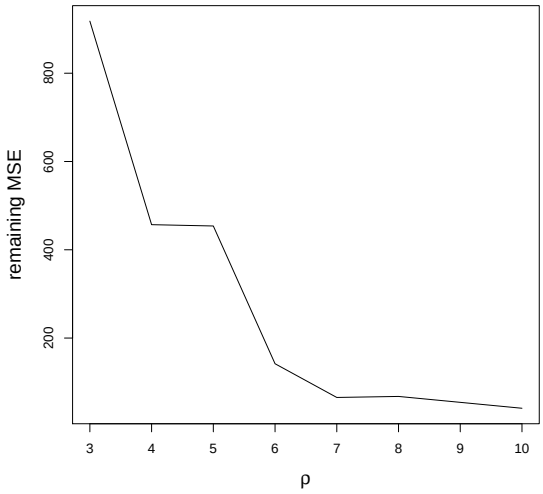

Figure 2.7: The MSE of the points in simulation E which are accepted by the threshold using different values of ρ.

not considered to be sparse visually. For this data set, the median of all the values calculated over this grid was determined as 2.54. This is very similar to the 3.12 exact value of ρ for trivariate data, and this justifies, at least to some degree, the use of asymptotics and the validity of (2.20). Although not a perfect justification, it suggests that the asymptotic approximation to the influence gives values of at least the right magnitude.

2.2 Performance of the density threshold

To fully analyse the success of this idea it is necessary to compare it to other available techniques used to smooth high dimensional data. Two such techniques are thin plate splines and additive models. Several data sets were simulated to test these different methods. Data clouds such as simulations A-I, which are denser in the middle, and gradually become sparser as you move away from the centre, are ideal for testing the value of a threshold since they provide the perfect mix of points of varying density in one data set.

For each data set a further 200 test data points (300 for E) were generated, with no error applied to the generated response values. The density of the training data was measured at each test data point, and the density threshold applied at each point individually. Tables 2.1-2.3 record the RSS of the estimates at these points, firstly for all 200, and secondly for only those points accepted by the threshold, using different methods of estimation. LP represents local polynomial, TPS thin plate splines, and AM additive models. To give each method an equal chance of success, optimal bandwidth parameters suggested by the respective R packages were used. The tables also show the number of points that fall below the threshold.

Thin plate splines were computed using the **fields** package, by Furrer, Nychka and Sain (2011), on R. This is a generalization of univariate smoothing splines in higher dimensions. According to Green and Silverman (1994), some, but not all, of the attractive aspects of spline smoothing in one dimension carry over.

Additive models were computed using the **gam** package, by Hastie (2011),

Method	LP	TPS	points omitted	LP	TPS
Threshold?	No	No	-	Yes	Yes
A	459	432	58	59	52
B	284	2249	25	100	2025
C	6720	976	44	77	211
D	59708	7187	119	77	406
E	NaN	4843668	184	52915	65504

Table 2.1: The values displayed here are the RSS for the estimates for simulations A-E, using local polynomial regression and thin plate splines. These include all 200 points, and only those accepted by the threshold. The table also shows the number of points omitted by the threshold out of the 200.

on R. This fits additive models using the method of Hastie and Tibshirani (1990). The algorithm iteratively fits additive models by backfitting. In this simulation the composite univariate functions used were splines, with bandwidth parameters chosen by generalized cross-validation.

These simulations were all successful in showing that local polynomial fitting is superior in areas accepted by the density threshold.

Table 2.1 shows that without applying a density threshold, local polynomial fitting is generally worse than thin plate splines, with higher RSS values. However, when points are discriminated against using the threshold, and estimation is only carried out at the points accepted by the threshold, the local polynomial fitting generally performs better. Table 2.2 shows similar results when comparing local polynomials with additive models. However, within the group of simulations in which the data-generating mechanism is additive, shown in Table 2.3, the additive models perform better as would be expected. Despite the additive data-generating mechanism local polynomial regression yielded the lower RSS with data set E, the five-dimensional data set which is by far the sparsest data set simulated. It appears that the sparser the data set, the more evident the usefulness of the threshold is, exemplified by data sets D and I. This is likely to be because the local

Method	LP	AM	points omitted	LP	AM
Threshold?	No	No	-	Yes	Yes
F	1889	6548	70	13	480
G	25137	48419	95	299	9514
H	308	156	59	96	77
I	1308	376	105	74	194

Table 2.2: The figures displayed here are the RSS for the estimates including all 200 points, and including only those accepted by the threshold, for local polynomial regression and additive models in those simulations (F-I) where the data-generating mechanism has interaction between covariates. The table also shows the number of points omitted by the threshold out of the 200.

Method	LP	AM	points omitted	LP	AM
Threshold?	No	No	-	Yes	Yes
A	459	39	58	59	13
B	284	64	25	100	24
C	6720	228	44	77	42
D	59708	2365	119	77	18
E	NaN	927753	184	52915	58047

Table 2.3: The figures displayed here are the RSS for the estimates including all 200 points, and including only those accepted by the threshold, for local polynomial regression and additive models in those simulations (A-E) where the data-generating mechanism is additive. The table also shows the number of points omitted by the threshold out of the 200.

polynomial estimator behaves very poorly, and can give extreme estimates, in sparser areas.

In most of the simulations, at least half of the data points are accepted by the threshold. This seems like a reasonable proportion, and makes it worthwhile applying the threshold, performing local linear regression, and benefiting from its advantages at the accepted points. The large number of points omitted from simulation E also represents a successful result for the threshold. Simulation E is a five dimensional data set, simulated through a t-distribution with 2 degrees of freedom. With this level of sparsity, and with these values of n and d, it seems likely that local polynomial regression is inappropriate, and the number of points omitted indicates that the threshold is a competent measure of this. However, for the 16 points accepted by the threshold, the local polynomial regression outperforms additive models and thin plate splines and so may still be useful if estimation at these points is of interest.

2.3 Discussion

The difficult and fundamental decision to make when designing the density threshold is just how dense must the data be to perform smoothing adequately. This choice is made here through the selection of the lower integral limit, a, in ρ i.e. by defining the area of interest to be $a = 0.85$ bandwidths away from the edge of the data cloud in each dimension. This is, in the author's opinion, justified both by the testing carried out for this value of a, and the feeling that this is approximately the region in which one would expect data sparsity to be becoming an issue. In any potential threshold developed there would always have to be a decision of this nature to be made, and the feeling is that in this case the theoretical justification, via the asymptotic approximation of the influence function, is good. The threshold formula developed, $T = \rho K(\mathbf{0})/n|\mathbf{H}|^{1/2}$, is neat in the sense that it takes the form of a multiple of the density of one point, where ρ is the multiple, so for example a value of $\rho = 3$ would represent a threshold that only allowed estimation at points at which there was a density equivalent to 3 data points

at that point. The threshold is effectively imposing a required equivalent number of data points at the point. The interpretability of this threshold is another advantage. Alternatively, one could argue that, for sufficient local estimation of a hyperplane with $p = d+1$ parameters, one needs effectively p pieces of information in the neighbourhood of $\mathbf{x}$. This could be achieved by having p observations situated exactly at $\mathbf{x}$, or, realistically, having a larger number of observations in the vicinity of $\mathbf{x}$ which essentially contribute the same amount of information. A threshold of this type would have the form

$$T_0 = \frac{(d+1)K(\mathbf{0})}{n|\mathbf{H}|^{1/2}}.$$

In practise, a threshold of this magnitude works well in lower dimensions and could work as an effective rule of thumb. However, it does not increase dramatically enough in higher dimensions, as is shown in Table 2.4. The values in this table are data-independent; so the table can be used for general reference.

Testing has suggested that if one uses T_0 for 16-dimensional data, then many points unsuitable for local regression are accepted by the threshold. The density threshold (2.18) is satisfactory in the way it adapts automatically to higher dimensions by becoming significantly larger. This is illustrated below using a real data set. This data set contains variables concerning 12000 chamois, which is a species of goat-antelope. The response is body mass and the 8 covariates are various climate variables, age and elevation. This can be used as training data while a further 2000 points can act as test data for which body mass can be estimated and compared with the observed values. The regression function $\hat{m}(\mathbf{x})$ is estimated at all 2000 test points using local linear regression for eight-dimensional covariates. The $h_j, j = 1, ..., 8$, are taken as the data range in each direction divided by 15, since bandwidth selection using a criterion is too time-consuming in 8 dimensions. The density at each point is also measured using kernel density estimation and T and T_0 are calculated to determine which points are accepted by the threshold developed from the asymptotic influence function, as well as the cruder version detailed above. For this data $T = 0.000000193$ which classifies 273 out of 2000 points acceptable for local polynomial re-

Dimension	p	ρ
1	2	1.50
2	3	2.19
3	4	3.12
4	5	4.46
5	6	6.10
6	7	8.35
7	8	11.22
8	9	15.34
9	10	20.41
10	11	27.82
11	12	36.94
12	13	51.13
13	14	68.72
14	15	88.72
15	16	110.49
16	17	147.30

Table 2.4: Comparing the number of parameters in the regression, p, with the corresponding value of ρ for $d = 1, ..., 16$ (data independent).

gression. Using $\rho = d + 1$, $T_0 = 0.000000113$ which classifies 599 points acceptable for smoothing. Figures 2.8-2.10 show the difference between the estimated values and the actual values, all plotted against density, for all 2000 points (Fig. 2.8), and for just those points accepted by the thresholds. The necessity for a threshold is highlighted in Fig. 2.8. The range of the body masses is approximately 40, and so some of the errors exhibited at points at which the density is lowest are clearly unacceptable products of the local regression. Fig 2.9 shows that, as expected for such a high dimension, some of the larger errors are still accepted by a threshold of the form of T_0 (the equivalent of $\rho = d + 1$ in T). The threshold developed in this thesis, with $\rho = 15.34$, only allows points at which the estimate is excellent as shown in Fig. 2.10. This exemplifies the need for a threshold

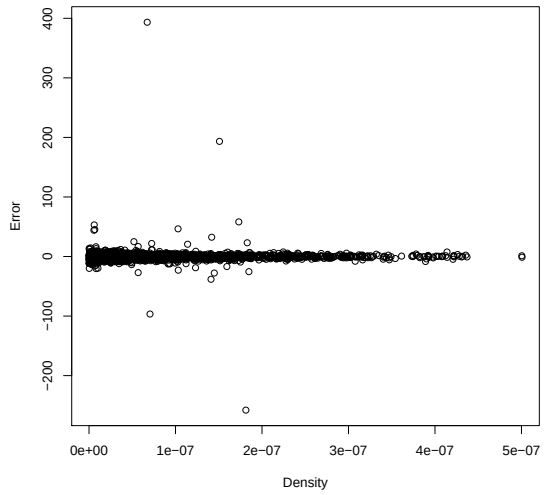

Figure 2.8: $\hat{m}(\mathbf{X}_i) - Y_i$ v. $\hat{f}(\mathbf{X}_i)$ for all 2000 test data points in the chamois data.

which increases substantially in higher dimensions.

Another related idea, with the aim of assessing data points, would be to restrict estimation to only those points which are less than th_j, where t is a constant, away from any $\mathbf{X}_i$. Whilst appealing due to its simplicity, a value of t must still be chosen. In the threshold (2.18) a similar process was carried out via asymptotic considerations, related to the density, and vigorous testing with data, to produce the values in Table 2.4. There seems to be no obvious theoretical path by which to determine a suitable value of t. These two ideas are similar, since a point which is th_j away from $\mathbf{x}$ causes a minimum density at $\mathbf{x}$, which is then effectively the minimum density that is considered sufficient for estimation to be considered reliable. However, this simpler concept has a further disadvantage when compared with (2.18), since, for any value of t, it would allow estimation at a point at which there was just one isolated $\mathbf{X}_i$ nearby, which, as has been shown in this chapter, is insufficient for data of any dimension.

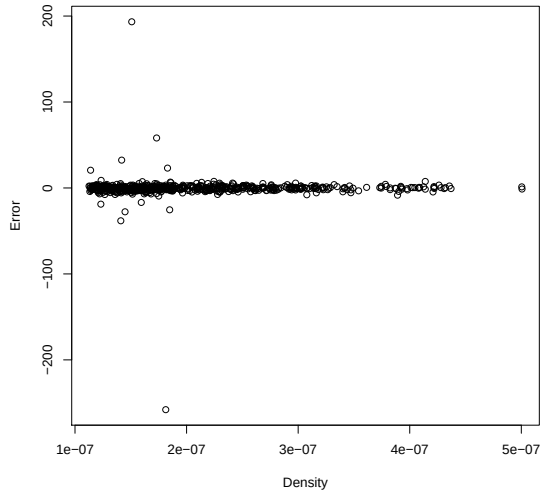

Figure 2.9: $\hat{m}(\mathbf{X}_i) - Y_i$ v. $\hat{f}(\mathbf{X}_i)$ for only those points, in the chamois test data, accepted by a similar threshold, T_0 (T with $\rho = d + 1$).

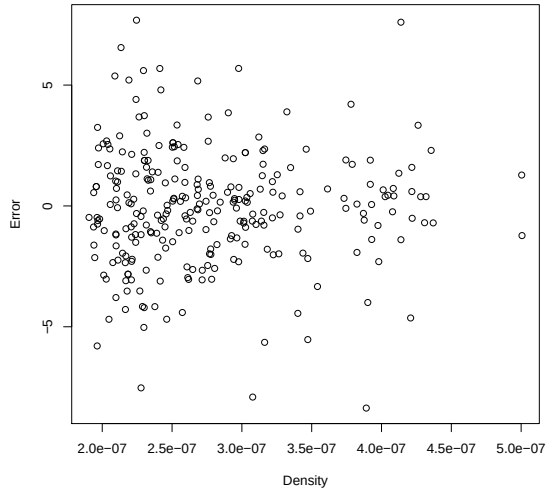

Figure 2.10: $\hat{m}(\mathbf{X}_i) - Y_i$ v. $\hat{f}(\mathbf{X}_i)$ for only those points, in the chamois test data, accepted by the threshold developed in this thesis, (2.18).

A different angle from which one might consider approaching this problem is through analysis of the standard error of the estimate, which is after all a measure of the uncertainty of the estimate. One would expect that a large standard error would be observed in regions of sparse data, indicating that the regression is not a sensible option in these areas. Unfortunately this approach is not suitable due to the curse of dimensionality. In local polynomial regression, the standard error is expressed, as in Hastie and Tibshirani (1990), as $\sqrt{\text{diag}[\mathbf{S}\mathbf{S}^T\sigma_\epsilon^2]}$, which contains the smoother matrix. The entries of the smoother matrix are affected adversely by the curse of dimensionality, and as a result the magnitude of the calculated value of the standard error may be completely different to the true magnitude of the error at that point. In other words, if an estimate at $\mathbf{x}$ is unreliable, then the standard error at $\mathbf{x}$ is also unreliable, and so no valid conclusions can be drawn. The threshold (2.18) solves this problem by determining the areas in which regression is feasible, without itself being affected by the curse of dimensionality.

When approaching a data set a decision must be made regarding the modelling strategy that will be adopted. There is a choice to be made between a simpler additive model, which lacks flexibility, but does not suffer from significant computational problems and reliability issues, and a local regression model, which gains flexibility but suffers from the curse of dimensionality and so larger uncertainty. In between, interactions can be included in additive models in order to form a compromise in this flexibility reliability trade-off. Whilst this thesis favours the local regression end of this spectrum, additive models (with and without interactions) should certainly not be dismissed since the reliability of a model is always important. An additional advantage when using additive models is that one can also gain insight into the individual effects of covariates. The threshold attempts to classify regions in which local linear regression is reliable. This then separates the data space into regions in which local linear regression should not be attempted, and so the use of additive models (using all the data) is advised, and regions in which local linear regression can be considered reliable. In these "reliable" regions the curse of dimensionality is not deemed to have

a significant effect and so the unreliability, which is the major disadvantage of this technique, is reduced. As a result, the option of a flexible model, possessing the advantages of local polynomial regression, as well as a certain amount of reliability, is provided. While this is not an option everywhere in the data space, it is certainly a competitive alternative to the additive model in some areas, as was demonstrated in section 2.2.

Of equal importance in devising a way of performing local polynomial fitting in higher dimensions, is the bandwidth matrix selection. To fully analyse the success of the density threshold it is necessary to be able to evaluate smoothing estimates fitted using optimal bandwidth values, otherwise observed large errors could be as a result of poor bandwidth selection rather than the curse of dimensionality. Therefore to fully test the threshold, AGCV was developed and is explained in Chapter 3. This method is used throughout this chapter unless otherwise stated. This bandwidth selection technique and the density threshold are designed to work together, since AGCV focuses specifically on the denser areas of the data. When used together, a powerful local regression tool in higher dimensions is realised.

It should be noted that all of the above analysis was carried out using the Gaussian kernel. However, the threshold is easily adapted to different choices of kernel function. In limited testing using the Epanechnikov kernel function, the threshold proved very capable of excluding all points where estimates were sufficiently poor. In theory, the threshold can also easily be extended to polynomials of different degree, but little work has been done with this aim due to the advantages of local linear regression in terms of bias and variance. As explained in Chapter 1, it could be beneficial to employ a local constant version of the threshold in areas where local linear regression is not considered reliable. Indeed, this threshold is implemented in Chapter 4.

When using (1.53) to calculate the density at a point, to examine using the threshold developed in this chapter, it is necessary to use the same $\mathbf{H}$ as will be used for the regression. As a result, certain quantities, usually only considered in the regression bandwidth selection procedure will affect

the bandwidth used in the density estimation, and so the density estimate on which the threshold is applied. It is interesting to briefly compare the optimal bandwidths for local linear regression and density estimation, in the simpler univariate case, in order to identify these quantities. Using the MSE, the asymptotically optimal bandwidth for density estimation is derived earlier and given as (1.64). The equivalent for local polynomial regression, following from (1.24)-(1.26), is

$$h_{opt} = \left[\frac{\sigma_\epsilon^2 \int [\kappa(u)]^2 du}{n(\int u^2 \kappa(u) du)^2 \int [m''(x)]^2 f(x) dx} \right]^{1/5} \tag{2.22}$$

where σ_ϵ^2 is the error variance of the regression at each X_i, assuming homoscedasticity (Simonoff, 1996). It is apparent that, in the selection of the regression bandwidths, in which one searches for a bandwidth as close as possible to the optimal (2.22), one is implicitly taking into account $m(x)$, the true mean function, and σ_ϵ^2. As a result, by using the same $\mathbf{H}$ in the density estimation, the same quantities are involved in the density estimation procedure despite having no association with the density. A similar issue would occur in the multivariate setting. However, this is unimportant in this context since an optimal density bandwidth is not the priority here, but rather a bandwidth which works with the threshold.

Interestingly, in the special case when

$$\int [m''(x)]^2 f(x) dx = \sigma_\epsilon^2 \int [f''(x)]^2 dx \tag{2.23}$$

is satisfied, the optimal bandwidths for regression and density estimation are equal. In this case, any bandwidth selection procedure for local polynomial regression, which seeks to approximate (2.22), will also, when the resulting bandwidth is applied in the density estimation, produce the optimal density estimate.

Chapter 3

Bandwidth matrix selection

The curse of dimensionality also causes problems in the area of bandwidth matrix selection. As was mentioned earlier, the use of a classical rather than a plug-in method is favoured in this thesis, due to the reliance on asymptotics of the latter. This was less of an issue in the previous chapter, where asymptotics were solely used to find an approximation of the influence function, but it is an issue here as the goal is now bandwidth selection itself. One such classical method, introduced in Chapter 1, is generalized cross-validation which is less precise than other cross-validation, but computationally less demanding. When implemented on R, GCV struggles greatly to cope with high dimensional data. GCV is a minimization problem, and it is the actual minimization which causes problems. A GCV value can easily be calculated for any $\mathbf{H}$ using (1.43) but a variety of issues arise through the minimization over d-dimensional data, carried out on R by the `optim` function (found in the base package). Often, extreme values will be suggested for h_j, significantly larger than even the data range. Alternatively R just returns an error message. Within the `optim` function, one must specify a starting point, in this case a starting set of h_j from which the Nelder-Mead algorithm, detailed in Nelder and Mead (1965), can start the minimization. Often this process is very sensitive to the starting point, and different parameters are suggested depending on the starting point. This is not a problem with GCV itself, but rather a problem of `optim` selecting

one of many minima, which sometimes occur in the GCV function, without it being necessarily the smallest as desired. Even if these problems are avoided and a selection which appears reasonable is made, often the chosen bandwidth matrix performs poorly and is consequently responsible for poor local polynomial regression.

3.1 AGCV

3.1.1 Adaptations

Here the original GCV, developed by Craven and Wahba (1979), has been adapted in two ways in order to alleviate the problems mentioned above. Both of these steps are taken to remove the influence of data points in less dense areas which otherwise may have a disproportionate effect on the procedure. This effect is more likely to occur as d increases.

Firstly, it is proposed that the median of the diagonal elements of the smoother matrix, $\mathbf{S}$, is used in the place of $\frac{\text{trace}(\mathbf{S})}{n}$ (effectively the mean of the diagonal elements). Denote the median of the diagonal elements of the smoother matrix as ψ. The introduction of the median eradicates the possibility of extremely large values of h_j being chosen. This is best shown through an example using the simulated data set E, detailed in section 2.1.3. GCV was carried out on this data set in order to select an optimal bandwidth matrix. Both the original GCV, (1.43), and the original GCV with $\frac{\text{trace}(\mathbf{S})}{n}$ replaced by ψ were used. The unaltered GCV selected extreme h_j values, while the altered one selected h_j values of a reasonable magnitude, as desired. Table 3.1 helps to show the cause of this, by displaying the impact on different parts of the GCV formula when different magnitudes of h_j are entered.

This demonstrates that the denominator of the altered GCV is relatively unaffected by the size of the bandwidths chosen here. In fact, this alters significantly only for very small values of h_j. In contrast, in the range of bandwidths tested here, $1 - \frac{\text{trace}(\mathbf{S})}{n}$ varies significantly, depending on the h_j. In the GCV minimization process, the larger the denominator the

	extremely large h_j	small h_j
$\frac{trace(\mathbf{S})}{n}$	0.02	0.108
ψ	0.01	0.038
$1 - \frac{trace(\mathbf{S})}{n}$	0.98	0.892
$1 - \psi$	0.99	0.962

Table 3.1: Comparing components of the GCV, with and without ψ, for data set E, with different sizes of h_j.

smaller the GCV becomes. In the case of the altered GCV, in which the denominator remains relatively unchanged regardless of h_j, the numerator can, rightly, have an influence in choosing the optimal parameters through the minimization. However, in the original GCV, there is a significant difference in possible denominators, depending on h_j, and if this is significantly greater than the difference in the numerator (between favourable small h_j and extreme h_j), then the GCV will be minimized by extreme h_j values, without the numerator having any significant input. This example shows how at times the unaltered GCV can select extreme h_j.

The value of 0.108 recorded in Table 3.1 is caused by some extreme (close to 1) influence values, which indicates that it is the curse of dimensionality causing this issue. The extreme nature of these points is ignored when the median is used as in the altered version of GCV. By including the median, small bandwidth values, which contribute to an increase in influence values, are penalized less harshly, and so bandwidths of a reasonable magnitude can be chosen. By softening this penalization slightly, extreme values of h_j can never enjoy the advantage they possess in the denominator in the original GCV, and their poor performance in the numerator will correctly see them discounted as possible bandwidths.

Figs. 3.1-2 show graphically the effect of using the median. Both of these display the GCV value calculated over a grid of values for a simulated bivariate data set. A t-distribution with 1.3 degrees of freedom was used to create some very sparse areas of data, since the effect being demonstrated here would not usually occur as frequently in such a low dimension. Fig. 3.1

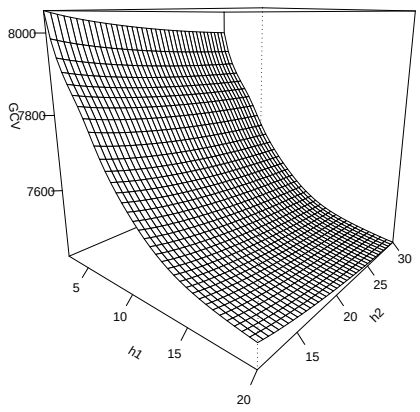

Figure 3.1: GCV function for a very sparse simulated bivariate data set, using the unaltered version of GCV.

shows the unaltered GCV decreasing as the h_j increase, explaining why in this case the GCV minimization process chooses extremely high h_j. Fig. 3.2 shows how the alteration stops this from occurring, with a clear minimum at approximately (2.5,11).

The second adaptation proposed to GCV is the removal of isolated points from the process. In this setting, an isolated point is one at which no point other than itself contributes to its local regression estimate. Whether a point is isolated or not depends on the bandwidth matrix selected, but there are some points which will always be isolated for any reasonable $\mathbf{H}$. Often an isolated point will impose a computational constraint on the minimization process. Within the expression for GCV, the diagonal elements of $\mathbf{S}$ in the denominator, and the $\hat{m}(\mathbf{X}_i)$ in the numerator, are very sensitive to h_j. On R, it is computationally impossible to compute these at an isolated point if the h_j are not sufficiently large to make the point *not isolated*. This means that within `optim` on R, only values of h_j which achieve this, and stop computational error, will be considered. In effect the isolated points

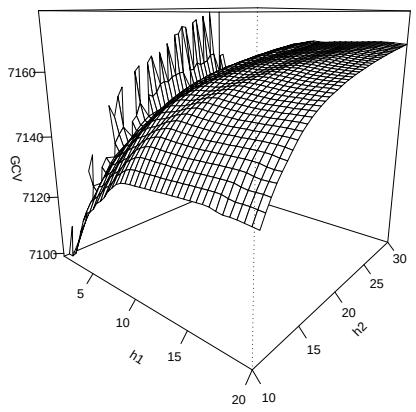

Figure 3.2: GCV function for a very sparse simulated bivariate data set, using the median within GCV.

are enforcing a minimum h_j which is in fact higher than the optimal h_j for the majority of the data. This has been observed over several trials. Silverman (1986) describes a similar effect caused by outliers in the context of likelihood cross-validation in univariate density estimation.

Applying these two adaptations to GCV, *adapted generalized cross-validation* (AGCV) is formulated, which is defined as follows;

$$AGCV(\mathbf{H}) = n^{-1} \sum_{i=1}^{n} \left\{ \frac{Y_i - \hat{m}(\mathbf{X}_i)}{1 - \psi_w} \right\}^2 w(\mathbf{X}_i) \tag{3.1}$$

where ψ_w is the median of the diagonal elements of the smoother matrix, $\mathbf{S}$, after excluding the elements contributed by the points for which $w(\mathbf{X}_i) = 0$. Set $w(\mathbf{X}_i) = 1$ for all i except the r points at which $f(\mathbf{X}_i)$ are smallest, at which it is 0. Set r as the number of points which could be considered isolated i.e. where the density at that point is equal to the density of just one data point. This is examined using kernel density estimation with Epanechnikov kernels. The bandwidth parameters to be used in the density estimation here should be the optimal values for density estimation,

calculated from an external source such as the **np** package by Hayfield and Racine (2008) in R.

Here the effects of the isolated points are avoided by excluding these points from both the numerator, via w, and the median in the denominator. In a simple example showing the effect of these points, consider a simulated five-dimensional data set, of size $n = 300$, simulated through a t-distribution with 1 degree of freedom. The response values were generated according to the model $m(\mathbf{X}_i) = -12\cos(X_{i1})+5\sin(5X_{i2})+10\sin(0.01X_{i3})+\cos(3X_{i4})+7\tan(X_{i5}) + 17$ and $\epsilon_i \sim N(0, 1)$. The altered GCV containing the median, but without the isolated points removed, is minimized by h_j values of (21.1, 3.45, 11.1, 0.8, 50.9), and here it is impossible for `optim` to select h_1 smaller than 20, and h_5 smaller than 50, due to the restrictions mentioned above caused by the points in less dense areas. If the 100 data points at which the density is smallest are removed from the procedure, equivalent to taking $r = 100$ in AGCV, then the AGCV can be minimized at $h_j = (2.5, 4.5, 2.4, 0.4, 1.6)$, parameters of a more reasonable size, given the range of the majority of the data.

3.1.2 Choice of r

Removing points is both a matter of removing any computational constraint imposed by points in sparser regions, and also fine-tuning by focussing on the denser regions of data, which are of interest. As discussed in Chapter 2, local polynomial regression is only possible in regions where there is sufficient data. Any points excluded from AGCV should be outside these regions. In this way AGCV is tailored towards finding optimal h_j for the areas accepted by the density threshold, (2.18). Choosing r is effectively choosing a *pilot region* in which local polynomial regression is considered feasible, before (2.18) defines a more accurate region. In practice r is the number of points for which the density at that point is equal to the density of just one data point. This means that r is sufficiently large to remove any points that impose a computational constraint in R, as mentioned above.

It is however possible to choose a larger value of r than this (as in the

illustrative example above), and such values will provide h_j values optimal for only denser parts of the data range. An r which includes just the points accepted by (2.18) would be ideal since the implementation of this would then provide the best regression estimates at those points. However, since at the bandwidth selection stage it is not known where the threshold will deem that local polynomial regression is reliable, since (2.18) depends on the h_j selected, an optimal r cannot be chosen. Thus the choice of r specified above acts as a useful rule of thumb. Another positive feature of this choice of r is that it usually leaves a reasonable number of points to be included in the GCV minimization, which can break down if too few points remain.

It would be neat to apply weighting with $w(\mathbf{X}_i) > 0$ to all n points, for example equivalent to $f(\mathbf{X}_i)$. However, this is not possible since this would still require $\hat{m}(\mathbf{X}_i)$ to be calculated for all points, including isolated ones, which would apply a restriction on the h_j selected by R, as mentioned above. For this reason it is preferable to remove these points completely from the process.

Epanechnikov kernels are used to calculate $\hat{f}(\mathbf{X}_i)$ for determining r because this results in less ambiguity concerning what can be considered an isolated point, compared with, for example, a Gaussian kernel.

3.1.3 Starting point selection

Since `optim` is used for this minimization, it is necessary to choose a starting point. This can be chosen automatically, but a successful minimization is more likely if this point is chosen with more care. The presence of more than one minimum is common, and makes the selection of the overall minimum more difficult. There is no way of guaranteeing that the overall minimum is selected, but chances of this are increased if the starting point is close to this minimum. From practical experience it is observed that a starting point smaller than the actual minimum is often more successful, but this is not justified theoretically here. It is often helpful to perform the minimization more than once using different starting values each time. These steps increase the reliability of the method, but due to the nature of `optim`,

the selection of the overall minimum can never be guaranteed. Hayfield and Racine (2008) also discuss the necessity to try several starting points, in order to adjust to the presence of local minima, in the bandwidth selection methods in the **np** package.

3.1.4 AGCV as a measure of error

Cross-validation is a measure of error to be minimized and is an improvement on the *average squared residual* which is inadequate since it is minimized by an interpolation of the data,

$$ASR(\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n}\left\{Y_i - \hat{m}(\mathbf{X}_i)\right\}^2. \tag{3.2}$$

Cross-validation can be expressed, as given earlier, as

$$CV(\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{Y_i - \hat{m}(\mathbf{X}_i)}{1 - S_{ii}}\right\}^2. \tag{3.3}$$

As explained earlier, Craven and Wahba (1979) introduced GCV, which is a computationally less costly version where the S_{ii} is replaced by the average $\frac{S_{ii}}{n}$.

$$GCV(\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{Y_i - \hat{m}(\mathbf{X}_i)}{1 - \frac{1}{n}\sum_{j=1}^{n} S_{jj}}\right\}^2 = ASR(\mathbf{H})\left(1 - \frac{1}{n}\sum_{i=1}^{n} S_{ii}\right)^{-2}. \tag{3.4}$$

As shown, this is the average squared residual, corrected by a factor. This is shown in Craven and Wahba (1979) as being effective in finding an estimate of the smoothing parameter which minimizes the mean squared error. Now

$$AGCV(\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{Y_i - \hat{m}(\mathbf{X}_i)}{1 - \psi_w}\right\}^2 w(\mathbf{X}_i) = AWSR(\mathbf{H})(1 - \psi_w)^{-2} \tag{3.5}$$

with the *average of weighted squared residuals*,

$$AWSR(\mathbf{H}) = \frac{1}{n}\sum_{i=1}^{n}\left\{Y_i - \hat{m}(\mathbf{X}_i)\right\}^2 w(\mathbf{X}_i). \tag{3.6}$$

So AGCV is the average of weighted squared residuals, corrected by a factor. The factors used in (3.4) and (3.5) perform exactly the same function. They

both calculate an average over the S_{ii} and subtract it from 1. The factor used in the AGCV is simply more robust, as explained previously. The only other difference between the GCV and the AGCV is that the AGCV approximates the average *weighted* squared residual rather than the *unweighted*, as GCV does. Again, this is used to make the procedure more robust.

AGCV can be justified as a legitimate approximation to the mean squared error since it works in exactly the same way as GCV, but in a more robust manner.

3.1.5 Simulation study

A rigorous simulation was carried out to measure the performance of AGCV against other bandwidth selection tools for multivariate data. Two trivariate data sets were generated with this purpose.

- P- 3-dimensional covariates simulated through a t-distribution with 5 degrees of freedom. The response values were generated according to the model $m(\mathbf{X}_i) = -12\cos(X_{i1}) + 5\sin(5X_{i2}) + 10\sin(X_{i3})$ and $\epsilon_i \sim N(0,1), i = 1, ..., 250$.

- Q- 3-dimensional covariates simulated through a t-distribution with 1.5 degrees of freedom. The response values were generated according to the model $m(\mathbf{X}_i) = -12\cos(X_{i1}) + 5\sin(5X_{i2}) + 10\sin(X_{i3})$ and $\epsilon_i \sim N(0,3), i = 1, ..., 250$.

The only difference between the two data sets is that Q contains much sparser regions of data.

Each of these data sets was simulated 100 times and the optimal smoothing parameters were calculated using four different methods; AGCV, GCV, LSCV (the default method in the **np** package) and GCV for thin plate splines (calculated using the **fields** package). The MSE was then calculated for estimates using each set of smoothing parameters. The MSE was calculated both including all 250 points and for just the densest 50 percent of each data set. The density was measured using kernel density estimation tools in the **np** package.

As mentioned in the discussion in Chapter 2, this technique is suited for use alongside the threshold, (2.18), since it is tailored towards use in the denser areas of the data. One cannot however use T here to compare procedures since different methods, selecting different h_j, would select different numbers of points with density higher than T, and so no fair comparison could be made between methods. For this reason, the same densest fifty percent of points were compared for all methods, ensuring fairness, whilst keeping in mind the philosophy of this thesis that local polynomial regression is only advisable in some regions of the data range. This is not ideal, in accordance with the threshold, but fairness is essential.

Other steps were taken to ensure fairness. AGCV, GCV and the method in the **np** package are all very dependent on the starting point selected by the user. Due to the computational time associated with bandwidth selection for each simulation, a maximum of 3 starting points was chosen for each method each time. These were chosen carefully to give each method the best chance of finding the optimal bandwidth parameters.

Analysis

AGCV consistently outperforms the other techniques, yielding a smaller median MSE. With the less sparse data, P, shown in Fig. 3.3, the AGCV and GCV perform best, with the AGCV performing better for the densest 50 percent as expected. The **np** and thin plate spline methods have larger MSEs as well as larger interquartile ranges. With the sparser data, Q, shown in Fig. 3.4, the AGCV and thin plate splines are the only techniques whose MSEs could be considered of a reasonable size given the magnitude of the response values. Among these, AGCV is marginally better with a slightly smaller median, which again improves when only including the densest 50 percent of the data. The GCV and the **np** least squares cross-validation both perform extremely poorly on this sparser data. Taking into account both P and Q, the AGCV is the only technique which consistently outperforms the others.

The plots in Fig. 3.5, which are all similar in trend, show how the

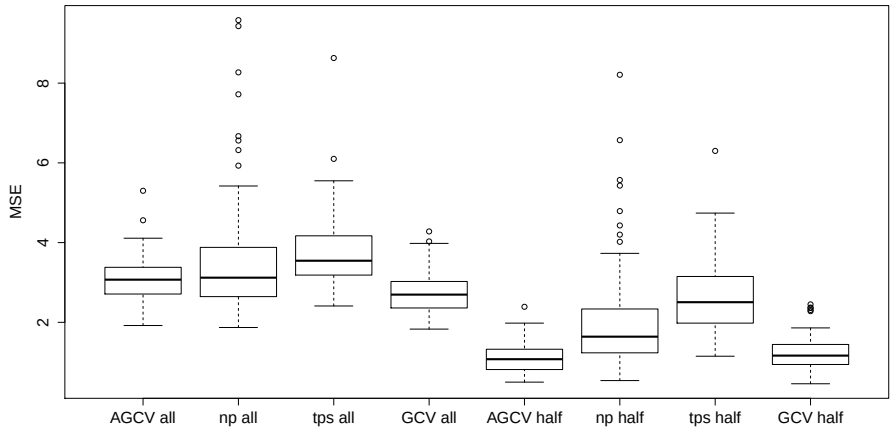

Figure 3.3: Each boxplot represents the 100 MSEs for simulation P for different bandwidth selection techniques. *all* represents the MSE of all n points, and *half* represents the MSE for the densest 50 percent.

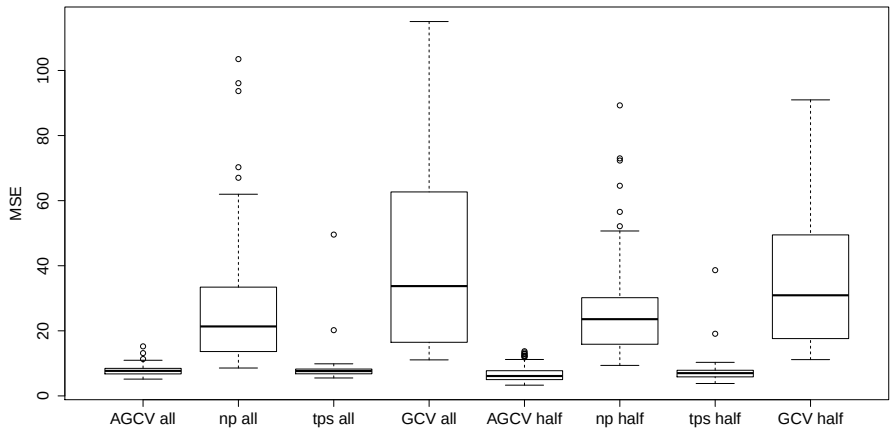

Figure 3.4: Each boxplot represents the 100 MSEs for simulation Q for different bandwidth selection techniques. *all* represents the MSE of all n points, and *half* represents the MSE for the densest 50 percent.

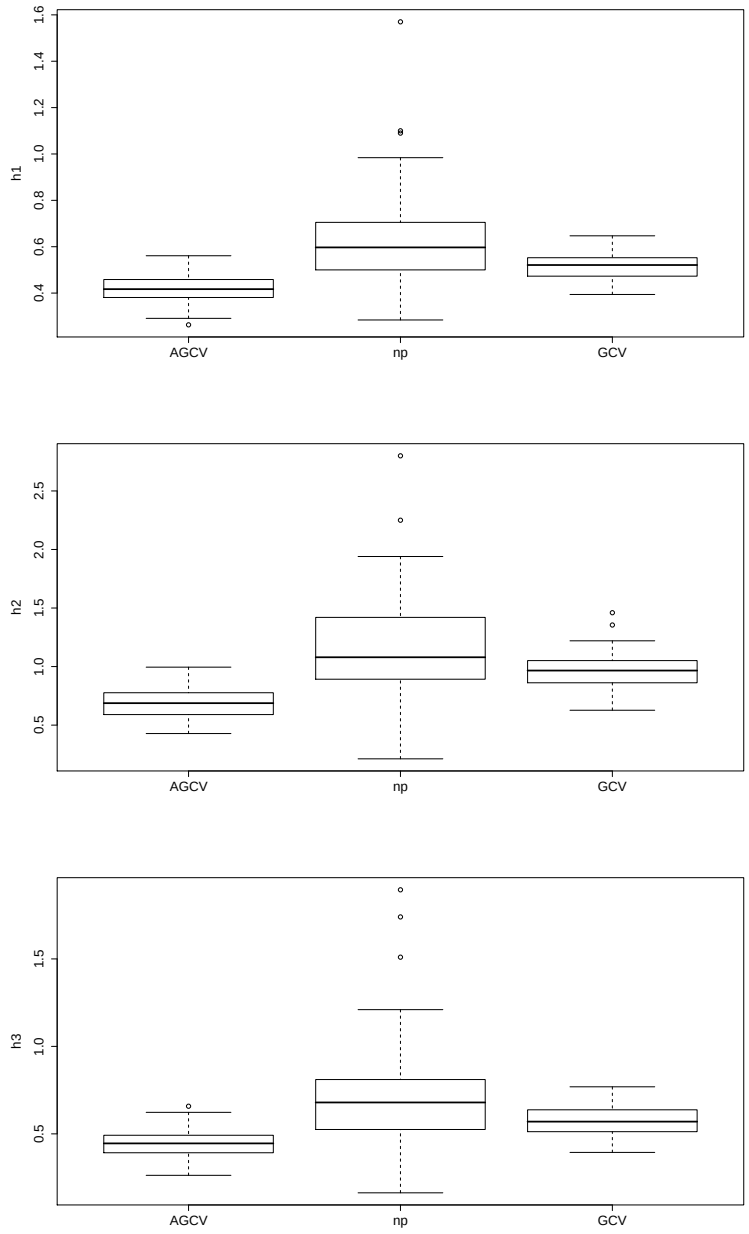

Figure 3.5: Each boxplot represents the 100 h_j values chosen by each bandwidth selection technique for simulation P. The top plot is h_1, the middle h_2 and the bottom h_3.

individual h_j values compare between different methods, for simulation P. They show how AGCV tends to pick smaller h_j values, which is likely to be due to the fact that the isolated points are ignored. This contributes to the improved regression carried out in the denser areas. These plots also reveal how relatively inconsistent the method in the **np** package is.

3.1.6 Discussion

The adaptations implemented in AGCV are effective not only in terms of providing more reliable parameter estimates and reducing the number of error messages in R, but also general performance. The minimization is much faster using AGCV when compared with GCV, and with higher dimensions this can be a significant amount of time. The dependence on the starting point, although still present, is much less of an issue with AGCV than with GCV, and so the overall minimum is much easier to find.

A thorough simulation study was carried out which demonstrates the way in which AGCV clearly outperforms competing methods. The main reason for this is that it is robust to the effects of points in sparse regions, and both of the adaptations made contribute towards this. Removing the isolated points in the way detailed is a robust enough step alone to be effective for most data sets, however the step of including the median provides extra assurance. This could be crucial since the r removed points are considered isolated *density*-wise when using bandwidth parameters chosen to be optimal in the density estimation. These bandwidth parameters define the neighbourhoods, which determine which points are considered isolated. It may be that the magnitudes of the regression bandwidth parameters are very different, and other points are isolated, in terms of the neighbourhoods defined by potential h_j, when GCV is carried out, which were not initially removed. The median importantly limits the issues that may arise as a result of this. The classification of an isolated point using density bandwidth parameters is not ideal but is the best that can be achieved at this initial stage.

The adjustments made to GCV here are made specifically in response

to problems encountered on R. The removal of isolated points in particular is to avoid the error messages encountered through a combination of sparse data and small h_j. Despite the fact that this is an adjustment developed in this way, it fits perfectly with the general solution to the curse of dimensionality expressed in this thesis, of excluding the areas of low density from consideration. The points that are ignored in AGCV are sufficiently isolated that they would never be accepted by the density threshold discussed in Chapter 2. In this way, the h_j selected by AGCV are more suited to the points accepted by the threshold, by not having to take into account other points excluded by it.

In practice in higher dimensions, smaller h_j, specifically selected for a smaller region of the data, give better estimates, for points in that region, than larger h_j, chosen for a greater area. This is particularly true when compared with the larger than normal bandwidths often chosen as a remedy for the curse of dimensionality. In high dimensions there is more space for variation in the nature of the data to occur and so h_j of different magnitudes could be suitable for different regions. This is more the case here than for the univariate equivalent, where varying bandwidths are already employed, as examined in Fan and Gijbels (1992). A variable bandwidth matrix is a potential way of adapting to this, as is already considered for kernel density estimation in Sain (2002). As mentioned in Chapter 1, multivariate variable bandwidth selection strategies do exist in local polynomial regression, such as the empirical-bias bandwidth selector and LOWESS. AGCV is similar to these in the way that suitably small h_j are selected in areas, determined by r, where the data is dense. AGCV differs by using the philosophy of the density threshold, and reducing the computational cost by incorporating *only* the points in dense areas, to be used, in conjunction with the threshold, only in these regions. In this way AGCV can be seen as a first step towards a variable bandwidth matrix, whilst avoiding the expensive computational costs associated with this.

Alternative classical bandwidth selection methods may benefit from similar adaptations to those proposed in this chapter. In the multivariate set-

ting, one such method, the Akaike information criterion (AIC), takes the form

$$AIC(\mathbf{H}) = \log\left[\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{m}_{\mathbf{H}}(\mathbf{X}_i)\right)^2\right] + \frac{2\text{trace}(\mathbf{S})}{n},$$

and is likely to suffer in a similar way to GCV. Isolated points would impose exactly the same constraints here, and the inclusion of the median could also be beneficial. Hurvich, Simonoff and Tsai (1998) propose a corrected version of the Akaike information criterion (AICc), which takes the form

$$AICc(\mathbf{H}) = \log\left[\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{m}_{\mathbf{H}}(\mathbf{X}_i)\right)^2\right] + \frac{1 + \frac{\text{trace}(\mathbf{S})}{n}}{1 - \frac{\text{trace}(\mathbf{S})+2}{n}},$$

which is also likely to suffer in a similar way to GCV. However it is possible, due to the position of trace($\mathbf{S}$) in both the numerator and the denominator that the introduction of the median is not necessary in AICc. This has not been tested, but it could be of interest to explore further.

It should be noted that on R it is practically very difficult to select h_j for d greater than 5. This is an issue of time, due to the extremely large parameter space that `optim` must search over in such high dimensions. A solution to this is to choose a constant h, selected by GCV, to be used in every entry of the diagonal bandwidth matrix, and so apply the same amount of smoothing in each covariate direction. In this case the covariates should first be standardized. This is significantly quicker to compute, but the quality of the $\hat{m}(\mathbf{X}_i)$ suffers as a result. It is also useful here to remove the most isolated points from the process for the same reason as in AGCV. This is very similar to the scaling approach, mentioned in Bowman and Azzalini (1997) and used in the context of multivariate density estimation. Alternatively, variable selection can be initially employed as discussed in Chapter 1.

Unfortunately, many bandwidth selection tools for density estimation, which are employed initially in AGCV in order to determine r, also struggle with a large value of d or n. In this situation it is useful to employ the rule of thumb for multivariate density estimation bandwidth selection of Scott (1992) at this initial stage.

3.2 Further approaches

There are many further angles from which the bandwidth selection problem can be approached. The investigation into two of these, which unfortunately are restricted in use, is described below.

3.2.1 OSCV

This section is motivated by Hart and Yi (1998), in which an alternative method, one-sided cross-validation (OSCV) is proposed for univariate local polynomial regression bandwidth selection. The aim here is to extend this method for use in the multivariate setting. Hart and Yi (1998) claim that OSCV possesses the same advantages as cross-validation, and is better statistically with a much lower bandwidth variance.

The method developed uses different types of regression estimators at the bandwidth selection and estimation stages, due to the observation, by Marron (1986) and others, that often cross-validation is more effective when applied to an inefficient regression estimator. The method described in Hart and Yi (1998) is outlined below. Consider the less efficient estimator, $\tilde{m}_b(X_i)$, with smoothing parameter b, which here is a local linear estimator using the data only on one side of the point at which estimation is taking place, for example $(X_1, Y_1), ..., (X_i, Y_i)$ where the X_i are ordered. The cross-validation for $\tilde{m}_b$, minimized by $\hat{b}$, is defined as

$$OSCV(b) = \frac{1}{n-l} \sum_{i=l+1}^{n} (\tilde{m}_b^i(X_i) - Y_i)^2 \tag{3.7}$$

where l is some small integer. This is a normal expression for cross-validation, applied to $\tilde{m}_b$. The minimizer of $OSCV(b)$ is approximately the same as that of the MASE (mean average squared error),

$$MASE^*(b) = E\left\{ \frac{1}{n-l} \sum_{i=l+1}^{n} (\tilde{m}_b^i(X_i) - m(X_i))^2 \right\}. \tag{3.8}$$

If m has two continuous derivatives, then asymptotically the minimizer of (3.8) is

$$b_n = C_{m,\sigma_\epsilon} \left[\frac{\int L(x)^2 dx}{\left[\int x^2 L(x) dx\right]^2} \right]^{\frac{1}{5}} n^{-\frac{1}{5}} \tag{3.9}$$

where L is the half-kernel used in OSCV which assigns weight only to data on the one side of the point being estimated, and C_{m,σ_ϵ} is a constant depending only on m and σ_ϵ.

Now considering the more efficient universal local polynomial regression, the mean average squared error of $\hat{m}$ is defined by

$$MASE(h) = E\left\{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{m}(X_i) - m(X_i)\right)^2\right\}. \tag{3.10}$$

The minimizer of $MASE(h)$ is asymptotic to

$$h_n = C_{m,\sigma_\epsilon}\left[\frac{\int \kappa(x)^2 dx}{\left[\int x^2\kappa(x)dx\right]^2}\right]^{\frac{1}{5}} n^{-\frac{1}{5}}. \tag{3.11}$$

So asymptotically,

$$\frac{h_n}{b_n} \to \left[\frac{\int \kappa(x)^2 dx}{\left[\int x^2\kappa(x)dx\right]^2}\frac{\left[\int x^2 L(x)dx\right]^2}{\int L(x)^2 dx}\right]^{\frac{1}{5}}. \tag{3.12}$$

This means a suitable value of h for use in the regression problem can be obtained by multiplying the $\hat{b}$ resulting from (3.7) by an adjusting constant C where

$$C = \left[\frac{\int \kappa(x)^2 dx}{\left[\int x^2\kappa(x)dx\right]^2}\frac{\left[\int x^2 L(x)dx\right]^2}{\int L(x)^2 dx}\right]^{\frac{1}{5}} \tag{3.13}$$

as confirmed in Yi (1996).

Multivariate OSCV

In order to solve the bandwidth matrix selection problems cited at the beginning of this chapter for data in higher dimensions and due to the success of this method in the univariate case, an extension to OSCV has been considered. Here the technique can be considered again *one-sided* through the choice of the inefficient initial estimator. Here, $\tilde{m}_{\mathbf{B}}(\mathbf{X}_i)$, with bandwidth matrix $\mathbf{B}$, is a local linear estimator which takes into account only the data points for which the covariate has a smaller Euclidean distance to the origin than that of the point at which estimation is taking place. This is represented in Fig 3.6. All other parts of the method are a straightforward

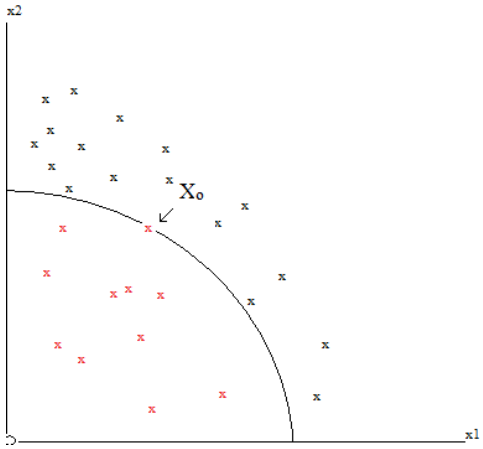

Figure 3.6: $\tilde{m}_{\mathbf{B}}(\mathbf{X}_i)$ is a local linear estimator based on the data only with a smaller Euclidean distance to the origin than that of the point at which estimation is taking place, $\mathbf{X}_0$. Here the data included is shown in red for a simple bivariate data set.

extension of those steps described above for univariate data. Here, analogously to (3.7),

$$OSCV(\mathbf{B}) = \frac{1}{n-l}\sum_{i=l+1}^{n}(\tilde{m}^i_{\mathbf{B}}(\mathbf{X}_i) - Y_i)^2. \tag{3.14}$$

It is necessary to find the multivariate equivalent to C and in order to do this it is necessary to find expressions equivalent to (3.9) and (3.11). In order to do this, the MISE is examined (the MASE is simply the empirical version of the MISE.) In order to simplify finding the optimal $\mathbf{H}$ from the asymptotic expression for the MISE, (1.18), only diagonal bandwidth matrices with $h_1 = ... = h_d = h$ are considered. This is suitable for use in finding C for OSCV, since each b_j will be multiplied by the same C regardless of the covariate direction, so the relative magnitudes of the different h_j are not important.

In condensing this into a single parameter minimization problem

$$\text{trace}\left\{\mathbf{H}\mathcal{H}_m(\mathbf{x})\right\} = h^2 \sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i}.$$

In this case

$$AMISE(\mathbf{H}) \approx \int \left((\frac{1}{2}\mu_2(K) h^2 \sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i})^2 + \frac{1}{nh^d} \int K(\mathbf{u})^2 d\mathbf{u} \frac{\sigma_\epsilon^2}{f(\mathbf{x})} \right) d\mathbf{x} \tag{3.15}$$

$$= \int \frac{1}{4}\mu_2(K)^2 h^4 \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x} + \int \frac{1}{nh^d} \int K(\mathbf{u})^2 d\mathbf{u} \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x}$$

$$= \frac{1}{4} h^4 \mu_2(K)^2 \int \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x} + \frac{1}{nh^d} \int K(\mathbf{u})^2 d\mathbf{u} \int \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x}$$

Minimization is performed in the usual way, by differentiating with respect to h, and equating the result to 0. Differentiating with respect to h yields

$$h^3 \mu_2(K)^2 \int \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x} - \frac{d}{nh^{d+1}} \int K(\mathbf{u})^2 d\mathbf{u} \int \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x} = 0$$

$$h^3 \mu_2(K)^2 \int \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x} = \frac{d}{nh^{d+1}} \int K(\mathbf{u})^2 d\mathbf{u} \int \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x}$$

$$h^{d+4} \mu_2(K)^2 \int \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x} = \frac{d}{n} \int K(\mathbf{u})^2 d\mathbf{u} \int \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x}$$

$$h^{d+4} = \frac{d \int K(\mathbf{u})^2 d\mathbf{u} \int \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x}}{n\mu_2(K)^2 \int \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x}}$$

$$h = n^{-\frac{1}{d+4}} \left[\frac{\int \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x}}{\int \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x}} \right]^{\frac{1}{d+4}} \left[\frac{d \int K(\mathbf{u})^2 d\mathbf{u}}{\mu_2(K)^2} \right]^{\frac{1}{d+4}} \tag{3.16}$$

So, analogously to the optimal h calculated for univariate local polynomial regression, and given in Chapter 2 as (2.22), the multivariate equivalent is

$$h = C_{m,\sigma_\epsilon} \left[\frac{d \int K(\mathbf{u})^2 d\mathbf{u}}{\mu_2(K)^2} \right]^{\frac{1}{d+4}} n^{-\frac{1}{d+4}} \tag{3.17}$$

where

$$C_{m,\sigma_\epsilon} = \left[\frac{\int \frac{\sigma_\epsilon^2}{f(\mathbf{x})} d\mathbf{x}}{\int \left[\sum_{i=1}^{d} \frac{\partial^2 m}{\partial x_i} \right]^2 d\mathbf{x}} \right]^{\frac{1}{d+4}} \tag{3.18}$$

In this way, one could find a multivariate equivalent to (3.11). However, an equivalent to (3.9) cannot be found due to the fact that the equivalent to L cannot be expressed easily as a kernel function, that is not dependent on the point of estimation, for this multivariate technique. For this reason any choice of C must be made in a less theoretical manner.

Simulations were carried out on simple bivariate data sets, for which bandwidth selection tools work without computational problems. Data sets were simulated with differing covariate and response distributions, and GCV and the multivariate OSCV, (3.14), were carried out and h_1, h_2, b_1 and b_2 were selected. The h_j are obtained from GCV under the assumption that it finds a sufficient approximation of the h_j which minimize the MASE (these are after all simple data sets on which GCV should not struggle), and the b_j are obtained from the multivariate OSCV. h_j/b_j was then examined to see if it was consistent, since this is the ratio which determines C.

The results of this simulation study are inconclusive. Throughout the simulations an encouraging outcome was that $h_1/b_1 \approx h_2/b_2$ for each data set. However, this value varied depending on the data set. Also it was not apparent which factor influenced the differing values since it seems that varying both the response and the covariates has an effect. The results are summarized in Table 3.2.

x_j distribution	Y	med. $\frac{h_1}{b_1}$	med. $\frac{h_2}{b_2}$
N(0,0.5)	$x_1^2 + x_2^2 + N(0,1)$	1	1.01
N(0,0.5)	$\sin(3x_1) + \sin(3x_2) + N(0,0.2)$	0.67	0.68
N(0,1)	$x_1^2 + x_2^2 + N(0,0.2)$	0.54	0.55

Table 3.2: Details of the simulated data used to determine C. The figures in the med. $\frac{h_1}{b_1}$ and med. $\frac{h_2}{b_2}$ columns represent the median for $\frac{h_1}{b_1}$ and $\frac{h_2}{b_2}$ (which determine C) from 100 simulations of each data set.

Table 3.2 shows the median value, which is approximately equal to the

mode in every case, of the 100 simulations for each data set. In conclusion, these simulations appear to suggest that C depends on the data set being used, but there is good evidence that it is close to 1, at least for bivariate data.

Discussion

Since no definitive value has been obtained for C in higher dimensions it is impossible to test the success of this multivariate OSCV technique accurately and determine whether it holds the same advantages as the univariate technique, discussed in Hart and Yi (1998). The technique mentioned above however is an improvement on other possible multivariate extensions of the univariate method tested. Amongst other possibilities is using a product kernel, taking the form of a product of half-kernels. It would certainly have been easier to find an equivalent to L in this case, however this was found to be more problematic in practice, for trivariate data at least. For some simple data sets, the multivariate OSCV surface (of the form put forward in this section) has been found to have the significant advantage of having only one minimum, while GCV has many minima. In these cases the overall minimum is clearly easier to find using OSCV. An example is shown through Figs. 3.7 and 3.8 which show GCV and OSCV values respectively for different two-dimensional bandwidth values using two covariates from the *California Air Pollution* data, introduced in Chapter 1. The OSCV function shown in Fig. 3.8 is much smoother, and it is this which makes the overall minimum easier to find. This is not the case generally, but makes OSCV attractive for some specific data sets. It is not immediately obvious which characteristics a data set should exhibit for OSCV to be the more appropriate technique.

Despite these advantages, OSCV has limited use in the multivariate setting, particularly when taking into account the variable transformation constant C. It could however be used as a rough indication of the magnitude of the optimal bandwidths, in order to select a suitable starting point for another procedure such as AGCV. This could be used in cases such as the

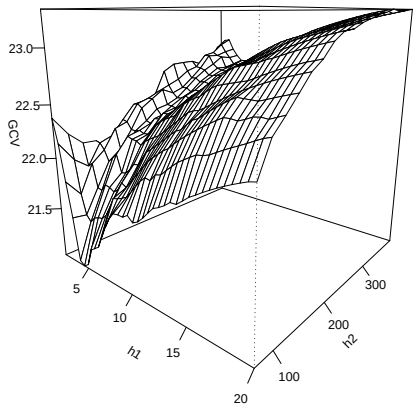

Figure 3.7: The GCV function for two covariates from the *California Air Pollution* data, displaying many local minima.

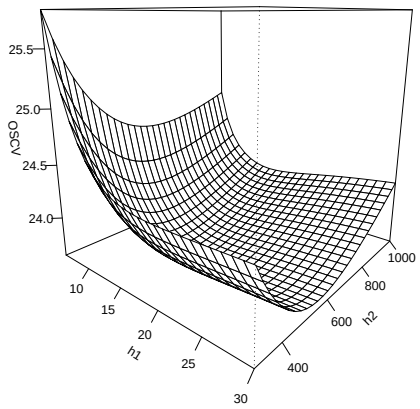

Figure 3.8: The OSCV function for two covariates from the *California Air Pollution* data, displaying one minimum.

above example where the OSCV surface has just one minimum compared to the many in the GCV surface. Unfortunately, in general, multivariate OSCV is also temperamental computationally, and would probably benefit from adaptations similar to those made in AGCV to combat data sparsity. It is successful in finding a minimum for some data sets but for others it returns multiple minima or error messages when using `optim` on R. There is also the additional complication that a value of l to be used in (3.14) must be chosen. In Hart and Yi (1998) different values of l are tested for univariate data, with the only constraint being that it is a *small* integer larger than 1. The same guidelines have been followed in the multivariate setting, however for some data, different values of l lead to dramatically different outputs for the same data set, with one value giving a reasonable output whilst another causes R to return error messages. Fortunately, with the relative success of AGCV, multivariate OSCV with its complications, does not have to be relied upon too heavily.

3.2.2 Univariate GCV via Newton-Raphson

As expressed earlier, `optim` on R can be unreliable. In an effort to avoid it the Newton-Raphson method can be employed in order to select the bandwidth for univariate local polynomial regression. The univariate case is focussed on initially since it is not obvious how to proceed with this in the multivariate case. The quantity to minimize is again the GCV, and in the univariate setting,

$$GCV(h) = \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{Y_i - \hat{m}(X_i)}{1 - \frac{1}{n}\text{trace}(\mathbf{S})}\right\}^2$$

The aim is to find the value of h for which the derivative of this is equal to zero. Differentiating the above by h one obtains

$$\frac{d}{dh}(GCV) = \frac{2}{n}\sum_{i=1}^{n}\left\{\frac{Y_i - \hat{m}(X_i)}{1 - \frac{1}{n}\text{trace}(\mathbf{S})}\right\}\frac{d}{dh}\left(\frac{Y_i - \hat{m}(X_i)}{1 - \frac{1}{n}\text{trace}(\mathbf{S})}\right).$$

Using the quotient rule for differentiating a quotient,

$$\frac{d}{dh}\left(\frac{u}{v}\right) = \frac{v\frac{du}{dh} - u\frac{dv}{dh}}{v^2}, \tag{3.19}$$

here $u = Y_i - \hat{m}(X_i)$, $v = 1 - \frac{\text{tr}(\mathbf{S})}{n}$, $\frac{du}{dh} = -\frac{d}{dh}\hat{m}(X_i)$, $\frac{dv}{dh} = -\frac{1}{n}\frac{d}{dh}\text{tr}(\mathbf{S})$. So here

$$\frac{v\frac{du}{dh} - u\frac{dv}{dh}}{v^2} = \frac{-\left(1 - \frac{\text{tr}(\mathbf{S})}{n}\right)\frac{d}{dh}\hat{m}(X_i) + \left(\frac{Y_i - \hat{m}(X_i)}{n}\right)\frac{d}{dh}\text{tr}(\mathbf{S})}{\left[1 - \frac{\text{tr}(\mathbf{S})}{n}\right]^2}.$$

So $\frac{d}{dh}(GCV)$ becomes

$$\frac{d}{dh}(GCV) = \frac{2}{n}\sum_{i=1}^{n}\left\{\frac{Y_i - \hat{m}(X_i)}{1 - \frac{1}{n}\text{tr}(\mathbf{S})}\right\}\left\{\frac{-\left(1 - \frac{\text{tr}(\mathbf{S})}{n}\right)\frac{d}{dh}\hat{m}(X_i) + \left(\frac{Y_i - \hat{m}(X_i)}{n}\right)\frac{d}{dh}\text{tr}(\mathbf{S})}{\left[1 - \frac{\text{tr}(\mathbf{S})}{n}\right]^2}\right\}$$

$$= \frac{2}{n\left[1 - \frac{\text{tr}(\mathbf{S})}{n}\right]^3}\sum_{i=1}^{n}\{Y_i - \hat{m}(X_i)\}\left\{\left(\frac{\text{tr}(\mathbf{S})}{n} - 1\right)\frac{d}{dh}\hat{m}(X_i) + \left(\frac{Y_i - \hat{m}(X_i)}{n}\right)\frac{d}{dh}\text{tr}(\mathbf{S})\right\} \tag{3.20}$$

This function is the function for which the zeroes are sought. Denote (3.20) as $s(h)$. The Newton-Raphson method, originally described by Isaac Newton, is carried out through iterations where the $(k+1)$th iteration is given by

$$h^{(k+1)} = h^{(k)} - \frac{s(h^{(k)})}{\frac{ds}{dh}(h^{(k)})} \tag{3.21}$$

and iterations are carried out until $\frac{||h^{(k+1)} - h^{(k)}||}{||h^{(k)}||} \leq \epsilon$ where $\epsilon > 0$ is small. With an appropriate h_0, chosen to be sufficiently large, Newton-Raphson can be tailored to almost always find the largest minimum, in terms of h, regardless of whether or not it is the minimum with the smallest overall GCV value. `optimize` (available in the base package on R), which is the univariate equivalent of `optim`, will always find a minimum, but it will not always be the minimum overall GCV value, or indeed the smallest or largest h value at which there is a minimum. This is the case for any starting point, and in this way it is more erratic than the use of Newton-Raphson described here. It may be unclear why choosing the largest minimizer could be an advantage, but Hart and Yi (1998) endorse the comments of Scott and Terrell (1987) and Park and Marron (1990) which suggest that this be used in density estimation in order to avoid undersmoothing. They also point out further justification in Hall and Marron (1991). If a similar view

is taken regarding regression then the Newton-Raphson procedure outlined below would appear to be effective and consistent enough at achieving this to be applied confidently.

To express the derivatives in (3.20) the following expansions, based on the definition of a derivative, are used;

$$\frac{d}{dh}\hat{m}(X_i) = \frac{\left[\hat{m}(X_i) - \hat{m}_{(1+\delta)h}(X_i)\right]}{h\delta} \tag{3.22}$$

$$\frac{d}{dh}\mathrm{tr}(\mathbf{S}) = \frac{\left[\mathrm{tr}(\mathbf{S}) - \mathrm{tr}(\mathbf{S}_{(1+\delta)h})\right]}{h\delta} \tag{3.23}$$

where $\mathbf{S}_{(1+\delta)h}$ is the smoother matrix and $\hat{m}_{(1+\delta)h}(X_i)$ the regression estimate, with a bandwidth of $(1+\delta)h$ employed in the place of h, and δ is small.

Including these changes

$$\begin{aligned} s(h) &= \frac{2}{n\left[1-\frac{\mathrm{tr}(\mathbf{S})}{n}\right]^3}\sum_{i=1}^{n}\{Y_i - \hat{m}(X_i)\}\left\{\left(\frac{\mathrm{tr}(\mathbf{S})}{n}-1\right)\left(\frac{1}{h\delta}\right)\left[\hat{m}(X_i) - \hat{m}_{(1+\delta)h}(X_i)\right]\right. \\ &\qquad \left. + \left(\frac{Y_i - \hat{m}(X_i)}{n}\right)\left(\frac{1}{h\delta}\right)\left[\mathrm{tr}(\mathbf{S}) - \mathrm{tr}(\mathbf{S}_{(1+\delta)h})\right]\right\} \\ &= \frac{2}{nh\delta\left[1-\frac{\mathrm{tr}(\mathbf{S})}{n}\right]^3}\sum_{i=1}^{n}\{Y_i - \hat{m}(X_i)\}\left\{\left(\frac{\mathrm{tr}(\mathbf{S})}{n}-1\right)\left[\hat{m}(X_i) - \hat{m}_{(1+\delta)h}(X_i)\right]\right. \\ &\qquad \left. + \left(\frac{Y_i - \hat{m}(X_i)}{n}\right)\left[\mathrm{tr}(\mathbf{S}) - \mathrm{tr}(\mathbf{S}_{(1+\delta)h})\right]\right\} \end{aligned} \tag{3.24}$$

In order to perform the Newton-Raphson procedure $\frac{ds}{dh}$ must be calculated. In an effort to simplify this $s(h)$ is split into two parts where $s(h) = s_1(h) + s_2(h)$.

$$s_1(h) = \frac{2}{nh\delta\left[1-\frac{\mathrm{tr}(\mathbf{S})}{n}\right]^3}\sum_{i=1}^{n}\{Y_i - \hat{m}(X_i)\}\left\{\left(\frac{\mathrm{tr}(\mathbf{S})}{n}-1\right)\left[\hat{m}(X_i) - \hat{m}_{(1+\delta)h}(X_i)\right]\right\} \tag{3.25}$$

$$s_2(h) = \frac{2}{nh\delta\left[1-\frac{\mathrm{tr}(\mathbf{S})}{n}\right]^3}\sum_{i=1}^{n}\{Y_i - \hat{m}(X_i)\}\left\{\left(\frac{Y_i - \hat{m}(X_i)}{n}\right)\left[\mathrm{tr}(\mathbf{S}) - \mathrm{tr}(\mathbf{S}_{(1+\delta)h})\right]\right\} \tag{3.26}$$

Now,

$$
\begin{aligned}
s_1(h) &= -2\sum_{i=1}^{n} \frac{1}{nh\delta\left[1-\frac{\mathrm{tr}(\mathbf{S})}{n}\right]^2}\left\{Y_i-\hat{m}(X_i)\right\}\left[\hat{m}(X_i)-\hat{m}_{(1+\delta)h}(X_i)\right] \\
&= -2\sum_{i=1}^{n} \frac{\left\{Y_i-\hat{m}(X_i)\right\}\left[\hat{m}(X_i)-\hat{m}_{(1+\delta)h}(X_i)\right]}{nh\delta\left[1-\frac{\mathrm{tr}(\mathbf{S})}{n}\right]\left[1-\frac{\mathrm{tr}(\mathbf{S})}{n}\right]} \\
&= \frac{-2}{\delta}\sum_{i=1}^{n} \frac{Y_i\hat{m}(X_i)-Y_i\hat{m}_{(1+\delta)h}(X_i)-\left[\hat{m}(X_i)\right]^2+\hat{m}(X_i)\hat{m}_{(1+\delta)h}(X_i)}{nh-2h\left[\mathrm{tr}(\mathbf{S})\right]+\frac{h\left[\mathrm{tr}(\mathbf{S})\right]^2}{n}}
\end{aligned}
\tag{3.27}
$$

Now $\frac{ds_1}{dh}$ is calculated using the quotient rule (3.19), where

$$u_1 = Y_i\hat{m}(X_i)-Y_i\hat{m}_{(1+\delta)h}(X_i)-\left[\hat{m}(X_i)\right]^2+\hat{m}(X_i)\hat{m}_{(1+\delta)h}(X_i)$$

$$v_1 = nh-2h\left[\mathrm{tr}(\mathbf{S})\right]+\frac{h\left[\mathrm{tr}(\mathbf{S})\right]^2}{n}$$

$$
\begin{aligned}
\frac{du_1}{dh} &= Y_i\frac{d}{dh}\hat{m}(X_i)-Y_i\frac{d}{dh}\hat{m}_{(1+\delta)h}(X_i)-2\hat{m}(X_i)\frac{d}{dh}\hat{m}(X_i) \\
&\quad +\hat{m}(X_i)\frac{d}{dh}\hat{m}_{(1+\delta)h}(X_i)+\frac{d}{dh}\hat{m}(X_i)\hat{m}_{(1+\delta)h}(X_i)
\end{aligned}
$$

$$\frac{dv_1}{dh} = n-2h\frac{d}{dh}\mathrm{tr}(\mathbf{S})-2\mathrm{tr}(\mathbf{S})+\frac{1}{n}\left[\left(\mathrm{tr}(\mathbf{S})\right)^2+2h\left(\mathrm{tr}(\mathbf{S})\right)\frac{d}{dh}\mathrm{tr}(\mathbf{S})\right]$$

In this way

$$\frac{ds_1}{dh} = \frac{-2}{\delta}\sum_{i=1}^{n}\frac{v_1\frac{du_1}{dh}-u_1\frac{dv_1}{dh}}{v_1^2} \tag{3.28}$$

$\frac{d}{dh}\hat{m}(X_i)$ and $\frac{d}{dh}\mathrm{tr}(\mathbf{S})$ are calculated as specified in (3.22) and (3.23), and similarly

$$\frac{d}{dh}\hat{m}_{(1+\delta)h}(X_i) = \frac{\left[\hat{m}_{(1+\delta)h}(X_i)-\hat{m}_{(1+2\delta)h}(X_i)\right]}{h\delta} \tag{3.29}$$

So

$$\frac{ds_1}{dh} = \frac{-2}{\delta}\sum_{i=1}^{n}\frac{A}{h^2\left[n-\mathrm{tr}(\mathbf{S})+\frac{[\mathrm{tr}(\mathbf{S})]^2}{n}\right]^2} \tag{3.30}$$

where

$$A = \left(n - 2\text{tr}(\mathbf{S}) + \frac{[\text{tr}(\mathbf{S})]^2}{n} \right) \frac{1}{\delta} \left\{ \hat{m}(X_i) \left[Y_i - 2\hat{m}(X_i) + 4\hat{m}_{(1+\delta)h}(X_i) - \hat{m}_{(1+2\delta)h}(X_i) \right] \right.$$
$$+ \hat{m}_{(1+\delta)h}(X_i) \left[-2Y_i - \hat{m}_{(1+\delta)h}(X_i) \right] + Y_i \hat{m}_{(1+2\delta)h}(X_i) \Big\}$$
$$- \left(\hat{m}(X_i) \left[Y_i - \hat{m}(X_i) + \hat{m}_{(1+\delta)h}(X_i) \right] - Y_i \hat{m}_{(1+\delta)h}(X_i) \right) \left\{ n - (\frac{2}{\delta} + 2)\text{tr}(\mathbf{S}) + \frac{2}{\delta}\text{tr}(\mathbf{S}_{(1+\delta)h}) \right.$$
$$\left. \left. + \frac{1}{n} \left[(\frac{2}{\delta} + 1) [\text{tr}(\mathbf{S})]^2 - \frac{2}{\delta}\text{tr}(\mathbf{S})\text{tr}(\mathbf{S}_{(1+\delta)h}) \right] \right\}$$

All of these terms can easily be calculated on R for given h.

The same procedure is carried out below for $s_2(h)$.

$$\begin{aligned} s_2(h) &= \frac{2}{n^2 h\delta \left[1 - \frac{\text{tr}(\mathbf{S})}{n} \right]^3} \sum_{i=1}^{n} \left(Y_i - \hat{m}(X_i) \right) \left(Y_i - \hat{m}(X_i) \right) \left(\text{tr}(\mathbf{S}) - \text{tr}(\mathbf{S}_{(1+\delta)h}) \right) \\ &= \frac{2}{n^2 h\delta \left[1 - \frac{\text{tr}(\mathbf{S})}{n} \right]^3} \sum_{i=1}^{n} \left(Y_i^2 + [\hat{m}(X_i)]^2 - 2Y_i \hat{m}(X_i) \right) \left(\text{tr}(\mathbf{S}) - \text{tr}(\mathbf{S}_{(1+\delta)h}) \right) \\ &= \frac{2}{n^2 h\delta \left[1 - \frac{\text{tr}(\mathbf{S})}{n} \right]^3} \sum_{i=1}^{n} \Big(Y_i^2 \text{tr}(\mathbf{S}) + [\hat{m}(X_i)]^2 \text{tr}(\mathbf{S}) - 2Y_i \hat{m}(X_i)\text{tr}(\mathbf{S}) \\ &\quad - Yi^2 \text{tr}(\mathbf{S}_{(1+\delta)h}) - [\hat{m}(X_i)]^2 \text{tr}(\mathbf{S}_{(1+\delta)h}) + 2Y_i \hat{m}(X_i)\text{tr}(\mathbf{S}_{(1+\delta)h}) \Big) \end{aligned} \tag{3.31}$$

Now $\frac{ds_2}{dh}$ is calculated using the quotient rule (3.19), where

$$u_2 = Y_i^2 \text{tr}(\mathbf{S}) + [\hat{m}(X_i)]^2 \text{tr}(\mathbf{S}) - 2Y_i \hat{m}(X_i)\text{tr}(\mathbf{S})$$
$$- Yi^2 \text{tr}(\mathbf{S}_{(1+\delta)h}) - [\hat{m}(X_i)]^2 \text{tr}(\mathbf{S}_{(1+\delta)h}) + 2Y_i \hat{m}(X_i)\text{tr}(\mathbf{S}_{(1+\delta)h})$$

$$v_2 = n^2 h + 3h [\text{tr}(\mathbf{S})]^2 - 3nh\text{tr}(\mathbf{S}) - \frac{h [\text{tr}(\mathbf{S})]^3}{n}$$

$$\begin{aligned} \frac{du_2}{dh} =& Y_i^2 \frac{d}{dh}\text{tr}(\mathbf{S}) + 2\hat{m}(X_i) \frac{d}{dh}\hat{m}(X_i)\text{tr}(\mathbf{S}) + [\hat{m}(X_i)]^2 \frac{d}{dh}\text{tr}(\mathbf{S}) - 2Y_i \frac{d}{dh}\hat{m}(X_i)\text{tr}(\mathbf{S}) \\ &- 2Y_i \hat{m}(X_i) \frac{d}{dh}\text{tr}(\mathbf{S}) - Y_i^2 \frac{d}{dh}\text{tr}(\mathbf{S}_{(1+\delta)h}) - 2\hat{m}(X_i) \frac{d}{dh}\hat{m}(X_i)\text{tr}(\mathbf{S}_{(1+\delta)h}) \\ &- [\hat{m}(X_i)]^2 \frac{d}{dh}\text{tr}(\mathbf{S}_{(1+\delta)h}) + 2Y_i \hat{m}(X_i) \frac{d}{dh}\text{tr}(\mathbf{S}_{(1+\delta)h}) + 2Y_i \frac{d}{dh}\hat{m}(X_i)\text{tr}(\mathbf{S}_{(1+\delta)h}) \end{aligned}$$

$$\begin{aligned}\frac{dv_2}{dh} =& n^2 + 3\left[\mathrm{tr}(\mathbf{S})\right]^2 + 6h\left[\mathrm{tr}(\mathbf{S})\right]\frac{d}{dh}\mathrm{tr}(\mathbf{S}) - 3n\left[\mathrm{tr}(\mathbf{S})\right] \\ & - 3nh\frac{d}{dh}\mathrm{tr}(\mathbf{S}) - \frac{\left[\mathrm{tr}(\mathbf{S})\right]^3}{n} - \frac{3h}{n}\left[\mathrm{tr}(\mathbf{S})\right]^2\frac{d}{dh}\mathrm{tr}(\mathbf{S})\end{aligned}$$

in which the substitution

$$\frac{d}{dh}\mathrm{tr}(\mathbf{S}_{(1+\delta)h}) = \frac{\left[\mathrm{tr}(\mathbf{S}_{(1+\delta)h}) - \mathrm{tr}(\mathbf{S}_{(1+2\delta)h})\right]}{h\delta}$$

is applied in calculation, analogously to (3.22), (3.23) and (3.29).

As a result,

$$\frac{ds_2}{dh} = \frac{2}{\delta}\sum_{i=1}^{n}\frac{v_2\frac{du_2}{dh} - u_2\frac{dv_2}{dh}}{n^4h^2\left[1 - \frac{\mathrm{tr}(\mathbf{S})}{n}\right]^6} \tag{3.32}$$

For an estimate of $\frac{ds}{dh}$, (3.32) and (3.30) are combined

$$\frac{ds}{dh} = \frac{ds_1}{dh} + \frac{ds_2}{dh} \tag{3.33}$$

In this way everything necessary for the Newton-Raphson algorithm, (3.21), is obtained. This has been implemented on R satisfactorily, using $\delta = \frac{1}{100}$, and consequently h via GCV is chosen.

Problems can occur in the implementation of univariate GCV on R when the GCV function is relatively flat with more than one minimum. In this case `optimize` is inconsistent in selecting the overall minimum and is very dependent on the starting point.

In contrast, the use of Newton-Raphson has been observed to show a degree of consistency. Newton-Raphson is also dependent on the initial h, h_0, used at the start of the algorithm, but as mentioned earlier this can be chosen so that the largest value of h at which a minimum in the GCV function is observed, is usually selected. An h_0 of 0.5 times the x data range has been trialled with some success. This is best illustrated through an example.

One-hundred x-values were simulated through a normal distribution with mean 50 and standard deviation 25. The response values were generated according to the model $m(X_i) = \log(X_i) + X_i$ and $\epsilon_i \sim N(0, 1)$. Fig. 3.9

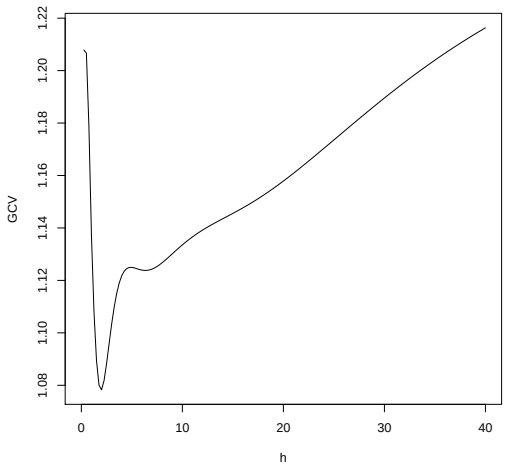

Figure 3.9: The GCV function for the simulated data set $m(X_i) = \log(X_i) + X_i + \epsilon_i$.

shows the GCV function for this data, displaying how it varies for different values of h. There are two minima, one at approximately $h = 2$ and one at approximately $h = 6.5$. Clearly the minimum at $h = 2$ leads to a smaller GCV value and so would be the natural choice of smoothing parameter. In this example, `optimize` selects the $h = 1.941$ value whereas Newton-Raphson selects the $h = 6.356$ value. This is how Newton-Raphson behaves consistently-with an h_0 of 0.5 times the data range, the algorithm usually stops at the larger minimum in terms of h. This may not always be optimal in terms of GCV, but the consistency is valuable.

This starting point is thought to be effective in achieving this for most data sets. At a starting point of this magnitude the GCV value is high relative to that at smaller h. This suggests that after the Newton-Raphson algorithm iterates down the slope of the GCV the first stationary point, at which the algorithm stops, will be a minimum, rather than a maximum. It is assumed that the optimal h is smaller than 0.5 times the data range and an alteration is made in the R code to ensure that initially the Newton-

Raphson algorithm iterates towards smaller values of h rather than those greater than h_0.

The Newton-Raphson algorithm can be used with any h_0 but, without sufficient care, it is difficult to determine at which minimum the algorithm is likely to stop, and so it no longer has an advantage over `optimize`, and is in fact less likely to choose the overall minimum.

It should be noted that, while in the author's experience this algorithm has been successful at identifying the largest local minimum, it has not been proven mathematically that this will be the case for any data set.

Chapter 4

Modal regression

In this chapter an alternative regression method is discussed. In previous chapters the focus has been on mean regression, but now modal regression is assessed as a possible alternative. As is nicely expressed in Scott (1992), the mode summarizes the "most likely" conditional values rather than the conditional average. Again the focus is on multivariate data, and the basic methodology discussed here is a multivariate extension of the univariate techniques of Einbeck and Tutz (2006). Modal regression has advantages, which will be discussed after first explaining the methodology, although it is worth mentioning initially that one of the main benefits of modal regression is its ability to represent a multimodal response. Modal regression has received little attention in the literature, and virtually none in the multivariate case. Scott (1992) and others propose it in the univariate case, but little methodology is given on how to actually implement it. Einbeck and Tutz (2006) fill this gap using *mean shift* which will be explained with regards to multivariate data later.

Simply put, modal regression uses the mode of the y values at x as the regression estimate at x. There could be more than one mode, and hence more than one regression estimate at x. Scott (1992) defines the regression estimate in the following way

$$\hat{m}(x) = \text{args max}_y \hat{f}(y|x). \tag{4.1}$$

Therefore, of crucial importance is the conditional density function, $f(y|x)$.

Now, in the nonparametric setting and for multivariate data,

$$\hat{f}(y|\mathbf{x}) = \frac{\hat{f}(\mathbf{x}, y)}{\hat{f}(\mathbf{x})} = \frac{\sum_{i=1}^{n} G\left(\frac{Y_i - y}{b}\right) \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right)}{b \sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right)}, \tag{4.2}$$

where G and κ are univariate (e.g. Gaussian) kernels, and the subscript j denotes the $j-$th component of the corresponding vector. The values b and h_j are bandwidth parameters to be selected. The maxima at $\mathbf{x}$ of function (4.2) form the regression estimates at $\mathbf{x}$. It is therefore the derivative of the multivariate conditional density estimator, (4.2), that is important, and in order to calculate this it is assumed that G in (4.2) is a radially symmetric kernel function of the form

$$G(.) = C_g g((.)^2),$$

where C_g is a positive constant and g is called the *profile* of G. Estimator (4.2) can then be re-written as

$$\hat{f}(y|\mathbf{x}) = \frac{C_g}{b} \sum_{i=1}^{n} w_i(\mathbf{x}) g\left(\left(\frac{Y_i - y}{b}\right)^2\right) \tag{4.3}$$

where

$$w_i(\mathbf{x}) = \frac{\prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right)}{\sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right)}. \tag{4.4}$$

An example of a conditional density function is given in Fig. 4.1, for one value of $\mathbf{x}$, for a data set that will be introduced later. Here the regression estimates would be approximately 1 and 3. The idea of using the maxima of the conditional kernel density estimate as estimators for the conditional modes is supported by Samanta and Thavaneswaran (1990) and Berlinet, Gannoun and Matzner-Løber (1998), who demonstrate that this estimator is "*consistent and asymptotically normally distributed under suitable regularity conditions*" (Einbeck and Tutz, 2006).

Modal regression in any dimension can be justified theoretically. It can be seen as the solution to a minimization problem in the same way that mean regression minimizes the MSE. Fan, Hu and Truong (1994) detail that the minimization problem

$$m_l(x) = \arg\min_{\alpha} E(l(Y - \alpha)|X = x) \tag{4.5}$$

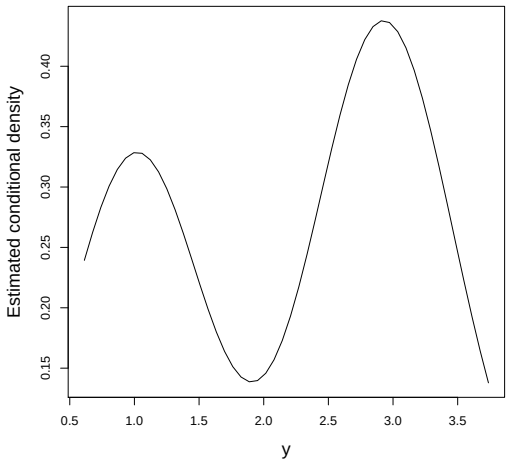

Figure 4.1: An example conditional density function, with regression estimates at approximately 1 and 3. This is at $\mathbf{x}$=(0.75,0.5) for simulation B (this data will be introduced in Section 4.2).

is solved by the mean if the loss function $l(z) = z^2$ (the MSE) and the median if $l(z) = |z|$. If $l(z) = -\delta(z)$ where $\delta(.)$ is the delta-function ($\delta(x) = 0$ for $x \neq 0$ and $\int \delta(x)dx = 1$), then (4.5) is solved by the mode, as detailed in Einbeck and Tutz (2006).

4.1 Conditional mean shift

As mentioned, modal regression as a technique has been suggested in the literature, but with few details on how to implement it. As Einbeck and Tutz (2006) point out, finding the maxima of a density function is a well established problem, but relatively little has been written on finding the maxima of a *conditional* density function. No mention at all has been made of relating these few techniques to modal regression. Scott (1992) and Carreira-Perpiñan (2000) both suggest methods for finding the maxima of a conditional density function but these are fairly complicated. It is worth

noting that a grid search is a possible solution to this problem whereby for each $\mathbf{x}$ a search is performed over y, however this is computationally very expensive, particularly when the data is multivariate. Einbeck and Tutz (2006) reflect on this and then successfully employ conditional mean shift on data with univariate predictors to find the maxima and hence perform modal regression.

Extending their work to the case of multivariate predictors results in the following. The maxima exist where the derivative of (4.3) is equal to 0;

$$\frac{\partial \hat{f}(y|\mathbf{x})}{\partial y} = \frac{2C_g}{b^3} \sum_{i=1}^{n} w_i(\mathbf{x}) g' \left(\left(\frac{Y_i - y}{b} \right)^2 \right) (y - Y_i) = 0.$$

Rearranging, leads to the following as an estimator for the conditional modes, y_m, at $\mathbf{x}$

$$y_m = \frac{\sum_{i=1}^{n} w_i(\mathbf{x}) g' \left(\left(\frac{Y_i - y_m}{b} \right)^2 \right) Y_i}{\sum_{i=1}^{n} w_i(\mathbf{x}) g' \left(\left(\frac{Y_i - y_m}{b} \right)^2 \right)}. \tag{4.6}$$

Let

$$h(.) = -g'(.)$$

where h is a kernel profile belonging to

$$H(.) = C_h h((.)^2).$$

Using this, (4.6) can be rewritten as

$$y_m = \frac{\sum_{i=1}^{n} H \left(\frac{Y_i - y_m}{b} \right) \prod_{j=1}^{d} \kappa \left(\frac{X_{ij} - x_j}{h_j} \right) Y_i}{\sum_{i=1}^{n} H \left(\frac{Y_i - y_m}{b} \right) \prod_{j=1}^{d} \kappa \left(\frac{X_{ij} - x_j}{h_j} \right)}. \tag{4.7}$$

In the examples presented in this chapter κ and G in (4.2) are Gaussian kernels, and as a result H is also Gaussian. This is easily shown by examining the profile of G. As a Gaussian kernel $G(u) = \frac{1}{\sqrt{2\pi}} \exp\left\{ -\frac{u^2}{2} \right\}$ which in the above notation means $C_g = \frac{1}{\sqrt{2\pi}}$ and $g(u) = \exp\left\{ -\frac{u}{2} \right\}$. In this case $h(u) = -g'(u) = -\frac{1}{2} \exp\left\{ -\frac{u}{2} \right\}$ and if one takes C_h as $\frac{-2}{\sqrt{2\pi}}$, then $H(u) = C_h h((u)^2) = G(u)$, the Gaussian kernel.

Equation (4.7) expresses the conditional mode, y_m, as a function of itself, denoted henceforth as $\mu(y_m)$. Since this cannot be solved analytically, it is solved iteratively using the result in Cheng (1995) that, starting from any $y_0 \in \mathbb{R}$, the mean shift procedure $y_{\ell+1} = \mu(y_\ell)$ converges to a nearby conditional mode. The term *mean shift* describes $\mu(y) - y$, the distance moved at each iteration by the procedure. Cheng (1995) describes how with each iteration, one shifts from a data point "*to the average of data points in its neighbourhood.*" According to Comaniciu and Meer (2002) the mean shift "*always points towards the direction of maximum increase in the density*", leading to a stationary point. They go on to reveal that in areas of high density the mean shift steps are smaller in comparison with areas of low density, meaning a more detailed analysis takes place in the high density area leading to a more accurate estimate of the mode which inevitably falls there.

Mean shift is a relatively unknown technique in the statistics community. It first appeared in Fukunaga and Hostetler (1975), in the context of pattern recognition. It was then largely ignored until Cheng (1995) highlighted the benefits of using mean shift to find the mode of a density. More recently it has been used in computer vision and feature space analysis such as in Comaniciu and Meer (2002). It was then used in Einbeck and Tutz (2006) in the univariate equivalent of the above. It is clearly very suitable there and here, since a method to seek a mode iteratively is both what is sought and a description of mean shift.

In order to detect more than one mode for each $\mathbf{x}$ it is necessary to specify more than one starting point for the mean shift, typically two. To identify modes in an M-modal conditional distribution, for a given multivariate $\mathbf{x}$, choose a set of starting points in the y-direction and then from each of these iterate $y_{l+1}^{(j)}(\mathbf{x}) = \mu(y_l^{(j)}(\mathbf{x}))$ until convergence is reached. The resulting $\hat{y}_m^{(1)}(\mathbf{x}), ..., \hat{y}_m^{(M)}(\mathbf{x})$ are then the M regression estimates at $\mathbf{x}$. In plots of the type in Fig. 4.4, there would then be M surfaces at $\mathbf{x}$. As in the univariate case, it is often sensible to set the number of starting points greater than M. More than one starting point can converge to the same mode, so if

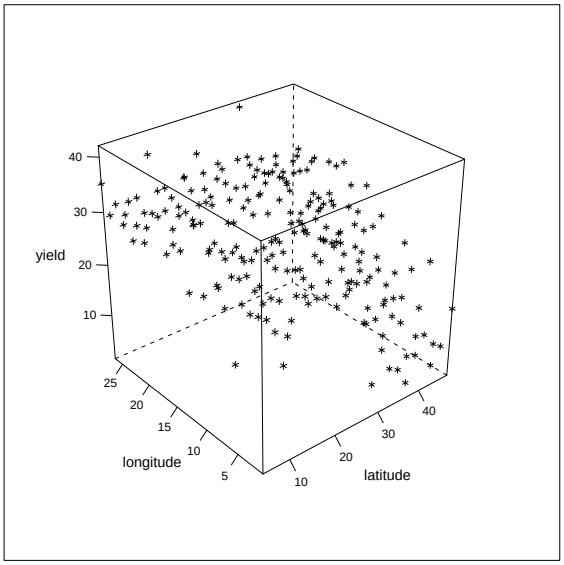

Figure 4.2: The bivariate wheat yield data set.

unsure about the actual number or location of modes, there is no harm in choosing a higher number of starting points from which all modes will be reached at least once. For the univariate case, Einbeck and Tutz (2006) state that a conditional mode is almost always reached after 30 iterations, and that this occurs fairly quickly. In the examples contributing to this chapter, the multivariate case has also been observed to behave satisfactorily in this sense.

4.2 Examples and properties

Fig. 4.2 shows data from a wheat yield trial, where latitude and longitude serve as covariates (the data are part of R package **nlme**, Pinheiro et al. (2011)). Fig. 4.3 provides the surface formed after 30 iterations of the mean shift process on the data set. Here $h_1 = 3.18$, $h_2 = 3.18$ and $b = 5.61$ after using the bandwidth selection methods described later.

Fig. 4.4 illustrates the characteristics of this smoothing technique through simulated bivariate data sets of size $n = 200$. Data set A is simulated from

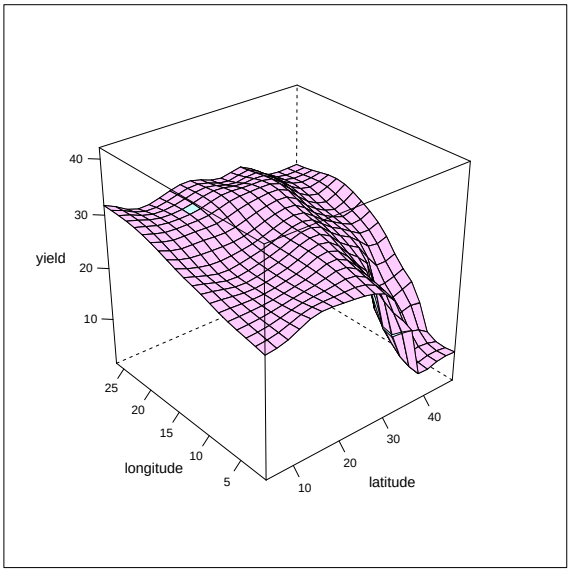

Figure 4.3: The modal regression estimate for the wheat yield data set, using conditional mean shift with 30 iterations.

the function $y = \sin(0.2x_1) + \cos(x_2)$ and subjected to Gaussian error with standard deviation 0.05. Data set B has a partially bimodal response, which splits for $x_1 \geq 0.5$ into two branches. For $x_1 < 0.5$ the response is simulated from the univariate function $y = 1.5 + 3x_1$ with Gaussian error of standard deviation 0.4. For $x_1 \geq 0.5$, the upper plane is centred at $y = 3$ and the lower plane at $y = 1$; the error standard deviation is 0.2 each. One observes from Fig. 4.4 how the estimated surfaces develop after different numbers of iterations, ℓ, with starting points positioned *above* (upper estimated surface) and *below* (lower estimated surface) all responses. For bivariate predictors, if y_0 is (for all $\mathbf{x}$) set greater than all Y_i, the simultaneous iterative execution of the mean shift resembles visually a net falling onto the data and forming a surface. Of course, if y_0 is below rather than above all Y_i, one would talk about a "rising" net. In the instance where there are more than two modes in the response distribution, these will clearly not all be detected by the "falling" and "rising" nets. These can be thought of instead as being detected by further nets, which, starting at points in between these two,

either rise or fall depending on the nearest conditional modes.

The technique is clearly visually appealing for bivariate data. The method can be applied to perform modal regression to data of any dimension in theory, but this has not been thoroughly examined, and in any case would not possess the same visual advantages. The right hand column of Fig. 4.4 demonstrates clearly the main advantages of modal regression. It is able to identify multiple modes when the underlying conditional distribution is multimodal, where other regression techniques could not successfully describe it. As is also mentioned in Scott (1992), modal regression is also edge-preserving, an important benefit when comparing it to mean regression. It is important to emphasize that the techniques proposed in this section do neither require the estimation of any density function, nor the solution of any optimization problem (such as least squares) at any stage; all computational work is carried out by the mean shift.

A further interesting property is that for a b value of $b = \infty$, the modal regression estimate is equal to the Nadaraya-Watson estimate. This can be demonstrated by examining the modal regression estimate, the conditional mode estimator, (4.7),

$$y_m = \frac{\sum_{i=1}^{n} H\left(\frac{Y_i - y_m}{b}\right) \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right) Y_i}{\sum_{i=1}^{n} H\left(\frac{Y_i - y_m}{b}\right) \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right)}.$$

If $b = \infty$, then for all Y_i,

$$H\left(\frac{Y_i - y_m}{b}\right) = H(0).$$

This means

$$\begin{aligned} y_m &= \frac{H(0) \sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right) Y_i}{H(0) \sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right)} \\ &= \frac{\sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right) Y_i}{\sum_{i=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - x_j}{h_j}\right)}, \end{aligned}$$

which is the multivariate Nadaraya-Watson estimate at $\mathbf{x}$ (see (1.29)). The implications of this in bandwidth selection are discussed later.

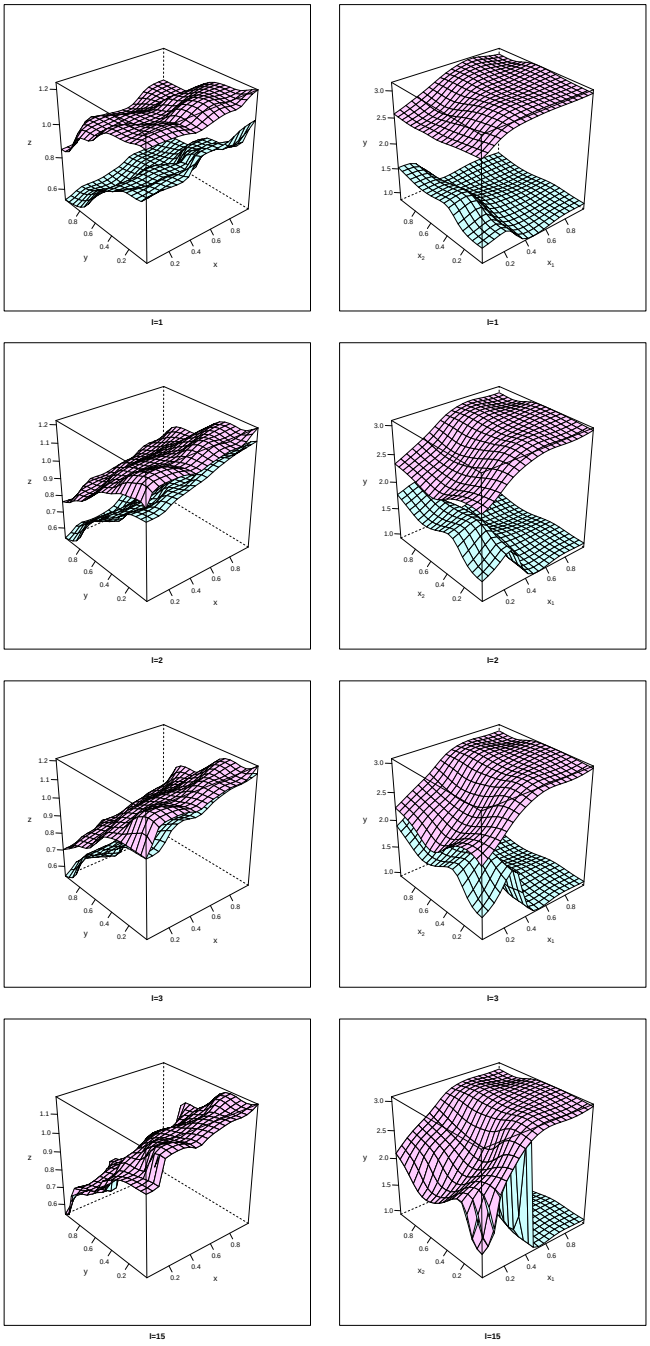

Figure 4.4: The left column displays modal regression surfaces for simulation A, for $\ell = 1, 2, 3, 15$ (from top to bottom). The right column shows the same for simulation B. The pink surfaces are comprised of modes captured by mean shift with starting points *above* all data, and the green surfaces, with starting points *below* all data.

4.3 Bandwidth selection

As with any smoothing technique, smoothing parameters must be chosen. The importance of this process, and the impact it has on the overall regression estimate was highlighted earlier in the context of local linear regression, and it is no less important here. The conditional mode estimator, equation (4.7), contains two types of smoothing parameters. The h_j are the manifestation of a diagonal bandwidth matrix $\mathbf{H}$, as adopted in local polynomial regression earlier. The h_j together describe the neighbourhood in the covariate space from which data is contributed to the modal regression. The parameter b influences the amount of smoothing applied in the response direction, the *vertical* direction in plots such as those in Figs. 4.3-4. Effectively, this determines the amount of smoothing applied to the actual conditional density curves, such as that in Fig. 4.1.

Equation (4.7) is essentially a manipulation of the conditional kernel density estimation formula (4.2). For this reason it makes sense to use bandwidth selection techniques developed for this purpose here. Amongst relatively little literature written on bandwidth selection for conditional density estimation, the most comprehensive variety of methods and discussion is given in Bashtannyk and Hyndman (2001) and Hyndman and Yao (2002). Although only univariate methods are discussed, extensions to multivariate versions are alluded to. The methods covered here include reference rules, a bootstrap approach, a regression-based approach as well as a combination of these methods. Of these methods, the regression-based approach seems the most straightforward to extend to the multivariate case, whilst performing well in the univariate simulations presented in Bashtannyk and Hyndman (2001). The regression-based approach was also reported as being less time-consuming than some others. The disadvantage of this technique is that it only calculates an optimal h, given b. For this reason a different strategy is needed to calculate b, which will be discussed after first explaining the extension of the *regression-based bandwidth selector* for use with multivariate data.

4.3.1 Estimating h_j

In this section the regression-based bandwidth selector of Bashtannyk and Hyndman (2001) is extended. This method centres around the penalized average square prediction error and is motivated by the use of such a measure by Härdle (1991) in selecting a bandwidth for regression. The conditional density estimator can itself be expressed as a regression problem as noted by Fan, Yao and Tong (1996), and in this way Bashtannyk and Hyndman (2001) exploited the bandwidth selection technique for regression for their purpose. Equally in the multivariate case, $\hat{f}(y|\mathbf{x})$ can be expressed as the value of β which minimizes

$$\sum_{i=1}^{n} w_i(\mathbf{x}) \left\{ \frac{1}{b} G\left(\frac{Y_i - y}{b} \right) - \beta \right\}^2 \tag{4.8}$$

where $w_i(\mathbf{x})$ is as in (4.4). By re-writing the conditional density estimate as a regression problem in the multivariate case one can use the multivariate penalized average square prediction error, $Q(h)$, in bandwidth selection. This is defined as

$$\begin{aligned} Q(h) &= \frac{\Delta}{n} \sum_{k=1}^{N} \sum_{i=1}^{n} \left\{ \frac{1}{b} \kappa\left(\frac{Y_i - y'_k}{b} \right) - \hat{f}(y'_k|\mathbf{X}_i) \right\}^2 \\ &\quad \times p\left(\frac{(\kappa(0))^d}{\sum_{l=1}^{n} \prod_{j=1}^{d} \kappa\left(\frac{X_{ij} - X_{lj}}{h} \right)} \right) \end{aligned} \tag{4.9}$$

where $\{y'_1, ..., y'_N\}$ are equally spaced over the sample space Y with $y'_{i+1} - y'_i = \Delta$ and where $p(u) = (1-u)^{-2}$ is a penalty function. Here one seeks an optimal $h = h_1 = ... = h_d$ which minimizes (4.9), keeping b fixed. In practice, the covariates are standardized prior to the minimization, and then the resulting h is unstandardized in each co-ordinate direction along with the covariates prior to the actual regression.

Bashtannyk and Hyndman (2001) states that minimizing the univariate equivalent of (4.9), with respect to h, is the same as minimizing the MISE, which is defined therein in the conditional density estimation context as

$$MISE(h, b; \hat{f}, f) = \int \int E\left\{ \hat{f}(y|x) - f(y|x) \right\}^2 f(x) dx dy. \tag{4.10}$$

They then explicitly suggest extending the univariate equivalent in the way carried out above, suggesting that they believe the minimization of (4.9) is an effective approximation to the minimization of the MISE in the multivariate case.

The R code for the univariate regression-based bandwidth selector is provided in the package **hdrcde**, Hyndman (2010), in the function `cde.bandwidths(method=3)` and is fairly straightforward to extend to the multivariate setting. The penalty function $p(u) = (1-u)^{-2}$ (the same as in GCV) suggested above for use with the multivariate $Q(h)$, (4.9), differs from the one used typically in the univariate case, since this was found to perform badly when applied here in the multivariate setting. It was found not to penalize very small values of h strongly enough. This alternative $p(u)$ is suggested in the `cde.bandwidths` code.

This method of selecting the h_j seems to work satisfactorily, as Figs. 4.3-4 suggest. These were constructed using h_j values obtained from this technique, and appear to show an appropriate amount of smoothing in the horizontal direction.

4.3.2 Estimating b

As mentioned, one cannot select values of h_j before first having a value of b to use in (4.9). When analysing the univariate regression-based bandwidth selector, Bashtannyk and Hyndman (2001) mainly use the normal reference rule detailed in their article to perform this task. This rule was inspired by work which uses reference distributions in bandwidth selection for kernel density estimation such as Silverman (1986). Such work is discussed in much more depth in Chapter 5. Bashtannyk and Hyndman (2001) formulate the reference rule by first calculating the optimal bandwidths in terms of MISE (the following uses the same notation as in their article). They then assume that $f(y|x)$ is normal with linear mean $c+dx$ and standard deviation p, and that $f(x)$ is a truncated normal density with mean μ_h and standard deviation σ_h. Using these assumptions, everything that is required to calculate the optimal bandwidths becomes available through some manipulation. After

which, the optimal bandwidth for b is given by

$$b_{NR} = \left\{ \frac{d^2 v(k)}{3\sqrt{2}\sigma_h^5 \lambda(k)} \right\}^{\frac{1}{4}} \left\{ \frac{16k \int \kappa^4(u) du p^5 (288\pi^9 \sigma_h^{58} \lambda^2(k))^{\frac{1}{8}}}{n \int u^4 \kappa^2(u) du d^{\frac{5}{2}} v^{\frac{3}{4}}(k) \left[v^{\frac{1}{2}}(k) + d(18\pi\sigma_h^{10}\lambda^2(k))^{\frac{1}{4}} \right]} \right\}^{\frac{1}{6}} \tag{4.11}$$

where

$$\lambda(k) = \int_{-k}^{k} \phi(t) dt,$$

with $\phi(t)$ the standard normal density function, and

$$v(k) = \sqrt{2\pi}\sigma_h^3 (3d^2\sigma_h^2 + 8p^2)\lambda(k) - 16k\sigma_h^2 p^2 e^{\frac{-k^2}{2}}$$

with k controlling the size of the sample space in the x direction.

This method is also implemented in `cde.bandwidths`, and is also used by Einbeck and Tutz (2006). There is also a normal reference rule which selects h, but this is omitted here due to the success of the regression-based bandwidth selector. Unfortunately, extending this normal reference rule for b to the multivariate setting is extremely difficult, so instead the univariate rule is modified. This seems acceptable since the rule itself, (4.11), does not depend directly on the number of dimensions in the data. Performing (4.11) for each covariate x_j separately, yielding b_j, and then setting

$$b_N = \frac{1}{d} \sum_{j=1}^{d} b_j \tag{4.12}$$

is generally effective here.

Figs. 4.5-6 show the results of using (4.12) in conjunction with (4.9) for the selection of the bandwidths. Fig. 4.5 shows modal regression via mean shift carried out on some air quality data measuring the air quality in New York in 1973. 117 observations are given on the response of ozone, measured in ppb, to the wind speed in mph and the temperature in F (this data is available in the R base package). The bandwidths selected here are $h_1 = 1.51, h_2 = 4.01$ and $b = 22.15$.

Fig. 4.6 shows the result of modal regression on a simulated bivariate data set J of size $n = 200$. The response is simulated from the function

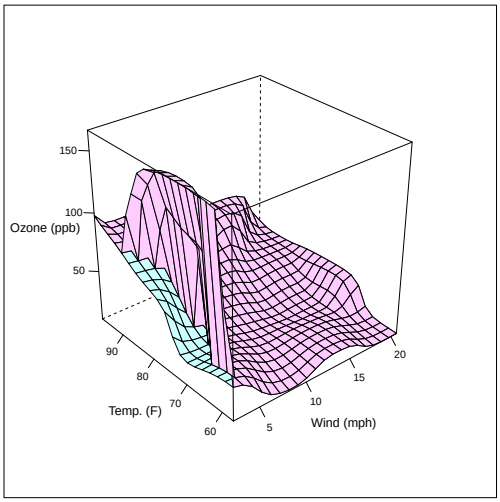

Figure 4.5: The modal regression estimate for the air quality data set, using conditional mean shift. The bandwidths were calculated using (4.9) and (4.12).

$y = \sin(12x_1 + 0.2) + \sin(12x_2)$ and subjected to Gaussian error with standard deviation 0.5. This data has a unimodal response distribution. The top plot shows the result when the bandwidths developed using the methods above are implemented ($h_1 = 0.072, h_2 = 0.072, b = 0.34$), and the bottom figure shows a surface formed using Nadaraya-Watson kernel regression. This bottom figure can be considered to give a true representation of the shape of the data.

Figs. 4.5-6 give an insight into the weakness of this bandwidth selection strategy – it struggles to deal with data with a unimodal response which has a significant amount of curvature. Although the air quality data regression is satisfactory in general, there is still a hint that things might not be perfect on the front left hand side of the surface where it splits. This could be a genuine feature of the data, or more likely a result of a poor choice of b. It should be noted that the two surfaces do not fail to meet due to an insufficient number of iterations in the mean shift, rather these are the surfaces settled upon after any reasonable number of iterations. In effect,

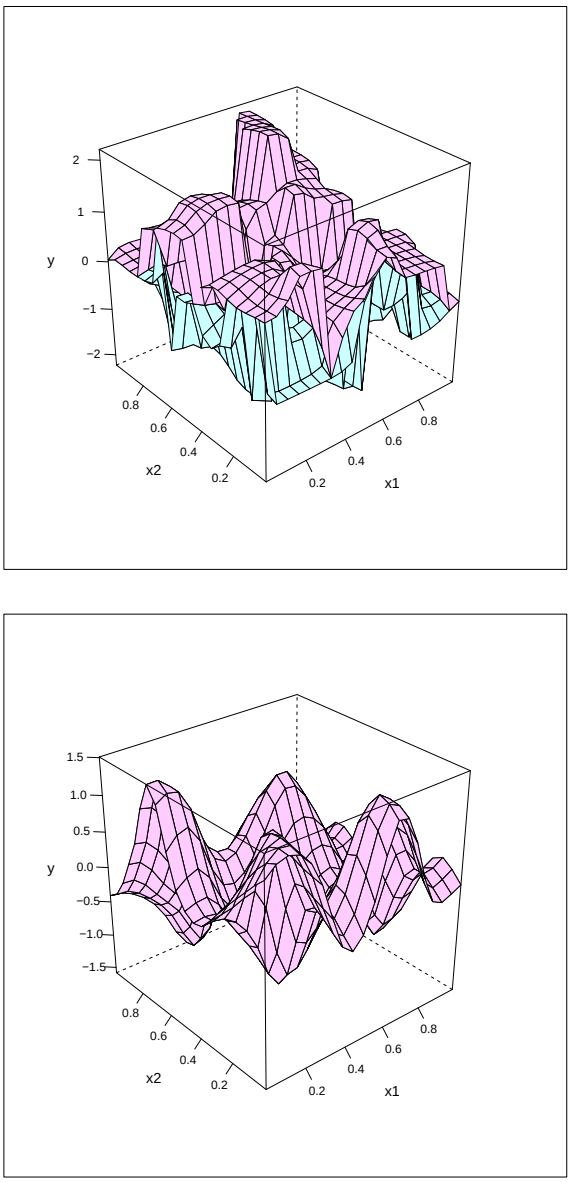

Figure 4.6: The top figure shows modal regression on data set J, which is known to have a unimodal response distribution, using the b obtained from (4.12). The bottom figure shows the same data represented by a Nadaraya-Watson kernel regression surface.

this bandwidth selection method is effective when the data has a multimodal response distribution, but in practise when it is unimodal, the value of b selected is too small. One could say that when using the modified normal reference rule the data is undersmoothed. If anything, an increase in b is necessary, which is in contrast to the conclusions in Chapter 5 in the discussion on univariate density estimation. This is also in contrast to the opinion of Einbeck and Tutz (2006) who show that the normal reference rule oversmooths in the univariate version of modal regression via mean shift. It is unsurprising that the use of (4.12) to find b is not perfect. This is after all a univariate technique, which is being used in the multivariate setting, and so it is likely that it does not adapt sufficiently when encountering the curse of dimensionality. However, in this shift to the multivariate case it has become suitable for use with multimodal responses. It should be mentioned that Fig. 4.6 acts as a slightly unfair comparison, since Nadaraya-Watson regression will always outperform modal regression on a data set where the data is smooth with no edges. Nevertheless, Fig. 4.6 (top) highlights a weakness of modal regression using b_N.

Adaptation for data which is known to have a unimodal response distribution

The method described above should be implemented when the data of interest has a multimodal response distribution. However, in order to combat the problems encountered when the response is unimodal throughout, if one is certain that this is the case, an adaptation is proposed to ensure that only one regression estimate results at each $\mathbf{x}$. In this case a variable vertical bandwidth is proposed of the form

$$b(\mathbf{x}) = \max\left\{b_N, \min\left(b : \hat{y}_m^{(1)}(\mathbf{x}) = ... = \hat{y}_m^{(M)}(\mathbf{x})\right)\right\}. \tag{4.13}$$

Firstly, the modified normal reference rule is carried out as above. For any $\mathbf{x}$ at which only one regression estimate is produced ($\hat{y}_m^{(1)}(\mathbf{x}) = ... = \hat{y}_m^{(M)}(\mathbf{x})$), b remains at (4.12) when performing regression at that point. At all other points, i.e. any point on a plot, such as Fig. 4.6 (top), at which the surfaces

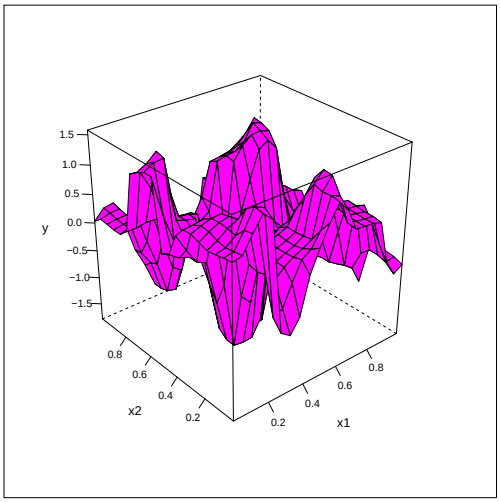

Figure 4.7: Modal regression on data set J, by conditional mean shift. A variable vertical bandwidth b, (4.13), is implemented.

do not meet, b is gradually increased until the two surfaces meet, resulting in a single regression surface throughout. At any $\mathbf{x}$, as soon as $\hat{y}_m^{(1)}(\mathbf{x}) = ... = \hat{y}_m^{(M)}(\mathbf{x})$, the minimum value of b which achieves this is taken as the bandwidth at that $\mathbf{x}$. In this way, $b(\mathbf{x})$ varies over the whole covariate space. Despite having a variable vertical bandwidth, it still works well to specify $b = b_N$ when calculating the h_j, given b, using (4.9).

Increasing the value of b increases the amount of smoothing in a conditional density estimate, such as Fig. 4.1, until eventually only one mode is present. This procedure forces the existence of only one regression estimate at each $\mathbf{x}$ whilst also retaining the edge-preserving quality which is such an important feature in modal regression. This is demonstrated in Fig. 4.7, which shows the same simulated data as in Fig. 4.6, but examined using (4.13). The function which generated this simulated data set J is known to have a unimodal response distribution, and so this adaptation is applicable here. It is clear in Fig. 4.7 that a much more appropriate regression surface is produced, particularly when compared to the plot produced when using the fixed value of b, Fig. 4.6 (top).

As well as being methods of bandwidth selection for this type of modal regression, the techniques described above are equally suitable for the wider field of bandwidth selection for multivariate conditional density estimation.

Investigating alternative methods of selecting b

The above bandwidth strategy was developed after first pursuing another direction of research, which itself had some interesting results. Of interest was determining if the optimal b value is independent of the number of dimensions in the data, d. In order to do this, a simulation study was carried out analysing the MSE of the modal regression estimate for different values of b on various simulated functions. However, it quickly became apparent that this study was not going to be successful, when it appeared that the use of very large values of b always yielded the smallest MSE values. It was whilst reflecting on this further that the observation was made that modal regression with $b = \infty$ is equivalent to Nadarya-Watson regression. This is an estimate of the *mean* at $\mathbf{x}$ and it makes sense that this estimate would minimize the MSE when compared to any estimate of the *mode*, particularly since the optimal h_j are of a similar magnitude to the optimal bandwidths for Nadaraya-Watson. For this reason, simulations analysing MSE are not appropriate when considering modal regression.

On a positive note, one cannot make a huge error if one chooses a large value of b. In fact the worst result would be the Nadaraya-Watson estimate and this is after all optimal in terms of MSE. For this reason, if given a choice of bandwidths it is probably sensible to choose the slightly larger value of b, since the quality of the regression estimate is unlikely to worsen. The problem with choosing a larger value of b is that the estimate loses the very qualities which make modal regression different i.e. the ability to represent a multimodal response and provide an edge-preserving surface.

A crude measure of estimating b was also examined. This focused on attempting to capture the same proportion of the multivariate covariate space with h_j and b, as the univariate bandwidths of Bashtannyk and Hyndman (2001) capture in the univariate space. This measure also often suggested

very large bandwidths which were favourable in terms of MSE, but failed in terms of producing an edge-preserving estimate.

There are also bandwidth selection tools already available for multivariate conditional density estimation in the **np** package by Hayfield and Racine (2008). However, in general, these select values which are too small as both h_j and b, provoking undersmoothing in every direction.

4.4 Relevance of a mode

This section is an extension of the univariate work by Einbeck and Tutz (2006). When there exist more than one mode of the conditional response distribution for a given $\mathbf{x}$, it is interesting to evaluate the relevance of the different modes. To estimate the probability associated with a conditional mode, one integrates numerically over the part of the estimated conditional density which forms that modal peak. The conditional density estimate in Fig. 4.1 is for $\mathbf{x} = (0.75, 0.5)$ for simulation B (the data on the right hand side of Fig. 4.4). This plot indicates clearly that this point falls in the half of this data set which has a bimodal response. The area covered by the peak on the left represents the probability that the data at $\mathbf{x} = (0.75, 0.5)$ has a response value of approximately $y = 1$, and the area covered by the peak on the right represents the same but for a response value of approximately $y = 3$. Einbeck and Tutz (2006) state that the search for the minimum and the integration can be performed simultaneously, by descending in small steps from the modes and increasing the integral, until either the boundary or the next dip separating the modes is reached. They note that this method of integration, although not being the most sophisticated, is "*surprisingly accurate.*" For simulation B, Fig. 4.8 displays a surface of probabilities, calculated as described for every $\mathbf{x}$, showing the probability that data is present in the mode captured by the "falling net". Fig. 4.9 shows the same for the "rising net." For this data set, the plots show a probability of 1 for approximately half of all values of $\mathbf{x}$; this is expected since the response is unimodal for these $\mathbf{x}$. Note that in these two figures, the orientation is rotated in order to allow for a better view of the probability surface.

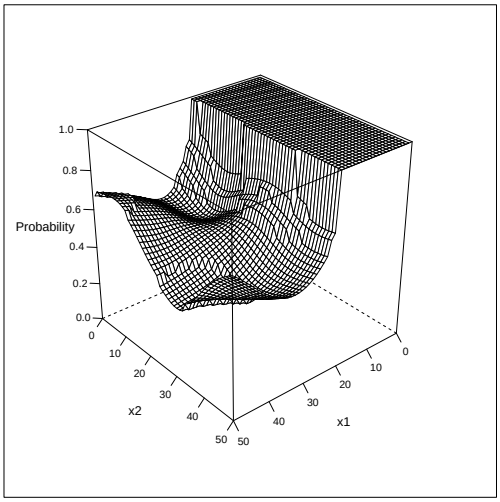

Figure 4.8: Bivariate probability plot for the "falling net" for the fitted surface from Fig. 4.4 (bottom right). For each $\mathbf{x}$, this displays the probability associated with the mode captured by the pink surface in Fig. 4.4.

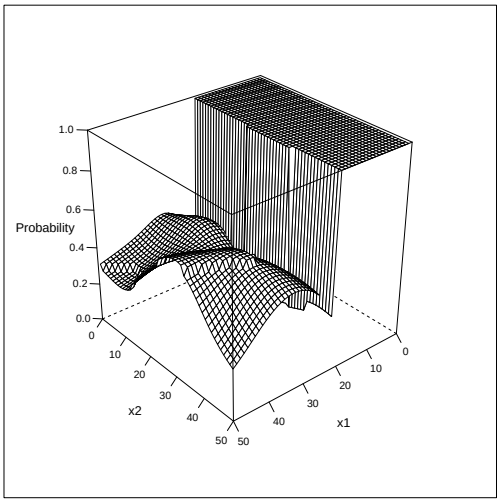

Figure 4.9: Bivariate probability plot for the "rising net" for the fitted surface from Fig. 4.4 (bottom right). For each $\mathbf{x}$, this displays the probability associated with the mode captured by the green surface in Fig. 4.4.

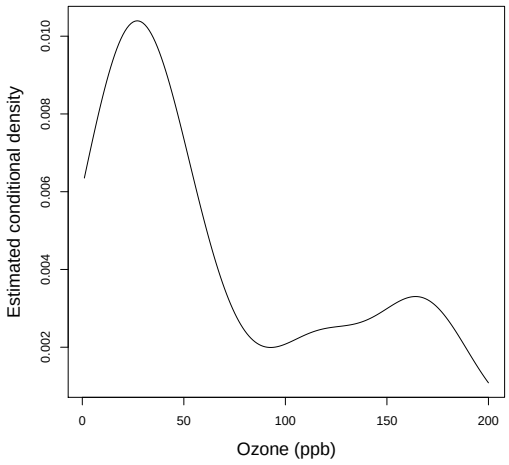

Figure 4.10: Estimated conditional density function at wind=3 and temperature=71 for the air quality data set.

Fig. 4.10 shows the conditional density estimate at wind=3, temperature=71 for the air quality data, previously exhibited in Fig. 4.5. This value of $\mathbf{x}$ is of interest due to its location in the part of the regression surface which splits into two branches. If one calculates the probabilities associated with each mode, as described above, one can estimate that the probability that the data at this point belongs to the lower (green) branch is 0.698, and the probability to the higher (pink) branch is 0.302. This suggests that the existence of two surfaces is justified here, but it is important to remember that these probabilities themselves are dependant on the choice of b, and thus a poor choice of b will yield poor estimates of the probabilities.

Whilst these surfaces of probabilities are neat, inference of a more traditional type is also possible. Since the modes of the conditional density function at $\mathbf{x}$ are the regression estimates at $\mathbf{x}$ it is the properties of these conditional modes which must be considered. Samanta and Thavaneswaran (1990) show that a conditional mode, estimated using kernels as in this chapter, is asymptotically normally distributed, and Berlinet, Gannoun and Matzner-Løber (1998) study this property in the context of confidence in-

tervals. This implies that such inference is possible for this modal regression technique.

4.5 Discussion

In summary, the advantages of this method of performing modal regression are that it can capture a multimodal response and is edge-preserving while also being a computationally simple and visually appealing procedure. However, it should be admitted that multivariate data with a multimodal response is relatively rare, and that multiple modes in the response distribution may be an indicator that important covariates have been omitted from the model. Nevertheless, the presented approach may still serve to detect and visualize situations of this type. In any case, Einbeck and Tutz (2006) note that in the univariate case there are parametric approaches to representing a multimodal response, such as those given by Wedel and Kamakura (1995) and Cherkassy and Ma (2005). They also note that any existing nonparametric methods require knowledge of which mode each data point is associated with, and so the mean shift technique is clearly different to, and holds advantages over, other existing methods attempting to perform the same task.

A bandwidth selection strategy was set out above, including an action to deal with the scenario where multiple regression estimates are given at a point at which one knows that the true response is unimodal. If one obtains a single regression estimate at $\mathbf{x}$, with $b = b_N$, one can be quite sure that the true response distribution is unimodal at this point, since the value of b, which broadly speaking determines the number of modes, has only been found to undersmooth when using (4.12), and never oversmooth. However there is one outcome that has not been examined thoroughly – if one obtains more than one regression estimate at $\mathbf{x}$, but does not know for sure how many modes the response should have at that point. At $\mathbf{x}$, more than one regression estimate could arise for one or more of three main reasons:

- Data sparsity
- Response distribution is truly multimodal
- The value of b is too small.

An example of this situation is illustrated in the regression surface of the air quality data in Fig. 4.5.

Data sparsity can affect the regression estimate in two main ways. The first is the presence of an outlier in the y-direction. As Scott (1992) reports, modal regression in its purest form is resistant to outliers. Provided that one sets enough starting points in the mean shift, the true function will be detected as a conditional mode(s), unaffected by an outlier, which may also be represented by its own mode. The mode representing the outlier can then be ignored, however one has the inconvenience of determining whether each mode is caused by an outlier or not. Alternatively, if one implements the variable $b(\mathbf{x})$, detailed in (4.13), in an attempt to force the existence of only one regression estimate at each $\mathbf{x}$, and in doing so preventing an outlier causing a mode, the outlier is having an influence by imposing a higher value of b than would be necessary without it, and so changing the regression estimate as a result. So outliers are inconvenient in both situations, either by adding another layer to the regression surface which is inconvenient for anyone who wishes to analyse the true function, or by influencing the regression detrimentally when one imposes (4.13). For this reason, it is recommended that one removes outliers from the process prior to the mean shift. This can be done using any standard outlier detection and removal process. A further option is to adopt the approach which Breiman, Meisel and Purcell (1977) take in the context of multivariate density estimation. Here the bandwidth varies depending on data sparsity, and a larger bandwidth in an area of greater data sparsity ensures that outliers do not have an unwanted effect on the density estimate. Implementing a technique similar to this and employing a variable b at each $\mathbf{x}$ in modal regression could prevent the problem of outliers described above.

4.5.1 Implementing the density threshold (2.18)

The second way in which data sparsity influences this technique is through the curse of dimensionality, which was widely discussed earlier in this thesis. Around the edges of the data, modal regression is particularly sensitive to individual data points. This leads to multiple regression estimates being produced more commonly around the edges than in the centre of the data. It makes sense to attempt to implement the threshold developed earlier, (2.18), particularly given the relationship between this technique and Nadaraya-Watson smoothing on which this threshold can be easily used. Briefly, in order to compare Nadaraya-Watson smoothing and modal regression via mean shift directly, assume that both use the same h_j in the covariate directions (in fact they are usually of a similar magnitude). The neighbourhood at $\mathbf{x}$ described by the kernels is always larger for Nadaraya-Watson, since it encapsulates the same space in the covariate space as well as the entire space in the response direction. The modal regression is limited in the response direction by the bandwidth b, reducing the neighbourhood size. Therefore, if (2.18) considers that Nadaraya-Watson smoothing is not reliable at $\mathbf{x}$, it seems a sensible indication that modal regression, with a smaller neighbourhood, will also not be reliable. In this way, (2.18) can be used to dismiss some areas of sparse data, without having to develop a new threshold especially for modal regression. For reference, the value of ρ to be used in the threshold, (2.18), for Nadaraya-Watson smoothing with bivariate data is 1.55. When applying this to the air quality data in Fig. 4.5, the density threshold is approximately 0.00035. The density is smaller than this at only three of the actual data points and these are all for high values of wind speed, and not in the area in the front left where the surface splits. When examining grid values in the area of the surface which splits, the density at the majority of these is also greater than 0.00035. The exception seems to be in the area around wind=3, temperature=70, where the density is approximately 0.00004. Here, the split in the surface could be due to data sparsity, however further back, at higher temperature values, the bimodal response is probably as a result of something else.

4.5.2 Further discussion

Having taken into account the methods above in order to diminish the effects of data sparsity, it is then necessary to decide whether any remaining $\mathbf{x}$ values, at which more than one regression estimate is produced, are the result of a truly multimodal underlying response distribution or a value of b which is too small. Of course if one knows for sure that the response should be unimodal this would be the time to implement the variable bandwidth, (4.13), without it suffering from any data sparsity related issues.

It should be stressed that modal regression via mean shift should not be used itself to determine the number of modes in the response at $\mathbf{x}$, since the bandwidth selection technique above favours a multimodal response, nor should the probability plots discussed in the last section be used as an indication, since these also depend on b. Essentially, the perfect bandwidth selection technique will choose a bandwidth which produces the correct number of modes in the response, but in order to know which technique to use it is necessary to know the modality of the response. If the modality of the response varies throughout the data then the bandwidth selection technique should vary accordingly. The idea of anticipating the number of modes prior to the bandwidth selection is the central theme of the next chapter, where this is applied in kernel density estimation. It is a common problem in density estimation to be unsure of the exact modality of data. This should however be seen as less of a problem to have in the regression context, since with most other methods only one estimate is possible at each $\mathbf{x}$. Silverman (1981) and Müller and Sawitzki (1991) describe strategies for detecting the number of modes in a univariate distribution, but these do not extend to *multivariate conditional* distributions. As a result there is no fixed answer to how one should act in this situation, it is simply important that one acts with caution when interpreting a modal regression surface generated using the techniques described in this chapter.

Chapter 5

Bandwidth selection for multimodal kernel density estimation

In the previous chapter it would have been useful to be able to adapt the normal reference rule to suit conditional density estimates of varying modality. In this chapter two possible methods are discussed in order to do exactly this in the simpler setting of kernel density estimation. To clarify, the work in this chapter is restricted solely to univariate data.

It is the kernel density estimator, (1.54), for a univariate random variable X, with standard deviation σ, which is of interest, and of particular interest is the selection of the bandwidth h. This estimator was originally proposed by Rosenblatt (1956) and has been investigated thoroughly in the literature since, notably in Parzen (1962) and Silverman (1978). The issue of bandwidth selection itself has been studied just as thoroughly, and is just as crucial in kernel density estimation as in the other areas in which it has been discussed in this thesis. As well as the normal reference rule, developed by Silverman (1986), there are also various other bandwidth selection techniques, but as discussed in Zhang and Wang (2009), they each suffer from at least one problem. Least squares cross-validation (LSCV) was introduced by Rudemo (1982) and Bowman (1984), but under the alternative name of

unbiased cross-validation. This technique, which selects h by minimizing the empirically estimated quantity whose expectation is identical to the MISE, is the best for minimizing the asymptotic MISE according to Stone (1984), but as noted by Zhang and Wang (2009) it tends to be highly variable as well as undersmooth the density. Zhang and Wang (2009) also review other methods introduced by Scott and Terrell (1987) and Sheather and Jones (1991), claiming that these suffer from computational problems and from not being robust to outliers. They themselves come up with one solution to these problems by developing a bandwidth selector which "*adapts to different types of densities.*" A recent paper by Srihera and Stute (2011) develops a density estimation and bandwidth selection tool which also adapts to the data at hand, however here it is the kernel function which is adjusted to suit f. This appears successful but only limited testing has been carried out.

In this chapter the focus is on the simplest of all these methods, the normal reference rule. This attempts to approximate the asymptotically optimal bandwidth, (1.64), which in turn minimizes an asymptotic version of the MISE. The unknown quantity in (1.64) is $\int \left[f''(x)\right]^2 dx$, and in order to form the normal reference rule, Silverman proposed using the normal density, $\phi(x) = (2\pi)^{-1/2} \exp^{-x^2/2}$ with standard deviation σ, as an approximation of $f(x)$, i.e.

$$\int \left[f''(x)\right]^2 dx \approx \sigma^{-5} \int \left[\phi''(x)\right]^2 dx = \frac{3}{8\sqrt{\pi}}\sigma^{-5} \approx 0.212\sigma^{-5}. \tag{5.1}$$

When κ is Gaussian $\kappa_0 = 0.776$ in (1.64), and using this normal approximation, Silverman derived the normal reference bandwidth selector,

$$h_{NR} = 1.06sn^{-1/5} \tag{5.2}$$

where the sample standard deviation, s, is used to approximate σ.

Silverman (1986) observed that this rule "*will work well if the population is unimodal, it may oversmooth somewhat if the population is multimodal.*" It is this oversmoothing in the multimodal densities that is addressed in this chapter. This problem has been tackled elsewhere previously in a number of ways. Firstly, different approaches are taken to approximating σ. Silverman (1986) proposes a more robust measure of spread, $A = \min(s, IQR/1.34)$,

which seeks to avoid oversmoothing in multimodal data as well as skew data. Zhang and Wang (2009) adapt this by using a quantile-based measure of spread. Secondly, adaptations are proposed to the constant 1.06 which was derived above from the normal reference assumption. Silverman (1986) proposes replacing 1.06 by 0.9, without providing any reasoning behind this other than that 0.9 is smaller than 1.06. This fits with the intuitive notion that the more modes the density has, the smaller the bandwidth should be in order to enable an adequate degree of resolution. This is indeed the line of thinking that is followed in this chapter. Here however, the aim is to provide a bandwidth selection method, with justification, which quantifies how much smaller the bandwidth should be for a density with m modes. If the data are multimodal, the normal reference rule will underestimate $\int [f''(x)]^2 \, dx$ and so overestimate h. Here, alternative ways of approximating $\int [f''(x)]^2 \, dx$ are considered to address this problem.

5.1 Approaches to bandwidth selection with reference to a Gaussian mixture

5.1.1 Reference to a fitted Gaussian mixture

One possible method of approximating $\int [f''(x)]^2 \, dx$ more accurately is to replace the concept of making reference to a normal density with making reference to a mixture of normal densities. Here it is sensible to have m normal densities ϕ_{μ_k,σ_k} centred at locations μ_k, with standard deviations σ_k, and associated mixture probabilities π_k, $k = 1, \ldots, m$. For a given data set, these parameters can be estimated to form a mixture density close to the density of the data. These parameters can be estimated using the EM algorithm of Laird (1978) which is implemented in R in packages such as **npmlreg** by Einbeck, Darnell and Hinde (2009). In this case the estimated density is then given by

$$\hat{f}_m(x) = \sum_{k=1}^{m} \hat{\pi}_k \phi_{\hat{\mu}_k,\hat{\sigma}_k}(x). \tag{5.3}$$

Now, the quantity of interest is $\int [f''(x)]^2 \, dx$. For this density

$$\hat{f}'_m(x) = \frac{1}{\sqrt{2\pi}} \sum_{k=1}^{m} \frac{\hat{\pi}_k}{\hat{\sigma}_k^3} (\hat{\mu}_k - x) \exp\left\{ -\frac{1}{2} \left(\frac{x - \hat{\mu}_k}{\hat{\sigma}_k} \right)^2 \right\},$$

and

$$\hat{f}''_m(x) = \frac{1}{\sqrt{2\pi}} \sum_{k=1}^{m} \frac{\hat{\pi}_k}{\hat{\sigma}_k^3} \left(\left(\frac{\hat{\mu}_k - x}{\hat{\sigma_k}} \right)^2 - 1 \right) \exp\left\{ -\frac{1}{2} \left(\frac{x - \hat{\mu}_k}{\hat{\sigma}_k} \right)^2 \right\}.$$

Hence,

$$\int \left[\hat{f}''_m(x) \right]^2 dx = \frac{1}{2\pi} \left[\sum_{k=1}^{m} \frac{\hat{\pi}_k^2}{\hat{\sigma}_k^6} \int \left(\left(\frac{\hat{\mu}_k - x}{\hat{\sigma}_k} \right)^2 - 1 \right)^2 \exp\left\{ - \left(\frac{x - \hat{\mu}_k}{\hat{\sigma}_k} \right)^2 \right\} dx \right.$$

$$\left. + \sum_{k \neq l} \frac{\hat{\pi}_k \hat{\pi}_l}{\hat{\sigma}_k^3 \hat{\sigma}_l^3} \int \left(\left(\frac{\hat{\mu}_k - x}{\hat{\sigma}_k} \right)^2 - 1 \right) \left(\left(\frac{\hat{\mu}_l - x}{\hat{\sigma}_l} \right)^2 - 1 \right) \exp^{-\frac{1}{2}\left(\left(\frac{x - \hat{\mu}_k}{\hat{\sigma}_k} \right)^2 + \left(\frac{x - \hat{\mu}_l}{\hat{\sigma}_l} \right)^2 \right)} dx \right].$$

This integral can be calculated explicitly through convolutions of normal densities (see Theorem 4.1 of Marron and Wand (1992)), or by using software such as Mathematica. In this way, the asymptotic optimal bandwidth, (1.64), can be approximated by

$$h_m = \kappa_0 \left\{ \int \left[\hat{f}''_m(x) \right]^2 dx \right\}^{-1/5} n^{-1/5}. \tag{5.4}$$

When approximating f by a mixture density one has the option of how large to make m before applying the EM algorithm. In this way, one specifies an expected modality prior to the density estimation, and so it is necessary that one has some prior knowledge of approximately how many modes the data should have in order for this method to be effective. This will be discussed in more detail later. It should be noted that choosing m densities to make up the mixture does not necessarily translate into m modes. In fact, this leads to *at most* m modes in the mixture density, and will often result in less.

5.1.2 Rule of thumb

A further bandwidth selection method has been developed which approximates $\int [f''(x)]^2 \, dx$ differently again. This also uses a mixture density but

does not require the actual fitting of a mixture or the complicated integration. This can be thought of as a rule of thumb, since it is sufficiently simple and completely data independent. In order to create this rule of thumb, some simplifying assumptions are required. The shape of the mixture density is restricted to an equal mixture of m normal densities, each with identical standard deviation σ_c, which are placed at equidistant locations $\mu_k, k = 1, \ldots, m$.

In this specific case, the density takes the form

$$\hat{f}(x) = \frac{1}{m} \sum_{k=1}^{m} \frac{1}{\sqrt{2\pi\sigma_c^2}} \exp\left\{ -\frac{1}{2} \left(\frac{x - \mu_k}{\sigma_c} \right)^2 \right\}. \tag{5.5}$$

Applying the calculations from the previous sub-section,

$$\begin{aligned} \int \left[\hat{f}''(x) \right]^2 dx &= \frac{1}{2\pi m^2 \sigma_c^6} \left[\sum_{k=1}^{m} \int \left(\left(\frac{\mu_k - x}{\sigma_c} \right)^2 - 1 \right)^2 \exp\left\{ - \left(\frac{x - \mu_k}{\sigma_c} \right)^2 \right\} dx \right. \\ &\quad \left. + \sum_{k \neq l} \int \left(\left(\frac{\mu_k - x}{\sigma_c} \right)^2 - 1 \right) \left(\left(\frac{\mu_l - x}{\sigma_c} \right)^2 - 1 \right) \exp^{-\frac{1}{2}\left(\left(\frac{x-\mu_k}{\sigma_c} \right)^2 + \left(\frac{x-\mu_l}{\sigma_c} \right)^2 \right)} dx \right]. \end{aligned}$$

As the integral over the squared second derivatives is a location invariant functional, the position of the locations can be written w.l.o.g. as $\mu_k = kd\sigma_c$, with a distance parameter d. For instance, a value of $d = 2$ means that all modes are two component standard deviations away from each other. Using the fact that the μ_k are set at a distance of $d\sigma_c$ apart, substitute $\frac{x-\mu_k}{\sigma_c} = u$ and $\frac{x-\mu_l}{\sigma_c} = u + (k-l)d$. This leads to

$$\int \left[\hat{f}''(x) \right]^2 dx = \frac{1}{2\pi m^2 \sigma_c^5} \left[\frac{3\sqrt{\pi} m}{4} + \sum_{k \neq l} \int (u^2 - 1) \left[(u + (k-l)d)^2 - 1 \right] \exp^{-\frac{1}{2}\left[u^2 + (u + (k-l)d)^2 \right]} du \right]. \tag{5.6}$$

Examining everything to the right hand side of the $\sum$ in (5.6),

$$\begin{aligned} &\int (u^2 - 1) \left[(u + (k-l)d)^2 - 1 \right] \exp^{-\frac{1}{2}\left[u^2 + (u + (k-l)d)^2 \right]} du \\ &= \int (u^2 - 1) \left[u^2 + 2u(k-l)d + (k-l)^2 d^2 - 1 \right] \exp^{-\frac{1}{2}\left[u^2 + (u + (k-l)d)^2 \right]} du \\ &= \exp\left\{ -\frac{(k-l)^2 d^2}{4} \right\} \int \left(u^4 + 2u^3 (k-l)d + u^2 (k-l)^2 d^2 - 2u^2 - 2u(k-l)d - (k-l)^2 d^2 + 1 \right) \\ &\quad \times \exp\left\{ - \left(u + \frac{(k-l)d}{2} \right)^2 \right\} du. \end{aligned} \tag{5.7}$$

By substituting $t = u + \frac{(k-l)d}{2}$, manipulation yields (5.7) to be equal to

$$\begin{aligned} & \exp\left\{-\frac{(k-l)^2d^2}{4}\right\} \int \left[t^4 - \left(2 + \frac{(k-l)^2d^2}{2}\right) t^2 + 1 - \frac{(k-l)^2d^2}{2} + \frac{(k-l)^4d^4}{16}\right] \exp^{-t^2} dt \\ = \quad & \exp\left\{-\frac{(k-l)^2d^2}{4}\right\} \left[\int t^4 \exp^{-t^2} dt - \left(2 + \frac{(k-l)^2d^2}{2}\right) \int t^2 \exp^{-t^2} dt \right. \\ + \quad & \left. \left(1 - \frac{(k-l)^2d^2}{2} + \frac{(k-l)^4d^4}{16}\right) \int \exp^{-t^2} dt\right]. \end{aligned} \tag{5.8}$$

For Gaussian integrals it is well established that

$$\int_{-\infty}^{\infty} t^n \exp^{-t^2} dt = \Gamma\left(\frac{n+1}{2}\right). \tag{5.9}$$

As a result

$$\int t^4 \exp^{-t^2} dt = \Gamma\left(\frac{5}{2}\right) = \frac{3\sqrt{\pi}}{4}$$

$$\int t^2 \exp^{-t^2} dt = \Gamma\left(\frac{3}{2}\right) = \frac{\sqrt{\pi}}{2}$$

and

$$\int \exp^{-t^2} dt = \Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}.$$

Using these, one can rewrite (5.8) as

$$\begin{aligned} & \exp^{-\left(\frac{(k-l)^2d^2}{4}\right)} \left[\frac{3\sqrt{\pi}}{4} - \frac{\sqrt{\pi}}{2}\left(2 + \frac{(k-l)^2d^2}{2}\right) + \sqrt{\pi}\left(1 - \frac{(k-l)^2d^2}{2} + \frac{(k-l)^4d^4}{16}\right)\right] \\ = \quad & \frac{3\sqrt{\pi}}{4} \exp^{-\left(\frac{(k-l)^2d^2}{4}\right)} \left[1 - (k-l)^2d^2 + \frac{(k-l)^4d^4}{12}\right] \end{aligned} \tag{5.10}$$

If one substitutes (5.10) back into (5.6) one now has

$$\int \left[\hat{f}''(x)\right]^2 dx = \frac{3}{8\sqrt{\pi}m^2\sigma_c^5} \left[m + \sum_{k \neq l} \exp^{-\left(\frac{(k-l)^2d^2}{4}\right)} \left[1 - (k-l)^2d^2 + \frac{(k-l)^4d^4}{12}\right]\right].$$

Now substituting $s = k - l$,

$$\int \left[\hat{f}''(x)\right]^2 dx = \frac{3}{8\sqrt{\pi}m\sigma_c^5} \left[1 + \frac{1}{m} \sum_{s=1}^{m-1} 2(m-s) \exp^{-\frac{d^2s^2}{4}} \left[1 - s^2d^2 + \frac{s^4d^4}{12}\right]\right]$$

which can be rewritten as

$$\int \left[\hat{f}''(x)\right]^2 dx = \frac{3}{8\sqrt{\pi}m\sigma_c^5} \left[1 + F(m,d)\right], \tag{5.11}$$

where

$$F(m,d) = \frac{1}{m}\sum_{s=1}^{m-1} 2(m-s)\exp^{-\frac{d^2s^2}{4}}\left[1 - s^2d^2 + \frac{s^4d^4}{12}\right].$$

[In the special case $m = 2$, an equivalent formulation of this result was provided by Zhang and Wang (2009).] Substituting (5.11) into the expression for h_{opt}, (1.64), one obtains

$$h_{\text{opt}} = \kappa_0\left(\frac{8\sqrt{\pi}}{3}\right)^{1/5} m^{1/5}n^{-1/5}\sigma_c\left[1 + F(m,d)\right]^{-1/5}. \tag{5.12}$$

It is important to remember that here σ_c is the *component* standard deviation, which is different from the *overall* standard deviation. However, the following shows how the component standard deviation can be written in terms of the overall one.

With $\hat{f}(x)$ as (5.5), a mixture of normal densities, the expectation can be written as

$$E(X) = \frac{1}{m}\sum_{k=1}^{m}\mu_k.$$

Equally,

$$E(X^2) = \frac{1}{m}\left(\sum_{k=1}^{m}\mu_k^2 + m\sigma_c^2\right),$$

and so the variance

$$\begin{aligned}
\text{Var}(X) &= E(X^2) - [E(X)]^2 \\
&= \frac{1}{m}\left(\sum_{k=1}^{m}\mu_k^2 + m\sigma_c^2\right) - \left(\frac{1}{m}\sum_{k=1}^{m}\mu_k\right)^2 \\
&= \frac{1}{m}\left(\sum_{k=1}^{m}\mu_k^2 + m\sigma_c^2 - \frac{1}{m}\sum_{k=1}^{m}\mu_k^2 - \frac{2}{m}\sum_{k<l}\mu_k\mu_l\right) \\
&= \frac{m-1}{m^2}\left(\sum_{k=1}^{m}\mu_k^2 - \frac{2}{m-1}\sum_{k<l}\mu_k\mu_l\right) + \sigma_c^2.
\end{aligned}$$

Using the same substitution as earlier, $\mu_k = kd\sigma_c$ and $\mu_l = ld\sigma_c$,

$$\begin{aligned}
\text{Var}(X) &= \frac{m-1}{m^2}\left(\sum_{k=1}^{m}k^2d^2\sigma_c^2 - \frac{2}{m-1}\sum_{k<l}kld^2\sigma_c^2\right) + \sigma_c^2 \\
&= \frac{m-1}{m^2}\left(d^2\sigma_c^2\sum_{k=1}^{m}k^2 - \frac{2d^2\sigma_c^2}{m-1}\sum_{k<l}kl\right) + \sigma_c^2. \qquad (5.13)
\end{aligned}$$

$\sum_{k<l} kl$ can be rewritten as $\sum_{i=1}^{m-1} i \sum_{k=i+1}^{m} k$. The following uses some common results for summations.

$$\begin{aligned}
\sum_{i=1}^{m-1} i \sum_{k=i+1}^{m} k &= \sum_{i=1}^{m-1} i \left(\frac{m(m+1)}{2} - \frac{i(i+1)}{2} \right) \\
&= \frac{m^2(m-1)(m+1)}{4} - \frac{1}{2} \sum_{i=1}^{m-1} i^2(i+1) \\
&= \frac{m^2(m-1)(m+1)}{4} - \frac{1}{2} \sum_{i=1}^{m-1} i^3 - \frac{1}{2} \sum_{i=1}^{m-1} i^2 \\
&= \frac{m^2(m^2-1)}{4} - \frac{m^2(m-1)^2}{8} - \frac{m(m-1)(2m-1)}{12} \\
&= \frac{m(3m^3+2m^2-3m-2)}{24} \qquad (5.14)
\end{aligned}$$

This can then be substituted into (5.13), as well as the well-known result

$$\sum_{k=1}^{m} k^2 = \frac{m(m+1)(2m+1)}{6}.$$

The variance then becomes

$$\begin{aligned}
\mathrm{Var}(X) &= \frac{(m-1)d^2\sigma_c^2}{m^2} \left(\frac{m(m+1)(2m+1)}{6} - \frac{m(3m^3+2m^2-3m-2)}{12(m-1)} \right) + \sigma_c^2 \\
&= \frac{(m-1)d^2\sigma_c^2}{6m} \left((m+1)(2m+1) - \frac{1}{2}(m+1)(3m+2) \right) + \sigma_c^2 \\
&= \sigma_c^2 \left(1 + (m^2-1)\frac{d^2}{12} \right) \qquad (5.15)
\end{aligned}$$

So, σ_c^2 can be estimated by $s^2/(1+(m^2-1)\frac{d^2}{12})$, where s is the overall sample standard deviation. Substituting this into (5.12), and using now $\kappa_0 = 0.776$ for a Gaussian kernel, yields

$$h_{\mathrm{opt}} = 1.06 m^{-\frac{4}{5}} n^{-\frac{1}{5}} s \frac{2\sqrt{3}}{d\sqrt{1+(\frac{12}{d^2}-1)/m^2}\,[1+F(m,d)]^{\frac{1}{5}}}. \qquad (5.16)$$

This expression still contains the unknown d, and a tool which does not involve the estimation of this, or the computation of an expression as cumbersome as (5.16), would be preferred. One area where further simplification is possible is by specifying a value of d. A value of $d = 2\sqrt{3}$ has a number of advantages, as well as representing a fairly typical distribution, where the

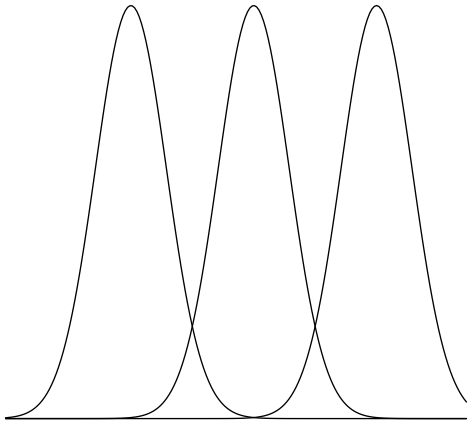

Figure 5.1: Three normals, each separated by a distance of $d = 2\sqrt{3}$ component standard deviations.

modes overlap slightly. Fig. 5.1 shows a distribution of this shape. This value of d considerably simplifies (5.15) to $\mathrm{Var}(X) = m^2\sigma_c^2$. It also performs favourably when examining $F(m, d)$, as is shown by the curves in Fig. 5.2 in which the function is plotted over a range of d values. Each curve represents a different m value for $m = 1, ..., 10$. For $d = 2\sqrt{3}$, $F(m, d)$ is approximately zero for all sensible values of m, for example $F(2, 2\sqrt{3}) = 0.050$, $F(3, 2\sqrt{3}) = 0.067$, and $F(4, 2\sqrt{3}) = 0.076$. Also, $F(m, d)$ is only included in (5.16) in a fifth root, and so it is safe to assume that $[1 + F(m, d)]^{\frac{1}{5}} \approx 1$. In taking these simplifications into account (5.16) becomes

$$h_m^* = 1.06 m^{-\frac{4}{5}} s n^{-\frac{1}{5}}. \tag{5.17}$$

This is a simple rule of thumb, which, in the same way as the normal reference rule does, only makes use of the spread of the data. In fact, it only differs from it by a factor of $m^{-4/5}$.

Table 5.1 shows the factor in h_m^* which is dependent on m and shows how it decreases as m increases. These factors differ significantly from the

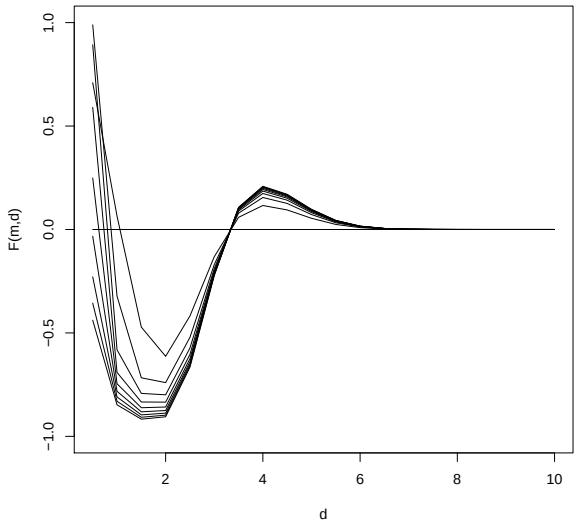

Figure 5.2: The function $F(m, d)$ plotted over a range of d values. Each curve represents a different m value, $m = 1, ..., 10$.

m	1	2	3	4	5	6	7	8
$m^{-4/5}$	1.000	0.574	0.415	0.329	0.276	0.235	0.211	0.189

Table 5.1: Multimodal correction factor $m^{-4/5}$ for $m = 1, ..., 8$ modes.

equivalent 0.85 suggested by Silverman (1986) to cope with any modality. Similarly to (5.4), it is necessary to anticipate prior to the density estimation how many modes one expects the data to have in order to choose a bandwidth tailored to the data set.

5.2 Investigating these methods using real data sets

These two bandwidth selection methods were tested on a variety of real data sets. Here the full analysis and results are presented for a traffic flow data set, before summarizing the results for the other data sets in Table

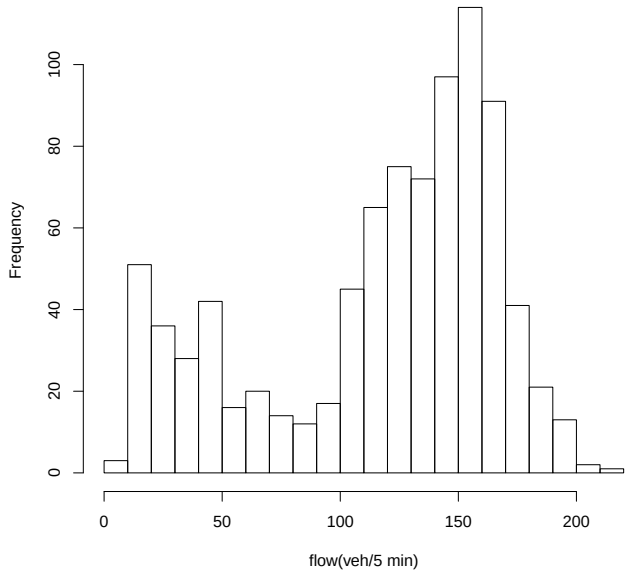

Figure 5.3: A histogram showing the *traffic flow* data.

5.3. The traffic data consists of $n = 876$ measurements of traffic flow (in vehicles/ 5 min.), taken from 10-12/07/07 on a Californian freeway. This data is retrieved from PeMS (http://pems.dot.ca.gov/). Fig. 5.3 shows a histogram of this data which suggests that the approximate shape of the data is bimodal. In any case it is almost certainly multimodal, which makes either (5.4) or (5.17) more suitable than the simple normal reference rule. Traffic engineers believe that such data tend to have at least two modes, one corresponding to freeflow, and another one to busy traffic.

Firstly, the fitting of a Gaussian mixture bandwidth selection method, (5.4), is examined. Using **npmlreg**, mixture parameters were estimated for $m = 1, ..., 4$. These are displayed in Table 5.2. Fig. 5.4 then displays these Gaussian mixtures. These are the densities which are being used in (5.4) as an approximation of the true density. When $m = 1$, h_m is identical to the normal reference rule, and this is clearly the worst approximation of the true data out of the four. Of interest are the bottom two plots in Fig. 5.4,

m	$\hat{\mu}_k$	$\hat{\sigma}_k$	$\hat{\pi}_k$
1	117.64	50.83	1.00
2	36.34, 142.16	18.36, 25.76	0.23, 0.77
3	34.87, 137.42, 160.21	17.14, 27.56, 8.70	0.22, 0.65, 0.13
4	17.54, 44.15, 138.92, 160.41	3.69, 15.39, 26.85, 7.00	0.07, 0.16, 0.67, 0.11

Table 5.2: The mixture parameters estimated for $m = 1, ..., 4$ using the **npmlreg** package for the *traffic flow* data.

which both show a mixture density with one less mode than the number of densities used to generate it. This highlights the fact that the number of mixture components is an upper bound on the number of modes. It is unimportant that the value of m specified is not replicated in the number of modes in the mixture, and the reason for this will be explained in the discussion at the end of this chapter.

With the Gaussian mixtures estimated, the bandwidths $h_1, ..., h_4$ can then be calculated. Fig. 5.5 (top) shows the estimated density estimate when incorporating each of these. It appears that using h_1, equivalent to h_{NR}, the density estimate is oversmoothed. The estimates using both h_2 and h_3 are likely to be of a more adequate resolution. Anticipating $m = 2$ reveals a third mode for small flow values, and anticipating $m = 3$ reveals potential fourth and fifth modes at flow values of approximately 70 and 125 veh/5 min. The possible existence of these further modes is completely missed when h_{NR} is implemented. By choosing a value of m which is too high, such as $m = 4$, the density estimate clearly becomes undersmoothed.

The rule of thumb, (5.17), was also trialled on the *traffic flow* data set. This method is the subject of a simulation study later, so only a brief analysis will be offered here. This is much more straightforward to implement and requires no fitting of a mixture, instead only the value of m within (5.17) itself needs to be varied. Fig. 5.5 (bottom) shows the density estimates using $h_1^*, ..., h_4^*$. This is the same analysis as given in Fig. 5.5 (top), but using a different bandwidth selection method. The results and conclusions are very similar to those for Fig. 5.5 (top), with the only real difference

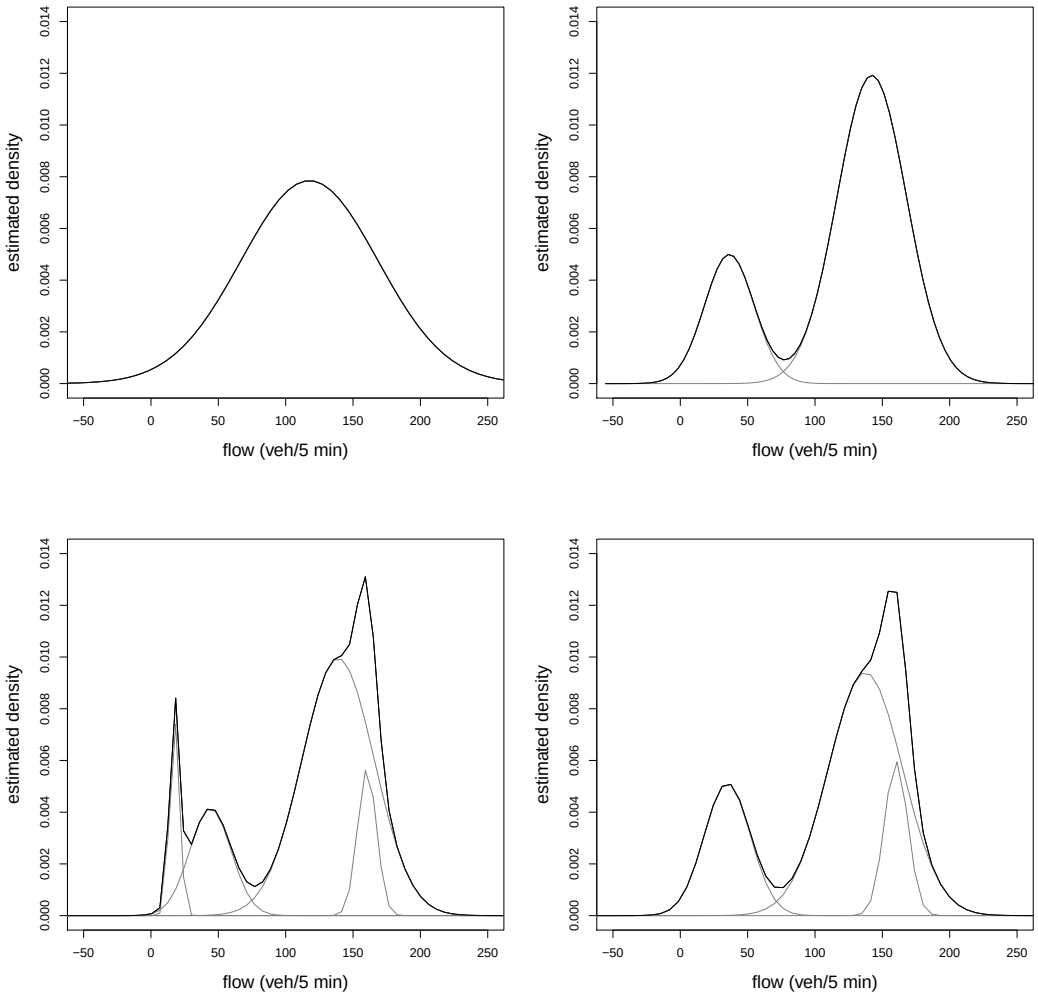

Figure 5.4: The Gaussian mixtures generated by **npmlreg** for the *traffic flow* data from $m = 1$ (top left) to $m = 4$ displayed clockwise. In each plot the black curve is the mixture density and the grey curves are the individual component densities.

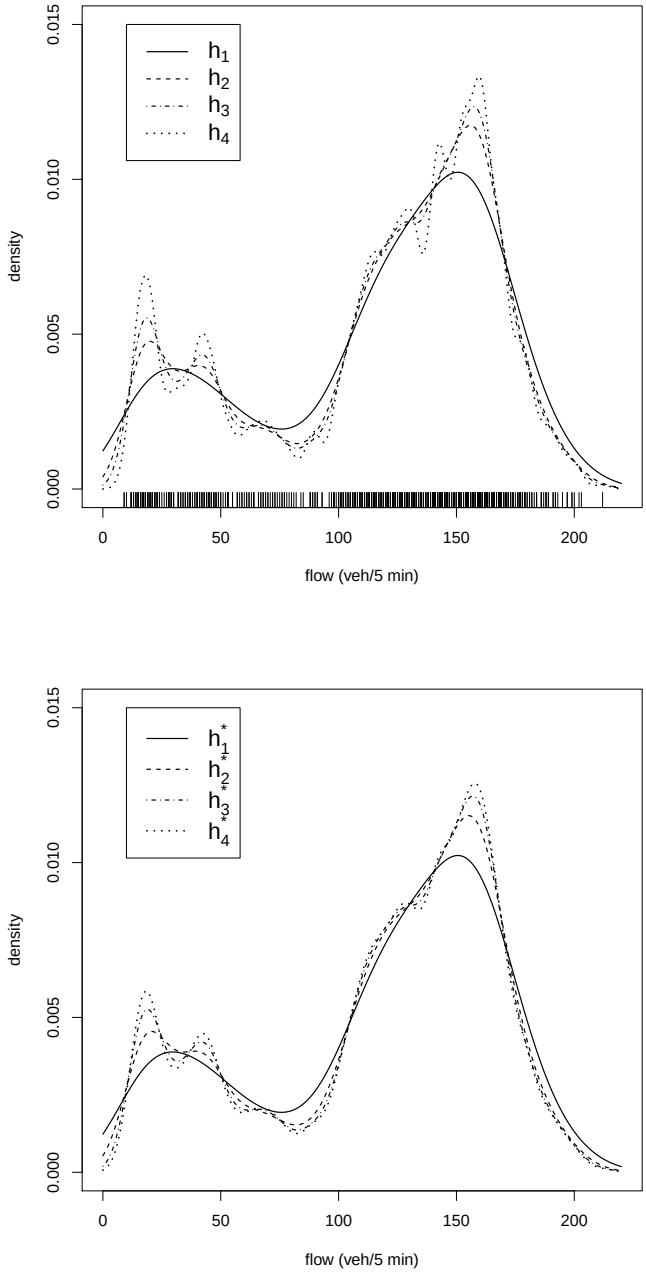

Figure 5.5: Top: Estimated densities for *traffic flow* using $h_1, ..., h_4$. Bottom: Estimated densities for *traffic flow* using $h_1^*, ..., h_4^*$.

being that the rule of thumb method is less temperamental with a higher value of m.

This analysis was carried out on several further data sets. These are described briefly below and the results are presented in Table 5.3.

- *Traffic speed* is data concerning the same $n = 876$ traffic measurements as in the *traffic flow* data above. Here the variable of interest is speed in m.p.h.

- *Galaxy* is a well-known data set from the **MASS** package on R (see Venables and Ripley (2002)), comprising of the velocities in 1000km/s of $n = 82$ galaxies from an unfilled survey of the Corona Borealis region.

- *Penny thickness* is from the **locfit** package, by Loader (2010), on R measuring the thickness of two U.S. pennies every year from 1945 to 1989.

- *Eruptions* is the eruptions variable from the well-known *faithful* data set, which measures the eruption time in minutes of $n = 272$ eruptions of the Old Faithful geyser in Yellowstone National Park.

- *Energy use* is the log-energy consumption, in kg oil equivalent per capita in the year 2007, for a sample of $n = 135$ countries. This data was retrieved from the World Bank data base. See http://data.worldbank.org/indicator/EG.USE.PCAP.KG.OE.

For each of the data sets in Table 5.3, both h_m and h_m^* were calculated for $m = 1, ..., 4$. The number of modes was then observed when $\hat{f}(x)$ was plotted using each of these bandwidths. Throughout the data sets, it is clear that for both bandwidth selection methods the number of modes observed is rarely equal to the number of modes anticipated. One can also see that h_m^* is fairly effective as a rule of thumb since it shows similar behaviour to the data dependent h_m. Generally, the two bandwidth selectors decrease at a similar rate as m increases and one almost always observes at least as many modes as one anticipates. Increasing m to be greater than 1 usually increases the

Data	m	1	2	3	4
Traffic flow	$\int[\hat{f}_m''(x)]^2dx$ in h_m	6.24e-10	1.75e-08	8.44e-08	1.74e-06
	h_m	13.89	7.13	5.20	2.84
	Modes observed (h_m)	2	3	5	8
	h_m^*	13.89	7.97	5.77	4.57
	Modes observed (h_m^*)	2	3	3	5
Traffic speed	$\int[\hat{f}_m''(x)]^2dx$ in h_m	4.86e-08	0.00058	0.0032	0.0037
	h_m	5.81	0.89	0.63	0.61
	Modes observed (h_m)	2	12	16	17
	h_m^*	5.81	3.34	2.41	1.92
	Modes observed (h_m^*)	2	3	3	3
Galaxy	$\int[\hat{f}_m''(x)]^2dx$ in h_m	0.000107	0.00486	0.118	0.12
	h_m	2.00	0.93	0.49	0.49
	Modes observed (h_m)	3	4	7	7
	h_m^*	2.00	1.15	0.83	0.66
	Modes observed (h_m^*)	3	3	5	5
Penny thickness	$\int[\hat{f}_m''(x)]^2dx$ in h_m	0.01	0.0496	0.257	0.284
	h_m	0.79	0.57	0.41	0.41
	Modes observed (h_m)	1	1	3	3
	h_m^*	0.79	0.46	0.33	0.26
	Modes observed (h_m^*)	1	3	6	7
Eruptions	$\int[\hat{f}_m''(x)]^2dx$ in h_m	0.11	38.03	62	5056.5
	h_m	0.39	0.12	0.11	0.05
	Modes observed (h_m)	2	3	3	14
	h_m^*	0.39	0.23	0.16	0.13
	Modes observed (h_m^*)	2	2	2	2
Energy use	$\int[\hat{f}_m''(x)]^2dx$ in h_m	0.151	0.961	2.93	2.98
	h_m	0.43	0.29	0.24	0.23
	Modes observed (h_m)	2	2	2	2
	h_m^*	0.43	0.24	0.18	0.14
	Modes observed (h_m^*)	2	2	3	4

Table 5.3: h_m and h_m^* for various data sets for $m = 1, ..., 4$ and the number of modes observed in each case.

number of modes produced which suggests the oversmoothing problem of h_{NR} is avoided. However, this does not always happen, as is exemplified by the *energy use* data which is promising in itself since it shows a certain robustness to the choice of m. One difference between the two methods is that h_m^* would appear to be significantly less temperamental. This is evident in the *traffic flow*, *eruptions* and *traffic speed* data, where an unrealistically high number of modes is observed for $m = 4$ when using h_m. This represents an obvious overfitting, and a clear disadvantage when compared with the rule of thumb.

5.3 Simulation study

The following simulation study demonstrates further the efficiency of the rule of thumb, (5.17). In order to do this it is important to remember that the goal is to produce the best quality density estimate possible, not to produce in the density estimate the number of modes that were anticipated when m was selected. In any case, as was demonstrated with the real data sets, it is relatively rare to attain a number of modes equal to m. To measure the quality of the density estimate the MSE is examined. Here

$$MSE(f, \hat{f}) = \frac{1}{n} \sum_{i=1}^{n} \left\{ \hat{f}(X_i) - f(X_i) \right\}^2 \tag{5.18}$$

is an empirical version of (1.55). The question of interest is whether one attains, for a data set generated from a distribution of known modality, the best density estimate, in terms of MSE, when the value of m used in (5.17) is equal to the number of modes in that data set. The following results suggest that this is the case.

Data sets of size $n = 500$ were generated from Gaussian mixtures made up of a number of component densities. In this study, the number of components, c, varies from 1 to 4. In total, data sets were generated from 8 different distributions. The specifications from which these were generated are given in Table 5.4. Data sets (a)-(d) are generated from an *ideal* scenario i.e. the scenario under which the rule of thumb was derived. Here,

Density	c	μ_k	σ_k	π_k
(a)	1	0	1	1
(b)	2	0, $2\sqrt{3}$	1, 1	0.5, 0.5
(c)	3	0, $2\sqrt{3}$, $4\sqrt{3}$	1, 1, 1	0.33, 0.33, 0.33
(d)	4	0, $2\sqrt{3}$, $4\sqrt{3}$, $6\sqrt{3}$	1, 1, 1, 1	0.25, 0.25, 0.25, 0.25
(e)	2	0, 2	1, 0.5	0.8, 0.2
(f)	2	0, 0.7	0.2, 0.4	0.4, 0.6
(g)	3	0, 2, 3	0.8, 0.3, 0.3	0.1, 0.4, 0.5
(h)	4	0, 1, 2, 3	0.3, 0.3, 0.3, 0.3	0.2, 0.3, 0.1, 0.4

Table 5.4: The mixture parameters, and the number of components, used to generate the simulated densities (a)-(h).

the data are simulated from an equal mixture of c Gaussian densities with equal standard deviation and with a distance of $2\sqrt{3}$ component standard deviations between them. For these data sets the rule of thumb produces the asymptotically optimal bandwidth. Data sets (e)-(h) are more complex, and for these h_m^* is indeed only a rule of thumb.

Each data set was generated 200 times, and the MSE was calculated each time as in (5.18) with h_m^* as the bandwidth in the density estimate. This was done for $m = 1, ..., 6$ for each data set. Fig. 5.6-7 show the results of this study, where Fig. 5.6 includes the ideal densities, and Fig. 5.7 the less ideal. The left hand column of each of these shows the mixture densities, and alongside each of these is a box plot displaying the 200 MSEs for each value of m for that mixture. For comparison, the rule of thumb of Silverman (1986), whereby one replaces 1.06 by 0.9 in h_{NR}, is also included, denoted S.

Additionally, Table 5.5 shows the percentage of times, out of the 200 simulations of each data set, that each value of m, when used in h_m^*, led to the smallest MSE for that simulation. Here, the suggestion from Silverman (1986) is not included. These figures and the table all suggest that the rule of thumb is successful. Firstly, for the ideal scenarios in Fig. 5.6, as would be expected, when $m = c$ (the modality is anticipated correctly in

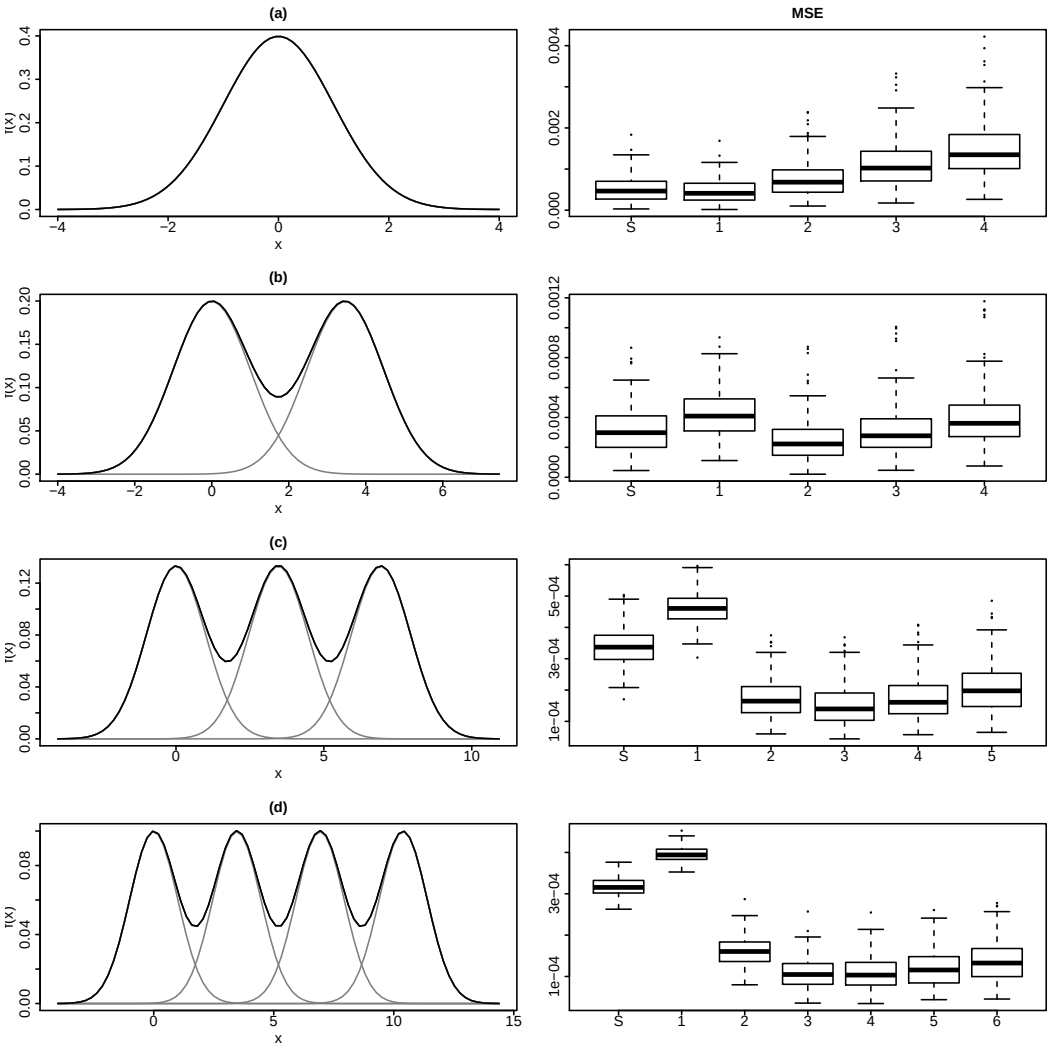

Figure 5.6: Results for (a)-(d). Left: The generating densities with the individual component densities which form these shown in grey. Right: Box plots representing the 200 MSEs using the rule of thumb, (5.17), for $m = 1, ..., 6$. Silverman's rule of thumb, whereby one replaces 1.06 by 0.9 in h_{NR}, is also included, denoted S.

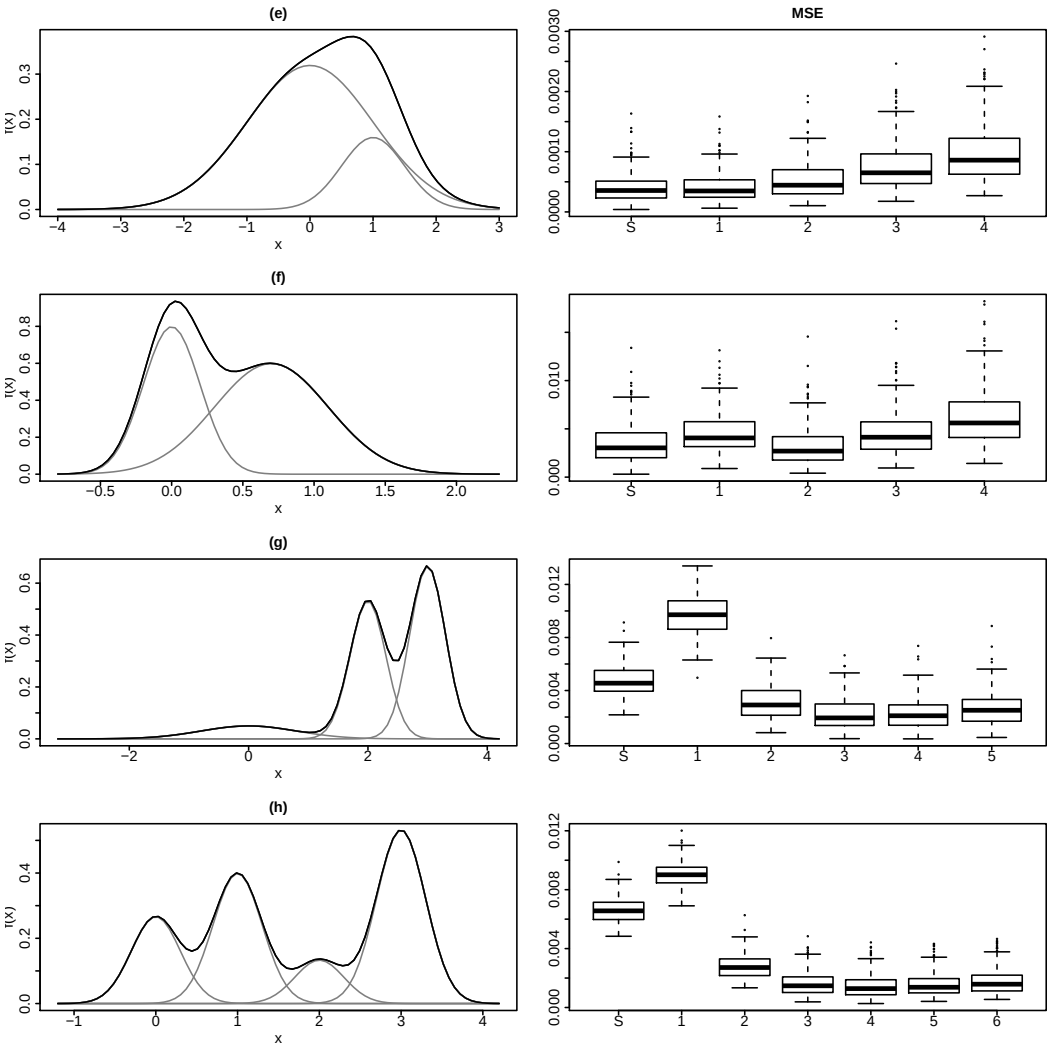

Figure 5.7: Results for (e)-(h). Left: The generating densities with the individual component densities which form these shown in grey. Right: Box plots representing the 200 MSEs using the rule of thumb, (5.17), for $m = 1, ..., 6$. Silverman's rule of thumb, whereby one replaces 1.06 by 0.9 in h_{NR}, is also included, denoted S.

Density	c	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$	$m=6$
(a)	1	**90**	10	0	0	0	0
(b)	2	6	**81**	13	0	0	0
(c)	3	0	18	**69**	13	0	0
(d)	4	0	0	38	**52**	9	1
(e)	2	**72**	28	0	0	0	0
(f)	2	15	**81**	4	0	0	0
(g)	3	0	6	**48**	36	9	1
(h)	4	0	0	22	**52**	23	3

Table 5.5: The percentage of times, out of the 200 simulations of each data set, that each value of m, when used in the rule of thumb (5.17), led to the smallest MSE for that simulation. The largest percentage for each density is expressed in bold.

the bandwidth selection), the MSE tends to be lowest. This is supported by the first four lines of Table 5.5. Encouragingly, the same behaviour is exhibited by the more complex densities where for every density, with the exception of (e), the MSE is minimal most frequently for $m = c$. However, the results for density (e) are also positive, and the reason for this can be seen by examining the plot of the generating mixture density. This reveals that although $c = 2$, the actual modality is only one, since one component is swamped by the other. For this data set, the bandwidth which performs best is h_1^* i.e. when one anticipates that the modality is one. Since the true modality here is one, this means that the bandwidth technique has performed best throughout when m is chosen as the *modality* of the generating mixture density. This highlights that it is the number of modes which is important rather than the number of components, which is useful since one is more likely to have an idea about the modality of a data set rather than the number of component densities which forms it. This study shows that, when the modality is anticipated accurately, this rule of thumb outperforms the normal reference rule ($m = 1$) as well as Silverman's suggested adaptation, when the true modality of the density is greater than

one.

5.4 Discussion

Two potential bandwidth selection methods have been presented in this chapter, and, whilst they have both been shown to be effective, there are several reasons why the rule of thumb should be favoured over the fitting of a Gaussian mixture. The main reason is that fitting a mixture requires significantly more effort, with very few apparent advantages. The fitted mixture is a density estimate in itself so it seems unnecessary to then use this as one step in another density estimation technique. Also, the integration required in this process is relatively demanding. As was shown with the real data sets earlier, the rule of thumb is also less temperamental when larger values of m are specified. This is an advantage in a type of technique which depends on the choice of m, a choice which often will be made without knowing exactly the true modality. It is useful that h_m^* is somewhat robust to misspecification of m.

An important point for discussion is the idea of anticipating the modality. To use either of these methods it is important to have some idea of the modality. It is important that one should not solely use some other density estimation tool, such as a histogram, to ascertain the modality of the data set, since this also depends on an initial bandwidth choice, which may not suit the true modality. Therefore, one needs some sort of notion of what the modality should be from an external source. This is the case with the *traffic flow* data, as discussed earlier, and similar ideas exist for many other data sets. As mentioned, h_m^* is somewhat robust to misspecification of m, and so there is some margin for error.

As was explained in the introduction to the simulation study, the modality is anticipated in order to produce the best possible density estimate, not to produce a density estimate of the anticipated modality. Indeed the aim of density estimation is not usually to assess how many modes it has. The simulation study showed that the rule of thumb did indeed achieve the best density estimates for a suitable choice of m, which indicates that the con-

cept of modality-dependent bandwidths is sensible. Recall that the quantity approximated in these methods is $\int [f''(x)]^2 \, dx$, which is a functional of the *curvature* of the density. Therefore the rule of thumb quantifies how much curvature one would expect in a typical m-modal density, and effectively introduces this amount of curvature into the estimated density, via (5.17). The Gaussian mixture fitting method works in the same way, but using an even closer, data-dependent, approximation of the curvature in the true density. It then depends on the data set, how many modes are caused by incorporating a certain amount of curvature into the density estimate. The amount of curvature required for the most accurate density estimate is not necessarily the amount of curvature which gives exactly m modes in the estimate and this explains why it is not realistic to expect the number of anticipated modes to appear in the density estimate. In the derivation of the rule of thumb, if one takes $d = 3$, this can sometimes lead to the number of modes observed being closer to the number anticipated, however since this is not the priority, and this makes the rule of thumb considerably less neat, the value of $d = 2\sqrt{3}$ is retained.

When viewed from this perspective the normal reference rule of Silverman (1986) appears extremely restrictive. It incorporates an amount of curvature in to the density estimate which is typical in a normal unimodal density. This is clearly insufficient for many data sets, and explains the oversmoothing which often results from the application of this bandwidth selector. In comparison, h_m^*, as well as h_m, is capable of producing a density estimate of a more appropriate resolution. This was shown in the *traffic flow* data, in which features of the density were revealed, which were missed when h_{NR} was implemented. Since (5.17) is a rule of thumb method, for which the priorities are simplicity and convenience, it is only fair to compare its performance with that of existing simple procedures, such as h_{NR}. Due to the nature of rule of thumb methods in general, it is likely that more sophisticated methods, such as those discussed at the start of the chapter, will outperform (5.17) in estimating the true density. However it should be noted that, due to its simplicity, it fares well when compared to these

sophisticated methods, in terms of computational problems and robustness to outliers. Zhang and Wang (2009) describe the same advantages for their simple procedure.

It would have been useful to be able to apply the bandwidth selection techniques, developed in this chapter, to the work on conditional density estimation bandwidth selection in the previous chapter. Unfortunately, in that setting, the modified normal reference rule undersmooths unimodal conditional densities and consequently works well for multimodal densities. If this was not the case then the methods in this chapter may be applicable, however due to the fact that these methods select a bandwidth smaller than the normal reference rule for a multimodal density and do not adjust the strategy for a unimodal density, these techniques could not improve the bandwidth selection methods of the previous chapter as they stand.

Chapter 6

Overview and applications

The purpose of this chapter is to give an overview of the techniques presented in this thesis and provide some useful comparisons and potential applications.

An important consideration in every setting in this thesis is bandwidth selection, either in scalar or matrix form. Chapter 5 is devoted to bandwidth selection in univariate kernel density estimation. This is the only chapter dedicated to univariate techniques, but it has relevance when considered alongside the conditional density estimation in Chapter 4. The rule of thumb method (5.17) presented in Chapter 5 compares favourably with other kernel density bandwidth selection techniques, such as the normal reference rule and LSCV. An important concept here is that one must have a notion, prior to estimation, of the true number of modes in the data. For some data, such as the *traffic flow* data, industry experts have an idea of an expected modality. In other cases, one might have the notion that the density of a data set is definitely not unimodal, in which case applying the rule of thumb with $m = 2$ will give a more accurate density estimate than the normal reference rule. Also, as mentioned earlier, there are various methods in the literature which aim to estimate the modality of a distribution, so these could possibly be applied here.

The issue of modality is also relevant in Chapter 4 in the setting of *multivariate conditional* density estimation. Here, one needs a notion of the

modality of the response distribution to choose an appropriate bandwidth selection tool for b. It seems likely that the multimodal correction factor, $m^{-4/5}$ (see Table 5.1), presented in Chapter 5 would be useful for any kernel density estimation, whether conditional or not. It seems sensible that the amount of curvature assigned by the rule of thumb to an $m-$modal kernel density estimate would apply equally to its conditional counterpart. However, it is difficult to confirm this since the bandwidth selection tool presented in Chapter 4 already undersmooths, and so a further reduction in the bandwidth for anticipated higher modality is not appropriate. In both Chapters 4 and 5, the only reason that adaptations, depending on modality, to the bandwidth selection are needed, is that the method initially proposed is not perfect, and either undersmooths (Chapter 4) or oversmooths (Chapter 5). Having said that, the tools presented here should not be used for the purpose of determining the modality of a distribution.

The other bandwidth selection technique introduced in this thesis, in Chapter 3, is AGCV, (3.1), for multivariate local polynomial regression. Through the removal of isolated data points and the introduction of the median, this method has become robust to sparse regions of data. The removal of isolated points could also be applied in $Q(h)$, (4.9), the bandwidth selection tool associated with the covariates in modal regression, in Chapter 4. This method has not been trialled with data of $d > 2$ although it can be performed in higher dimensions, with the only limitation being that it is not as attractive to visualize as when bivariate data is examined. If this was to be carried out, it is likely that adaptations to $Q(h)$ would be valuable.

It has been demonstrated that AGCV is more effective, quicker and less sensitive to the minimization starting point than competing methods. The two adjustments made to GCV were in response to computational difficulties encountered on R, however they are both helpful when using any software. The inclusion of the median ensures that extremely large bandwidth parameters are not chosen, and the removal of isolated points is a sensible measure when considering the threshold developed in this thesis. As mentioned earlier, the estimation at points accepted by the density threshold is improved

by removing sufficiently isolated points from the bandwidth selection process. This allows relatively small values of h_j to be chosen, which yield improved estimates in denser regions, when compared with larger bandwidths which others may employ as a solution to the curse of dimensionality.

It is useful that the threshold developed in Chapter 2, as the primary method of tackling the curse of dimensionality, works so neatly with AGCV. It was shown in Chapter 2 that the density threshold, derived from a bound on the influence, and so a bound on the variance, was successful in distinguishing where over a data range local polynomial regression could be considered sensible. Due to the relationship, mentioned in Chapter 4, between modal regression and Nadaraya-Watson regression, this threshold can also be used in this context to try to prevent problems arising from data sparsity in modal regression. The formula for the threshold, (2.18), neatly takes the form of a multiple of the density of one point, and in this way reflects the amount of information required at $\mathbf{x}$ for regression to be considered feasible.

Both mean and modal regression are covered in this thesis as potential nonparametric methods of multivariate regression. It is advisable to use mean regression in most circumstances, since it is optimal in terms of MSE, and this explains why it is covered significantly more than modal regression in the literature. In the author's experience, local linear regression performs better in terms of MSE over the whole data range, apart from on an *edge* (within the interior) where the modal regression outperforms it. As a general rule, it is sensible to use modal regression only when there is a specific reason to do so. This could be that the response is multimodal, or that there are edges in the true function which would suit an edge-preserving technique. However, Chu, Glad, Godtliebsen and Marron (1998) describe a compromise between modal and mean regression. The estimate at x given by the sigma filter, described in this article, is effectively one mean shift iteration from the Y value at x. This is used in image processing and may provide some competition for modal regression when a function with edges is evaluated.

Whilst local mean and modal regression are both studied here, local

median regression is not touched upon. In the univariate setting this has been explored as an alternative and its strength, according to Truong (1989) and others, is its robustness to outliers in the y-direction. Truong (1989) shows that local conditional median regression also has favourable asymptotic properties. The same paper suggests that this type of regression is particularly suitable for data with an asymmetric conditional response distribution (examples given are income and housing data) since the resulting regression is easier to understand. Median regression does not appear to have been developed substantially in the multivariate setting, and so this in-between stage is a possible area of interest.

As discussed, the topics covered in Chapters 4 and 5 are useful for data sets with specific characteristics. The density threshold and AGCV are applicable more widely. Within the subject of local polynomial regression they are suitable for use with any kernel function, degree of polynomial or type of bandwidth matrix. Additionally, the density threshold can be used with data of any dimension, and a desirable property of this threshold is the substantial nature with which it increases as the dimension of the data increases. This was shown using the chamois data in Chapter 2. This characteristic reflects how inappropriate local regression is for $d > 5$ without a large sample size. The threshold itself will work with any combination of sample size and dimension of data, although it is unlikely to find any regions where regression is feasible if an unsuitable combination of these is present. As an approximate guide, for $d \leq 5$, the threshold will typically yield regions where regression is considered sensible for a data set where n is in the hundreds, and for d greater than this realistically the magnitude of n needs to be in the thousands. Of course there may always be individual $\mathbf{x}$-points within the data range where the data is sufficiently clustered for the threshold to accept when n is smaller.

An interesting source of data, for which the sample size is often more likely to be suitable for local regression in high dimensions, is computer-generated data. One of the computer-generated data sets examined with these methods is the *gaia* data from the **LPCM** package, by Einbeck and

Evers (2011). Here $n = 8286$ and up to 19 different variables can be included. The threshold does not discriminate against either data generation method but the use of computer-generated data may lead to some unexpected results unless the user is properly informed. To analyse the performance of the density threshold with data of this type, it would typically be split into training and test data. Frequently in this situation, at test points rejected by the threshold, the regression estimates have been observed to be as good, when compared with the simulated response at that point, as at those accepted by the threshold. The reasons for this are not immediately clear, but it is likely that this is mainly due to the small error variance typically used in computer-generated data. At the points rejected by the threshold, the variance of the regression estimate is large, as one would expect, but this is not reflected obviously in the quality of the estimate. The data is typically clustered and so any test data will not be sufficiently isolated for the estimator to suffer from any serious computational instability. Instead, it is likely that the estimate at such a point is influenced solely by the response value of the nearest training point. Since this response value was generated with a very small error variance, the regression estimate, despite suffering such a high level of variance, could still be considered a reasonable approximation to the response value of the test point, also generated with a small error variance. In any case, the regression estimates at these points do not often appear to be significantly worse than those at points accepted by the threshold. The threshold can still be employed on data sets of this type, and it is still effective at ruling out points at which NaN is returned, as well as points at which estimates with high variance are produced, but it is sensible to apply caution when analysing the performance of the threshold in such a scenario.

The choice of bandwidth matrix is the other factor which influences the magnitude of sample size required to yield regions of feasibility. If the bandwidth parameters are larger, data points further away from $\mathbf{x}$ can be included in estimation at that point and so a smaller value of n is likely to produce points at which estimation is possible. AGCV works well with any

medium-sized data set. If a data set is too large, the computation will be too time-consuming, and this also occurs if $d > 5$, unless standardization is applied and the bandwidth selection is reduced to a single parameter problem. For this reason, the implementation of AGCV and the density threshold combined is best suited to medium-sized data sets of $2 \leq d \leq 5$. However, these techniques can potentially be applied to data of a very high dimension.

There are many data sets for which d is very large, and in some cases $d > n$, such as with genomic data in computational biology. Here, one could use variable selection, as mentioned earlier, to reduce the data set into one which AGCV can handle. Functional data can also be thought of as a form of almost infinitely high dimensional data. Ferraty, Hall and Vieu (2010) introduce an algorithm which reduces functional data, again via variable selection, into a local linear regression problem, with the intention of improving prediction. This local linear problem typically has $2 \leq d \leq 10$ and so once such an algorithm has been implemented, AGCV and the density threshold could be employed to further improve performance in functional data analysis. The threshold could avoid the problems of "*numerical instability*" which Ferraty, Hall and Vieu (2010) describe as an issue for larger values of d.

Some of the ideas presented in this thesis have also been submitted to journals for publication. The topics in Chapters 2 and 3 have been submitted under the title *Challenging the curse of dimensionality in local linear regression* (Taylor and Einbeck), and are also included in the conference proceedings of the International Workshop on Statistical Modelling 2010, under the title *Strategies for local smoothing in high dimensions: using density thresholds and adapted GCV* (Taylor and Einbeck). The ideas in Chapter 5 have been submitted under the title *A mixture-of-normals reference rule for density estimation under multimodality* (Einbeck and Taylor). The material in Chapter 4 is included in the conference proceedings of the International Workshop on Statistical Modelling 2011, under the title *Multivariate regression smoothing through the "falling" net* (Taylor and Einbeck). All of the publications mentioned above in which I am the first author are solely my work, while both authors contributed equally to the paper in which I am the second author.

Bibliography

[1] ADLER, D. and MURDOCH, D. (2011). rgl: 3D visualization device system (OpenGL). R package version 0.92.798. *http://CRAN.R-project.org/package=rgl*

[2] BASHTANNYK, D. and HYNDMAN, R. (2001). Bandwidth selection for kernel conditional density estimation. *Computational Statistics and Data Analysis* **36**, 279–298.

[3] BERLINET, A., GANNOUN, A. and MATZNER-LØBER, E. (1998). Normalité asymptotique d'estimateurs convergents du mode conditionnel. *Canadian Journal of Statistics* **26**, 365–380.

[4] BOWMAN, A.W. (1984). An alternative method of cross-validation for the smoothing of density estimates. *Biometrika* **71**, 353–360.

[5] BOWMAN, A.W. and AZZALINI, A. (1997) *Applied smoothing techniques for data analysis.* Oxford University Press Inc., New York.

[6] BREIMAN, L., MEISEL, W. and PURCELL, E. (1977). Variable kernel estimates of multivariate densities. *Technometrics* **19**, 135–144.

[7] BUJA, A., HASTIE, T. and TIBSHIRANI, R. (1989). Linear smoothers and additive models (with discussion). *Annals of Statistics* **17**, 453–555.

[8] CARREIRA-PERPIÑAN, M.A. (2000). Reconstruction of sequential data with probabilistic models and continuity constraints. In *Advances in Neural Information Processing Systems*(eds. S.A. Solla, T.K. Leen and K.R. Müller), pp. 414-420. MIT Press, Cambridge.

[9] CHACÓN, J.E. (2009). Data-driven choice of the smoothing parametrization for kernel density estimators. *Canadian Journal of Statistics* **37**, 249–265.

[10] CHENG, Y. (1995). Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **17**, 790–799.

[11] CHERKASSY, V. and MA, Y. (2005). Multiple model regression estimation. *IEEE Transactions on Neural Networks* **16**, 785–797.

[12] CHU, C.K., GLAD, I.K., GODTLIEBSEN, F. and MARRON, J.S. (1998). Edge-preserving smoothers for image processing (with discussion). *Journal of the American Statistical Association* **93**, 526–541.

[13] CHU, C.K. and MARRON, J. (1991). Choosing a kernel regression estimator (with discussion). *Statistica Sinica* **6**, 404–436.

[14] CLEVELAND, W.S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association* **74**, 829–836.

[15] CLEVELAND, W.S. and DEVLIN, S.J. (1988). Locally weighted regression: an approach to regression analysis by local fitting. *Journal of the American Statistical Association* **83**, 596–610.

[16] CLEVELAND, W.S. and LOADER, C. (1996). *Smoothing by local regression: Principles and methods (Tech. Rep.).* AT&T Bell Laboratories, Murray Hill.

[17] COMANICIU, D. and MEER, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**, 603–619.

[18] CORNILLON, P.A., HENGARTNER, N. and MATZNER-LØBER, E. (2011). *Recursive bias estimation for multivariate regression smoothers (Tech. Rep.).* arXiv.

[19] CRAVEN, P. and WAHBA, G. (1979). Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik* **3**, 377–403.

[20] DELAIGLE, A. and MEISTER, A. (2011). Nonparametric function estimation under Fourier-oscillating noise. *Statistica Sinica* **21(3)**, 1065–1092.

[21] DUCHON, J. (1977). Splines minimizing rotation-invariant semi-norms in Sobolev spaces. *Constructive Theory of Functions of Several Variables,* W. Schempp and K. Zeller, eds., 85–100. Springer-Verlag, Berlin.

[22] EILERS, P.H.C. and MARX, B.D. (1996). Flexible smoothing with B-splines and penalties. *Statistical Science* **11**, 89–121.

[23] EINBECK, J. and EVERS, L. (2011). LPCM: Local principal curve methods. R package version 0.44-5. *http://CRAN.R-project.org/package=LPCM*

[24] EINBECK, J., DARNELL, R. and HINDE, J. (2009). npmlreg: Nonparametric maximum likelihood estimation for random effect models. R package version 0.44. *http://CRAN.R-project.org/package=npmlreg*

[25] EINBECK, J. and TUTZ, G. (2006). Modelling beyond regression functions: An application of multimodal regression to speed-flow data. *Applied Statistics* **55**, 461–475.

[26] FAHRMEIR, L. and TUTZ, G. (2001). *Multivariate Statistical Modelling Based on Generalized Linear Models.* Springer, New York.

[27] FAN, J. (1992). Design-adaptive nonparametric regression. *Journal of the American Statistical Association* **87**, 998–1004.

[28] FAN, J. (1993). Local linear regression smoothers and their minimax efficiencies. *Annals of Statistics* **21**, 196–216.

[29] FAN, J. and GIJBELS, I. (1992). Variable bandwidth and local linear regression smoothers. *Annals of Statistics* **20**, 2008–2036.

[30] FAN, J. and GIJBELS, I. (1996). *Local polynomial modelling and its applications.* Chapman and Hall, London.

[31] FAN, J., HU, T. and TRUONG, Y.K. (1994). Robust nonparametric function estimation. *Scandinavian Journal of Statistics* **21**, 433–446.

[32] FAN, J., YAO, Q. and TONG, H. (1996). Estimation of conditional densities and sensitivity measures in nonlinear dynamical systems. *Biometrika* **83**, 189–206.

[33] FERRATY, F., HALL, P. and VIEU, P. (2010). Most-predictive design points for functional data predictors. *Biometrika* **97**, 807–824.

[34] FOWLKES, E.B. (1987). Some diagnostics for binary logistic regression via smoothing (with discussion). *Proceedings of the Statistical Computing Section, American Statistical Association* **1**, 54–56.

[35] FRIEDMAN, J.H. and STUETZLE, W. (1981). Project pursuit regression. *Journal of the American Statistical Association* **76**, 817–823.

[36] FUKUNAGA, K. and HOSTETLER, L.D. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory* **21**, 32–40.

[37] FURRER, R., NYCHKA, D. and SAIN, S. (2011). fields: Tools for spatial data. R package version 6.6.1. *http://CRAN.R-project.org/package=fields*

[38] GENTLE, J.E., HÄRDLE, W. and MORI, Y. (2004). *Handbook of computational statistics: Concepts and methods.* Springer-Verlag, Berlin.

[39] GREEN, P.J. and SILVERMAN, B.W. (1994). *Nonparametric regression and generalized linear models: A roughness penalty approach.* Chapman and Hall, London.

[40] HÄRDLE, W. (1991). *Smoothing techniques with implementation in S.* Springer, New York.

[41] HÄRDLE, W., MÜLLER, M., SPERLICH, S. and WERWATZ, W. (2004). *Nonparametric and semiparametric models.* Springer-Verlag, Berlin.

[42] HALL, P. and MARRON, J.S. (1991). Lower bounds for bandwidth selection in density estimation. *Probability Theory and Related Fields* **90**, 149–173.

[43] HART, J.D. and YI, S. (1998). One-sided cross-validation. *Journal of the American Statistical Association* **93**, 620–631.

[44] HASTIE, T. (2011). gam: Generalized additive models. R package version 1.04.1. *http://CRAN.R-project.org/package=gam*

[45] HASTIE, T. and LOADER, C.R. (1993a). Rejoinder to: "Local regression: Automatic kernel carpentry." *Statistical Science* **8**, 139–143.

[46] HASTIE, T. and LOADER, C.R. (1993b). Local regression: Automatic kernel carpentry. *Statistical Science* **8**, 120–129.

[47] HASTIE, T. and TIBSHIRANI, R. (1990). *Generalized Additive Models.* Chapman and Hall, London.

[48] HASTIE, T., TIBSHIRANI, R. and FRIEDMAN, J. (2001). *The Elements of Statistical Learning.* Springer, New York.

[49] HAYFIELD, T. and RACINE, J.S. (2008). Nonparametric econometrics: The np package. *Journal of Statistical Software* **27(5)**, 1–32.

[50] HORNG, W. (2004). Kernel estimates of the derivative of regression curves. *Journal of National Hsin Chu Teachers College* **18**, 259–274.

[51] HURVICH, C., SIMONOFF, J. and TSAI, C. (1998). Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion. *Journal of the Royal Statistical Society series B* **60**, 271–293.

[52] HYNDMAN, R.J. (2010). hdrcde: Highest density regions and conditional density estimation. R package version 2.15. *http://CRAN.R-project.org/package=hdrcde*.

[53] HYNDMAN, R.J., and YAO, Q. (2002). Nonparametric estimation and symmetry tests for conditional density functions. *Journal of Nonparametric Statistics* **14**, 259–278.

[54] LAFFERTY, J. and WASSERMAN, L. (2008). Rodeo: sparse, greedy nonparametric regression. *Annals of Statistics* **36(1)**, 28–63.

[55] LAIRD, N.M. (1978). Nonparametric maximum likelihood estimation of a mixing distribution. *Journal of the American Statistical Association* **73**, 805–811.

[56] LEPSKI, O. (1991). Asymptotically minimax adaptive estimation. I: Upper bounds. Optimally adaptive estimates. *Theory of Probability and its Applications* **37**, 682–697.

[57] LI, Q. and RACINE, J.S. (2004). Cross-validated local linear nonparametric regression. *Statistica Sinica* **14(2)**, 485–512.

[58] LOADER, C.R. (1999). *Local regression and likelihood.* Springer, New York.

[59] LOADER, C.R. (2010). locfit: Local regression, likelihood and density estimation. R package version 1.5-6. *http://CRAN.R-project.org/package=locfit*

[60] MARRON, J.S. (1986). Will the art of smoothing ever become a science? *Communications in Contemporary Mathematics* **9**, 169–178.

[61] MARRON, J.S. and WAND, M.P. (1992). Exact mean integrated squared error. *Annals of Statistics* **20**, 712–736.

[62] MÜLLER, D.W. and SAWITZKI, G. (1991). Excess mass estimates and tests for multimodality. *Journal of the American Statistical Association* **86**, 738–746.

[63] NELDER, J.A. and MEAD, R. (1965). A simplex method for function minimization. *Computer Journal* **7**, 308–313.

[64] NEWELL, J. and EINBECK, J. (2007). A comparative study of nonparametric derivative estimators. *Proceedings of the 22nd International Workshop on Statistical Modelling.* 453–456.

[65] PARK, B.U. and MARRON, J.S. (1990). Comparison of data-driven bandwidth selectors. *Journal of the American Statistical Association* **85**, 66–72.

[66] PARZEN, E. (1962). On estimation of a probability density function and mode. *Annals of Mathematical Statistics* **33**, 1065–1076.

[67] PETERSEN, K.B. and PEDERSEN, M.S. (2008). The Matrix Cookbook. *http://matrixcookbook.com/.*

[68] PINHEIRO, J., BATES, D., DEBROY, S., SARKAR, D. and R DEVELOPMENT CORE TEAM (2011). nlme: Linear and nonlinear mixed effects models. R package version 3.1-98.

[69] R DEVELOPMENT CORE TEAM (2010). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

[70] RAMSAY, J.O. and SILVERMAN, B.W. (2005). *Functional Data Analysis.* Springer, New York.

[71] ROSENBLATT, M. (1956). Remarks on some nonparametric estimates of a density function. *Annals of Mathematical Statistics* **27**, 832–837.

[72] RUDEMO, M. (1982). Empirical choice of histograms and kernel density estimators. *Scandinavian Journal of Statistics* **9**, 65–78.

[73] RUPPERT, D. (1997). Empirical-bias bandwidths for local polynomial nonparametric regression and density estimation. *Journal of the American Statistical Association* **92**, 1049–1062.

[74] RUPPERT, D., SHEATHER, S.J. and WAND, M.P. (1995) An effective bandwidth selector for local least squares regression. *Journal of the American Statistical Association* **90**, 1257–1270.

[75] RUPPERT, D. and WAND, M.P. (1994). Multivariate locally weighted least squares regression. *Annals of Statistics* **22**, 1346–1370.

[76] SAIN, S.R. (2002). Multivariate locally adaptive density estimation. *Computational Statistics and Data Analysis* **39**, 165–186.

[77] SAMANTA, M. and THAVANESWARAN, A. (1990). Non-parametric estimation of the conditional mode. *Communications in Statistics - Theory and Methods* **19**, 4515–4524.

[78] SCOTT, D.W. (1992). *Multivariate Density Estimation.* Wiley, New York.

[79] SCOTT, D.W. and TERRELL, G.R. (1987). Biased and unbiased cross-validation in density estimation. *Journal of the American Statistical Association* **82**, 1131–1146.

[80] SEIFERT, B. and GASSER, T. (2000). Data adaptive ridging in local polynomial regression. *Journal of Computational and Graphical Statistics* **9(2)**, 338–360.

[81] SHEATHER, S.J. and JONES, M.C. (1991). A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society series B* **53**, 683–690.

[82] SILVERMAN, B.W. (1978). Weak and strong uniform consistency of the kernel estimate of a density and its derivatives. *Annals of Statistics* **6**, 177–184.

[83] SILVERMAN, B.W. (1981). Using kernel density estimates to investigate multimodality. *Journal of the Royal Statistical Society series B* **43**, 97–99.

[84] SILVERMAN, B.W. (1986). *Density estimation for statistics and data analysis.* Chapman and Hall, London.

[85] SIMONOFF, J.S. (1996). *Smoothing methods in statistics.* Springer, New York.

[86] SRIHERA, R. and STUTE, W. (2011). Kernel adjusted density estimation. *Statistics and Probability Letters* **81(5)**, 571–579.

[87] STONE, C.J. (1977). Consistent nonparametric regression. *Annals of Statistics* **5**, 595–645.

[88] STONE, C.J. (1980). Optimal rates of convergence for nonparametric estimators. *Annals of Statistics* **8**, 1348–1360.

[89] STONE, C.J. (1984). An asymptotically optimal window selection rule for kernel density estimates. *Annals of Statistics* **12**, 1285–1297.

[90] TRUONG, Y.K. (1989). Asymptotic properties of kernel estimators based on local medians. *Annals of Statistics* **17**, 606–617.

[91] VENABLES, W.N. and RIPLEY, B.D. (2002). *Modern applied statistics with S.* Springer, New York.

[92] VIDAURRE, D., BIELZA, C. and LARRAÑAGA, P. (2011). Lazy lasso for local regression. *Computational Statistics* Online first.

[93] WAHBA, G. and WALD, S. (1975). A completely automatic French curve. *Communications in statistics* **4**, 1–17.

[94] WAND, M.P. (2010). SemiPar: Semiparametic regression. R package version 1.0-3. *http://CRAN.R-project.org/package=SemiPar.*

[95] WAND, M.P. and JONES, M.C. (1995). *Kernel smoothing.* Chapman and Hall, London.

[96] WAND, M.P. and RIPLEY, B. (2010). KernSmooth: Functions for kernel smoothing for Wand & Jones (1995). R package version 2.23-4. *http://CRAN.R-project.org/package=KernSmooth.*

[97] WEDEL, M. and KAMAKURA, W.A. (1995). A mixture likelihood approach for generalized linear models. *Journal of Classification* **12**, 21–55.

[98] WHITTAKER, E.T. (1923). On a new method of graduation. *Proceedings of the Edinburgh Mathematical Society* **41**, 63–75.

[99] YANG, L. and TSCHERNIG, R. (1999). Multivariate bandwidth selection for local linear regression. *Journal of the Royal Statistical Society series B* **61(4)**, 793–815.

[100] YI, S. (1996). *On one-sided cross-validation in nonparametric regression.* Ph.D. dissertation, Department of Statistics, Texas A&M University.

[101] ZHANG, J. and WANG, X. (2009). Robust normal reference bandwidth for kernel density estimation. *Statistica Neerlandica* **63(1)**, 13–23.

Appendix A

A.1 Multivariate Taylor's theorem

The multivariate version of Taylor's Theorem is used frequently in this thesis, particularly in Chapter 2, and so it is included here in full for completion. The following is exactly as it is expressed in Wand and Jones (1995).

Let m be a d-variate function, and α_n be a sequence of $d \times 1$ vectors with all components tending to zero. Also let $\nabla g(\mathbf{x})$ be the vector of first-order partial derivatives of m and $\mathcal{H}_m(\mathbf{x})$ be the Hessian matrix of m, the dxd matrix having (i, j) entry equal to

$$\frac{\delta^2 m(\mathbf{x})}{\delta x_i \delta x_j}.$$

Then, assuming that all entries of $\mathcal{H}_m(\mathbf{x})$ are continuous in a neighbourhood of $\mathbf{x}$,

$$m(\mathbf{x} + \alpha_n) = m(\mathbf{x}) + \alpha_n^T \nabla g(\mathbf{x}) + \frac{1}{2}\alpha_n^T \mathcal{H}_m(\mathbf{x})\alpha_n + o(\alpha_n^T \alpha_n).$$

A.2 Quotients of summations

Bivariate and trivariate data can be expressed in the form (1.28).

For bivariate data,

$$w_i = K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x}) \times$$

$$\{[K_{11}K_{22} - K_{12}K_{12}] + (X_{i1} - x_1)\,[K_{12}K_2 - K_{22}K_1] + (X_{i2} - x_2)\,[K_{12}K_1 - K_{11}K_2]\}$$

where

$$K_a = \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(X_{ia} - x_a),$$

and

$$K_{ab} = \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(X_{ia} - x_a)(X_{ib} - x_b).$$

For trivariate data,

$$w_i = K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\varsigma - \tau + \upsilon - \omega) \tag{A.1}$$

where

$$\varsigma = K_{11}(K_{22}K_{33} - K_{23}K_{23}) - K_{12}(K_{12}K_{33} - K_{13}K_{23}) + K_{13}(K_{12}K_{23} - K_{13}K_{22}),$$

$$\tau = (X_{i1} - x_1)\left[K_1(K_{22}K_{33} - K_{23}K_{23}) - K_{12}(K_2K_{33} - K_3K_{23}) + K_{13}(K_2K_{23} - K_3K_{22})\right],$$

$$\upsilon = (X_{i2} - x_2)\left[K_1(K_{12}K_{33} - K_{23}K_{13}) - K_{11}(K_2K_{33} - K_3K_{23}) + K_{13}(K_2K_{13} - K_3K_{12})\right]$$

and

$$\omega = (X_{i3} - x_3)\left[K_1(K_{12}K_{23} - K_{13}K_{22}) - K_{11}(K_2K_{23} - K_3K_{22}) + K_{12}(K_2K_{13} - K_3K_{12})\right].$$

A.3 Proof of (1.30)

As given in Ruppert and Wand (1994),

$$\mathbf{X}^T\mathbf{W}\mathbf{X} = \begin{bmatrix} \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x}) & \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \\ \\ \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x}) & \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \end{bmatrix} \tag{A.2}$$

and $\mathbf{X}^T\mathbf{W}\mathbf{R}$ is $\mathbf{X}^T\mathbf{W}\mathbf{X}$ but with the adjustment of including the Y_i in the summations in the left column,

$$\mathbf{X}^T\mathbf{W}\mathbf{R} = \begin{bmatrix} \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})Y_i & \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \\ \\ \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})Y_i & \sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})^T \end{bmatrix} \tag{A.3}$$

Now,

$$\begin{aligned}\hat{m}(\mathbf{x}) &= \mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{Y} \\ &= \frac{\mathbf{e}_1^T\text{adj}(\mathbf{X}^T\mathbf{W}\mathbf{X})\mathbf{X}^T\mathbf{W}\mathbf{Y}}{\det(\mathbf{X}^T\mathbf{W}\mathbf{X})} \end{aligned} \tag{A.4}$$

since, generally,

$$\mathbf{B}^{-1} = \frac{\text{adj}(\mathbf{B})}{\det(\mathbf{B})} \tag{A.5}$$

where adj($\mathbf{B}$) is the adjugate matrix.

Now, due to the way the adjugate matrix is calculated by taking the transpose of the matrix of cofactors, the first row of the adjugate matrix of $\mathbf{B}$ depends on all entries other than the first column of $\mathbf{B}$. In this way

$$\mathbf{e}_1^T\text{adj}(\mathbf{X}^T\mathbf{W}\mathbf{R}) = \mathbf{e}_1^T\text{adj}(\mathbf{X}^T\mathbf{W}\mathbf{X})$$

since $\mathbf{X}^T\mathbf{W}\mathbf{R}$ and $\mathbf{X}^T\mathbf{W}\mathbf{X}$ are identical except for the first column, which is not involved here.

Substituting this into (A.4), one obtains

$$\begin{aligned}\hat{m}(\mathbf{x}) &= \frac{\mathbf{e}_1^T\text{adj}(\mathbf{X}^T\mathbf{W}\mathbf{R})\mathbf{X}^T\mathbf{W}\mathbf{Y}}{\det(\mathbf{X}^T\mathbf{W}\mathbf{X})} \\ &= \frac{\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{R})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{Y}\det(\mathbf{X}^T\mathbf{W}\mathbf{R})}{\det(\mathbf{X}^T\mathbf{W}\mathbf{X})} \end{aligned} \tag{A.6}$$

again using (A.5).

Now,

$$\mathbf{X}^T\mathbf{W}\mathbf{Y} = \begin{bmatrix} \sum_{i=1}^n K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})Y_i \\ \\ \sum_{i=1}^n K_{\mathbf{H}}(\mathbf{X}_i - \mathbf{x})(\mathbf{X}_i - \mathbf{x})Y_i \end{bmatrix}$$

which is identical to the first column of $\mathbf{X}^T\mathbf{W}\mathbf{R}$ i.e.

$$\mathbf{X}^T\mathbf{W}\mathbf{Y} = \mathbf{X}^T\mathbf{W}\mathbf{R}\mathbf{e}_1.$$

Substituting this into (A.6), one obtains

$$\begin{aligned}\hat{m}(\mathbf{x}) &= \frac{\mathbf{e}_1^T(\mathbf{X}^T\mathbf{W}\mathbf{R})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{R}\mathbf{e}_1\det(\mathbf{X}^T\mathbf{W}\mathbf{R})}{\det(\mathbf{X}^T\mathbf{W}\mathbf{X})} \\ &= \frac{\mathbf{e}_1^T\mathbf{I}\mathbf{e}_1\det(\mathbf{X}^T\mathbf{W}\mathbf{R})}{\det(\mathbf{X}^T\mathbf{W}\mathbf{X})} \\ &= \frac{\det(\mathbf{X}^T\mathbf{W}\mathbf{R})}{\det(\mathbf{X}^T\mathbf{W}\mathbf{X})}. \end{aligned}$$