Bandwidth selectors for multivariate kernel density estimation[1]

Tarn Duong
School of Mathematics and Statistics

1 October 2004

[1]This thesis is presented for the degree of Doctor of Philosophy at the University of Western Australia.

This thesis is my own account of my research and contains, as its main content, work that has not been previously submitted for a degree at any university.

Tarn Duong
October 2004

Acknowledgements

Thanks to my supervisor, Martin Hazelton, for guiding me through this journey; to my friends, Isabel Casas Villalba, Sandra Pereira, Csaba Schneider and Ranjodh Singh, for providing companionship; to other members of the School of Mathematics and Statistics, particularly Berwin Turlach; for financial support provided by an Australian Postgraduate Award; and most of all, to my family who have supported me throughout everything.

Abstract

Kernel density estimation is an important data smoothing technique. It has been applied most successfully for univariate data whilst for multivariate data its development and implementation have been relatively limited. The performance of kernel density estimators depends crucially on the bandwidth selection. Bandwidth selection in the univariate case involves selecting a scalar parameter which controls the amount of smoothing. In the multivariate case, the bandwidth matrix controls both the degree and direction of smoothing so its selection is more difficult. So far most of the research effort has been expended on automatic, data-driven selectors for univariate data. There is, on the other hand, a relative paucity of multivariate counterparts. Most of these multivariate bandwidth selectors are focused on the restricted case of diagonal matrices. In this thesis practical algorithms are constructed, with supporting theoretical justifications, for unconstrained bandwidth matrices.

The two main classes of univariate bandwidth selectors are plug-in and cross validation. These unidimensional selectors are generalised to the multidimensional case. The univariate framework for theoretically analysing kernel density estimators is extended to a general multivariate version. This framework has at its core the quantification of the relative rates of convergence which provide a guide to the asymptotic behaviour of bandwidth selectors. Simulation studies and real data analysis are employed to illustrate their finite sample behaviour. It is found that unconstrained selectors possess good asymptotic and finite sample properties in a wide range of situations.

Buoyed by this success, two extensions are embarked upon. The first is variable bandwidth selection, generalising the above case where the bandwidth is fixed throughout the sample space. The variation of the bandwidths is controlled by the local properties of the data. The novel contribution is to use non-parametric clustering to summarise these local properties, along with unconstrained bandwidth matrices. The second is in kernel discriminant analysis where unconstrained bandwidth matrices are shown to produce more accurate discrimination.

Publications arising from thesis

Duong, T. & Hazelton M. L. (2003), 'Plug-in bandwidth matrices for bivariate kernel density estimation', *Journal of Nonparametric Statistics* **15**, 17–30.

Duong, T. & Hazelton M. L. (2004), 'Convergence rates for unconstrained bandwidth matrix selectors in multivariate kernel density estimation'. To appear in *Journal of Multivariate Analysis*.

Duong, T. & Hazelton M. L. (2004), 'Cross-validation bandwidth matrices for multivariate kernel density estimation'. Submitted for publication.

Contents

List of Tables

List of Figures

Chapter 1

Kernel density estimation

1.1 Introduction

Data smoothing is an important class of fundamental techniques in statistics which allow us to take a sample of data and from it construct a continuous estimator. Estimating probability density functions can be considered the simplest data smoothing situation. Historically, in order to reduce the computational burden for this estimation, a functional or parametric form is imposed on the density estimate. This functional form is largely subjective but imposing it does greatly simplify the problem. All that remains is to estimate the parameters. These estimated parameters plus the functional form give a *parametric* density estimator. The most common parametric estimators are maximum likelihood estimators, and these are useful in a wide range of situations.

Nonetheless there are still many situations where parametric estimation is not applicable. In these cases, it is appropriate to use *non-parametric* density estimators. These do *not* require a functional form to be imposed on the density estimate. As a trade-off for their increased flexibility, most non-parametric density estimators are more computationally intensive and this has restricted their widespread use until the advent of easily available fast computing power in the late twentieth century. Subsequent to this, there has been vast body of research conducted on non-parametric density estimators.

As the title of this thesis suggests, we will concentrate on one class of non-parametric density estimators, namely kernel density estimators. Other types of non-parametric density estimators include histograms, frequency polygons, spline estimators, orthogonal series estimators and penalised likelihood estimators. These estimators are discussed in Silverman (1986), Scott (1992) and Simonoff (1996). We concentrate on kernel density estimators because they are easy to interpret and to implement. Within their intuitively and mathematically simple framework, we can more clearly ascertain the key issues, many of which can be carried over to the other density estimators. Kernel density estimators are most practicable for low to moderate number of dimensions. Six dimensional data are

typically a practical upper limit since at higher dimensions the sparsity of data leads to unstable estimation, see Scott (1992, Section 7.2).

Kernel density estimation is an important smoothing technique in its own right with direct applications such as exploratory data analysis and data visualisation. Its usefulness is not limited to these direct applications. It can be applied indirectly to other non-parametric problems, e.g. discriminant analysis, goodness-of-fit testing, hazard rate estimation, intensity function estimation and regression. Kernel smoothers can also serve as a testing ground for developing analogous smoothing techniques since ideas from the former can be easily transferred to latter. See Silverman (1986), Wand & Jones (1995), Simonoff (1996) and Schimek (2000) for a discussion of related techniques in a united smoothing framework.

A univariate kernel density estimator, for a random sample $X_1, X_2, \ldots X_n$, drawn from a common (smooth) density f, is

$$\hat{f}(x; h) = n^{-1} \sum_{i=1}^{n} K_h(x - X_i). \tag{1.1}$$

Here K is the unscaled kernel function which is typically is a symmetric probability density function with finite variance. K_h is the scaled kernel function and h is the (fixed) bandwidth which is a positive, non-random number. The scaled and unscaled kernels are related by $K_h(x) = h^{-1}K(h^{-1}x)$. At each data point, we place a scaled kernel function of probability mass n^{-1}. These are then summed together to give a composite curve. This composite curve is the kernel density estimate as illustrated in Figure 1.1.

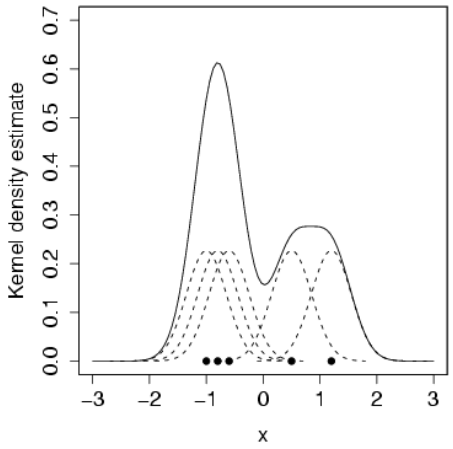

Figure 1.1: Univariate kernel density estimate: solid line – kernel density estimate, dashed lines – individual kernels

The data points are $X_1 = -1, X_2 = -0.8, X_3 = -0.6, X_4 = 0.5, X_5 = 1.2$, marked

on the x-axis. The kernel K is the standard normal pdf (the dotted lines are the scaled kernels). We see that the kernel density estimate is bimodal, reflecting the structure of the data. The bandwidth used is $h = 0.3517$, chosen subjectively here. In common with all smoothing problems, the most important factor is to determine the amount of smoothing: for kernel density estimators the amount of smoothing is controlled by the bandwidth. The crucial task is thus to find an *automatic, data-driven* bandwidth selector.

The general form of the d-dimensional multivariate kernel density estimator, for a random sample $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ drawn from a common (smooth) density f, is

$$\hat{f}(\boldsymbol{x}; \mathbf{H}) = n^{-1} \sum_{i=1}^{n} K_{\mathbf{H}}(\boldsymbol{x} - \boldsymbol{X}_i) \tag{1.2}$$

where $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)^T$and $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{id})^T, i = 1, 2, \ldots, n$. Here K is the unscaled kernel, $K_{\mathbf{H}}$ is the scaled kernel and $\mathbf{H}$ is the $d \times d$ (fixed) bandwidth matrix, which is non-random, symmetric and positive definite. The scaled and unscaled kernels are related by $K_{\mathbf{H}}(\boldsymbol{x}) = |\mathbf{H}|^{-1/2} K(\mathbf{H}^{-1/2}\boldsymbol{x})$. This formulation is a little different to the univariate case since the 1×1 bandwidth matrix is $\mathbf{H} = h^2$ so we are dealing with 'squared bandwidths' here. Though the basic principle, of placing a scaled kernel of mass n^{-1} at each data point and then aggregating to form the density estimate, carries over unchanged from the univariate case, as illustrated in Figure 1.2: we have a sample data set $\boldsymbol{X}_1 = (7, 3), \boldsymbol{X}_2 = (2, 4), \boldsymbol{X}_3 = (4, 4), \boldsymbol{X}_4 = (5, 2)$ and $\boldsymbol{X}_5 = (5.5, 6.5)$ with a bandwidth matrix $\mathbf{H} = \begin{bmatrix} 1 & 0.7 \\ 0.7 & 1 \end{bmatrix}$. On the left are the individual scaled kernels, centred at each data point and on the right is the density estimate.

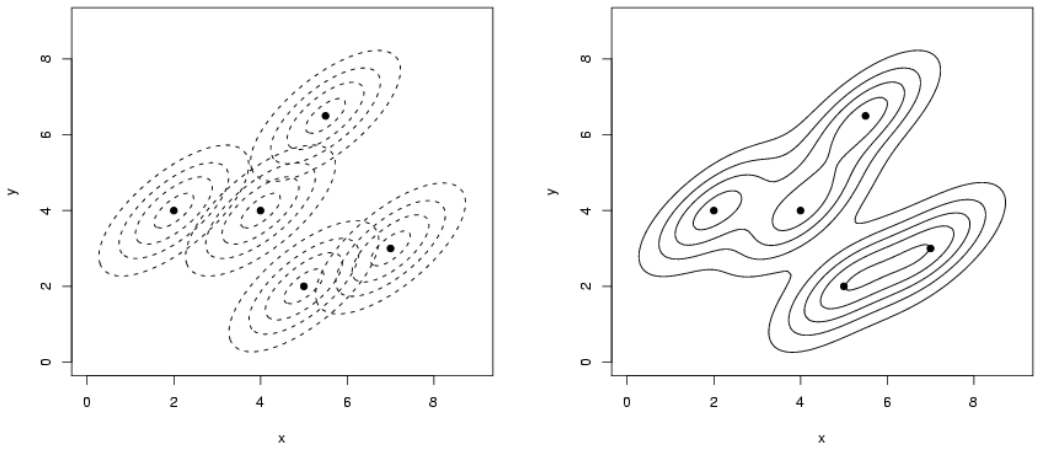

Figure 1.2: Bivariate kernel density estimate: solid line – kernel density estimate, dashed lines – individual kernels

We will restrict our attention to kernel functions K that are spherically symmetric probability density functions (i.e. second order kernels). By using second order kernels, the resulting density estimate $\hat{f}$ is also a probability density function. Moreover, we will mostly use normal kernels throughout this thesis for two reasons: they lead to smooth density estimates and they simplify the mathematical analysis.

We will develop theory for the central problem of bandwidth selection for the general multivariate case. This will form the bulk of the thesis. However we will focus on the important bivariate case when looking at particular examples of putting this theory into practice for two reasons. First, bivariate kernel density estimators, like their univariate counterparts, can be easily visualised on a 2-dimensional page through contour/slice plots and perspective/wire-frame plots. Second, they have important features (e.g. kernel orientation as determined by the bandwidth matrix) which their univariate counterparts lack but which can be easily extended to higher dimensions.

1.2 Error criteria

The bandwidth selector plays a central role in determining the performance of kernel density estimators. Thus we wish to select bandwidths which give the optimal performance. Performance is measured by the closeness of a kernel density estimate to its target density. There are many possible error criteria from which to choose. A common global error criterion is the Integrated Squared Error or ISE. This is the integrated squared distance between an estimate $\hat{f}$ and the target density f:

$$\operatorname{ISE} \hat{f}(\cdot; \mathbf{H}) = \int_{\mathbb{R}^d} [\hat{f}(\boldsymbol{x}; \mathbf{H}) - f(\boldsymbol{x})]^2 \, d\boldsymbol{x}. \tag{1.3}$$

The ISE is a random variable and is difficult to predict. An alternative is the Mean Integrated Squared Error or MISE, defined as

$$\operatorname{MISE}(\mathbf{H}) \equiv \operatorname{MISE} \hat{f}(\cdot; \mathbf{H}) = \mathbb{E} \operatorname{ISE} \hat{f}(\cdot; \mathbf{H}) = \mathbb{E} \int_{\mathbb{R}^d} [\hat{f}(\boldsymbol{x}; \mathbf{H}) - f(\boldsymbol{x})]^2 \, d\boldsymbol{x}. \tag{1.4}$$

See Jones (1991),Turlach (1993), Grund et al. (1994) for a discussion on the relative merits of using the ISE and MISE. Other authors have used other error criteria. See Devroye & Györfi (1985) for a thorough treatment of the Mean Integrated Absolute Error (MIAE) which replaces the square in the MISE with the absolute value:

$$\operatorname{MIAE} \hat{f}(\cdot; \mathbf{H}) = \mathbb{E} \int_{\mathbb{R}^d} |\hat{f}(\boldsymbol{x}; \mathbf{H}) - f(\boldsymbol{x})| \, d\boldsymbol{x}.$$

Marron & Tsybakov (1995) deal with error criteria that are more akin to visual interpretations of closeness.

From these criteria, we choose the MISE as it is the most mathematically tractable criterion and is the most widely used in practice. We thus wish to find

$$\mathbf{H}_{\mathrm{MISE}} = \operatorname*{argmin}_{\mathbf{H}\in\mathcal{H}} \operatorname{MISE} \hat{f}(\cdot;\mathbf{H})$$

where $\mathcal{H}$ is the space of symmetric, positive definite $d\times d$ matrices. As MISE does not have a closed form, except if f is a normal mixture and K is the normal kernel (see Wand & Jones (1995)), finding $\mathbf{H}_{\mathrm{MISE}}$ is in general extremely difficult. The usual approach is to find a tractable approximation to the MISE. The first step in determining this approximation is to rewrite the MISE. Under some mild regularity conditions, which will assume to hold throughout the thesis, we are able to exchange the integral and expectation operators:

$$\begin{aligned}\operatorname{MISE} \hat{f}(\cdot;\mathbf{H}) &= \int_{\mathbb{R}^d} \operatorname{MSE}\ \hat{f}(\boldsymbol{x};\mathbf{H})\ d\boldsymbol{x} \\ &= \int_{\mathbb{R}^d} \operatorname{Var} \hat{f}(\boldsymbol{x};\mathbf{H})\ d\boldsymbol{x} + \int_{\mathbb{R}^d} \operatorname{Bias}^2 \hat{f}(\boldsymbol{x};\mathbf{H})\ d\boldsymbol{x}.\end{aligned}$$

As the expected value of the kernel density estimate is

$$\mathbb{E}\, \hat{f}(\boldsymbol{x};\mathbf{H}) = \mathbb{E}\, K_{\mathbf{H}}(\boldsymbol{x}-\boldsymbol{X}) = \int_{\mathbb{R}^d} K_{\mathbf{H}}(\boldsymbol{x}-\boldsymbol{y}) f(\boldsymbol{y})\ d\boldsymbol{y} = (K_{\mathbf{H}} * f)(\boldsymbol{x})$$

(where * is the convolution operator) then the bias is

$$\operatorname{Bias} \hat{f}(\boldsymbol{x};\mathbf{H}) = (K_{\mathbf{H}} * f)(\boldsymbol{x}) - f(\boldsymbol{x}).$$

The variance is

$$\operatorname{Var} \hat{f}(\boldsymbol{x};\mathbf{H}) = n^{-1}[(K^2_{\mathbf{H}} * f)(\boldsymbol{x}) - (K_{\mathbf{H}} * f)(\boldsymbol{x})^2].$$

Combining the squared bias and the variance we have

$$\begin{aligned}&\operatorname{MISE} \hat{f}(\cdot;\mathbf{H}) \\ &\quad= n^{-1} \int_{\mathbb{R}^d} [(K^2_{\mathbf{H}} * f)(\boldsymbol{x}) - (K_{\mathbf{H}} * f)(\boldsymbol{x})^2]\ d\boldsymbol{x} - \int_{\mathbb{R}^d} [(K_{\mathbf{H}} * f)(\boldsymbol{x}) - f(\boldsymbol{x})]^2\ d\boldsymbol{x} \\ &\quad= n^{-1} R(K) |\mathbf{H}|^{-1/2} + (1-n^{-1}) \int_{\mathbb{R}^d} (K_{\mathbf{H}} * f)(\boldsymbol{x})^2\ d\boldsymbol{x} - 2\int_{\mathbb{R}^d} (K_{\mathbf{H}} * f)(\boldsymbol{x}) f(\boldsymbol{x})\ d\boldsymbol{x} \\ &\qquad + R(f)\end{aligned}$$

where $R(g) = \int_{\mathbb{R}^d} g(\boldsymbol{x})^2\ d\boldsymbol{x}$ for any square integrable function g. From this form of the MISE, we proceed to an asymptotic approximation of the MISE, known as the AMISE. As the AMISE is a tractable expression we can find $\mathbf{H}_{\mathrm{AMISE}}$, the minimiser of AMISE, more easily than $\mathbf{H}_{\mathrm{MISE}}$.

We now introduce some more notation that will assist us in determining an expression for AMISE. The vec (vector) operator takes the elements of a $d \times d$ matrix and stacks them column-wise into a vector. The vech (vector half) operator takes the elements of the

lower triangular half of a $d \times d$ matrix, and stacks them column-wise into a vector. For example

$$\operatorname{vec} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a \\ c \\ b \\ d \end{bmatrix}, \operatorname{vech} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a \\ c \\ d \end{bmatrix}.$$

Hence a vec'ed matrix is length d^2 and a vech'ed one is of length $\frac{1}{2}d(d+1)$. The vec and vech of a symmetric matrix $\mathbf{A}$ are related in the following ways:

$$\operatorname{vec} \mathbf{A} = \mathbf{D}_d \operatorname{vech} \mathbf{A}$$
$$\mathbf{D}_d^T \operatorname{vec} \mathbf{A} = 2\mathbf{A} - \operatorname{dg} \mathbf{A}$$

where $\mathbf{D}_d$ is the duplication matrix of order d and $\operatorname{dg} \mathbf{A}$ is matrix $\mathbf{A}$ with all of its non-diagonal elements set to zero. For example

$$\mathbf{D}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \operatorname{dg} \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & d \end{bmatrix}.$$

The following form of the AMISE is derived by Wand & Jones (1995, pp. 94 – 101):

$$\operatorname{AMISE}(\mathbf{H}) \equiv \operatorname{AMISE} \hat{f}(\cdot; \mathbf{H}) = n^{-1} R(K) |\mathbf{H}|^{-1/2} + \tfrac{1}{4} \mu_2(K)^2 \int_{\mathbb{R}^d} \operatorname{tr}^2(\mathbf{H} D^2 f(\boldsymbol{x}))\, d\boldsymbol{x}$$

where $\int_{\mathbb{R}^d} \boldsymbol{x}\boldsymbol{x}^T K(\boldsymbol{x})\, d\boldsymbol{x} = \mu_2(K)\mathbf{I}_d$ with $\mu_2(K) < \infty$ and $\mathbf{I}_d$ is the $d \times d$ identity matrix; and $D^2 f(\boldsymbol{x})$ is the Hessian matrix of f. The first term in the AMISE is the asymptotic integrated variance and the second term is the asymptotic integrated squared bias. The rate of convergence of the AMISE to the MISE is given by

$$\operatorname{MISE} \hat{f}(\cdot; \mathbf{H}) = \operatorname{AMISE} \hat{f}(\cdot; \mathbf{H}) + o(n^{-1}|\mathbf{H}|^{-1/2} + \|\operatorname{vech} \mathbf{H}\|^2)$$

provided that all entries in $D^2 f(\boldsymbol{x})$ are piecewise continuous and square integrable, and all entries of $\mathbf{H} \to 0$ and $n^{-1}|\mathbf{H}|^{-1/2} \to 0$, as $n \to \infty$. An alternative form of the AMISE is

$$\operatorname{AMISE} \hat{f}(\cdot; \mathbf{H}) = n^{-1} R(K) |\mathbf{H}|^{-1/2} + \tfrac{1}{4} \mu_2(K)^2 (\operatorname{vech}^T \mathbf{H}) \boldsymbol{\Psi}_4 (\operatorname{vech} \mathbf{H}) \tag{1.5}$$

where $\boldsymbol{\Psi}_4$ is the $\frac{1}{2}d(d+1) \times \frac{1}{2}d(d+1)$ matrix given by

$$\boldsymbol{\Psi}_4 = \int_{\mathbb{R}^d} \operatorname{vech}(2D^2 f(\boldsymbol{x}) - \operatorname{dg} D^2 f(\boldsymbol{x})) \operatorname{vech}^T (2D^2 f(\boldsymbol{x}) - \operatorname{dg} D^2 f(\boldsymbol{x}))\, d\boldsymbol{x}. \tag{1.6}$$

(Note that the subscript 4 on $\boldsymbol{\Psi}$ indicates the order of the derivatives involved.) This form of the AMISE arises as $\int_{\mathbb{R}^d} \operatorname{tr}^2(\mathbf{H} D^2 f(\boldsymbol{x}))\, d\boldsymbol{x} = (\operatorname{vech}^T \mathbf{H}) \boldsymbol{\Psi}_4 (\operatorname{vech} \mathbf{H})$ under the above regularity conditions.

We can explicitly state an expression for $\boldsymbol{\Psi}_4$ in terms of its individual elements using the following notation. Let $\boldsymbol{r} = (r_1, r_2, \ldots, r_d)$ where the $r_1, r_2, \ldots, r_d$ are non-negative integers. Let $|\boldsymbol{r}| = r_1 + r_2 + \cdots + r_d$ then the $\boldsymbol{r}$-th partial derivative of f can be written as

$$f^{(\boldsymbol{r})}(\boldsymbol{x}) = \frac{\partial^{|\boldsymbol{r}|}}{\partial_{x_1}^{r_1} \ldots \partial_{x_d}^{r_d}} f(\boldsymbol{x}).$$

Define the integrated density derivative functional as

$$\psi_{\boldsymbol{r}} = \int_{\mathbb{R}^d} f^{(\boldsymbol{r})}(\boldsymbol{x}) f(\boldsymbol{x})\ d\boldsymbol{x}. \tag{1.7}$$

This then implies that each element in $\boldsymbol{\Psi}_4$ is a $\psi_{\boldsymbol{r}}$ functional.

To be more explicit, we look more closely at the vech operator and its inverse. Suppose we have a $d \times d$ symmetric matrix $\mathbf{A}$ then the (i,j)-th element of $\mathbf{A}$, $[\mathbf{A}]_{ij}, i, j = 1, 2, \ldots, d$ is mapped to the the k-th element of vech $\mathbf{A}$, $[\operatorname{vech} \mathbf{A}]_k, k = 1, 2, \ldots, d'$ where $d' = \frac{1}{2}d(d+1)$ and

$$k = (j-1)d - \tfrac{1}{2}j(j-1) + i.$$

Conversely suppose that we have a vector vech $\mathbf{A}$ of length d' then $[\operatorname{vech} \mathbf{A}]_k$ is mapped to $[\mathbf{A}]_{ij}$ where

$$\begin{aligned} &j : (j-1)d - \tfrac{1}{2}(j-1)(j-2) < k \leq jd - \tfrac{1}{2}j(j-1) \\ &i = k - (j-1)d + \tfrac{1}{2}j(j-1). \end{aligned}$$

We have that $[\boldsymbol{\Psi}_4]_{k,k'} = [\operatorname{vech} D^2 f(\boldsymbol{x})]_k [\operatorname{vech} D^2 f(\boldsymbol{x})]_{k'}, k, k' = 1, 2, \ldots, d'$. Since we have $[D^2 f(\boldsymbol{x})]_{ij} = f^{(\boldsymbol{e}_i + \boldsymbol{e}_j)}(\boldsymbol{x})$ then $[\boldsymbol{\Psi}_4]_{k,k'}$ contains the functional $\psi_{\boldsymbol{e}_i + \boldsymbol{e}_{i'} + \boldsymbol{e}_j + \boldsymbol{e}_{j'}}$, where $\boldsymbol{e}_i$ is a d-dimensional elementary vector i.e. it has 1 as its i-th element and 0 elsewhere. The coefficient of this functional is given in

$$[\boldsymbol{\Psi}_4]_{k,k'} = [2 - 1\{i = j\}][2 - 1\{i' = j'\}] \psi_{\boldsymbol{e}_i + \boldsymbol{e}_{i'} + \boldsymbol{e}_j + \boldsymbol{e}_{j'}}$$

where $1\{\cdot\}$ is the indicator function. Following the above algorithm, for $d = 2$,

$$\boldsymbol{\Psi}_4 = \begin{bmatrix} \psi_{40} & 2\psi_{31} & \psi_{22} \\ 2\psi_{31} & 4\psi_{22} & 2\psi_{13} \\ \psi_{22} & 2\psi_{13} & \psi_{04} \end{bmatrix}.$$

It is important to note that all we have done so far is to write down various alternative expressions for MISE and AMISE. We must remember that they remain unknown in practice as they depend on the unknown density f. The next step is to find an estimate of (A)MISE, $\widehat{\text{(A)MISE}}$, from the data and then find its minimiser i.e.

$$\hat{\mathbf{H}} = \operatorname*{argmin}_{\mathbf{H} \in \mathcal{H}} \widehat{\text{(A)MISE}}$$

which is known as a *bandwidth selector*. This serves as our surrogate for $\mathbf{H}_{\text{(A)MISE}}$. In the next section, we review the various methods that have been used so far in the search for data-driven bandwidth selectors based on various estimators of (A)MISE.

1.3 Bandwidth selectors

1.3.1 Univariate bandwidth selectors

Since Rosenblatt (1956) and Parzen (1962) introduced univariate kernel density estimators, there has been a vast body of research conducted on them and their bandwidth selectors. See Silverman (1986), Scott (1992), Wand & Jones (1995), Simonoff (1996) and Bowman & Azzalini (1997) for a summary. Wand & Jones (1995, Chapter 3) contains a comprehensive history of univariate bandwidth selectors with an extended bibliography. These authors provide references to all of the original developments of the major types of bandwidth selectors, including most importantly plug-in and cross validation selectors. What is given below is a summarised version, highlighting the main ideas. The reader interested in the more detailed account should peruse Wand & Jones (1995).

Ideas for plug-in selection have been around in many different guises since the 1970s but they all share the basic idea of using the AMISE

$$\text{AMISE}\, \hat{f}(\cdot; h) = n^{-1}h^{-1}R(K) + \tfrac{1}{4}h^4\mu_2(K)^2\psi_4$$

as a starting point. Here we require that $h \to 0$ and $n^{-1}h^{-1} \to 0$ as $n \to \infty$ and that f'' is piecewise continuous and square integrable. The critical step is to estimate $\psi_4 = \int_{-\infty}^{\infty} f^{(4)}(x)f(x)\ dx$. We then plug this estimate $\hat{\psi}_4$ in the previous equation to obtain the *plug-in* estimate of the AMISE:

$$\text{PI}(h) = n^{-1}h^{-1}R(K) + \tfrac{1}{4}h^4\mu_2(K)^2\hat{\psi}_4.$$

This advantage of this plug-in approach is that we have a closed form solution for the selector that minimises this PI:

$$\hat{h}_{\text{PI}} = \left[\frac{R(K)}{\mu_2(K)^2\hat{\psi}_4 n}\right]^{1/5}.$$

The most commonly used method of estimating ψ_4 was introduced by Sheather & Jones (1991). These authors observe that if X has density f then $\psi_4 = \mathbb{E}\, f^{(4)}(X)$ and a 'natural estimator' would be the sample mean of the fourth derivative of a *pilot* kernel density estimate of f

$$\hat{f}_P(x; g) = n^{-1}\sum_{j=1}^{n} L_g(x - X_j)$$

where L is the pilot kernel and g is the pilot bandwidth. So

$$\hat{\psi}_4(g) = n^{-1}\sum_{i=1}^{n}\hat{f}_P^{(4)}(X_i; g) = n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n} L_g^{(4)}(X_i - X_j).$$

Sheather & Jones (1991) also provide an algorithm for selecting the most appropriate pilot bandwidth g.

Cross validation methods make use of leave-one-out estimators of the form

$$\hat{f}_{-i}(X_i; h) = (n-1)^{-1} \sum_{\substack{j=1 \\ j \neq i}}^{n} K_h(X_i - X_j).$$

Here we leave out the i-th data value, compute a kernel density estimate on the rest of the data and then evaluate it at the missing data value. This way we check against or cross validate the estimate of f. If our estimate is appropriate then $\hat{f}_{-i}(X_i; h)$ should be non-zero since we already have a data point at X_i.

Least squares cross validation (LSCV) was developed independently by Rudemo (1982) and Bowman (1984). It attempts to find the bandwidth which minimises

$$\mathrm{LSCV}(h) = \int_{-\infty}^{\infty} \hat{f}(x; h)^2 \, dx - 2n^{-1} \sum_{i=1}^{n} \hat{f}_{-i}(X_i; h).$$

It can be shown that $\mathbb{E}\,\mathrm{LSCV}(h) = \mathrm{MISE}\,\hat{f}(\cdot; h) - R(f)$. Due to its unbiasedness, the LSCV selector is sometimes called the unbiased cross validation (UCV) selector. This unbiasedness, along with its simple interpretation and implementation has assured its widespread use since its introduction. Also contributing to its widespread use is that it does not rely on asymptotic expansions unlike the plug-in methods above and the biased and smoothed cross validation methods below.

Biased cross validation (BCV) was introduced by Scott & Terrell (1987). It is similar to plug-in selectors in that it attempts to find the bandwidth which minimises an estimate of the AMISE. The essential differences are in the estimator of ψ_4 and in the selection of the pilot bandwidth g. Here we set $g = h$ and so

$$\mathrm{BCV}(h) = n^{-1} h^{-1} R(K) + \tfrac{1}{4} h^4 \mu_2(K)^2 \check{\psi}_4(h)$$

where

$$\check{\psi}_4(h) = n^{-1}(n-1)^{-1} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} (K_h'' * K_h'')(X_i - X_j).$$

The estimator $\check{\psi}_4(h)$ is obtained by replacing f with $\hat{f}_{-i}(\cdot; h)$ and taking the sample mean, noting that $\psi_4 = \int_{-\infty}^{\infty} f^{(4)}(x) f(x) \, dx = \int_{-\infty}^{\infty} f''(x) f''(x) \, dx$ under the same regularity conditions on f'' for the AMISE expansion.

Smoothed cross validation (SCV), devised by Hall et al. (1992), can be thought of as a hybrid of estimating the MISE and AMISE. It comprises the asymptotic integrated variance $n^{-1} h^{-1} R(K)$ and an estimate of the exact (non-asymptotic) integrated squared bias. An expression for the exact integrated squared bias is $\int_{-\infty}^{\infty} [(K_h * f)(x) - f(x)]^2 \, dx$ and so an estimate is

$$\int_{-\infty}^{\infty} [(K_h * \hat{f}_P(\cdot; g))(x) - \hat{f}_P(x; g))]^2 \, dx$$

where the target density f has been replaced by its pilot kernel estimate $\hat{f}_P$. Then

$$\begin{aligned}\mathrm{SCV}(h) &= n^{-1}h^{-1}R(K) + \int_{-\infty}^{\infty} [(K_h * \hat{f}_P(\cdot; g))(x) - \hat{f}_P(x; g))]^2 \, dx \\ &= n^{-1}h^{-1}R(K) + n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(K_h * K_h * L_g * L_g - 2K_h * L_g * L_g \\ &\quad + L_g * L_g)(X_i - X_j)\end{aligned}$$

It turns out that if we use the leave-one-out version of the pilot estimator $\hat{f}_{P,-i}(x; g)$ instead, we still have an asymptotically equivalent expression for SCV.

With the SCV in this form, a connection with LSCV is more easily ascertained. The LSCV can be expressed as

$$\mathrm{LSCV}(h) = n^{-1}h^{-1}R(K) + n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{j=1}^{n}(K_h * K_h - 2K_h)(X_i - X_j).$$

So if there are no replications in the data (which occurs with probability 1 for continuous data), then this is $\mathrm{SCV}(h)$ with $g = 0$ (since L_0 can be thought of as the Dirac delta function).

Hall et al. (1992) show that the SCV is also asymptotically equivalent to the smoothed bootstrap of Taylor (1989) and Faraway & Jhun (1990). The smoothed bootstrap is based on resampling from a pilot kernel density estimate $\hat{f}_P(x; g)$ to estimate the MISE and its minimiser. Let $X_1^*, X_2^*, \ldots, X_n^*$ be a bootstrap sample taken from $\hat{f}_P(x; g)$ with $L = K$. Let the bootstrap kernel density estimate be

$$\hat{f}^*(x; h) = n^{-1}\sum_{i=1}^{n} K_h(x - X_i^*)$$

and $\mathbb{E}^*$ and Var^* be the expected value and variance with respect to the bootstrap density $\hat{f}_P$ then the bootstrap estimate of the MISE is

$$\begin{aligned}\mathrm{MISE}^*\, \hat{f}^*(\cdot; h) &= \int_{-\infty}^{\infty} \mathrm{Var}^* \hat{f}^*(x; h) + [\mathbb{E}^*\, \hat{f}^*(x; h) - f(x)]^2 \, dx \\ &= \mathrm{SCV}(h) + o(n^{-1}h^{-1}).\end{aligned}$$

There are two main ways we look at the performance of these different selectors. One is their asymptotic relative convergence rate and the other is their finite sample behaviour. The relative convergence rate of a selector $\hat{h}$ to the MISE-optimal bandwidth h_{MISE} is $n^{-\alpha}$ if

$$(\hat{h} - h_{\mathrm{MISE}})/h_{\mathrm{MISE}} = O_p(n^{-\alpha}) \tag{1.8}$$

for some $\alpha > 0$. A considerable proportion of the literature is devoted to deriving these relative convergence rates.

Sheather & Jones (1991) show that the Sheather-Jones plug-in selector has relative rate $n^{-5/14}$. Hall et al. (1991) show that by using higher order kernels, this rate can be increased to $n^{-1/2}$. Hall & Marron (1991) show that the rate $n^{-1/2}$ is the fastest possible rate relative to h_{MISE} for *any* selector. For the LSCV selector, Hall & Marron (1987) derive the rate of $n^{-1/10}$. Scott & Terrell (1987) show that the BCV selector has the same rate of convergence $n^{-1/10}$ using similar techniques. For SCV, Jones et al. (1991) show that if the pilot bandwidth g is independent of h then the rate of convergence is $n^{-5/14}$ whereas for a judicious choice of dependency between g and h can lead to $n^{-1/2}$ convergence. Wand & Jones (1995, pp. 79–86) contains summary derivations of all these rates. For the above selectors (except those with rate $n^{-1/2}$), the rates of convergence remain the same if we consider them with respect to h_{AMISE} rather than h_{MISE}. This is because the relative discrepancy between h_{AMISE} and h_{MISE} is of order $n^{-2/5}$ which is negligible when compared to the slower convergence rates mentioned above.

Authors who have made comparative simulation studies of univariate bandwidth selectors are Park & Marron (1990), Park & Turlach (1992), Cao et al. (1994), Chiu (1996), Jones et al. (1996). Turlach (1993) and Wand & Jones (1995) collate the results from existing simulation studies (including many of those above), whilst also adding their own insights. Sheather (1992) looks at various selectors for real datasets. Chiu (1991) looks at simulation results from a frequency domain point of view. Loader (1999) takes a different approach to the other authors and makes some iconoclastic observations.

The most important conclusion from these review papers is that there is no uniformly best bandwidth selector for all target densities. The shape and structure of the target density heavily influence which selectors perform well. Nonetheless most of these authors agree that plug-in (in particular the Sheather-Jones version) and smoothed cross validation methods have the widest range of usefulness though least squares cross validation, because of its non-reliance on asymptotics, can still be useful in some cases.

1.3.2 Multivariate bandwidth selectors

The main reasons that multivariate kernel density estimators have been relatively neglected is that they, in their most general form, are far more computationally and mathematically involved than univariate estimators. Selecting a bandwidth matrix rather than just a scalar bandwidth raises difficulties that have no direct analogue in the univariate case. Most important of these is that a bandwidth matrix induces an orientation of the kernel function. The monographs of Bowman & Azzalini (1997), Scott (1992), Silverman (1986), Simonoff (1996) and Wand & Jones (1995) provide an overview of the research already carried out in multivariate density estimation. These contain relatively superficial treatments of multivariate bandwidth selectors when compared to their univariate

counterparts. We need to delve into the journal literature to trace, in a more detailed manner, the development of multivariate kernel density estimators and their bandwidth matrix selectors.

The type of orientation of the kernel function is controlled by the parameterisation of the bandwidth matrix. Wand & Jones (1993) consider parameterisation for bivariate bandwidth matrices. There are respectively three main classes (i) – (iii) and three hybrid classes (iv) – (vi) of parameterisation:

(i) the class of all symmetric, positive definite matrices: $\mathbf{H} = \begin{bmatrix} h_1^2 & h_{12} \\ h_{12} & h_2^2 \end{bmatrix}$

(ii) the class of all diagonal, positive definite matrices: $\operatorname{dg} \mathbf{H} = \begin{bmatrix} h_1^2 & 0 \\ 0 & h_2^2 \end{bmatrix}$

(iii) the class of all positive constants times the identity matrix: $h^2\mathbf{I} = \begin{bmatrix} h^2 & 0 \\ 0 & h^2 \end{bmatrix}$

(iv) the class of all positive constants times the sample variance $\mathbf{S}$: $h^2\mathbf{S} = \begin{bmatrix} h^2 S_1^2 & h^2 S_{12} \\ h^2 S_{12} & h^2 S_2^2 \end{bmatrix}$

(v) the class of all positive constants times $\operatorname{dg} \mathbf{S}$: $h^2 \operatorname{dg} \mathbf{S} = \begin{bmatrix} h^2 S_1^2 & 0 \\ 0 & h^2 S_2^2 \end{bmatrix}$

(vi) the class of matrices obtained by using the correlation coefficient ρ_{12} to determine the rotation: $\begin{bmatrix} h_1^2 & \rho_{12} h_1 h_2 \\ \rho_{12} h_1 h_2 & h_2^2 \end{bmatrix}$

The diagonal matrix parameterisation (ii), which is the most commonly used one, is inappropriate in cases like Figure 1.3(a). Most of the probability mass of the target density is obliquely oriented but the kernel maintains an orientation to the axes. For general use, (iii) $h^2\mathbf{I}$ is too restrictive. As an example consider Figure 1.3(b). The target density has different amounts of spreading in the co-ordinate directions and its contours are ellipses whereas the kernel's contours are circular. Of the hybrid parameterisations (iv) – (vi), the first two (iv) – (v) are inadvisable for general use with a global bandwidth matrix. These parameterisations lead to kernels that align themselves according to the variance matrix of the target density as seen in Figure 1.3(c). They have contours that are horizontal ellipses whereas the components of the target density have vertical elliptical contours. The third hybrid parameterisation (vi) depends on the appropriateness of the correlation coefficient as a measure of orientation of the density, so again it is not generally used. In Figure 1.3(d), the kernel is oriented according to the correlation matrix, almost in a perpendicular direction to the individual components of the density. Since we wish to derive an automatic bandwidth selector for the widest possible range of situations, we focus on the most general parameterisation i.e. (i) full bandwidth matrices.

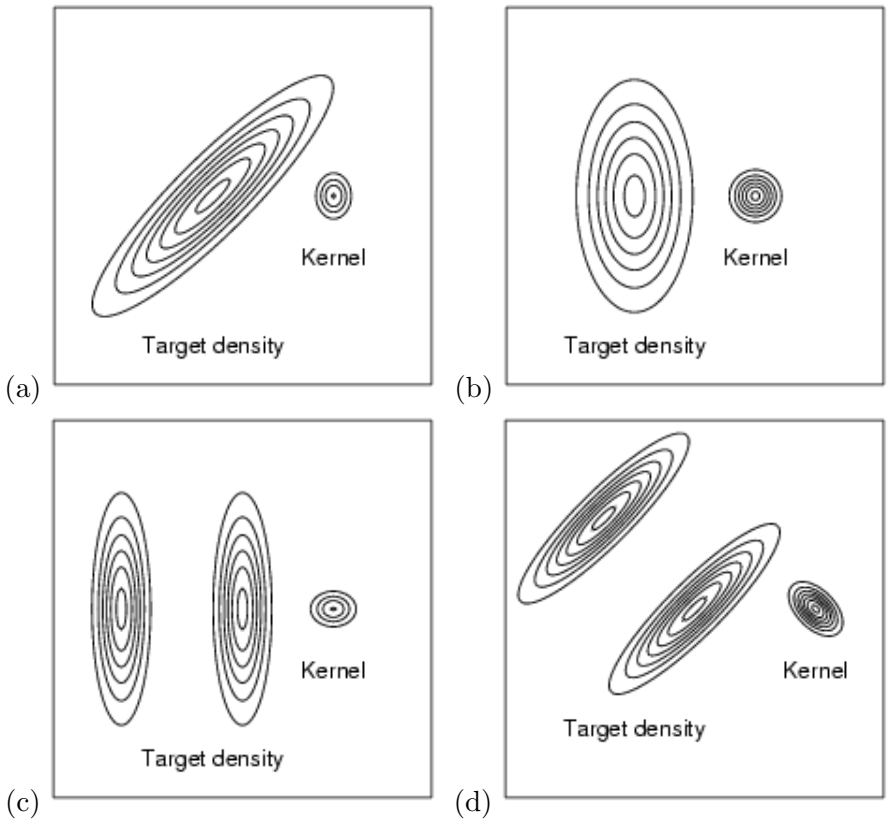

Figure 1.3: Bandwidth matrix parameterisations: target density and kernel shapes

The first foray into multivariate kernel density estimation in the current framework is by Cacoullos (1966), who mostly investigates bandwidth matrices of the parameterisation $h^2\mathbf{I}$. Using this parameterisation the kernel density estimator is

$$\hat{f}(\boldsymbol{x}; h) = n^{-1}h^{-d}\sum_{i=1}^{n} K(h^{-1}(\boldsymbol{x} - \boldsymbol{X}_i)).$$

The asymptotic mean squared error (AMSE) of $\hat{f}$ is

$$\operatorname{AMSE} \hat{f}(\boldsymbol{x}; h) = n^{-1}h^{-d}R(K)f(\boldsymbol{x}) + \tfrac{1}{4}h^4\mu_2(K)^2 \operatorname{tr}(D^2 f(\boldsymbol{x})).$$

It is straightforward to see that the minimiser of this is order $n^{-1/(d+4)}$. The consistency and asymptotic bounds for the bias and mean squared error of $\hat{f}$ using this type of bandwidth matrix are derived. Some of these results are extended to diagonal bandwidth matrices of the form $\operatorname{dg}\mathbf{H}$ or $\operatorname{diag}(h_1^2, h_2^2, \ldots, h_d^2)$. It is important to note that closed forms for the AMSE optimal bandwidths are no longer available for $d > 2$. Despite this lack of closed form solutions, the diagonal case is more appropriate when the components of the data vector have incommensurable characteristics.

Epanechnikov (1969) extends the work of Cacoullos (1966) in the context of the AMISE rather than AMSE. Epanechnikov attempts to optimise the choice of both the bandwidths

and the kernel function. A closed form solution is only available if $h_1 = \cdots = h_d = h$:

$$h_{\text{AMISE}} = \left[\frac{dR(K)}{n\mu_2(K)^2 \int_{-\infty}^{\infty} \text{tr}^2(D^2 f(\boldsymbol{x}))\ d\boldsymbol{x}} \right]^{1/(d+4)}.$$

Having found an optimal bandwidth, the author then proceeds to find an optimal kernel. This optimal kernel is now known as the Epanechnikov kernel. This is followed up by an examination of the behaviour of the AMISE of the kernel density estimator using both the optimal bandwidth and optimal kernel. We choose not to use the Epanechnikov kernel, even though it is optimal, because it is not sufficiently smooth for our purposes. Fortunately the loss in efficiency in using the most other common kernels (including the normal) is small – see Wand & Jones (1995, Section 2.7).

Deheuvels (1977) examines full bandwidth matrices of the form $h^2\mathbf{H}'$ where $\mathbf{H}'$ is an orthogonal matrix which does not depend on the sample size n. (This case subsumes the $h^2\mathbf{I}$ case.) Deheveuels then derives an optimal choice of h. In common with Cacoullos (1966) and Epanechnikov (1969), this is a solution to an essentially univariate problem. These three early works also have in common that in the formulas for their optimal bandwidths there remain quantities that depend on f and the estimation of these unknown quantities is not considered. Thus they establish a theoretical basis for practical bandwidth selectors without supplying data-based algorithms.

We now turn to the literature in which attempts to build these algorithms are explored. Stone (1984) looks at the multivariate least squares cross validation criterion. It is a straightforward generalisation of the univariate form:

$$\text{LSCV}(\mathbf{H}) = \int_{\mathbb{R}^d} \hat{f}(\boldsymbol{x}; \mathbf{H})^2\ d\boldsymbol{x} - 2n^{-1} \sum_{i=1}^{n} \hat{f}_{-i}(\boldsymbol{X}_i; \mathbf{H}).$$

Stone shows that the LSCV selector converges asymptotically in probability to $\mathbf{H}_{\text{MISE}}$ (in the context of a diagonal matrix selector) if the density f and its marginal densities are bounded. The multivariate LSCV selector retains the characteristics of its univariate counterpart i.e. simple interpretation and implementation, and non-reliance on asymptotic expansions for its computation.

Sain et al. (1994) re-examine LSCV selectors as well as generalising the biased cross validation, and bootstrap and smoothed cross validation selectors. These authors only consider the case of product kernels which is equivalent to using diagonal bandwidth matrices. The BCV criterion that they use is

$$\text{BCV}(\mathbf{H}) = n^{-1}R(K)|\mathbf{H}|^{-1/2} + \tfrac{1}{4}(\text{vech}^T\, \mathbf{H})\tilde{\mathbf{\Psi}}_4(\text{vech}\, \mathbf{H})$$

where $\tilde{\mathbf{\Psi}}_4$ is an estimator of $\mathbf{\Psi}_4$ and is made up of estimates of the type, for $|\boldsymbol{r}| = 4$,

$$\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H}) = n^{-1} \sum_{i=1}^{n} \hat{f}_{-i}^{(\boldsymbol{r})}(\boldsymbol{X}_i; \mathbf{H}) = n^{-1}(n-1)^{-1} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} K_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_i - \boldsymbol{X}_j).$$

This uses a different estimator than the univariate BCV selector of Scott & Terrell (1987). The general multivariate SCV criterion is

$$\begin{aligned}\mathrm{SCV}(\mathbf{H}) = n^{-1}R(K)|\mathbf{H}|^{-1/2} + n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(K_{\mathbf{H}} * K_{\mathbf{H}} * L_{\mathbf{G}} * L_{\mathbf{G}} - 2K_{\mathbf{H}} * L_{\mathbf{G}} * L_{\mathbf{G}} \\ + L_{\mathbf{G}} * L_{\mathbf{G}})(\boldsymbol{X}_i - \boldsymbol{X}_j)\end{aligned}$$

where L is a pilot kernel and $\mathbf{G}$ is a pilot bandwidth matrix. Sain et al. (1994) use a less general version, as they set $\mathbf{G} = \mathbf{H}$. Based on their asymptotic results and simulation study, they recommend the BCV selector. However their SCV selector is suboptimal since they ignore the possibility of optimally selecting the pilot $\mathbf{G}$. It is not clear whether the BCV selector would still perform better than the SCV selector with an appropriately chosen pilot bandwidth.

Plug-in selectors were generalised to the multivariate case by Wand & Jones (1994), extending the approach taken by Sheather & Jones (1991). Plug-in selectors are similar to BCV selectors except for the way that is used to estimate $\boldsymbol{\Psi}_4$:

$$\mathrm{PI}(\mathbf{H}) = n^{-1}R(K)|\mathbf{H}|^{-1/2} + \tfrac{1}{4}(\mathrm{vech}^T\,\mathbf{H})\hat{\boldsymbol{\Psi}}_4(\mathrm{vech}\,\mathbf{H})$$

where $\hat{\boldsymbol{\Psi}}_4$ is made up of estimates of the type, for $|\boldsymbol{r}| = 4$,

$$\hat{\psi}_{\boldsymbol{r}}(\mathbf{G}) = n^{-1}\sum_{i=1}^{n}\hat{f}^{(\boldsymbol{r})}(\boldsymbol{X}_i;\mathbf{G}) = n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}K_{\mathbf{G}}^{(\boldsymbol{r})}(\boldsymbol{X}_i - \boldsymbol{X}_j).$$

Here $\mathbf{G}$ may be different to and independent of $\mathbf{H}$. By using a different pilot bandwidth matrix, we have more scope than BCV selectors but this leaves us with the problem of selecting an appropriate pilot. Wand and Jones develop an algorithm to find such a pilot bandwidth. Furthermore they show, with their theoretical analysis and simulation study, that the good properties of one dimensional plug-in selectors mostly carry over to the multi-dimensional case. This is done in detail for diagonal bandwidth matrices though they supply an outline for full bandwidth matrices.

Cross validation and plug-in selectors are the most commonly used selectors. Another type of selector, less frequently used, is introduced by Terrell (1990): the maximal smoothing selector. This is the selector that induces the smoothest density estimate that is consistent with the data scale. Terrell uses the parameterisation $h^2\mathbf{H}'$ where $|\mathbf{H}'| = 1$ and a kernel K such that $\int_{\mathbb{R}^d}\boldsymbol{x}\boldsymbol{x}^T K(\boldsymbol{x})\,d\boldsymbol{x} = \mathbf{I}_d$ then the AMISE is

$$\mathrm{AMISE}\,\hat{f}(\cdot;h) = n^{-1}h^{-d}R(K) + \tfrac{1}{4}h^4\int_{\mathbb{R}^d}\mathrm{tr}^2(\mathbf{H}'D^2f(\boldsymbol{x}))\,d\boldsymbol{x}$$

which has a minimum at

$$h = \left[\frac{dR(K)}{n\int_{\mathbb{R}^d}\mathrm{tr}^2(\mathbf{H}'D^2f(\boldsymbol{x}))\,d\boldsymbol{x}}\right]^{1/(d+4)}.$$

Now we proceed by a minimax approach: first we find the density f (with variance $\mathbf{I}_d$) that gives the maximum value of the integral in the denominator and then minimise it over $\mathbf{H}'$. We then set the maximally smoothed selector to this value, which is

$$\hat{\mathbf{H}}_{\mathrm{MS}} = \left[\frac{(d+8)^{(d+6)/2}\pi^{d/2}R(K)}{16(d+2)n\Gamma(d/2+4)}\right]^{2/(d+4)}\mathbf{S}.$$

It is Terrell's opinion that we should use a conservative approach so as not to produce spurious features in the data and that the onus is to provide evidence for the existence of any features. Notice that (a) this the only multivariate bandwidth selector that has a closed form and (b) it is of the form $h^2\mathbf{S}$ which in general is not advisable, as noted earlier.

Authors who have supplied convergence rates are Sain et al. (1994), $n^{-d/(2d+8)}$ for their cross validation selectors; and Wand & Jones (1994), $n^{-\min(8,d+4)/(2d+12)}$ for their plug-in selectors. These authors provide the details of the derivations for $h^2\mathbf{I}$ type matrices though they outline how to extend them to more general bandwidth matrices.

Cwik & Koronacki (1997*b*) perform a simulation study of a variety of multivariate density estimators including a kernel density estimator with a Wand & Jones (1994) type plug-in selector, a Friedman type projection pursuit estimator and an EM type clustering estimator developed by Cwik & Koronacki (1997*a*). These authors' conclusion is that the EM clustering estimator is best overall but as all the test densities are normal mixtures (assuming the number of mixture components is known) this is not entirely unexpected. To date, there have been no large scale simulation studies of multivariate bandwidth selectors, similar to those for univariate selectors.

1.3.3 Variable bandwidth selectors

We have now covered the main developments in *fixed* bandwidth selectors. Next we cover generalisations of these fixed bandwidth selectors to variable bandwidth selectors. We momentarily return to the univariate case for the exposition of these ideas. There are two main classes of variable bandwidth selectors. In both cases we have a bandwidth function, rather than a constant bandwidth, where either

- the bandwidth is different at each estimation point x : $h(x)$
- the bandwidth is different at each data point X_i: $h_i = \omega(X_i), i = 1, \ldots, n$.

Here, the functions $h(\cdot)$ and $\omega(\cdot)$ are considered to be non-random functions, in much the same way that we consider a single bandwidth to be a non-random number. We will use the terminology used by Sain & Scott (1996) and refer to these selectors as balloon and sample-point selectors. The kernel density estimators arising from these selectors are known as balloon and sample-point kernel density estimators. Other authors use the terms

local and variable estimators. The former were introduced by Schucany (1989), building on work done by authors such as Loftsgaarden & Quesenberry (1965). The latter were introduced independently by Wagner (1975), Victor (1976) and Breiman et al. (1977).

The balloon estimator is

$$\hat{f}_B(x; h(x)) = n^{-1} \sum_{i=1}^{n} K_{h(x)}(x - X_i).$$

If we look at $\hat{f}_B$ at a single estimation point x_0 then $\hat{f}_B(x; h(x_0))$ is exactly the same as $\hat{f}(x; h(x_0))$, a fixed kernel density estimator with bandwidth $h(x_0)$. The bandwidth is a function of the estimation point and for a given point x_0, all the kernels have the same bandwidth $h(x_0)$. An illustration of this is in Figure 1.4. The data are the same as in Figure 1.1. The (arbitrary) bandwidth function is $h(x) = 0.0176 + 1/(x^2 + 1)$ and we look at two estimation points -0.5 and 1. The bandwidths are $h(-0.5) = 0.8176$ and $h(1) = 0.5176$. The dashed lines are the kernels corresponding to $h(-0.5)$ and the dotted lines are for $h(1)$. The balloon kernel density estimate is given by the solid line. Balloon estimators typically do not integrate to 1 so they are not true density functions, a result from focusing on estimating locally rather than globally. See Terrell & Scott (1992).

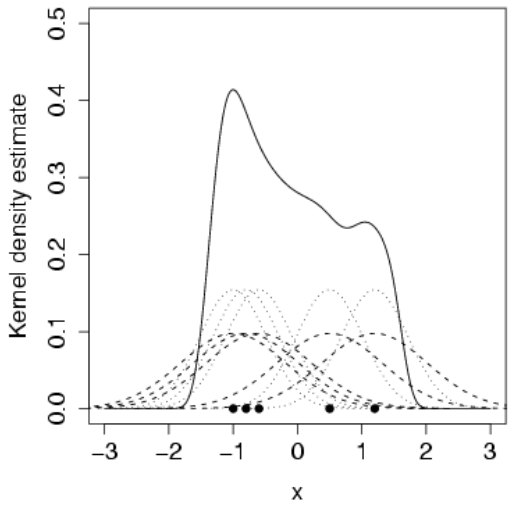

Figure 1.4: Univariate balloon kernel density estimate: solid line – kernel density estimate, dotted and dashed lines – individual kernels

Sample point estimators are given by

$$\hat{f}_{\mathrm{SP}}(x; \omega) = n^{-1} \sum_{i=1}^{n} K_{h_i}(x - X_i)$$

where $h_i = \omega(X_i), i = 1, 2, \ldots, n$. The difference between a sample point estimator and a fixed kernel density estimator is that for the former, each kernel has a different bandwidth.

It is also different from the balloon estimator as the bandwidths change at each of the data points rather than at each estimation point. We look at Figure 1.5. The data points have associated with them bandwidths $h_1 = 0.5070, h_2 = 0.6168, h_3 = 0.7423, h_4 = 0.8070, h_5 = 0.4169$. So the kernels are all normal kernels with different bandwidths (the dashed lines). To form the sample point kernel density estimator (the solid line), we sum these kernels and divide by n. Since each of the kernels is a density function, the sample point estimator remains a density function.

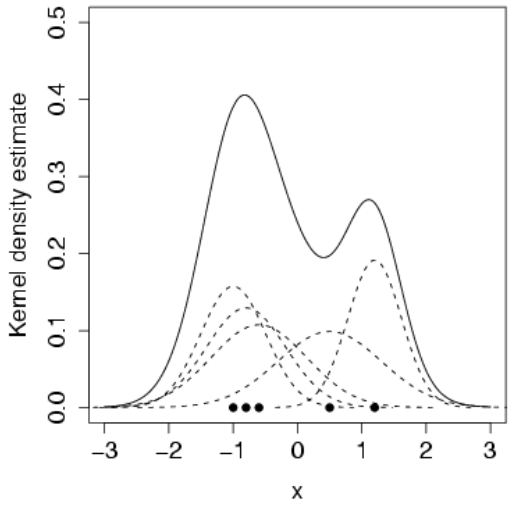

Figure 1.5: Univariate sample point kernel density estimate: solid line – kernel density estimate, dashed lines – individual kernels

In these methods, we need to select a bandwidth function $h(\cdot)$ or $\omega(\cdot)$. For the balloon estimators, the most common choice is to build up a bandwidth function by collating locally optimal bandwidths at each estimation point x. See Hazelton (1996), Hazelton (1999). For sample point estimators, Abramson (1982) shows that if $\omega(X_i) = hf(X_i)^{-1/2}$, where h is a constant, then this leads to an $O(h^4)$ bias rather than the usual $O(h^2)$ bias for fixed bandwidth estimators. This form of the bandwidth function appeals intuitively since it states that the smaller bandwidths should be used in those parts of the data set with high density of points (which is controlled by the value of f) and larger bandwidths in parts with lower density. This combination of small bandwidths near the modes and large bandwidths in the tails should be able to detect fine features near the former and prevent spurious features in the latter. Abramson's suggestion is to use a pilot estimate $\hat{f}_P$ to give $\hat{\omega}(X_i) = h\hat{f}_P(X_i)^{-1/2}$.

The theoretical improvement of using these variable bandwidth selectors is measured by changes in the rate of convergence of the MISE of the resulting kernel density esti-

mates. Recall that $h_{\text{AMISE}} = O(n^{-1/5})$ and at a single estimation point x_0, we have $\inf_{h>0} \text{MSE}\, \hat{f}(x_0; h) = O(n^{-4/5})$. This serves as our benchmark to compare the performance of the variable bandwidth selectors. From Jones (1990), the best possible convergence rate of the MSE of the balloon estimator $\inf_{h(x_0)>0} \text{MSE}\, \hat{f}_B(x_0; h(x_0))$ is $n^{-4/5}$ (i.e. the same as the fixed kernel density estimator) though it has a smaller constant of proportionality. The sample point estimator has a best possible rate (using the Abramson selector) for $\inf_{\omega(\cdot)>0} \text{MSE}\, \hat{f}_{\text{SP}}(x_0; \omega)$ of $n^{-8/9}$. Examples of studies of improvements with finite samples are given in Foster (1995) for balloon selectors and Sain & Scott (1996) for sample point selectors.

Terrell & Scott (1992) develop multivariate generalised kernel density estimators which unify the fixed kernel density estimator, balloon and sample point kernel estimators as well as other non-parametric density estimators (like frequency polygons and histograms), though they focus on balloon estimators. They generalise the sample point estimator of Breiman et al. (1977). They generalise the nearest neighbour estimator of Loftsgaarden & Quesenberry (1965) and develop a balloon version of the estimator from Deheuvels (1977) by using the curvature of f as well as the level of f. For another approach to balloon estimators, see Abdous & Berlinet (1998) and their Rao-Blackwellised estimator.

The general multivariate sample point estimator is

$$\hat{f}_{\text{SP}}(\boldsymbol{x}; \boldsymbol{\Omega}) = n^{-1} \sum_{i=1}^{n} K_{\boldsymbol{\Omega}(\boldsymbol{X}_i)}(\boldsymbol{x} - \boldsymbol{X}_i).$$

There are many choices for this $\boldsymbol{\Omega}$ function. The commonly used form attributed to Abramson (1982) is $\boldsymbol{\Omega}(\boldsymbol{X}_i) = h^2 f(\boldsymbol{X}_i)^{-1}\mathbf{I}$. Using the reciprocal of f leads to a higher order convergence for the bias, as in the univariate case. The problem then becomes producing an appropriate pilot estimate of f before selecting h. Breiman et al. (1977) use $\boldsymbol{\Omega}(\cdot)$ to be the k-th nearest neighbour function of $\boldsymbol{X}_i$ multiplied by the identity matrix. This requires us to choose the number of nearest neighbours (which can be viewed as an analogue to the bandwidth). Sain (2002) chooses $\boldsymbol{\Omega}(\cdot)$ to be a piecewise constant function, following from Sain & Scott (1996), over a partition of the data into m bins i.e. $\boldsymbol{\Omega}(\boldsymbol{X}_i) = \mathbf{H}_j$ if $\boldsymbol{X}_i \in$ bin j. Then a modified version of the LSCV is minimised to select appropriate bandwidth matrices.

Jones (1990) observes that we need not be restricted to exclusively to either of these classes of variable bandwidth selectors, that it is possible to combine these two approaches so that we have a bandwidth which depends on the data point and the estimation point. Another combination is taken by Cwik & Koronacki (1997*a*) who extend the univariate filtered kernel density estimate of Marchette et al. (1996) to higher dimensions. These authors use a partitioned bandwidth selector similar to Sain (2002): instead of smoothing at $\boldsymbol{X}_i$ according to $K_{\mathbf{H}_j}(\boldsymbol{x} - \boldsymbol{X}_i)$ only, smoothing is controlled by a weighted sum of

$K_{\mathbf{H}_1}(\boldsymbol{x} - \boldsymbol{X}_i), \ldots, K_{\mathbf{H}_m}(\boldsymbol{x} - \boldsymbol{X}_i)$. In effect all the different bandwidth matrices affect estimation at $\boldsymbol{X}_i$. The weights are determined by what the authors denote as filtering functions. Their algorithm to estimate these filtering functions assumes f to be a finite mixture density with known number of mixture components. In the simulation studies of Cwik & Koronacki (1997*b*), only normal mixture test densities are considered and they an EM type algorithm to fit normal mixtures. So it is not clear how this method will fare on other test densities.

1.4 Structure of thesis

At the moment, the most significant gap in the knowledge of fixed multivariate selectors is a concerted study of full bandwidth matrix selectors. In Chapter 2 we develop a fixed full bandwidth matrix selector using plug-in methods. We supply rates of convergence, a simulation study and applications to real data. In Chapter 3, we produce equivalents for cross validation selectors. In Chapter 4 we take a by-way into variable bandwidth selection, focusing on the partitioned selector which has a constant bandwidth within each partition class. We select these bandwidths by drawing upon the knowledge from the previous two chapters. In Chapter 5, we take a different by-way, this time into kernel discriminant analysis, applying kernel density estimation with full bandwidth matrices to this problem. In Chapter 6, we summarise all the results developed in this thesis and suggest future avenues of research. In the appendices, there are a list of notation, supplementary tables of results too detailed to fit into the main text and a description of the software developed by the author for data analysis.

Chapter 2

Plug-in bandwidth selectors

2.1 Introduction

Plug-in bandwidth selectors are based on the AMISE, implemented with pilot kernel estimates of functionals of the unknown target density f. Most important of these are the fourth order functionals in $\mathbf{\Psi}_4$ which are part of the asymptotic integrated squared bias. Plug-in selectors are already widely used for univariate kernel density estimation as they have demonstrated good theoretical and practical properties; they have a fast rate of convergence and have low variability. Multivariate plug-in selectors in comparison are less well studied and less widely used.

Current methods of plug-in bandwidth matrix selection are mostly for diagonal bandwidth matrices. Diagonal bandwidth matrices do indeed dramatically simplify the problem since it is considerably easier to select a diagonal matrix than a full one. However, we are now restricted to using kernels that are aligned to the co-ordinate axes and this will not be adequate for densities which have large probability mass not parallel to the axes. This was explored in Section 1.3.

To devise full plug-in selectors, we generalise existing diagonal plug-in selectors. We encounter some problems with the lack of positive definiteness of $\hat{\mathbf{\Psi}}_4$ if we simply use the pilot plug-in selectors of Wand & Jones (1994). Its positive definiteness is essential to the minimisation of the $\widehat{\text{AMISE}}$. We formulate a new pilot selector that guarantees the positive definiteness of $\hat{\mathbf{\Psi}}_4$ in Section 2.2. We supply the asymptotic analysis of the bandwidth selectors using these pilot selectors in Section 2.3 by examining the relative rate of convergence to the AMISE-optimal bandwidth matrix. We set up a general framework to compute asymptotic relative rates of convergence that will be used repeatedly throughout this thesis. This is followed by, in Section 2.5, an investigation of their finite sample properties with a simulation study and real data analysis. Whilst these lack the mathematical rigour of the asymptotic results, they *do* provide information at realistic sample sizes.

2.2 Optimal pilot bandwidth selectors

We develop a full bandwidth matrix selector in the following way. Let the plug-in criterion be

$$\mathrm{PI}(\mathbf{H}) = n^{-1}R(K)|\mathbf{H}|^{-1/2} + \tfrac{1}{4}\mu_2(K)^2(\mathrm{vech}^T\,\mathbf{H})\hat{\boldsymbol{\Psi}}_4(\mathrm{vech}\,\mathbf{H}). \tag{2.1}$$

This is the AMISE, Equation (1.5), with $\boldsymbol{\Psi}_4$ replaced by its estimate $\hat{\boldsymbol{\Psi}}_4$. Thus we wish to find $\hat{\mathbf{H}}_{\mathrm{PI}}$, the minimiser of $\mathrm{PI}(\mathbf{H})$. In order to do this, we need to compute $\hat{\boldsymbol{\Psi}}_4$. This is done via estimates of the $\psi_{\boldsymbol{r}}$ functionals, $\hat{\psi}_{\boldsymbol{r}}(\mathbf{G})$, where $\mathbf{G}$ is a pilot bandwidth matrix, usually different from $\mathbf{H}$. These are then substituted or plugged-into $\boldsymbol{\Psi}_4$. This procedure gives plug-in methods their name. The first step is to consider the problem of estimating integrated density derivative functionals i.e. how to compute $\hat{\psi}_{\boldsymbol{r}}(\mathbf{G})$ and how to select $\mathbf{G}$.

2.2.1 AMSE pilot bandwidth selectors

If we note that $\psi_{\boldsymbol{r}} = \int_{\mathbb{R}^d} f^{(\boldsymbol{r})}(\boldsymbol{x})f(\boldsymbol{x})\;d\boldsymbol{x} = \mathbb{E}\,f^{(\boldsymbol{r})}(\boldsymbol{X})$ where $\boldsymbol{X}$ has density f then the natural estimator of $\psi_{\boldsymbol{r}}$ is the sample mean of $\hat{f}^{(\boldsymbol{r})}(\boldsymbol{X}_i)$:

$$\hat{\psi}_{\boldsymbol{r}}(\mathbf{G}) = n^{-1}\sum_{i=1}^{n}\hat{f}^{(\boldsymbol{r})}(\boldsymbol{X}_i;\mathbf{G}) = n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}K_{\mathbf{G}}^{(\boldsymbol{r})}(\boldsymbol{X}_i-\boldsymbol{X}_j). \tag{2.2}$$

This is known as the leave-in-diagonals estimator as it includes all the non-stochastic terms where $i=j$. The bias of this estimator is

$$\begin{aligned}\mathrm{Bias}\,\hat{\psi}_{\boldsymbol{r}}(\mathbf{G}) &= n^{-1}K_{\mathbf{G}}^{(\boldsymbol{r})}(\mathbf{0}) + \tfrac{1}{2}\mu_2(K)\int_{\mathbb{R}^d}\mathrm{tr}(\mathbf{G}D^2f(\boldsymbol{x}))f^{(\boldsymbol{r})}(\boldsymbol{x})\;d\boldsymbol{x}\\ &\quad + o(n^{-1}|\mathbf{G}|^{-|\boldsymbol{r}|/2} + \|\mathrm{vech}\,\mathbf{G}\|)\end{aligned}$$

and the variance is

$$\begin{aligned}\mathrm{Var}\,\hat{\psi}_{\boldsymbol{r}}(\mathbf{G}) &= 2n^{-2}\psi_{\mathbf{0}}\int_{\mathbb{R}^d}K_{\mathbf{G}}^{(\boldsymbol{r})}(\boldsymbol{x})^2\;d\boldsymbol{x} + 4n^{-1}\left[\int_{\mathbb{R}^d}f^{(\boldsymbol{r})}(\boldsymbol{x})^2f(\boldsymbol{x})\;d\boldsymbol{x} - \psi_{\boldsymbol{r}}^2\right]\\ &\quad + o(n^{-2}|\mathbf{G}|^{-1/2}\|\mathrm{vech}\,\mathbf{G}^{-|\boldsymbol{r}|}\|+n^{-1}).\end{aligned}$$

Both expressions are taken from Wand & Jones (1995). Once again, we encounter the problem of choosing the parameterisation of a matrix selector: this time it is for the pilot bandwidth $\mathbf{G}$. Recall from Section 1.3 that the $h^2\mathbf{I}$ parameterisation was considered too restrictive for the final bandwidth $\mathbf{H}$. We relax this restriction for $\mathbf{G}$, following Wand & Jones (1994), as otherwise the symbolic manipulations become unwieldy. So we parameterise $\mathbf{G}$ as $g^2\mathbf{I}$. Now it appears that this will defeat the purpose of using full matrices for $\mathbf{H}$ but this is not the case. First, pilot bandwidths need not be specified to the same degree of accuracy as final bandwidths. Second, with appropriate pre-transforming of the data (discussed in Section 2.2.3), the effects of this more restricted parameterisation can

be somewhat mitigated. Third, the parameterisation of $\mathbf{G}$ does not affect the convergence rate of $\hat{\psi}_{\boldsymbol{r}}(\mathbf{G})$. So this is a suitable compromise between tractability and flexibility.

Let $\mathbf{G}$ be in the form $g^2\mathbf{I}$. Let $|\boldsymbol{r}| = j$ then the bias simplifies to

$$\operatorname{Bias}\hat{\psi}_{\boldsymbol{r}}(g) = n^{-1}g^{-d-j}K^{(\boldsymbol{r})}(\mathbf{0}) + \tfrac{1}{2}g^2\mu_2(K)\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i} + o(n^{-1}g^{-d-j} + g^2). \tag{2.3}$$

The variance simplifies to

$$\operatorname{Var}\hat{\psi}_{\boldsymbol{r}}(g) = 2n^{-2}g^{-d-2j}\psi_{\mathbf{0}}R(K^{(\boldsymbol{r})}) + o(n^{-2}g^{-d-2j}) \tag{2.4}$$

provided that $K^{(\boldsymbol{r})}$ is square integrable and $g = g_n \to 0$ and $n^{-1}g^{-d-2j} \to 0$ as $n \to \infty$. This leads to

$$\begin{aligned}\operatorname{AMSE}\hat{\psi}_{\boldsymbol{r}}(g) &= 2n^{-2}g^{-d-2j}\psi_{\mathbf{0}}R(K^{(\boldsymbol{r})}) \\ &\quad + \left[n^{-1}g^{-d-j}K^{(\boldsymbol{r})}(\mathbf{0}) + \tfrac{1}{2}g^2\mu_2(K)\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\right]^2.\end{aligned} \tag{2.5}$$

Thus we are seeking

$$g_{\boldsymbol{r},\mathrm{AMSE}} = \operatorname*{argmin}_{g>0}\operatorname{AMSE}\hat{\psi}_{\boldsymbol{r}}(g).$$

The following expressions for AMSE optimal pilot selectors are taken from Wand & Jones (1994). For most common kernels, including the normal kernel, if all the elements of $\boldsymbol{r}$ are even then $K^{(\boldsymbol{r})}(\mathbf{0})$ and $\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}$ will be of opposite sign, for $i = 1, 2, \ldots, d$. Then the bias terms will cancel each other if g is equal to

$$g_{\boldsymbol{r},\mathrm{AMSE}} = \left[\frac{-2K^{(\boldsymbol{r})}(\mathbf{0})}{\mu_2(K)\big(\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\big)n}\right]^{1/(d+j+2)}. \tag{2.6}$$

If at least one of the elements of $\boldsymbol{r}$ is odd then $K^{(\boldsymbol{r})}(\mathbf{0}) = 0$. In this case, we find the minimum AMSE if g is equal to

$$g_{\boldsymbol{r},\mathrm{AMSE}} = \left[\frac{2\psi_{\mathbf{0}}(2|\boldsymbol{r}| + d)R(K^{(\boldsymbol{r})})}{\mu_2(K)^2\big(\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\big)^2n^2}\right]^{1/(d+2j+4)}. \tag{2.7}$$

These expressions $g_{\boldsymbol{r},\mathrm{AMSE}}$ involve higher order $\psi_{\boldsymbol{r}}$ functionals. This dependency continues for all $\boldsymbol{r}$ so we need a way to resolve this problem. One convenient way is to use normal reference approximations. This is just

$$\hat{\psi}_{\boldsymbol{r}}^{\mathrm{NR}} = (-1)^{|\boldsymbol{r}|}\phi_{2\mathbf{S}}^{(\boldsymbol{r})}(\mathbf{0}) \tag{2.8}$$

where $\mathbf{S}$ is the sample variance. So starting with normal reference approximations of all $\psi_{\boldsymbol{r}}$ functionals for a given order, we can proceed to find estimates of the lower order $\psi_{\boldsymbol{r}}$ functionals.

This method of computing $\hat{\psi}_{\boldsymbol{r}}$ thus requires one pilot bandwidth for each functional. This means that computing $\hat{\boldsymbol{\Psi}}_4$ requires many separate pilot bandwidths. This is not a problem for diagonal bandwidth matrices. It is however a potential problem for full bandwidth matrices as the $\hat{\boldsymbol{\Psi}}_4$ estimated in this element-wise way is not guaranteed to be positive definite. This estimator could be non-positive definite and would lead to no solution to the optimisation of the PI($\mathbf{H}$) or it could be nearly singular and would lead to numerical instabilities. Hence using appropriate estimators of each element of a matrix will not necessarily lead to an appropriate estimator of the matrix as a whole. This motivates us to create a new pilot selector which does not suffer from this drawback i.e. we are, in effect, attempting to estimate a matrix in its entirety rather than element-wise. Positive-definiteness can be guaranteed by using a single, common pilot bandwidth for all $\psi_{\boldsymbol{r}}$ functionals, as we now demonstrate.

Lemma 1. *If a single pilot bandwidth matrix and normal kernels are used to estimate all the $\psi_{\boldsymbol{r}}$ functionals then $\hat{\boldsymbol{\Psi}}_4$ is positive definite.*

Proof. We notice that if we replace f with $\hat{f}(\cdot; \frac{1}{2}\mathbf{G})$ in $\psi_{\boldsymbol{r}}$, $|\boldsymbol{r}| = 4$, in Equation (1.7) then we have $\hat{\psi}_{\boldsymbol{r}}(\mathbf{G})$:

$$\begin{aligned}\int_{\mathbb{R}^d} \hat{f}^{(\boldsymbol{r})}(\boldsymbol{x}; \tfrac{1}{2}\mathbf{G}) \hat{f}(\boldsymbol{x}; \tfrac{1}{2}\mathbf{G})\, d\boldsymbol{x} &= n^{-2} \sum_{i=1}^{n}\sum_{j=1}^{n} \int_{\mathbb{R}^d} \phi^{(\boldsymbol{r})}_{\frac{1}{2}\mathbf{G}}(\boldsymbol{x} - \boldsymbol{X}_i) \phi_{\frac{1}{2}\mathbf{G}}(\boldsymbol{x} - \boldsymbol{X}_j)\, d\boldsymbol{x} \\ &= (-1)^{|\boldsymbol{r}|} n^{-2} \sum_{i=1}^{n}\sum_{j=1}^{n} \phi^{(\boldsymbol{r})}_{\mathbf{G}}(\boldsymbol{X}_i - \boldsymbol{X}_j) \\ &= \hat{\psi}_4(\mathbf{G}).\end{aligned}$$

This implies that $\hat{\boldsymbol{\Psi}}_4$ is obtained by replacing f with $\hat{f}(\cdot; \frac{1}{2}\mathbf{G})$ in $\boldsymbol{\Psi}_4$. From Equation (1.6), $\boldsymbol{\Psi}_4$ is positive definite by definition for all densities f. Since $\hat{f}(\cdot; \frac{1}{2}\mathbf{G})$ is a density function itself, $\hat{\boldsymbol{\Psi}}_4$ is positive definite. □

2.2.2 SAMSE pilot bandwidth selector

Modifying AMSE pilot selectors, we derive a SAMSE (Sum of Asymptotic Mean Squared Error) pilot selector. This type of selector has been specially devised to maintain the positive definiteness of $\hat{\boldsymbol{\Psi}}_4$ which is crucial to the numerical minimisation of the plug-in criterion PI. This selector is also simpler and more parsimonious than AMSE selectors.

We define SAMSE for the j-th order integrated density derivative functional estimators to be

$$\operatorname{SAMSE}_j(\mathbf{G}) = \sum_{\boldsymbol{r}:|\boldsymbol{r}|=j} \operatorname{AMSE} \hat{\psi}_{\boldsymbol{r}}(\mathbf{G}).$$

Since expressions for AMSE (and hence SAMSE) are difficult to derive for a full or even a diagonal $\mathbf{G}$ then we will again use the form $g^2\mathbf{I}$ as in Section 2.2.1. We wish to find

$$g_{j,\mathrm{SAMSE}} = \operatorname*{argmin}_{g>0} \mathrm{SAMSE}_j(g).$$

The SAMSE criterion is rewritten as follows:

$$\begin{aligned}\sum_{\boldsymbol{r}:|\boldsymbol{r}|=j} \mathrm{AMSE}\,\hat{\psi}_{\boldsymbol{r}}(\mathbf{G}) &= \sum_{\boldsymbol{r}:|\boldsymbol{r}|=j} 2n^{-2}g^{-2j-d}R(K^{(\boldsymbol{r})}) \\ &\quad + \sum_{\boldsymbol{r}:|\boldsymbol{r}|=j}\left[n^{-1}g^{-j-d}K^{(\boldsymbol{r})}(\mathbf{0}) + \tfrac{1}{2}g^2\mu_2(K)\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\right]^2 \\ &= 2n^{-2}g^{-2j-d}A_0 + n^{-2}g^{-2j-2d}A_1 + n^{-1}g^{-j-d+2}A_2 + \tfrac{1}{4}g^4A_3\end{aligned}$$

where A_0, A_1, A_2 and A_3 are constants (i.e. containing K and f but not n) defined by

$$\begin{aligned}A_0 &= \sum_{\boldsymbol{r}:|\boldsymbol{r}|=j} R(K^{(\boldsymbol{r})}) \\ A_1 &= \sum_{\boldsymbol{r}:|\boldsymbol{r}|=j} K^{(\boldsymbol{r})}(\mathbf{0})^2 \\ A_2 &= \mu_2(K)\sum_{\boldsymbol{r}:|\boldsymbol{r}|=j} K^{(\boldsymbol{r})}(\mathbf{0})\left(\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\right) \\ A_3 &= \mu_2(K)^2\sum_{\boldsymbol{r}:|\boldsymbol{r}|=j}\left(\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\right)^2.\end{aligned}$$

We can see that A_0, A_1 and A_3 are positive by construction. A_2 is negative because if all elements if $\boldsymbol{r}$ are even, $K^{(\boldsymbol{r})}(\mathbf{0})$ and $\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}$ are of opposite sign and if at least one of its elements is odd, $K^{(\boldsymbol{r})}(\mathbf{0}) = 0$.

We can simplify this expression as the first term is $O(n^{-2}g^{-2j-d})$ and the second term is $O(n^{-2}g^{-2j-2d})$ which means the latter always dominates the former. If we remove the first term (which is the asymptotic variance) we are left with

$$\mathrm{SAMSE}_j(g) = n^{-2}g^{-2j-2d}A_1 + n^{-1}g^{-j-d+2}A_2 + \tfrac{1}{4}g^4A_3. \tag{2.9}$$

In effect, we are only considering the contribution of the squared bias. Then differentiating this with respect to g gives

$$\frac{\partial}{\partial g}\mathrm{SAMSE}_j(g) = -(2j+2d)n^{-2}g^{-2j-2d-1}A_1 - (j+d-2)n^{-1}g^{-j-d+1}A_2 + g^3A_3.$$

This is a quadratic in $n^{-1}g^{-j-d-2}$ and has solution

$$g_{j,\mathrm{SAMSE}} = \left[\frac{(4j+4d)A_2}{\left((-j-d+2)A_2 + \sqrt{(-j-d+2)^2A_2^2 + (8j+8d)A_1A_3}\right)n}\right]^{1/(j+d+2)}. \tag{2.10}$$

This is the j-th order SAMSE pilot bandwidth. Lemma 1 demonstrates that under given conditions using any single, common pilot bandwidth selector does indeed guarantee the positive definiteness of $\hat{\mathbf{\Psi}}_4$. Thus it follows immediately that using the SAMSE pilot bandwidth guarantees positive definiteness.

The other main advantage of SAMSE pilot selectors is that they are more parsimonious than AMSE pilot selectors when we compare the number of pilot bandwidths (computed with a kernel estimate rather than with normal reference) and final bandwidths that each selector requires. An m-stage diagonal bandwidth matrix with AMSE pilots computes

$$\sum_{i=1}^{m} \sum_{j=0}^{\min(i,d-1)} \binom{i}{j} \binom{d}{j+1}$$

pilot plus d final bandwidths. An m-stage full bandwidth matrix with AMSE pilots computes

$$\nu_m + \sum_{i=1}^{m} \sum_{j=0}^{\min(2i,d-1)} \binom{2i+1}{j} \binom{d}{j+1}$$

pilot, where $\nu_1 = 0, \nu_1 = 1, \nu_2 = 3$ and for $m = 4, 5, 6, \ldots$

$$\nu_m = \sum_{i=1}^{m-3} \sum_{j=0}^{\min(i,d-1)} \binom{i}{j} \binom{d}{j+1},$$

plus $\frac{1}{2}d(d+1)$ final bandwidths. These expressions for the number of AMSE pilot bandwidths are taken from Wand & Jones (1994). An m-stage full bandwidth matrix with SAMSE pilots computes m pilots + $\frac{1}{2}d(d+1)$ final bandwidths. Table 2.2.2 contains these counts for $m = 2$ and for $d = 1, 2, \ldots, 6$. We can see that SAMSE selectors remain feasible for all dimensions listed in the table whilst AMSE selectors start to become infeasible for $d > 3$ since the number of bandwidths required grows combinatorially.

	Number of pilot plus final bandwidths					
	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
Diagonal $\mathbf{H}$ with AMSE pilots	3	9	19	34	55	83
Full $\mathbf{H}$ with AMSE pilots	3	16	50	130	296	610
Full $\mathbf{H}$ with SAMSE pilots	3	5	8	12	17	23

Table 2.1: Number of pilot and final bandwidths for 2-stage plug-in selectors

2.2.3 Pre-scaling and pre-sphering

In the previous sections we parameterise $\mathbf{G}$ as $g^2\mathbf{I}$. To use this parameterisation effectively, each component of the data vector should be commensurate. So we transform the data $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ before any pilot bandwidth selection. A common transformation is *pre-scaling*. By pre-scaling, we transform the data so that they have unit variance in each

co-ordinate direction. Let $\boldsymbol{X}^*$ be the scaled version of $\boldsymbol{X}$, i.e. $\boldsymbol{X}^* = \mathbf{S}_{\mathcal{D}}^{-1/2}\boldsymbol{X}$ where $\mathbf{S}_{\mathcal{D}} = \operatorname{dg}\mathbf{S}$. This means that

$$\boldsymbol{X}^* = (S_1^{-1}X_1, S_2^{-1}X_2, \ldots, S_d^{-1}X_d)$$

where S_i^2 is the i-th marginal sample variance. Let $\mathbf{S}_{\mathcal{D}}^*$ be the sample variance of the scaled data then

$$\mathbf{S}_{\mathcal{D}}^* = \widehat{\operatorname{Var}\boldsymbol{X}}^* = \mathbf{S}_{\mathcal{D}}^{-1/2}(\widehat{\operatorname{Var}\boldsymbol{X}})\mathbf{S}_{\mathcal{D}}^{-1/2} = \mathbf{S}_{\mathcal{D}}^{-1/2}\mathbf{S}\mathbf{S}_{\mathcal{D}}^{-1/2} = \begin{bmatrix} 1 & \frac{S_{12}}{S_1S_2} & \cdots & \frac{S_{1d}}{S_1S_d} \\ \vdots & & & \vdots \\ \frac{S_{1d}}{S_1S_d} & \frac{S_{2d}}{S_2S_d} & \cdots & 1 \end{bmatrix}.$$

Another transformation that could be applied to the data, before pilot bandwidth selection, is *pre-sphering*. Pre-sphering transforms the data so that their variance is now the identity matrix. So here the data are rotated as well as dilated/contracted whereas scaling only dilates/contracts the data. The sphering transformation is $\boldsymbol{X}^* = \mathbf{S}^{-1/2}\boldsymbol{X}$. Then the variance of the pre-sphered data is

$$\mathbf{S}^* = \widehat{\operatorname{Var}\boldsymbol{X}}^* = \mathbf{S}^{-1/2}(\widehat{\operatorname{Var}\boldsymbol{X}})\mathbf{S}^{-1/2} = \mathbf{S}^{-1/2}\mathbf{S}\mathbf{S}^{-1/2} = \mathbf{I}.$$

Once we have pre-transformed the data, we can find a bandwidth $\mathbf{H}^*$ on this transformed scale. The next lemma answers the question of how to find $\mathbf{H}$, the bandwidth on the original data scale, from $\mathbf{H}^*$.

Lemma 2. *If* $\mathbf{H}$ *is the bandwidth matrix for the original data and* $\mathbf{H}^*$ *is the bandwidth matrix for the pre-sphered data then*

$$\mathbf{H} = \mathbf{S}^{1/2}\mathbf{H}^*\mathbf{S}^{1/2}.$$

A corresponding result holds for pre-scaled data with $\mathbf{S}$ *replaced by* $\mathbf{S}_{\mathcal{D}}$.

Proof. We show this by first considering the kernel density estimate on the sphered data:

$$\begin{aligned}
\hat{f}^*(\boldsymbol{x}^*;\mathbf{H}^*) &= n^{-1}\sum_{i=1}^{n} K_{\mathbf{H}^*}\big(\boldsymbol{x}^* - \boldsymbol{X}_i^*\big) \\
&= n^{-1}|\mathbf{H}^*|^{-1/2}\sum_{i=1}^{n} K\big(\mathbf{H}^{*-1/2}(\boldsymbol{x}^* - \boldsymbol{X}_i^*)\big) \\
&= n^{-1}|\mathbf{H}^*|^{-1/2}\sum_{i=1}^{n} K\big((\mathbf{S}^{1/2}\mathbf{H}^{*1/2})^{-1}(\boldsymbol{x} - \boldsymbol{X}_i)\big) \\
&= n^{-1}|\mathbf{S}|^{1/2}|\mathbf{S}^{1/2}\mathbf{H}^*\mathbf{S}^{1/2}|^{-1/2}\sum_{i=1}^{n} K\big((\mathbf{S}^{1/2}\mathbf{H}^*\mathbf{S}^{1/2})^{-1/2}(\boldsymbol{x} - \boldsymbol{X}_i)\big).
\end{aligned}$$

The last equality follows from the result that if $\mathbf{A}$ and $\mathbf{B}$ are positive definite and symmetric matrices then $(\mathbf{B}^{1/2}\mathbf{A}\mathbf{B}^{1/2})^{1/2} = \mathbf{B}^{1/2}\mathbf{A}^{1/2}$. Since $\boldsymbol{x}^* = \mathbf{S}^{-1/2}\boldsymbol{x}$ is a change of variables, then $\hat{f}^*(\boldsymbol{x}^*;\mathbf{H}^*) = |\mathbf{S}|^{1/2}\hat{f}(\boldsymbol{x};\mathbf{H})$ and thus $\mathbf{H} = \mathbf{S}^{1/2}\mathbf{H}^*\mathbf{S}^{1/2}$. Furthermore, $\mathbf{S}$ can be replaced with $\mathbf{S}_{\mathcal{D}}$ to give a corresponding result for pre-scaling. □

2.3 Convergence rates for plug-in selectors

The performance of a bandwidth matrix selector can be assessed by its relative rate of convergence. We need to adapt the definition for the relative rate for a univariate selector in Equation (1.8): a matrix selector $\hat{\mathbf{H}}$ converges to $\mathbf{H}_{\text{AMISE}}$ with relative rate $n^{-\alpha}$ if

$$\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\text{AMISE}}) = O_p(\mathbf{J}_{d'} n^{-\alpha}) \operatorname{vech} \mathbf{H}_{\text{AMISE}} \tag{2.11}$$

where $\mathbf{J}_{d'}$ is the $d' \times d'$ matrix of ones and $d' = \frac{1}{2}d(d+1)$. Here we extend the asymptotic order notation to matrix sequences. Specifically let $\{\mathbf{A}_n\}$ and $\{\mathbf{B}_n\}$ be matrix sequences with $\mathbf{A}_n$ and $\mathbf{B}_n$ having the same dimensions. We write $\mathbf{A}_n = o(\mathbf{B}_n)$ if $[\mathbf{A}_n]_{ij} = o([\mathbf{B}_n]_{ij})$ for all elements $[\mathbf{A}_n]_{ij}$ of $\mathbf{A}_n$ and $[\mathbf{B}_n]_{ij}$ of $\mathbf{B}_n$. This definition, for the one dimensional case reduces to the usual relative rate of convergence, Equation (1.8). At first glance, it appears that the 'straightforward' multi-dimensional generalisation is $\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\text{AMISE}}) = O_p(\mathbf{I}_{d'} n^{-\alpha}) \operatorname{vech} \mathbf{H}_{\text{AMISE}}$ i.e. using $\mathbf{I}_{d'}$ rather than $\mathbf{J}_{d'}$. This is not adequate for cases when the off-diagonal elements of $\mathbf{H}_{\text{AMISE}}$ are identically zero (e.g. when the variance of f is a diagonal matrix) because then left hand side is identically zero and the relative rate then is undefined. Our definition using $\mathbf{J}_{d'}$ prevents such problems by taking linear combinations of elements of $\mathbf{H}_{\text{AMISE}}$ as these linear combinations include at least one non-zero diagonal element. So in effect we are defining rates of convergence based on the 'overall' order of $\mathbf{H}_{\text{AMISE}}$ rather than a purely element-wise order. Of course this notion of an overall order of $\mathbf{H}_{\text{AMISE}}$ relies on the fact that its elements are of the same order.

We also have corresponding definitions for O, o_p and O_p. The preceding definitions can all be defined in terms of $\mathbf{H}_{\text{MISE}}$ as well. Equation (2.11) can be unwieldy since we do not a closed form for $\hat{\mathbf{H}}$ in most cases. We now look for an alternative route to finding relative convergence rates using the next lemma which we will call the 'AMSE Lemma'.

Lemma 3 (AMSE). *Assume that*

(A1) All entries in $D^2 f(\boldsymbol{x})$ are bounded, continuous and square integrable.

(A2) All entries of $\mathbf{H} \to 0$ and $n^{-1}|\mathbf{H}|^{-1/2} \to 0$, as $n \to \infty$.

(A3) K is a spherically symmetric probability density.

Let $\hat{\mathbf{H}} = \operatorname*{argmin}_{\mathbf{H} \in \mathcal{H}} \widehat{\text{AMISE}}(\mathbf{H})$ be a bandwidth selector and define its mean squared error (MSE) by

$$\operatorname{MSE}(\operatorname{vech} \hat{\mathbf{H}}) = \mathbb{E}[\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\text{AMISE}}) \operatorname{vech}^T(\hat{\mathbf{H}} - \mathbf{H}_{\text{AMISE}})].$$

Then $\operatorname{MSE}(\operatorname{vech} \hat{\mathbf{H}}) = [\mathbf{I}_{d'} + o(\mathbf{J}_{d'})] \operatorname{AMSE}(\operatorname{vech} \hat{\mathbf{H}})$ where the asymptotic MSE can be written as

$$\operatorname{AMSE}(\operatorname{vech} \hat{\mathbf{H}}) = \operatorname{AVar}(\operatorname{vech} \hat{\mathbf{H}}) + [\operatorname{ABias}(\operatorname{vech} \hat{\mathbf{H}})][\operatorname{ABias}(\operatorname{vech} \hat{\mathbf{H}})]^T$$

in which

$$\begin{aligned}\operatorname{ABias}(\operatorname{vech}\hat{\mathbf{H}}) &= [D^2_{\mathbf{H}}\operatorname{AMISE}(\mathbf{H}_{\operatorname{AMISE}})]^{-1}\,\mathbb{E}[D_{\mathbf{H}}(\widehat{\operatorname{AMISE}} - \operatorname{AMISE})(\mathbf{H}_{\operatorname{AMISE}})]\\ \operatorname{AVar}(\operatorname{vech}\hat{\mathbf{H}}) &= [D^2_{\mathbf{H}}\operatorname{AMISE}(\mathbf{H}_{\operatorname{AMISE}})]^{-1}\operatorname{Var}[D_{\mathbf{H}}(\widehat{\operatorname{AMISE}} - \operatorname{AMISE})(\mathbf{H}_{\operatorname{AMISE}})]\\ &\quad\times [D^2_{\mathbf{H}}\operatorname{AMISE}(\mathbf{H}_{\operatorname{AMISE}})]^{-1}.\end{aligned}$$

Here $D_{\mathbf{H}}$ is the differential operator with respect to $\operatorname{vech}\mathbf{H}$ *and $D^2_{\mathbf{H}}$ is the corresponding Hessian operator.*

Proof. We may expand $D_{\mathbf{H}}\widehat{\operatorname{AMISE}}$ as follows:

$$\begin{aligned}D_{\mathbf{H}}\widehat{\operatorname{AMISE}}(\hat{\mathbf{H}}) &= D_{\mathbf{H}}(\widehat{\operatorname{AMISE}} - \operatorname{AMISE})(\hat{\mathbf{H}}) + D_{\mathbf{H}}\operatorname{AMISE}(\hat{\mathbf{H}})\\ &= [\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})]D_{\mathbf{H}}(\widehat{\operatorname{AMISE}} - \operatorname{AMISE})(\mathbf{H}_{\operatorname{AMISE}}) + \big\{D_{\mathbf{H}}\operatorname{AMISE}(\mathbf{H}_{\operatorname{AMISE}})\\ &\quad + [\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})]D^2_{\mathbf{H}}\operatorname{AMISE}(\mathbf{H}_{\operatorname{AMISE}})\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{AMISE}})\big\}.\end{aligned}$$

We have $D_{\mathbf{H}}\widehat{\operatorname{AMISE}}(\hat{\mathbf{H}}) = \mathbf{0}$ and $D_{\mathbf{H}}\operatorname{AMISE}(\mathbf{H}_{\operatorname{AMISE}}) = \mathbf{0}$. This implies that

$$\begin{aligned}\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{AMISE}}) &= -[\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})][D^2_{\mathbf{H}}\operatorname{AMISE}(\mathbf{H}_{\operatorname{AMISE}})]^{-1}\\ &\quad\times D_{\mathbf{H}}(\widehat{\operatorname{AMISE}} - \operatorname{AMISE})(\mathbf{H}_{\operatorname{AMISE}}).\end{aligned}$$

Taking expectations and variances respectively completes the proof.

We choose this particular expansion because we can ascertain from it that the closeness of $\hat{\mathbf{H}}$ to $\mathbf{H}_{\operatorname{AMISE}}$ is driven by the closeness of $\widehat{\operatorname{AMISE}}$ to AMISE i.e. our selector will be closer to its target if our estimate of the error criterion is better. □

The AMSE Lemma (Lemma 3) forms a central component of our strategy to compute the relative convergence rates of $\hat{\mathbf{H}}$ to $\mathbf{H}_{\operatorname{AMISE}}$:

1. Find expressions for the order of the expected value and variance of $D_{\mathbf{H}}(\widehat{\operatorname{AMISE}} - \operatorname{AMISE})(\mathbf{H}_{\operatorname{AMISE}})$. They are the same order as, and most importantly, *easier* to evaluate than $\operatorname{ABias}(\operatorname{vech}\hat{\mathbf{H}})$ and $\operatorname{AVar}(\operatorname{vech}\hat{\mathbf{H}})$.

2. Combine $\operatorname{ABias}(\operatorname{vech}\hat{\mathbf{H}})$ and $\operatorname{AVar}(\operatorname{vech}\hat{\mathbf{H}})$ into $\operatorname{AMSE}(\operatorname{vech}\hat{\mathbf{H}})$ and note that if $\operatorname{MSE}(\operatorname{vech}\hat{\mathbf{H}}) = O(\mathbf{J}_{d'}n^{-2\alpha})(\operatorname{vech}\mathbf{H}_{\operatorname{AMISE}})(\operatorname{vech}\mathbf{H}_{\operatorname{AMISE}})^T$ then $\hat{\mathbf{H}}$ has relative rate $n^{-\alpha}$.

The AMSE Lemma can be adapted to consider convergence to $\mathbf{H}_{\operatorname{MISE}}$ by replacing all references to AMISE by MISE. Nonetheless, it is generally simpler to consider convergence to $\mathbf{H}_{\operatorname{AMISE}}$ and then examine whether the discrepancy between $\mathbf{H}_{\operatorname{MISE}}$ and its asymptotic form is significant.

For the plug-in selectors, the estimate of AMISE is PI. We have

$$(\operatorname{PI} - \operatorname{AMISE})(\mathbf{H}) = \tfrac{1}{4}\mu_2(K)^2(\operatorname{vech}^T\mathbf{H})(\hat{\boldsymbol{\Psi}}_4 - \boldsymbol{\Psi}_4)(\operatorname{vech}\mathbf{H})[1 + o_p(1)]$$

so

$$D_{\mathbf{H}}(\mathrm{PI}-\mathrm{AMISE})(\mathbf{H}) = \tfrac{1}{2}\mu_2(K)^2[\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})](\hat{\mathbf{\Psi}}_4 - \mathbf{\Psi}_4)(\operatorname{vech}\mathbf{H}).$$

Then we have

$$\begin{aligned}\mathbb{E}[D_{\mathbf{H}}(\mathrm{PI}-\mathrm{AMISE})(\mathbf{H})] &= \tfrac{1}{2}\mu_2(K)^2[\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})](\operatorname{Bias}\hat{\mathbf{\Psi}}_4)(\operatorname{vech}\mathbf{H})\\ \operatorname{Var}[D_{\mathbf{H}}(\mathrm{PI}-\mathrm{AMISE})(\mathbf{H})] &= \tfrac{1}{4}\mu_2(K)^4[\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})]\operatorname{Var}[\hat{\mathbf{\Psi}}_4(\operatorname{vech}\mathbf{H})].\end{aligned}$$

These expressions will be used in the next two lemmas where we compute the asymptotic bias and variance of the AMSE and SAMSE plug-in selectors, which we denote as $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ and $\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$.

Lemma 4. *Assume A1 – A3 from Lemma 3. Further assume that $K^{(\boldsymbol{r})}$ is square integrable and that if $|\boldsymbol{r}| = 4$ then $K^{(\boldsymbol{r})}(\mathbf{0}) = 1$ if all elements of $\boldsymbol{r}$ are even and $K^{(\boldsymbol{r})}(\mathbf{0}) = 0$ otherwise. If we use the* AMSE *pilot bandwidths then*

$$\begin{aligned}\operatorname{ABias}(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}) &= O(\mathbf{J}_{d'}n^{-4/(d+12)})\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}\\ \operatorname{AVar}(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}) &= O(\mathbf{J}_{d'}n^{-8/(d+12)})(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}}).\end{aligned}$$

Proof. Following Wand & Jones (1994), let $|\boldsymbol{r}| = j$ then the bias and variance of $\hat{\psi}_{\boldsymbol{r}}(g)$ are respectively:

$$\begin{aligned}\operatorname{Bias}\hat{\psi}_{\boldsymbol{r}}(g) &= n^{-1}g^{-d-j}K^{(\boldsymbol{r})}(\mathbf{0}) + \tfrac{1}{2}g^2\mu_2(K)\sum_{i=1}^{d}\psi_{\boldsymbol{r}+\boldsymbol{e}_i} + O(g^4)\\ \operatorname{Var}\hat{\psi}_{\boldsymbol{r}}(g) &= 2n^{-2}g^{-d-2j}\psi_{\mathbf{0}}R(K^{(\boldsymbol{r})}) + o(n^{-2}g^{-d-2j}).\end{aligned}$$

There are two cases we need to consider. From Section 2.2.1, if all elements of $\boldsymbol{r}$ are even then the pilot bandwidth which minimises the AMSE is $g_{\boldsymbol{r},\mathrm{AMSE}} = O(n^{-1/(j+d+2)})$. This choice of g is a result from the annihilation of the leading terms of the bias so then

$$\begin{aligned}\operatorname{Bias}\hat{\psi}_{\boldsymbol{r}}(g_{\boldsymbol{r},\mathrm{AMSE}}) &= O(g_{\boldsymbol{r},\mathrm{AMSE}}^4) = O(n^{-4/(d+j+2)})\\ \operatorname{Var}\hat{\psi}_{\boldsymbol{r}}(g_{\boldsymbol{r},\mathrm{AMSE}}) &= O(n^{-2}g_{\boldsymbol{r},\mathrm{AMSE}}^{-d-2j}) = O(n^{-(d+4)/(d+j+2)}).\end{aligned}$$

On the other hand, if at least one element of $\boldsymbol{r}$ is odd then $K^{(\boldsymbol{r})}(\mathbf{0}) = 0$ and the pilot bandwidth which minimises the AMSE is $g_{\boldsymbol{r},\mathrm{AMSE}} = O(n^{-2/(d+2j+4)})$. Then the bias and variance are

$$\begin{aligned}\operatorname{Bias}\hat{\psi}_{\boldsymbol{r}}(g_{\boldsymbol{r},\mathrm{AMSE}}) &= O(g_{\boldsymbol{r},\mathrm{AMSE}}^2) = O(n^{-4/(d+2j+4)})\\ \operatorname{Var}\hat{\psi}_{\boldsymbol{r}}(g_{\boldsymbol{r},\mathrm{AMSE}}) &= O(n^{-2}g_{\boldsymbol{r},\mathrm{AMSE}}^{-d-2j}) = O(n^{-8/(d+2j+4)}).\end{aligned}$$

Combining these two cases together we have $\mathbb{E}\,\hat{\boldsymbol{\Psi}}_4 - \boldsymbol{\Psi}_4 = O(\mathbf{J}_{d'} n^{-4/(d+2j+2)})$ and $\operatorname{Var}[\hat{\boldsymbol{\Psi}}_4(\operatorname{vech}\mathbf{H})] = O(\mathbf{J}_{d'}(n^{-(d+4)/(d+j+2)} + n^{-8/(d+2j+4)}))(\operatorname{vech}\mathbf{H})(\operatorname{vech}^T\mathbf{H})$. Thus

$$\begin{aligned}\mathbb{E}[D_{\mathbf{H}}(\mathrm{PI}-\mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})] &= O(\mathbf{J}_{d'} n^{-4/(d+2j+2)})\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}\\ \operatorname{Var}[D_{\mathbf{H}}(\mathrm{PI}-\mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})] &= O(\mathbf{J}_{d'}(n^{-(d+4)/(d+j+2)} + n^{-8/(d+2j+4)}))\\ &\quad\times(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}}).\end{aligned}$$

The result follows as $j = 4$ and $D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H}_{\mathrm{AMISE}}) = O(\mathbf{J}_d)$. From Wand (1992) the Hessian matrix of $\mathrm{AMISE}(\mathbf{H})$ is

$$\begin{aligned}D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H}) &= \tfrac{1}{4}n^{-1}(4\pi)^{-d/2}|\mathbf{H}|^{-1/2}\mathbf{D}_d^T(\mathbf{H}^{-1}\otimes\mathbf{I}_d)\\ &\quad\times[(\operatorname{vec}\mathbf{I}_d)(\operatorname{vec}^T\mathbf{I}_d) + 2\mathbf{I}_{d^2}](\mathbf{I}_d\otimes\mathbf{H}^{-1})\mathbf{D}_d + \tfrac{1}{2}\boldsymbol{\Psi}_4.\end{aligned}$$

As $\mathbf{H}_{\mathrm{AMISE}} = O(\mathbf{J}_{d'} n^{-2/(d+4)})$ then $D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H})$ tends to a constant, positive definite matrix as $n\to\infty$. □

Lemma 5. *Assume A1 – A3 from Lemma 3. Further assume that $K^{(\boldsymbol{r})}$ is square integrable and that if $|\boldsymbol{r}| = 4$ then $K^{(\boldsymbol{r})}(\mathbf{0}) = 1$ if all elements of $\boldsymbol{r}$ are even and $K^{(\boldsymbol{r})}(\mathbf{0}) = 0$ otherwise. If we use the SAMSE pilot bandwidth then*

$$\begin{aligned}\mathrm{ABias}(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}) &= O_p(\mathbf{J}_{d'} n^{-2/(d+6)})\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}\\ \mathrm{AVar}(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}) &= O_p(\mathbf{J}_{d'} n^{-4/(d+6)})(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}}).\end{aligned}$$

Proof. From Section 2.2.2, the j-th order SAMSE pilot bandwidth is $g_{j,\mathrm{SAMSE}}$ is order $n^{-1/(j+d+2)}$. If all elements of $\boldsymbol{r}$ are even then the bias is

$$\operatorname{Bias}\hat{\psi}_{\boldsymbol{r}}(g_{j,\mathrm{SAMSE}}) = O(n^{-1}g_{j,\mathrm{SAMSE}}^{-d-j} + g_{j,\mathrm{SAMSE}}^2) = O(n^{-2/(d+j+2)}).$$

On the other hand, if at least one element of $\boldsymbol{r}$ is odd then $K^{(\boldsymbol{r})}(\mathbf{0}) = 0$ and the bias is

$$\operatorname{Bias}\hat{\psi}_{\boldsymbol{r}}(g_{j,\mathrm{SAMSE}}) = O(g_{j,\mathrm{SAMSE}}^2) = O(n^{-2/(d+j+2)}).$$

Combining these together we have that $\mathbb{E}\,\hat{\boldsymbol{\Psi}}_4 - \boldsymbol{\Psi}_4 = O(\mathbf{J}_{d'} n^{-2/(d+j+2)})$ and so

$$\mathbb{E}[D_{\mathbf{H}}(\mathrm{PI}-\mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})] = O(\mathbf{J}_{d'} n^{-2/(d+j+2)})\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}.$$

To form the SAMSE, we exclude the variances of $\hat{\psi}_{\boldsymbol{r}}$ as they are dominated by the leading terms of the squared bias i.e.

$$\operatorname{Var}[\hat{\boldsymbol{\Psi}}_4(\operatorname{vech}\mathbf{H})] = O((\mathbf{J}_{d'} n^{-4/(d+j+2)})(\operatorname{vech}\mathbf{H})(\operatorname{vech}^T\mathbf{H})$$

which implies that

$$\operatorname{Var}[D_{\mathbf{H}}(\mathrm{PI}-\mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})] = O(\mathbf{J}_{d'} n^{-4/(d+j+2)})(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}}).$$

Substituting $j = 4$ and $D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H}_{\mathrm{AMISE}}) = O(\mathbf{J}_d)$ completes the proof. □

Putting Lemmas 4 and 5 together with the AMSE Lemma (Lemma 3) we can state the following theorem about the convergence rates for plug-in selectors.

Theorem 1. *Under the conditions of Lemmas 4 and 5,*

1. *The relative rate of convergence of* $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ *is* $n^{-4/(d+12)}$.

2. *The relative rate of convergence of* $\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$ *is* $n^{-2/(d+6)}$.

The additional conditions on K in Lemmas 4 and 5 are satisfied by most common kernels including the normal kernel. The relative rate of convergence of $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ to $\mathbf{H}_{\mathrm{AMISE}}$ is slightly faster than that of $\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$. See Table 2.2 for the rates for d up to 6. For the important bivariate case, the rate for $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ is $n^{-2/7}$ and for $\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$ is $n^{-1/4}$. For a sample of size $n = 100\ 000$ the ratio of $n^{-2/7}$ to $n^{-1/4}$ is about 1.5, so just considering that convergence rates we will not offer compelling evidence of which plug-in selector to use in practice. Wand & Jones (1994) show that their diagonal the plug-in selector has rate $n^{-\min(8,d+4)/(2d+12)}$. The rate persists even if the $h^2\mathbf{I}$ parameterisation is used instead. Jones (1992, Table 3) contains convergence rates for selectors of the $h^2\mathbf{I}$ parameterisation. The rate here agrees with our rate for the diagonal $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$. This rate is faster than those for the full bandwidth selectors. This implies that selecting the off-diagonal elements of the full bandwidth matrix, which determine the orientation of the kernel, is the most difficult aspect of full plug-in selection. Also in this table is the rate for the Park & Marron (1990) plug-in selector which turns out to have the same $n^{-4/(d+12)}$ rate as the full $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ selector, even though they use different estimators for the $\psi_{\boldsymbol{r}}$ functionals. The final row in Table 2.2 is the relative discrepancy between $\mathbf{H}_{\mathrm{AMISE}}$ and $\mathbf{H}_{\mathrm{MISE}}$. It is straightforward to show that

$$\operatorname{vech}(\mathbf{H}_{\mathrm{AMISE}} - \mathbf{H}_{\mathrm{MISE}}) = O(\mathbf{I}_{d'} n^{-2/(d+4)}) \operatorname{vech} \mathbf{H}_{\mathrm{MISE}}.$$

If this discrepancy is smaller than the rate of convergence of $\hat{\mathbf{H}}$ to $\mathbf{H}_{\mathrm{AMISE}}$ then $\hat{\mathbf{H}}$ will have the same rate with respect to $\mathbf{H}_{\mathrm{MISE}}$. This is indeed the case for $\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$. However, the discrepancy between $\mathbf{H}_{\mathrm{AMISE}}$ and $\mathbf{H}_{\mathrm{MISE}}$ dominates the rate for $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ for $d > 4$.

	Convergence rate to $\mathbf{H}_{\mathrm{AMISE}}$						
Selector	d	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
$\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ (diagonal)	$n^{-\min(8,d+4)/(2d+12)}$	$n^{-5/14}$	$n^{-3/8}$	$n^{-7/18}$	$n^{-2/5}$	$n^{-4/11}$	$n^{-1/3}$
$\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$	$n^{-4/(d+12)}$	$n^{-4/13}$	$n^{-2/7}$	$n^{-4/15}$	$n^{-1/4}$	$n^{-4/17}$	$n^{-2/9}$
$\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$	$n^{-2/(d+6)}$	$n^{-2/7}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$	$n^{-2/11}$	$n^{-1/6}$
$\mathbf{H}_{\mathrm{AMISE}} - \mathbf{H}_{\mathrm{MISE}}$	$n^{-2/(d+4)}$	$n^{-2/5}$	$n^{-1/3}$	$n^{-2/7}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$

Table 2.2: Comparison of convergence rates for plug-in selectors

2.4 Estimating the optimal pilot bandwidths

The formulas for the optimal pilot bandwidths contain unknown quantities that depend on the target density f, mostly through the $\psi_{\boldsymbol{r}}$ functionals. To apply these formulas in practice will require us to estimate any unknown quantities. We show that the error introduced from estimation is sufficiently small that it does not affect the rates of convergence established previously using the ideal pilot selectors.

For the full AMSE bandwidth matrix the off-diagonal terms dominate the diagonal terms. These off-diagonal terms can be estimated using pilot bandwidths $g_{\boldsymbol{r},\mathrm{AMSE}}$ in Equation (2.7). These pilot bandwidths are calculated from a bias minimisation procedure since the squared bias is dominant over the variance. We also use bias minimisation for the same reasons to compute $g_{j,\mathrm{SAMSE}}$ in Equation (2.10). So all we have to show in order to establish that the relative rates of convergence remain the same using the estimated pilot bandwidths is to show that the estimated pilot $\hat{g}$ is relatively consistent for the true pilot g. This is true if the relative rate of convergence is $n^{-\alpha}$ for some $\alpha > 0$ i.e.

$$(\hat{g} - g)/g = O_p(n^{-\alpha}).$$

Lemma 6. *Let $\hat{g}$ be an estimate of a pilot bandwidth g, constructed by replacing $\psi_{\boldsymbol{r}}$ with $\hat{\psi}_{\boldsymbol{r}}$. Under the conditions of Lemmas 4 and 5:*

1. *For the full AMSE optimal pilot for $|\boldsymbol{r}| = 4$, the relative rate of convergence of $\hat{g}_{\boldsymbol{r},\mathrm{AMSE}}$ to $g_{\boldsymbol{r},\mathrm{AMSE}}$ is $n^{-4/(d+16)}$.*

2. *For the SAMSE optimal pilot for order 4, the relative rate of convergence of $\hat{g}_{4,\mathrm{SAMSE}}$ to $g_{4,\mathrm{SAMSE}}$ is $n^{-2/(d+8)}$.*

Proof. As $\hat{g} = O_p(g)$ then

$$\hat{g}^p - g^p = (\hat{g} - g)(\hat{g}^{p-1} + \hat{g}^{p-2}g + \cdots + \hat{g}g^{p-2} + g^{p-1}) = (\hat{g} - g)O(g^{p-1})$$

and so

$$\frac{\hat{g} - g}{g} = (\hat{g}^p - g^p)O(g^{-p}). \tag{2.12}$$

For the full AMSE selector, the off-diagonal $g_{\boldsymbol{r},\mathrm{AMSE}}$ (i.e. for odd $\boldsymbol{r}$) dominate the diagonal terms (i.e. even $\boldsymbol{r}$). We have, for the former, with $|\boldsymbol{r}| = 4$, from Equation (2.7)

$$g = O\left(\left(n\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\right)^{-2/(d+12)}\right).$$

Here we have left out the quantities that are not affected by the data i.e. those that do

not involve n or are not required to be estimated. So

$$\begin{aligned}\hat{g}^{(d+12)/2} - g^{(d+12)/2} &= O_p\bigg(\bigg(n\sum_{i=1}^{d}\hat{\psi}_{\boldsymbol{r}+2\boldsymbol{e}_i}(g')\bigg)^{-1} - \bigg(n\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\bigg)^{-1}\bigg)\\ &= O_p\bigg(n^{-1}\bigg(\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\bigg)^{-1}\bigg(\sum_{i=1}^{d}\hat{\psi}_{\boldsymbol{r}+2\boldsymbol{e}_i}(g')\bigg)^{-1}\bigg)\\ &\quad\times O_p\bigg(\sum_{i=1}^{d}\Big(\psi_{\boldsymbol{r}+2\boldsymbol{e}_i} - \hat{\psi}_{\boldsymbol{r}+2\boldsymbol{e}_i}(g')\Big)\bigg)\\ &= O_p(n^{-1}g'^2)\end{aligned}$$

since $\psi_{\boldsymbol{r}} = O(1), \hat{\psi}_{\boldsymbol{r}}(g') = O_p(1)$ and $\mathbb{E}\,\hat{\psi}_{\boldsymbol{r}}(g') - \psi_{\boldsymbol{r}} = O(g'^2)$ from Equation (2.3), $|\boldsymbol{r}| = 6$. Noting that $g = O(n^{-2/(d+12)})$ and $g' = O(n^{-2/(d+16)})$, from Equation (2.7) with $|\boldsymbol{r}| = 6$, we have

$$\frac{\hat{g} - g}{g} = O_p(n^{-1}n^{-4/(d+16)})O(n) = O_p(n^{-4/(d+16)}).$$

For the SAMSE pilot, we start with, from Equation (2.10),

$$g = O\bigg(\bigg(n\sum_{\boldsymbol{r}:|\boldsymbol{r}|=4}\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\bigg)^{-1/(d+6)}\bigg)$$

and so

$$\begin{aligned}\frac{\hat{g} - g}{g} &= O_p(\hat{g}^{d+6} - g^{d+6})O(n)\\ &= O_p\bigg(n^{-1}\bigg(\sum_{\boldsymbol{r}:|\boldsymbol{r}|=4}\sum_{i=1}^{d}\hat{\psi}_{\boldsymbol{r}+2\boldsymbol{e}_i}(g')\bigg)^{-1} - n^{-1}\bigg(\sum_{\boldsymbol{r}:|\boldsymbol{r}|=4}\sum_{i=1}^{d}\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}\bigg)^{-1}\bigg)O(n)\\ &= O_p\bigg(\sum_{\boldsymbol{r}:|\boldsymbol{r}|=4}\sum_{i=1}^{d}\Big(\psi_{\boldsymbol{r}+2\boldsymbol{e}_i}O(n) - \hat{\psi}_{\boldsymbol{r}+2\boldsymbol{e}_i}(g')\Big)\bigg)\\ &= O_p(g'^2)\\ &= O_p(n^{-2/(d+8)})\end{aligned}$$

where $g' = O(n^{-1/(d+8)})$ in this case. □

2.5 Practical performance of plug-in bandwidth selectors

The asymptotic properties of plug-in selectors were examined in the previous section. In this section, we examine their finite sample properties.

2.5.1 Algorithms for plug-in bandwidth selectors

As the finite sample properties of bandwidth selectors do not admit a closed form analysis, we use simulations instead. To perform the simulations, we need to specify the algorithms

for the plug-in selectors i.e. the m-stage AMSE (diagonal and full) bandwidth matrices of Wand & Jones (1994) and the m-stage SAMSE full bandwidth selectors. Before these algorithms are employed, the data are usually pre-transformed. The plug-in bandwidth matrix $\hat{\mathbf{H}}^*_{\mathrm{PI}}$ for the pre-sphered or pre-scaled data can be back transformed to the original scale by $\hat{\mathbf{H}}_{\mathrm{PI}} = \mathbf{S}^{1/2}\hat{\mathbf{H}}^*_{\mathrm{PI}}\mathbf{S}^{1/2}$ or $\hat{\mathbf{H}}_{\mathrm{PI}} = \mathbf{S}^{1/2}_{\mathcal{D}}\hat{\mathbf{H}}^*_{\mathrm{PI}}\mathbf{S}^{1/2}_{\mathcal{D}}$.

Algorithm for m-stage AMSE bandwidth selectors

1. Set $j_{\max} = 2m + 4$. Obtain normal reference estimates $\hat{\psi}^{\mathrm{NR}}_{\boldsymbol{r}}$ for $|\boldsymbol{r}| = j_{\max}$. Plug these estimates into the AMSE pilot bandwidths $g_{\boldsymbol{r},\mathrm{AMSE}}, |\boldsymbol{r}| = j_{\max}-2$.

2. For $j = j_{\max} - 2, j_{\max} - 4, \ldots, 6$:

   (a) Calculate kernel estimates of $\psi_{\boldsymbol{r}}$ functionals of order $j = |\boldsymbol{r}|$ using plug-in estimate of $g_{\boldsymbol{r},\mathrm{SAMSE}}, |\boldsymbol{r}| = j$.

   (b) Substitute $\hat{\psi}_{\boldsymbol{r}}$ estimates into Equations (2.6) and (2.7) to give plug-in estimates of $g_{\boldsymbol{r},\mathrm{SAMSE}}, |\boldsymbol{r}| = j - 2$.

3. Employ $g_{\boldsymbol{r},\mathrm{SAMSE}}, |\boldsymbol{r}| = 4$ to produce kernel estimate $\hat{\boldsymbol{\Psi}}_4$. Plug this estimate into Equation (1.5) to give PI($\mathbf{H}$).

4. To obtain required plug-in bandwidth matrix $\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$:

   (a) If using diagonal bandwidth matrix and $d = 2$ then use

$$h_{1,\mathrm{AMISE}} = \left[\frac{\psi_{04}^{3/4} R(K)}{\mu_2(K)^2 \psi_{40}^{3/4}(\psi_{40}^{1/2}\psi_{04}^{1/2} + \psi_{22})n}\right]^{1/6}$$

$$h_{2,\mathrm{AMISE}} = \left[\frac{\psi_{40}^{3/4} R(K)}{\mu_2(K)^2 \psi_{40}^{3/4}(\psi_{40}^{1/2}\psi_{40}^{1/2} + \psi_{22})n}\right]^{1/6}$$

   (b) Otherwise numerically minimise PI($\mathbf{H}$).

Algorithm for m-stage SAMSE bandwidth selectors

1. Set $j_{\max} = 2m + 4$. Obtain normal reference estimates $\hat{\psi}^{\mathrm{NR}}_{\boldsymbol{r}}$ for $|\boldsymbol{r}| = j_{\max}$. Plug these estimates into the SAMSE pilot bandwidth $g_{j_{\max}-2,\mathrm{SAMSE}}$.

2. For $j = j_{\max} - 2, j_{\max} - 4, \ldots, 6$:

   (a) Calculate kernel estimates of $\psi_{\boldsymbol{r}}$ functionals of order $j = |\boldsymbol{r}|$ using plug-in estimate of $g_{j,\mathrm{SAMSE}}$.

   (b) Substitute $\hat{\psi}_{\boldsymbol{r}}$ estimates into Equation (2.10) to give plug-in estimate of pilot $g_{j-2,\mathrm{SAMSE}}$.

3. Employ $g_{4,\mathrm{SAMSE}}$ to produce kernel estimate $\hat{\mathbf{\Psi}}_4$. Plug this estimate into Equation (1.5) to give PI($\mathbf{H}$).

4. Numerically minimise PI($\mathbf{H}$) to obtain required plug-in bandwidth $\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$.

The code for these bandwidth selectors (and all subsequent selectors developed in this thesis) is written in `R`, R Development Core Team (2003), which is closely related to `Splus`, Mathsoft (1999). In practice we employ a quasi-Newton (variable metric) method of numerical minimisation at stage 4 of these algorithms, using the `optim` function in `R`. In the simulation study we did not encounter any significant computational difficulties using this approach. All the computer code is collected into an `R` library called `ks`. For more details on the `ks` library, see Appendix C.

2.5.2 Simulation results for normal mixture densities

For our simulation study, we now move away from the general multivariate case to the bivariate case, for the reasons stated earlier in Section 1.1 i.e. they are easily visualised on a two dimensional page but have properties that are easily extended to higher dimensions. To compare the performance of the plug-in bandwidth matrix selectors, we conduct a simulation study on 6 mixture normal densities, labelled A to F. All but density F are taken from Wand & Jones (1993). These were chosen as they exhibit a range of characteristics that we wish to detect using a kernel density estimator. The formulas for these densities are given in Table 2.3 and the contour plots are in Figure 2.1. Density A is a normal density with diagonal covariance matrix so it is a base case. Density B is bimodal, though its modes are not as widely separated as density C. The former has spherical components whereas the latter has elliptical components. Densities similar to density C are well-known to pose difficulties for kernel density estimators with fixed bandwidth matrices. Density D has spherical and oblique elliptical components and is also known to be difficult to estimate. Density E is trimodal, kurtotic with heavier tails. Density F is a rotated version of density A. Densities D, E and F all have probability mass oriented at an angle to the axes so they provide a testing ground whether full selectors are able to recover their structure better than diagonal selectors.

The advantage of using normal mixtures as our target densities is that we can compute exact, closed form ISE and MISE. Let f be a mixture normal density with m components, with each component having mean $\boldsymbol{\mu}_k$, variance $\mathbf{\Sigma}_k$ and mixing proportion w_k:

$$f(\boldsymbol{x}) = \sum_{k=1}^{m} w_k \phi_{\mathbf{\Sigma}_k}(\boldsymbol{x} - \boldsymbol{\mu}_k).$$

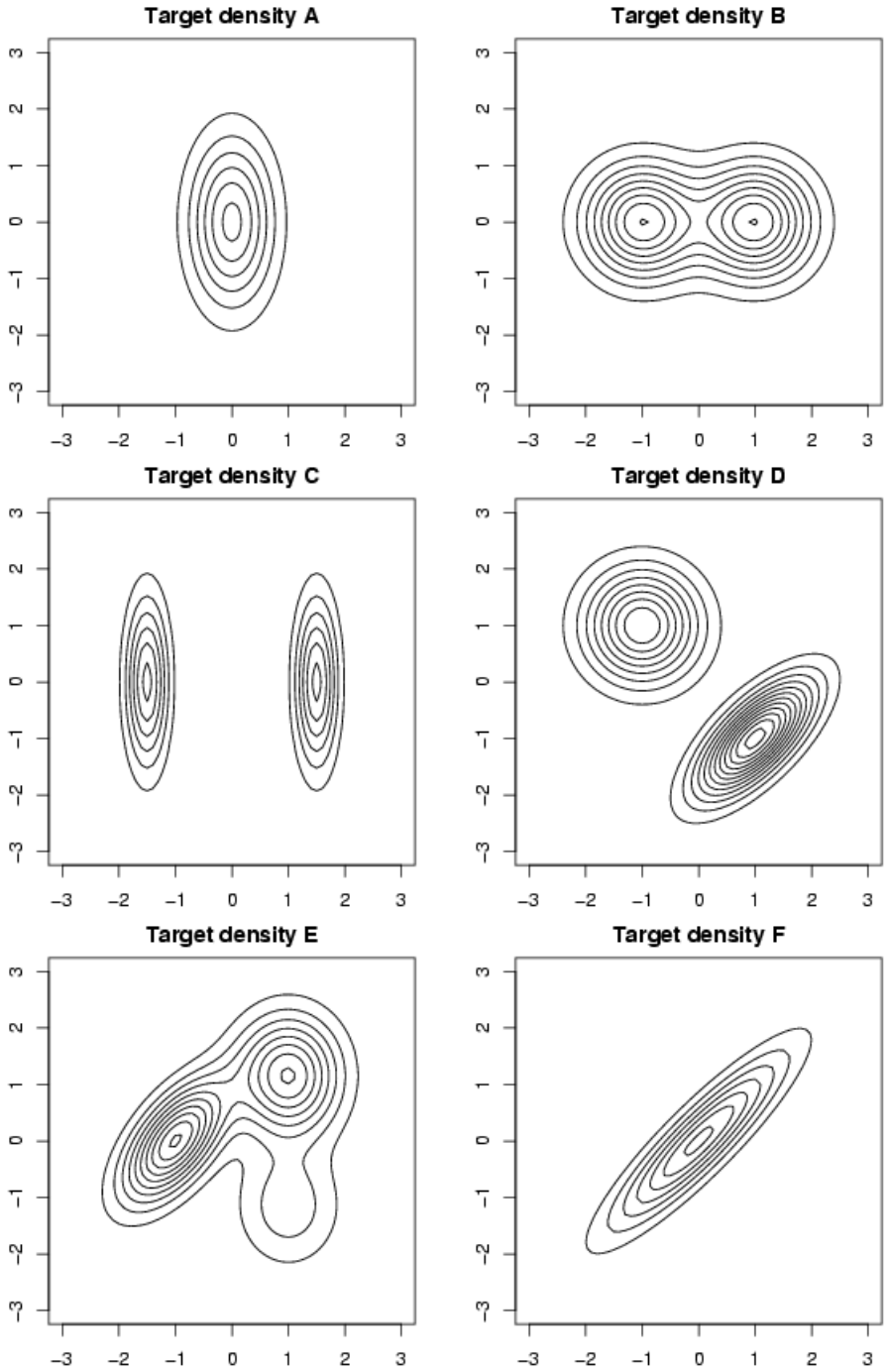

Figure 2.1: Contour plots for target densities A – F

Target density	Formula
A	$N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{1}{4} & 0 \\ 0 & 1 \end{bmatrix} \right)$
B	$\frac{1}{2}N\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & 0 \\ 0 & \frac{4}{9} \end{bmatrix} \right) + \frac{1}{2}N\left(\begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & 0 \\ 0 & \frac{4}{9} \end{bmatrix} \right)$
C	$\frac{1}{2}N\left(\begin{bmatrix} \frac{3}{2} \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{1}{16} & 0 \\ 0 & 1 \end{bmatrix} \right) + \frac{1}{2}N\left(\begin{bmatrix} -\frac{3}{2} \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{1}{16} & 0 \\ 0 & 1 \end{bmatrix} \right)$
D	$\frac{1}{2}N\left(\begin{bmatrix} 1 \\ -1 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & \frac{14}{45} \\ \frac{14}{45} & \frac{4}{9} \end{bmatrix} \right) + \frac{1}{2}N\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & 0 \\ 0 & \frac{4}{9} \end{bmatrix} \right)$
E	$\frac{3}{7}N\left(\begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{9}{25} & \frac{63}{250} \\ \frac{63}{250} & \frac{49}{100} \end{bmatrix} \right) + \frac{3}{7}N\left(\begin{bmatrix} 1 \\ \frac{2}{\sqrt{3}} \end{bmatrix}, \begin{bmatrix} \frac{9}{25} & 0 \\ 0 & \frac{49}{100} \end{bmatrix} \right) + \frac{1}{7}N\left(\begin{bmatrix} 1 \\ -\frac{2}{\sqrt{3}} \end{bmatrix}, \begin{bmatrix} \frac{9}{25} & 0 \\ 0 & \frac{49}{100} \end{bmatrix} \right)$
F	$N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \frac{9}{10} \\ \frac{9}{10} & 1 \end{bmatrix} \right)$

Table 2.3: Formulas for target densities A – F

Then the ISE for a density estimate normal mixture kernels and with bandwidth $\mathbf{H}$ is

$$\begin{aligned} \operatorname{ISE} \hat{f}(\cdot; \mathbf{H}) = n^{-2} \sum_{i=1}^{n} \sum_{i'=1}^{n} \phi_{2\mathbf{H}}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}) - 2n^{-1} \sum_{i=1}^{n} \sum_{k=1}^{m} w_k \phi_{\mathbf{H}+\boldsymbol{\Sigma}_k}(\boldsymbol{X}_i - \boldsymbol{\mu}_k) \\ + \sum_{k=1}^{m} \sum_{k'=1}^{m} w_k w_{k'} \phi_{\boldsymbol{\Sigma}_k+\boldsymbol{\Sigma}_{k'}}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k'}). \end{aligned}$$

Taking expected values, the MISE, as given by Wand & Jones (1995), is

$$\begin{aligned} \operatorname{MISE} \hat{f}(\cdot; \mathbf{H}) = n^{-1}(4\pi)^{-d/2}|\mathbf{H}|^{-1/2} + \sum_{k=1}^{m} \sum_{k'=1}^{m} w_k w_{k'} \big[(1 - n^{-1}) \phi_{2\mathbf{H}+\boldsymbol{\Sigma}_k+\boldsymbol{\Sigma}_{k'}}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k'}) \\ - 2\phi_{\mathbf{H}+\boldsymbol{\Sigma}_k+\boldsymbol{\Sigma}_{k'}}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k'}) + \phi_{\boldsymbol{\Sigma}_k+\boldsymbol{\Sigma}_{k'}}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k'}) \big]. \end{aligned}$$

To assess the efficacy of our bandwidth selectors, we first find the MISE-optimal bandwidth $\mathbf{H}_{\mathrm{MISE}}$ and compare it to $\hat{\mathbf{H}}_{\mathrm{PI}}$ from our simulations. We then compute ISE($\hat{\mathbf{H}}_{\mathrm{PI}}$) and compare it to the MISE($\mathbf{H}_{\mathrm{MISE}}$).

The selectors were run for two sample sizes, $n = 100$ and 1000, each for 400 trials. For each data set we constructed bivariate kernel density estimates using multivariate normal kernels and bandwidth matrix selected using the following methods:

- Wand & Jones (1994) 2-stage plug-in diagonal bandwidth matrix selector, which we label D2
- Wand & Jones (1994) 1-stage and 2-stage plug-in full bandwidth matrix selectors, labelled F1 and F2 respectively;
- Plug-in bandwidth matrix selectors using our 1-stage and 2-stage SAMSE based algorithm, labelled S1 and S2 respectively.

Diagonal selectors start with 'D', AMSE full selectors with 'F' and SAMSE full selectors with 'S'. The number that follows the letter indicates the number of stages. All but the diagonal bandwidth matrix selector were implemented using both pre-scaling and pre-sphering of the data. We add an asterisk superscript to the method label to indicate the latter type of transformation (e.g. F2*).

It is possible for AMSE full ('F') selectors to produce a non-positive definite estimate of $\mathbf{\Psi}_4$. The failure rate (as a percentage), classified by target density and sample size, is in Table 2.4. First, the failure rates of both F1 and F2 selectors are not negligible (for certain target densities) and will have implications for use in practical situations; as there is usually only one set of values available, not obtaining a finite bandwidth matrix poses a problem. Second, the failures occurred for the densities which are not oriented in parallel to the coordinate axes. Third, the failure rates do not appear to decrease with increasing sample size. The F1* and F2* selectors did not encounter such problems. Nonetheless we must keep in mind that we have only considered six normal mixture densities and that it remains theoretically possible for either of these selectors to fail for another density. This seems likely only when the structure of the target density is very intricate. For example, when f is composed of several components with long, thin elliptical contours at a variety of orientations to the coordinate axes.

		Target density					
Selector		A	B	C	D	E	F
F1	$n = 100$	0.00	0.00	0.50	0.50	6.75	0.00
	$n = 1000$	0.00	0.00	2.75	0.00	5.25	0.00
F2	$n = 100$	0.00	0.00	1.75	0.25	4.75	0.00
	$n = 1000$	0.00	0.00	4.75	0.00	3.25	0.00

Table 2.4: Percentage failure rates for F1 and F2 selectors.

For brevity, we present only in this section the box plots of the log(ISE) in Figure 2.2 for $n = 100$ and in Figure 2.3 for $n = 1000$. In Appendix B, refer to Tables B.1 and B.2 for the bandwidth matrix that attains the median ISE and Tables B.3 and B.4 for the means and standard deviations of the ISE.

Looking at the box plots, we see that there is no uniformly best selector - the performance of a selector depends largely on the target density shape. For densities A, B and E, all the selectors have similar performance (although 2 stage selectors have a slight advantage over their 1 stage counterparts for density E). For density C, the performance of the 1-stage selectors is markedly worse than the 2-stage selectors. This target density is clearly not well approximated by a single component normal density, and since the 1-stage pilot selectors depend heavily on the normality assumption, the resulting final bandwidth is inadequate. In contrast, for the 2-stage pilot selectors, the dependence on

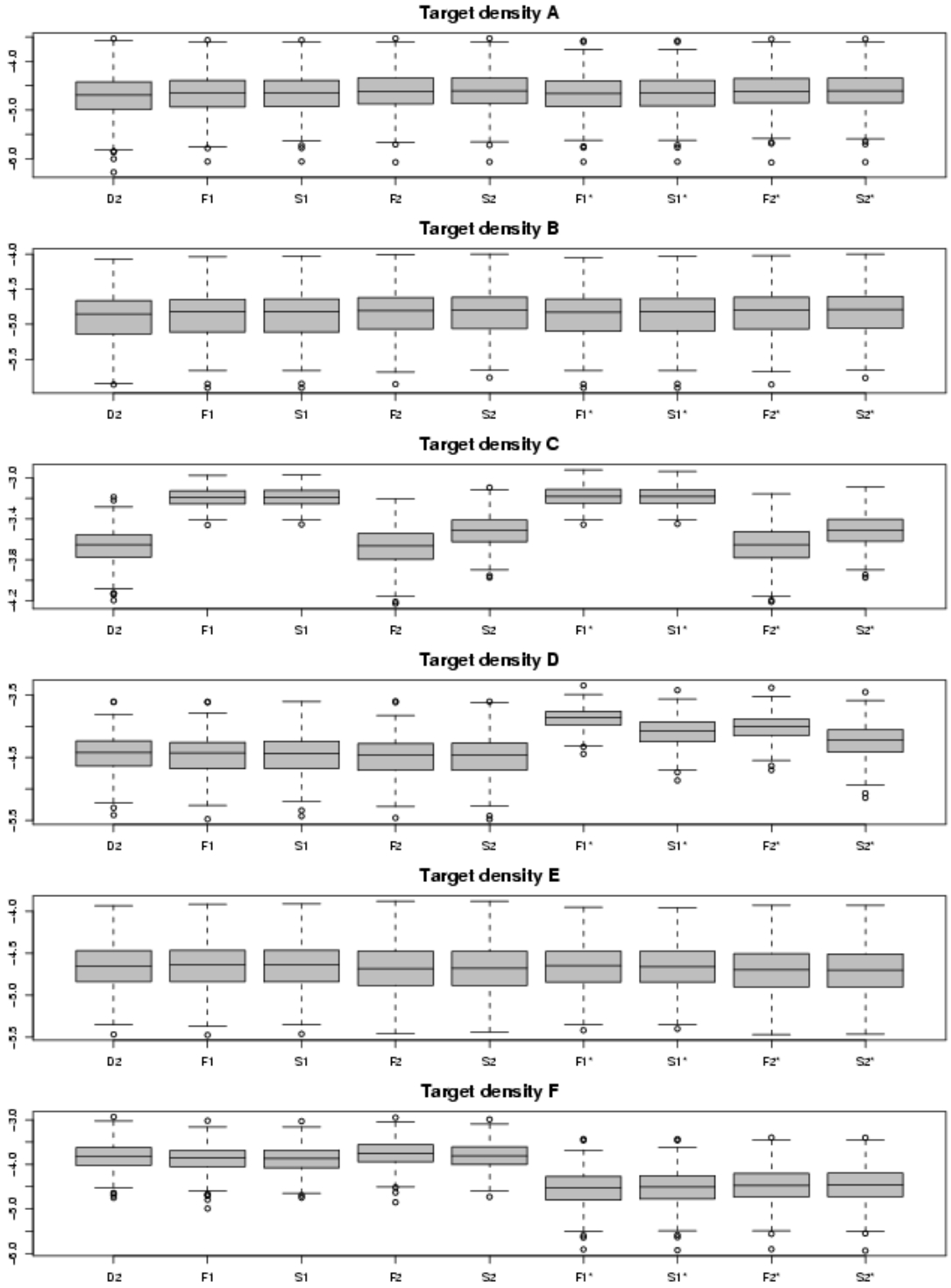

Figure 2.2: Box plots of log(ISE) for plug-in selectors, sample size $n = 100$

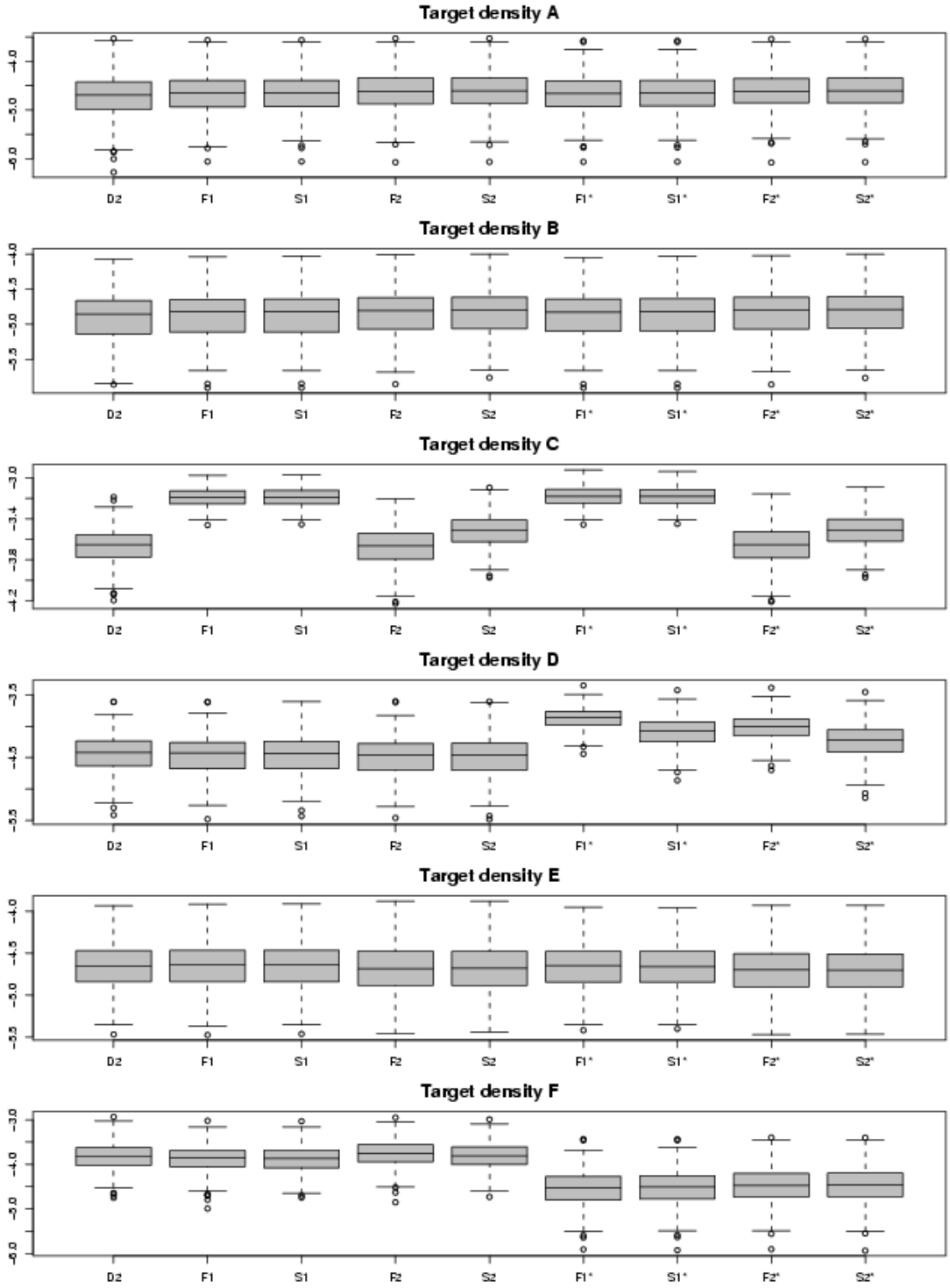

Figure 2.3: Box plots of log(ISE) for plug-in selectors, sample size $n = 1000$

normality is mitigated with an extra stage of pilot functional estimation and the resulting final bandwidth is more appropriate. Pre-sphering is most detrimental for density D, with S2* being the best of these. The reason that pre-sphered selectors perform badly here is that sphering corrupts important structure of the data: the overall correlation is -0.58 while the individual components have correlation zero and 0.7. For density F, the situation is the reverse for density D, the pre-sphered selectors outperform the pre-scaled selectors. This is expected as the density is aligned 45 degrees to the coordinate axes. We note that D2 does poorly with this target density: its performance could be improved by pre-sphering in this case. However, we are reminded by Wand & Jones (1993) that the implementation of a diagonal bandwidth matrix selector with pre-sphering is not generally advisable. This final comment is worth emphasising since it has important considerations in practice. The pre-sphering transformation uses the overall covariance structure of the data which may be different to the local covariance structure of certain regions (e.g. density D). If pre-sphering is combined with a diagonal bandwidth matrix, this can lead to situations where the smoothing in these regions will be in inappropriate directions since diagonal bandwidth matrices are not able induce kernel orientations other than parallel to the coordinate axes.

2.5.3 Results for real data

We analyse the 'Old Faithful' geyser data set from Simonoff (1996) (amongst many others). It consists of pairs of an eruption duration time and the time till the next eruption, both in minutes, of the 'Old Faithful' geyser in Yellowstone National Park, USA. They were collected from 222 eruptions from August 1978 to August 1979. This dataset has structure that is not oriented parallel to the axes so it is a good test case to compare full bandwidth selectors to diagonal selectors. The estimates of the bandwidth selectors are in Table 2.5.

F1*	F1	S1*	S1	
$\begin{bmatrix} 0.0319 & 0.0410 \\ 0.0410 & 6.428 \end{bmatrix}$	$\begin{bmatrix} 0.1086 & 0.9347 \\ 0.9347 & 12.18 \end{bmatrix}$	$\begin{bmatrix} 0.0761 & 0.7192 \\ 0.7192 & 14.022 \end{bmatrix}$	$\begin{bmatrix} 0.0321 & 0.0466 \\ 0.0466 & 6.442 \end{bmatrix}$	-
F2*	**F2**	**S2***	**S2**	**D2**
$\begin{bmatrix} 0.0811 & 0.6395 \\ 0.6395 & 8.186 \end{bmatrix}$	$\begin{bmatrix} 0.0260 & 0.0280 \\ 0.0280 & 5.704 \end{bmatrix}$	$\begin{bmatrix} 0.0565 & 0.5604 \\ 0.5604 & 10.503 \end{bmatrix}$	$\begin{bmatrix} 0.0284 & 0.0277 \\ 0.0277 & 10.503 \end{bmatrix}$	$\begin{bmatrix} 0.0282 & 0 \\ 0 & 5.211 \end{bmatrix}$

Table 2.5: Plug-in bandwidth matrices for 'Old Faithful' geyser data

The contour plots of the kernel density estimates for the 1-stage and 2-stage selectors are in Figures 2.4 and 2.5 respectively. We can see that using the pre-sphering with full selectors produce kernel density estimates that are similar to each other; whereas using pre-scaling with full or diagonal selectors produce kernel density estimates that are similar to each other. The latter group of methods provide density estimates in which the lower

left mode runs almost parallel to the waiting time axis. For the pre-sphered methods the orientation of this mode is at a marked angle to this axis. We also note that the elements of the bandwidth matrices are larger for the pre-sphered methods than the pre-scaled ones, producing smoother estimates.

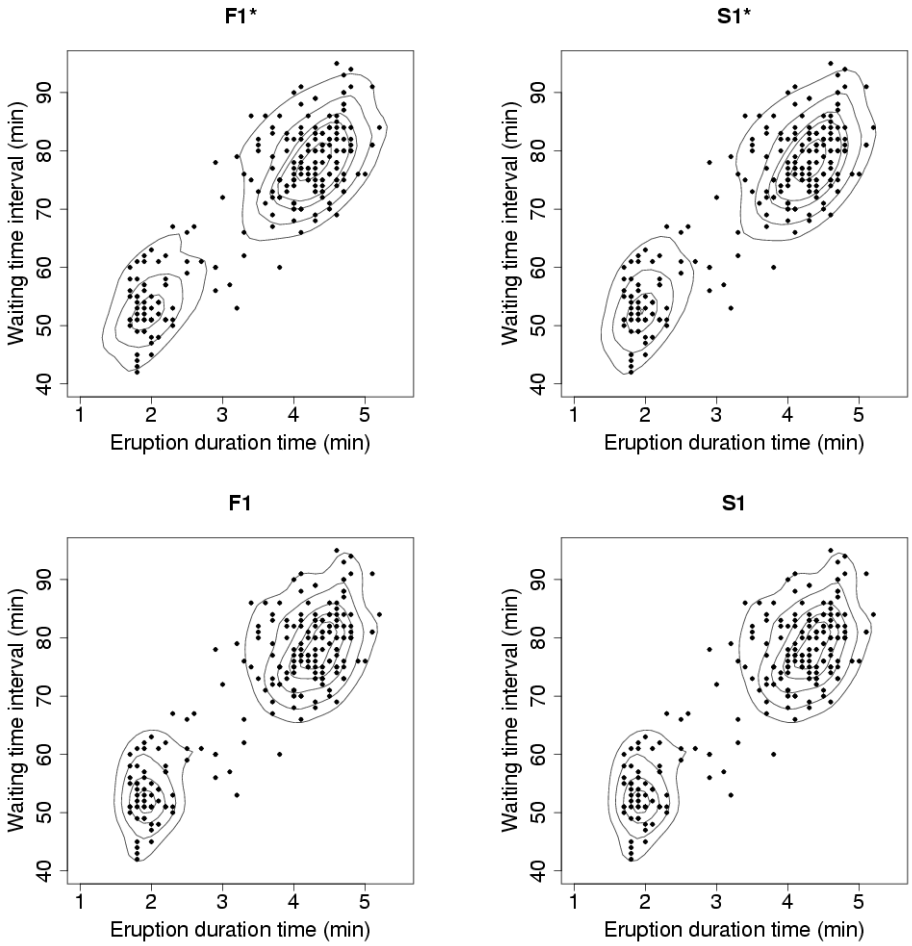

Figure 2.4: 'Old Faithful' geyser data contour plots - 1-stage plug-in selectors

Another data set that we analyse is taken from UNICEF (2003) (United Nations Children's Fund). It contains measurements of the under 5 (years of age) child mortality rate, i.e. the number of children under 5 dying per 1000 live births, and the expected life expectancy at birth (in years) for 73 countries. These countries have GNI (Gross National Income) of less than $US 1000 per person per year. From the analysis of the 'Old Faithful' geyser data, we recommend (at least) 2 stages of pilot estimation so we only produce estimates from these selectors in Table 2.6.

This dataset has probability mass oriented to the axes, though it is at a different angle to the 'Old Faithful' geyser data. We again expect that the full bandwidth selectors will be able to detect this obliqueness whereas the diagonal selector will not. This is verified

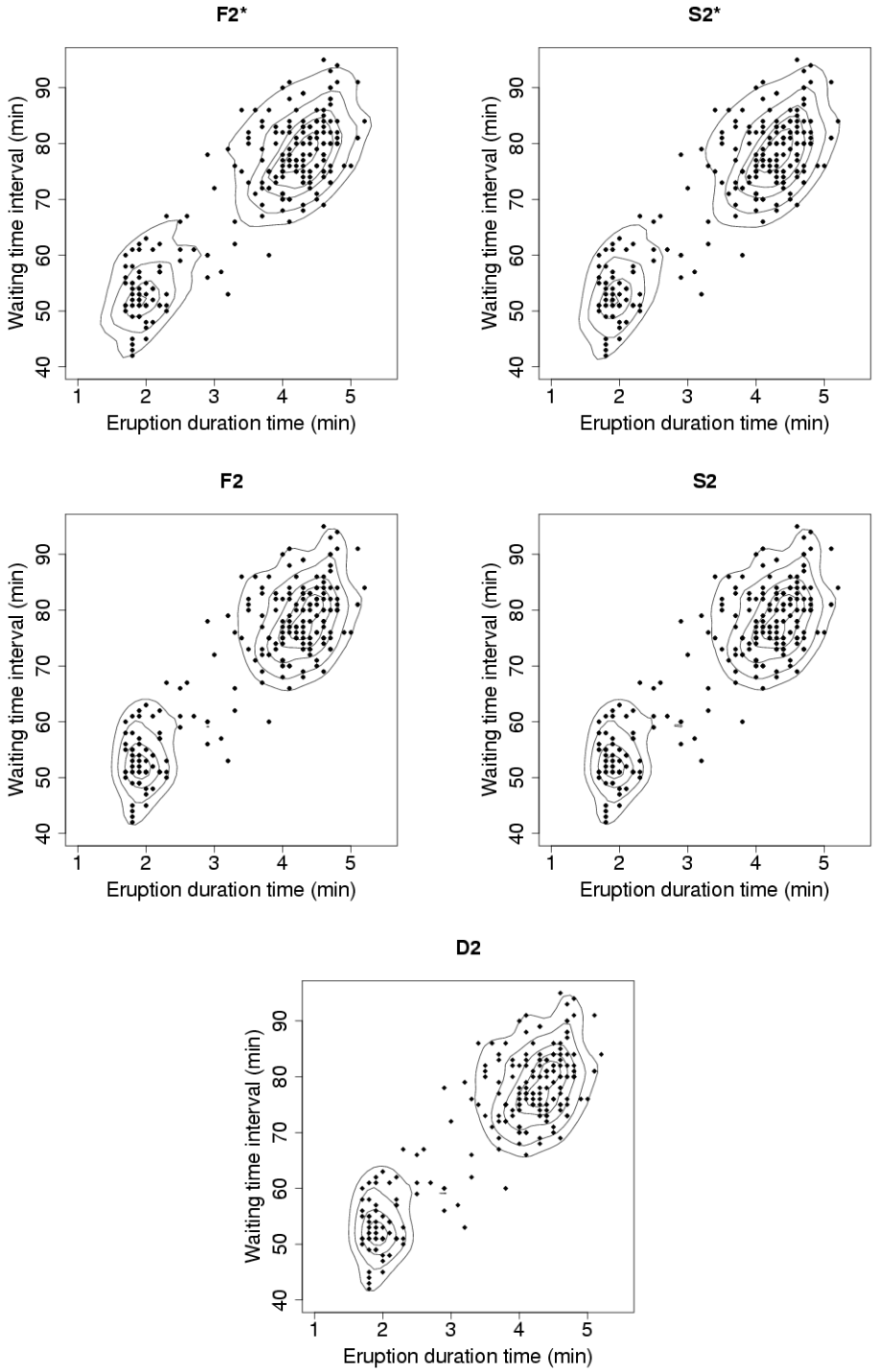

Figure 2.5: 'Old Faithful' geyser data contour plots - 2-stage plug-in selectors

$F2^*$	F2	$S2^*$	S2	D2
$\begin{bmatrix} 805.8 & -99.40 \\ -99.40 & 17.33 \end{bmatrix}$	$\begin{bmatrix} 237.7 & -15.34 \\ -15.3 & 7.232 \end{bmatrix}$	$\begin{bmatrix} 797.6 & -106.6 \\ -106.6 & 19.57 \end{bmatrix}$	$\begin{bmatrix} 245.8 & -11.07 \\ -11.07 & 6.674 \end{bmatrix}$	$\begin{bmatrix} 201.0 & 0 \\ 0 & 6.243 \end{bmatrix}$

Table 2.6: Plug-in bandwidth matrices for child mortality-life expectancy data

by the contour plots of the corresponding kernel density estimates in Figure 2.6. The plots for D2, F2, S2 appear to have several spurious features, with D2 being the most noisy whereas the plots for $F2^*$ and $S2^*$ are smoother.

2.6 Conclusion

Using a diagonal bandwidth matrix restricts us to using kernels that are aligned to the co-ordinate axes. In situations where the data are not oriented parallel to the co-ordinate axes using a full bandwidth matrix is more appropriate. We modified the existing pilot bandwidth selection stages for plug-in selectors, from an element-wise (AMSE pilots) procedure to a matrix-wise (SAMSE pilots) procedure. The SAMSE procedure is guaranteed to produce a finite pilot bandwidth and is more parsimonious. We derived the asymptotic properties for these plug-in selectors as well as looking at their finite sample behaviour. It appears 2 stages of pilot estimation along with pre-sphering ($S2^*$ and $F2^*$) are the best overall strategies (though we recall that the $S2^*$ is simpler to implement.) Moreover, they are both better than D2 which is currently the most widely used plug-in selector.

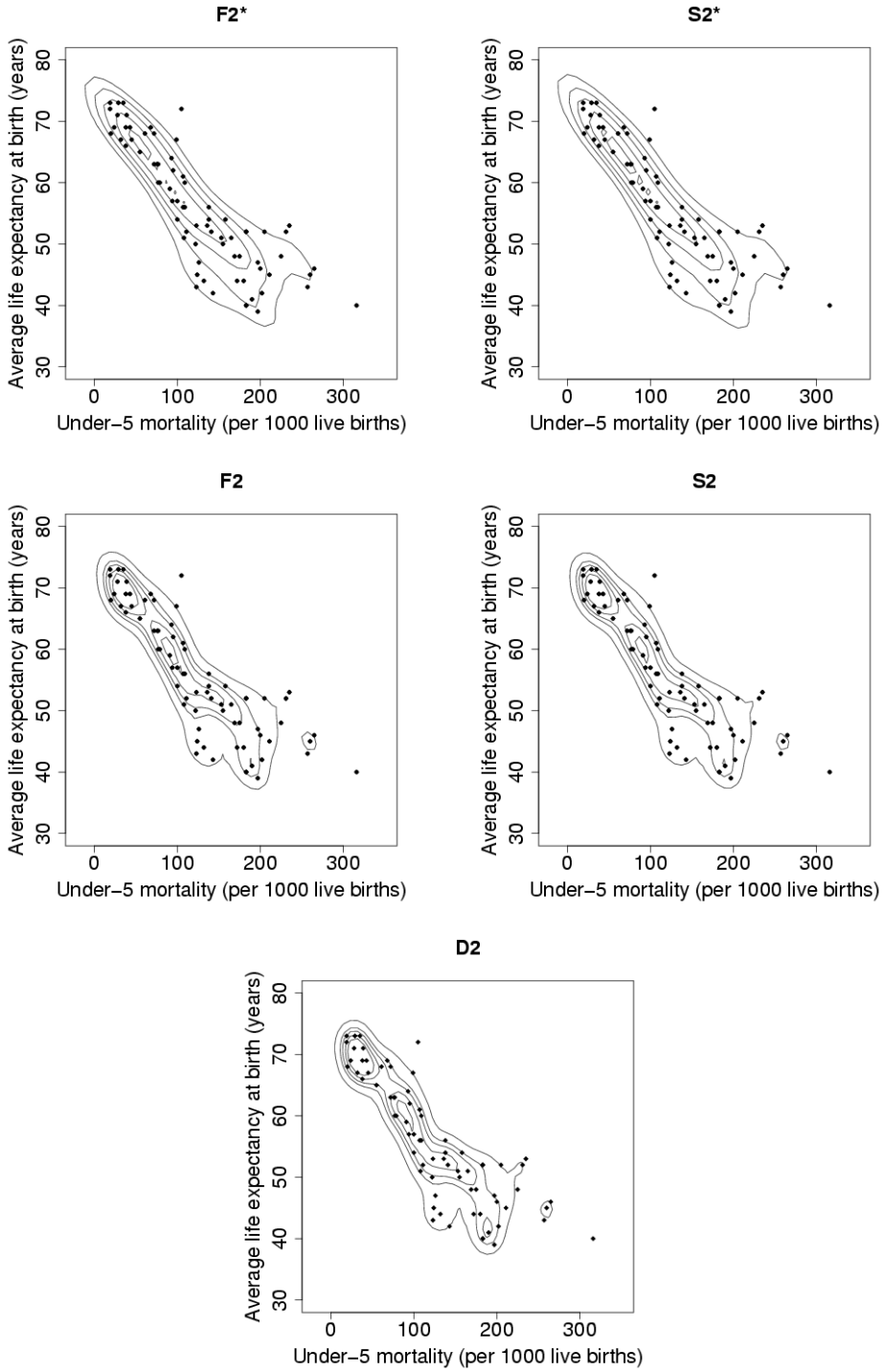

Figure 2.6: Child mortality-life expectancy data contour plots - 2-stage plug-in selectors

Chapter 3

Cross validation bandwidth selectors

3.1 Introduction

Cross validation selectors are the main alternative to plug-in selectors. Cross validation selectors are widely used in univariate kernel density estimation and, in a restricted way, in multivariate kernel density estimation. For the univariate case, like their plug-in counterparts, we have already a solid understanding of the performance of the cross validation selectors. There are three main types of cross-validation: least squares, biased and smoothed. Biased cross validation is dependent on the AMISE so its performance depends on the AMISE being appropriate approximation for the MISE. Least squares cross validation is not subject to this condition, though it has been shown to be more variable than other selectors in the univariate setting. These two cross validation methods are slower in terms of convergence rates than plug-in selectors. Smoothed cross validation, on the other hand, has convergence rate and variability that are comparable to plug-in selectors. It achieves this by using an exact estimate of the bias rather than relying on its asymptotic approximation.

In Chapter 2, we extended the existing diagonal plug-in selectors to full selectors. We attempt a similar extension for the cross validation selectors in this chapter. To generalise least squares and biased cross validation is fairly straightforward, as is shown in Sections 3.2 and 3.3. They are straightforward primarily because they do not require independent pilot bandwidths. However smoothed cross validation selectors *do* require independent pilot bandwidths: to generalise the selection of these pilot bandwidths is not trivial and is the main theoretical result of this chapter. See Section 3.4. Asymptotic relative convergence rates are computed, within each section, using the mathematical machinery developed in the previous chapter. The analysis of a simulation study and real data sets is used to compare finite sample properties in Section 3.5.

3.2 Least squares cross validation

The multivariate version of the least squares cross validation (LSCV) criterion is a straightforward generalisation of the univariate form devised by Rudemo (1982) and Bowman (1984):

$$\mathrm{LSCV}(\mathbf{H}) = \int_{\mathbb{R}^d} \hat{f}(\boldsymbol{x};\mathbf{H})^2 \, d\boldsymbol{x} - 2n^{-1}\sum_{i=1}^{n} \hat{f}_{-i}(\boldsymbol{X}_i;\mathbf{H})$$

where the leave-one-out estimator is

$$\hat{f}_{-i}(\boldsymbol{x};\mathbf{H}) = (n-1)^{-1}\sum_{\substack{j=1\\ j\neq i}}^{n} K_{\mathbf{H}}(\boldsymbol{x}-\boldsymbol{X}_j).$$

The LSCV selector $\hat{\mathbf{H}}_{\mathrm{LSCV}}$ is the minimiser of $\mathrm{LSCV}(\mathbf{H})$. This criterion attempts to estimate the MISE in a fairly directly manner since $\mathbb{E}\,\mathrm{LSCV}(\mathbf{H}) = \mathrm{MISE}\ \hat{f}(\cdot;\mathbf{H}) - R(f)$. Due to its unbiasedness, the LSCV selector is sometimes called the unbiased cross validation (UCV) selector. The LSCV can be expanded to give:

$$\begin{aligned}
&\mathrm{LSCV}(\mathbf{H})\\
&= n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(K_{\mathbf{H}} * K_{\mathbf{H}})(\boldsymbol{X}_i-\boldsymbol{X}_j) - 2n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n} K_{\mathbf{H}}(\boldsymbol{X}_i-\boldsymbol{X}_j)\\
&= n^{-1}R(K)|\mathbf{H}|^{-1/2} + n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n}(K_{\mathbf{H}} * K_{\mathbf{H}} - 2K_{\mathbf{H}})(\boldsymbol{X}_i-\boldsymbol{X}_j). \qquad (3.1)
\end{aligned}$$

(From this expression, we will see later that this is a special case of the smoothed cross validation criterion in Section 3.4.) For normal kernels, this expression simplifies further since $\phi_{\mathbf{H}} * \phi_{\mathbf{H}} = \phi_{2\mathbf{H}}$:

$$\mathrm{LSCV}(\mathbf{H}) = n^{-1}(4\pi)^{-d/2}|\mathbf{H}|^{-1/2} + n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n}(\phi_{2\mathbf{H}} - 2\phi_{\mathbf{H}})(\boldsymbol{X}_i-\boldsymbol{X}_j). \quad (3.2)$$

Some research has been carried out by Sain et al. (1994) on multivariate LSCV selectors. However they use only product kernels which is equivalent to using a diagonal bandwidth matrix with spherically symmetric kernels. These authors computed the relative rates of convergence for the diagonal selector which we now replicate for the full selector.

We follow our strategy in Section 2.3 to find the relative convergence rate for $\hat{\mathbf{H}}_{\mathrm{LSCV}}$ to $\mathbf{H}_{\mathrm{AMISE}}$. To find $\mathrm{ABias}(\mathrm{vech}\,\hat{\mathbf{H}}_{\mathrm{LSCV}})$ and $\mathrm{AVar}(\mathrm{vech}\,\hat{\mathbf{H}}_{\mathrm{LSCV}})$ we need the expected value and variance of $D_{\mathbf{H}}(\mathrm{LSCV}-\mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})$, calculated in Lemmas 7 and 8.

Lemma 7. *Assume A1 – A2 of the AMSE Lemma (Lemma 3), and that K is normal. Then*

$$\operatorname{ABias}(\operatorname{vech} \hat{\mathbf{H}}_{\mathrm{LSCV}}) = O(\mathbf{J}_{d'} n^{-2/(d+4)}) \operatorname{vech} \mathbf{H}_{\mathrm{AMISE}}$$

Proof. A higher order expansion of the MISE is

$$\begin{aligned} \operatorname{MISE} \hat{f}(\cdot; \mathbf{H}) = \operatorname{AMISE} \hat{f}(\cdot; \mathbf{H}) + \tfrac{1}{8} \int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H} D^2 f(\boldsymbol{x})) \operatorname{tr}(\mathbf{H}^2 (D^2)^2 f(\boldsymbol{x}))\, d\boldsymbol{x} \\ + o(\|\operatorname{vech} \mathbf{H}\|^3) \end{aligned} \tag{3.3}$$

where D^2 is the Hessian operator with respect to the free variable $\boldsymbol{x}$, so $(D^2)^2$ is obtained by 'multiplying' the Hessian operator with itself. This means that $(D^2)^2$ is matrix of fourth order partial differential operators.

As $\mathbb{E} \operatorname{LSCV}(\mathbf{H}) = \operatorname{MISE} \hat{f}(\cdot; \mathbf{H}) - R(f)$, and swapping the order of expectation and differentiation, yields

$$\begin{aligned} &\mathbb{E}[D_{\mathbf{H}}(\mathrm{LSCV} - \mathrm{AMISE})(\mathbf{H})] \\ &\quad = D_{\mathbf{H}}[\mathbb{E}(\mathrm{LSCV} - \mathrm{AMISE})(\mathbf{H})] \\ &\quad = D_{\mathbf{H}} \left[- R(f) - \tfrac{1}{8} \int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H} D^2 f(\boldsymbol{x})) \operatorname{tr}(\mathbf{H}^2 (D^2)^2 f(\boldsymbol{x}))\, d\boldsymbol{x} + o(\|\operatorname{vech} \mathbf{H}\|^3) \right] \\ &\quad = -\tfrac{1}{8} \int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^2 (D^2)^2 f(\boldsymbol{x})) \mathbf{D}_d^T \operatorname{vec} D^2 f(\boldsymbol{x})\, d\boldsymbol{x} \\ &\qquad - \tfrac{1}{4} \int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H} D^2 f(\boldsymbol{x})) \mathbf{D}_d^T \operatorname{vec}(\mathbf{H} (D^2)^2 f(\boldsymbol{x}))\, d\boldsymbol{x} + o(\|\operatorname{vech} \mathbf{H}\| \operatorname{vech} \mathbf{H}) \end{aligned}$$

as $D_{\mathbf{H}} \operatorname{tr}(\mathbf{A}\mathbf{H}) = \mathbf{D}_d^T \operatorname{vec} \mathbf{A}$ and $D_{\mathbf{H}} \operatorname{tr}(\mathbf{A}\mathbf{H}^2) = \mathbf{D}_d^T \operatorname{vec}(\mathbf{H}\mathbf{A})$ for a matrix $\mathbf{A}$ of appropriate dimensions. So $\operatorname{ABias}(\operatorname{vech} \hat{\mathbf{H}}_{\mathrm{LSCV}})$ is $O(\mathbf{J}_{d'} n^{-2/(d+4)}) \operatorname{vech} \mathbf{H}_{\mathrm{AMISE}}$. □

Lemma 8. *Assume A1 – A2 of AMSE Lemma (Lemma 3), and that K is normal. Then*

$$\operatorname{AVar}(\operatorname{vech} \hat{\mathbf{H}}_{\mathrm{LSCV}}) = O(\mathbf{J}_{d'} n^{-d/(d+4)}) (\operatorname{vech} \mathbf{H}_{\mathrm{AMISE}}) (\operatorname{vech}^T \mathbf{H}_{\mathrm{AMISE}}).$$

Proof. For the asymptotic variance, we start with

$$\begin{aligned} &\operatorname{Var}[D_{\mathbf{H}}(\mathrm{LSCV} - \mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})] \\ &\quad = \operatorname{Var}[D_{\mathbf{H}} \mathrm{LSCV}(\mathbf{H}_{\mathrm{AMISE}})] \\ &\quad = \operatorname{Var} \left[n^{-1}(n-1)^{-1} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} D_{\mathbf{H}} (\phi_{2\mathbf{H}} - 2\phi_{\mathbf{H}})(\boldsymbol{X}_i - \boldsymbol{X}_j) \right] \\ &\quad = \operatorname{Var} \left[n^{-2} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} (\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_i - \boldsymbol{X}_j) \right] [1 + o(n^{-1})] \end{aligned}$$

where

$$\boldsymbol{\varphi}_{\mathbf{A}}(\boldsymbol{X}) = \phi_{\mathbf{A}}(\boldsymbol{X})\mathbf{D}_d^T \operatorname{vec}(\mathbf{A}^{-1}\boldsymbol{X}\boldsymbol{X}^T\mathbf{A} - \mathbf{A}^{-1}). \tag{3.4}$$

We use this $\boldsymbol{\varphi}$ function because it is related to the derivative of the normal density via $D_{\mathbf{H}}\phi_{a\mathbf{H}}(\boldsymbol{x}) = \frac{1}{2}a\boldsymbol{\varphi}_{a\mathbf{H}}(\boldsymbol{x})$. As $\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}}$ is a symmetric function, the variance simplifies to

$$\begin{aligned}\operatorname{Var}[D_{\mathbf{H}}(\mathrm{LSCV} - \mathrm{AMISE})(\mathbf{H})] &= 2n^{-2}\operatorname{Var}[(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)] \\ &\quad + 4n^{-1}\operatorname{Cov}[(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2), (\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)].\end{aligned}$$

The first term of $\operatorname{Var}[D_{\mathbf{H}}(\mathrm{LSCV} - \mathrm{AMISE})(\mathbf{H})]$ comprises

$$\begin{aligned}\operatorname{Var}[(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)] &= \mathbb{E}\left\{[(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\,[(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]^T\right\} \\ &\quad - [\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\,[\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]^T.\end{aligned}$$

We have that $\mathbb{E}\{(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)[(\boldsymbol{\varphi}_{\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]^T\}$ contains expressions of the type

$$\begin{aligned}&\mathbb{E}\{\phi_{a\mathbf{H}}(\boldsymbol{X}_1 - \boldsymbol{X}_2)\mathbf{D}_d^T \operatorname{vec}[(a\mathbf{H})^{-1}(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T(a\mathbf{H})^{-1} - (a\mathbf{H})^{-1}] \\ &\quad \times \phi_{b\mathbf{H}}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \operatorname{vec}^T[(b\mathbf{H})^{-1}(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T(b\mathbf{H})^{-1} - (b\mathbf{H})^{-1}]\mathbf{D}_d\}.\end{aligned} \tag{3.5}$$

To simplify this expression, we note that $\phi_{a\mathbf{H}}(\boldsymbol{x})\phi_{b\mathbf{H}}(\boldsymbol{x}) = (2\pi)^{-d/2}|(a+b)\mathbf{H}|^{-1/2}\phi_{a'\mathbf{H}}(\boldsymbol{x})$ where $a' = ab/(a+b)$:

$$\begin{aligned}&\mathbb{E}\{\phi_{a\mathbf{H}}(\boldsymbol{X}_1 - \boldsymbol{X}_2)\mathbf{D}_d^T \operatorname{vec}[(a\mathbf{H})^{-1}(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T(a\mathbf{H})^{-1} - (a\mathbf{H})^{-1}] \\ &\quad \times \phi_{b\mathbf{H}}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \operatorname{vec}^T[(b\mathbf{H})^{-1}(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T(b\mathbf{H})^{-1} - (b\mathbf{H})^{-1}]\mathbf{D}_d\} \\ &= O(\mathbf{J}_{d'}|\mathbf{H}|^{-1/2}) \int_{\mathbb{R}^{2d}} \phi_{a'\mathbf{H}}(\boldsymbol{x} - \boldsymbol{y})\mathbf{D}_d^T \operatorname{vec}[(a\mathbf{H})^{-1}(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T(a\mathbf{H})^{-1} - (a\mathbf{H})^{-1}] \\ &\quad \times \operatorname{vec}^T[(b\mathbf{H})^{-1}(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T(b\mathbf{H})^{-1} - (b\mathbf{H})^{-1}]\mathbf{D}_d f(\boldsymbol{x})f(\boldsymbol{y})\, d\boldsymbol{x}d\boldsymbol{y} \\ &= O(\mathbf{J}_{d'}|\mathbf{H}|^{-1/2}) \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}(\boldsymbol{w})\mathbf{D}_d^T \operatorname{vec}[a^{-2}a'\mathbf{H}^{-1/2}\boldsymbol{w}\boldsymbol{w}^T\mathbf{H}^{-1/2} - a^{-1}\mathbf{H}^{-1}] \\ &\quad \times \operatorname{vec}^T[b^{-2}a'\mathbf{H}^{-1/2}\boldsymbol{w}\boldsymbol{w}^T\mathbf{H}^{-1/2} - b^{-1}\mathbf{H}^{-1}]\mathbf{D}_d f(\boldsymbol{y} + (a'\mathbf{H})^{1/2}\boldsymbol{w})f(\boldsymbol{y})\, d\boldsymbol{w}d\boldsymbol{y} \\ &= O(\mathbf{J}_{d'}|\mathbf{H}|^{-1/2}) \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}(\boldsymbol{w})\mathbf{D}_d^T \operatorname{vec}[a^{-2}a'\mathbf{H}^{-1/2}\boldsymbol{w}\boldsymbol{w}^T\mathbf{H}^{-1/2} - a^{-1}\mathbf{H}^{-1}] \\ &\quad \times \operatorname{vec}^T[b^{-2}a'\mathbf{H}^{-1/2}\boldsymbol{w}\boldsymbol{w}^T\mathbf{H}^{-1/2} - b^{-1}\mathbf{H}^{-1}]\mathbf{D}_d[f(\boldsymbol{y}) + o(1)]f(\boldsymbol{y})\, d\boldsymbol{w}d\boldsymbol{y} \\ &= O(\mathbf{J}_{d'}|\mathbf{H}|^{-1/2})(\operatorname{vech}\mathbf{H}^{-1})(\operatorname{vech}^T\mathbf{H}^{-1}).\end{aligned}$$

To completely determine an order expression for $\operatorname{Var}[(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]$, we find that

$$\begin{aligned}\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2) &= D_{\mathbf{H}}[\mathbb{E}(\phi_{2\mathbf{H}} - 2\phi_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)] \\ &= D_{\mathbf{H}}\left[\tfrac{1}{4}\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^2(D^2)^2 f(\boldsymbol{y}))f(\boldsymbol{y})\, d\boldsymbol{y} + o(\|\operatorname{vech}\mathbf{H}\|^2)\right] \\ &= \tfrac{1}{2}\int_{\mathbb{R}^d} \mathbf{D}_d^T \operatorname{vec}(\mathbf{H}(D^2)^2 f(\boldsymbol{y}))f(\boldsymbol{y})\, d\boldsymbol{y} + o(\operatorname{vech}\mathbf{H})\end{aligned}$$

since

$$\begin{aligned}\mathbb{E}\,\phi_{a\mathbf{H}}(\boldsymbol{X}_1-\boldsymbol{X}_2) &= \int_{\mathbb{R}^{2d}} \phi_{a\mathbf{H}}(\boldsymbol{x}-\boldsymbol{y})f(\boldsymbol{x})f(\boldsymbol{y})\,d\boldsymbol{x}d\boldsymbol{y}\\ &= \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}(\boldsymbol{w})f(\boldsymbol{y}+(a\mathbf{H})^{1/2}\boldsymbol{w})f(\boldsymbol{y})\,d\boldsymbol{w}d\boldsymbol{y}\\ &= \int_{\mathbb{R}^{d}} \left[f(\boldsymbol{y})+\tfrac{1}{2}a\,\mathrm{tr}(\mathbf{H}D^2f(\boldsymbol{y}))+\tfrac{1}{8}a^2\,\mathrm{tr}(\mathbf{A}^2(D^2)^2f(\boldsymbol{y}))\right]f(\boldsymbol{y})\,d\boldsymbol{y}\\ &\quad + o(\|\mathrm{vech}\,\mathbf{H}\|^2).\end{aligned}$$

so that

$$[\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1-\boldsymbol{X}_2)][\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1-\boldsymbol{X}_2)]^T = O(\mathbf{J}_{d'})(\mathrm{vech}\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H})$$

and thus

$$\mathrm{Var}[(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1-\boldsymbol{X}_2)] = O(\mathbf{J}_{d'}|\mathbf{H}|^{-1/2})(\mathrm{vech}\,\mathbf{H}^{-1})(\mathrm{vech}^T\,\mathbf{H}^{-1}). \tag{3.6}$$

The second term of $\mathrm{Var}[D_{\mathbf{H}}(\mathrm{LSCV}-\mathrm{AMISE})(\mathbf{H})]$ comprises

$$\begin{aligned}&\mathrm{Cov}[(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1-\boldsymbol{X}_2),(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_2-\boldsymbol{X}_3)]\\ &\quad= \mathbb{E}\left\{[(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1-\boldsymbol{X}_2)]\,[(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_2-\boldsymbol{X}_3)]^T\right\}\\ &\qquad- [\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1-\boldsymbol{X}_2)]\,[\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_2-\boldsymbol{X}_3)]^T.\end{aligned}$$

We have already derived an order expression for the latter term in this covariance. The former term $\mathbb{E}\{(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1-\boldsymbol{X}_2)[(\boldsymbol{\varphi}_{2\mathbf{H}}-\boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_2-\boldsymbol{X}_3)]^T\}$ contains expressions of the type

$$\begin{aligned}&\mathbb{E}\{\phi_{a\mathbf{H}}(\boldsymbol{X}_1-\boldsymbol{X}_2)\mathbf{D}_d^T\,\mathrm{vec}[(a\mathbf{H})^{-1}(\boldsymbol{X}_1-\boldsymbol{X}_2)(\boldsymbol{X}_1-\boldsymbol{X}_2)^T(a\mathbf{H})^{-1}-(a\mathbf{H})^{-1}]\\ &\quad\times\phi_{b\mathbf{H}}(\boldsymbol{X}_2-\boldsymbol{X}_3)\,\mathrm{vec}^T[(b\mathbf{H})^{-1}(\boldsymbol{X}_2-\boldsymbol{X}_3)(\boldsymbol{X}_2-\boldsymbol{X}_3)^T(b\mathbf{H})^{-1}-(b\mathbf{H})^{-1}]\mathbf{D}_d\}.\end{aligned}$$

We can simplify this expression:

$$\begin{aligned}&\int_{\mathbb{R}^{3d}} \phi_{a\mathbf{H}}(\boldsymbol{x}-\boldsymbol{y})\mathbf{D}_d^T\,\mathrm{vec}[(a\mathbf{H})^{-1}(\boldsymbol{x}-\boldsymbol{y})(\boldsymbol{x}-\boldsymbol{y})^T(a\mathbf{H})^{-1}-(a\mathbf{H})^{-1}]\\ &\times\phi_{b\mathbf{H}}(\boldsymbol{y}-\boldsymbol{z})\,\mathrm{vec}^T[(b\mathbf{H})^{-1}(\boldsymbol{y}-\boldsymbol{z})(\boldsymbol{y}-\boldsymbol{z})^T(b\mathbf{H})^{-1}-(b\mathbf{H})^{-1}]\mathbf{D}_d f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})\,d\boldsymbol{x}d\boldsymbol{y}d\boldsymbol{z}\\ &= \int_{\mathbb{R}^{3d}} \phi_{\mathbf{I}}(\boldsymbol{v})\mathbf{D}_d^T\,\mathrm{vec}[(a\mathbf{H})^{-1/2}\boldsymbol{v}\boldsymbol{v}^T(a\mathbf{H})^{-1/2}-(a\mathbf{H})^{-1}]\\ &\quad\times\phi_{\mathbf{I}}(\boldsymbol{w})\,\mathrm{vec}^T[(b\mathbf{H})^{-1/2}\boldsymbol{w}\boldsymbol{w}^T(b\mathbf{H})^{-1/2}-(b\mathbf{H})^{-1}]\\ &\quad\times f(\boldsymbol{y}+(a\mathbf{H})^{1/2}\boldsymbol{v})f(\boldsymbol{y})f(\boldsymbol{y}-(b\mathbf{H})^{1/2}\boldsymbol{w})\,d\boldsymbol{v}d\boldsymbol{w}d\boldsymbol{y}\\ &= \int_{\mathbb{R}^{3d}} \phi_{\mathbf{I}}(\boldsymbol{v})\mathbf{D}_d^T\,\mathrm{vec}[(a\mathbf{H})^{-1/2}\boldsymbol{v}\boldsymbol{v}^T(a\mathbf{H})^{-1/2}-(a\mathbf{H})^{-1}]\\ &\quad\times\phi_{\mathbf{I}}(\boldsymbol{w})\,\mathrm{vec}^T[(b\mathbf{H})^{-1/2}\boldsymbol{w}\boldsymbol{w}^T(b\mathbf{H})^{-1/2}-(b\mathbf{H})^{-1}]\\ &\quad\times[f(\boldsymbol{y})+O(\|\mathrm{vech}\,\mathbf{H}\|)]f(\boldsymbol{y})[f(\boldsymbol{y})+O(\|\mathrm{vech}\,\mathbf{H}\|)]\,d\boldsymbol{v}d\boldsymbol{w}d\boldsymbol{y}\\ &= O(\mathbf{J}_{d'})(\mathrm{vech}\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H})\end{aligned}$$

which means that

$$\mathrm{Cov}[(\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_1 - \boldsymbol{X}_2), (\boldsymbol{\varphi}_{2\mathbf{H}} - \boldsymbol{\varphi}_{\mathbf{H}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)] = O(\mathbf{J}_{d'})(\mathrm{vech}\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H}).$$

Combining the expression for this covariance with Equation (3.6) yields,

$$\begin{aligned}
&\mathrm{Var}[D_{\mathbf{H}}(\mathrm{LSCV} - \mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})] \\
&\quad = O(\mathbf{J}_{d'} n^{-2} |\mathbf{H}_{\mathrm{AMISE}}|^{-1/2})(\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}}^{-1})(\mathrm{vech}^T\,\mathbf{H}_{\mathrm{AMISE}}^{-1}) \\
&\qquad + O(\mathbf{J}_{d'} n^{-1})(\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}})(\mathrm{vech}^T\,\mathbf{H}_{\mathrm{AMISE}}) \\
&\quad = O(\mathbf{J}_{d'} n^{-d/(d+4)})(\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}})(\mathrm{vech}^T\,\mathbf{H}_{\mathrm{AMISE}})
\end{aligned}$$

as $\mathbf{H}_{\mathrm{AMISE}} = O(\mathbf{J}_{d'} n^{-2/(d+4)})$. Moreover, as $D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H}_{\mathrm{AMISE}}) = O(\mathbf{J}_d)$ then the result follows. □

The relative rate of convergence of the LSCV selector is obtained by combining the AMISE Lemma with Lemmas 7 and 8 to give Theorem 2.

Theorem 2. *Under the conditions of Lemmas 7 and 8 the relative rate of convergence of* $\hat{\mathbf{H}}_{\mathrm{LSCV}}$ *to* $\mathbf{H}_{\mathrm{AMISE}}$ *is* $n^{-\min(d,4)/(2d+8)}$.

The rate from Theorem 2 is for full bandwidth selectors. The rate remains the same for diagonal or $h^2\mathbf{I}$ selectors. Table 3.1 is an augmented version of Table 2.2 as we add the rate for the LSCV selector. The rate for the SAMSE plug-in selectors is faster than the LSCV for $d \leq 3$. For $d > 3$, the situation is reversed. For AMSE plug-in selectors it is much the same except that the change over point is at $d = 4$. The discrepancy of $\mathbf{H}_{\mathrm{AMISE}}$ and $\mathbf{H}_{\mathrm{MISE}}$ is dominated by the rate of the LSCV selector for $d \leq 3$. So for these dimensions, the rate of $\hat{\mathbf{H}}_{\mathrm{LSCV}}$ to $\mathbf{H}_{\mathrm{MISE}}$ and to $\mathbf{H}_{\mathrm{AMISE}}$ are the same. For $d \geq 4$, since the rate of $\mathbf{H}_{\mathrm{AMISE}} - \mathbf{H}_{\mathrm{MISE}}$ and rate of the LSCV selector to $\mathbf{H}_{\mathrm{AMISE}}$ are the same, it is not possible to ascertain directly the rate of convergence of $\hat{\mathbf{H}}_{\mathrm{LSCV}}$ to $\mathbf{H}_{\mathrm{MISE}}$ from this table.

	Convergence rate to $\mathbf{H}_{\mathrm{AMISE}}$						
Selector	d	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
$\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$ (diagonal)	$n^{-\min(8,d+4)/(2d+12)}$	$n^{-5/14}$	$n^{-3/8}$	$n^{-7/18}$	$n^{-2/5}$	$n^{-4/11}$	$n^{-1/3}$
$\hat{\mathbf{H}}_{\mathrm{PI,AMSE}}$	$n^{-4/(d+12)}$	$n^{-4/13}$	$n^{-2/7}$	$n^{-4/15}$	$n^{-1/4}$	$n^{-4/17}$	$n^{-2/9}$
$\hat{\mathbf{H}}_{\mathrm{PI,SAMSE}}$	$n^{-2/(d+6)}$	$n^{-2/7}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$	$n^{-2/11}$	$n^{-1/6}$
$\hat{\mathbf{H}}_{\mathrm{LSCV}}$	$n^{-\min(d,4)/(2d+8)}$	$n^{-1/10}$	$n^{-1/6}$	$n^{-3/14}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$
$\mathbf{H}_{\mathrm{AMISE}} - \mathbf{H}_{\mathrm{MISE}}$	$n^{-2/(d+4)}$	$n^{-2/5}$	$n^{-1/3}$	$n^{-2/7}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$

Table 3.1: Comparison of convergence rates

3.3 Biased cross validation

The LSCV selector relies on estimating the MISE. The approach taken by the biased cross validation (BCV) selector relies on estimating the AMISE:

$$\operatorname{AMISE} \hat{f}(\cdot; \mathbf{H}) = n^{-1} R(K) |\mathbf{H}|^{-1/2} + \tfrac{1}{4} \mu_2(K)^2 (\operatorname{vech}^T \mathbf{H}) \mathbf{\Psi}_4 (\operatorname{vech} \mathbf{H}).$$

As for the plug-in selectors in Chapter 2, we need to estimate $\mathbf{\Psi}_4$. Plug-in methods use a pilot bandwidth matrix/matrices that is/are independent of $\mathbf{H}$. For BCV, we set $\mathbf{G} = \mathbf{H}$ and use slightly different estimators. Since AMISE is a biased estimator of MISE then we expect that BCV is also biased for the MISE (although it is asymptotically unbiased). This gives BCV its name: the bias is introduced in an attempt reduce the variance.

There are two versions of BCV, depending on the estimator of $\psi_{\boldsymbol{r}}, |\boldsymbol{r}| = 4$, see Sain et al. (1994), Jones & Kappenman (1992). We can use

$$\check{\psi}_{\boldsymbol{r}}(\mathbf{H}) = n^{-2} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} (K_{\mathbf{H}}^{(\boldsymbol{r})} * K_{\mathbf{H}})(\boldsymbol{X}_i - \boldsymbol{X}_j) \tag{3.7}$$

or we could use

$$\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H}) = n^{-1} \sum_{i=1}^{n} \hat{f}_{-i}^{(\boldsymbol{r})}(\boldsymbol{X}_i; \mathbf{H}) = n^{-1}(n-1)^{-1} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} K_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_i - \boldsymbol{X}_j). \tag{3.8}$$

The motivation of $\tilde{\psi}_{\boldsymbol{r}}$ is fairly straight forward from its definition and follows from the fact that it is a sample mean of the $\hat{f}_{-i}^{(\boldsymbol{r})}(\boldsymbol{X}_i; \mathbf{H})$ and that $\psi_{\boldsymbol{r}} = \mathbb{E}\, f^{(\boldsymbol{r})}(\boldsymbol{X})$. The motivation of $\check{\psi}_{\boldsymbol{r}}$ is given by replacing f in $\psi_{\boldsymbol{r}} = \int_{\mathbb{R}^d} f^{(\boldsymbol{r})}(\boldsymbol{x}) f(\boldsymbol{x})\, d\boldsymbol{x}$ with $\hat{f}_{-i}(\boldsymbol{X}_i; \mathbf{H})$ and then taking the sample mean. (This was shown in Section 1.3.) We use these, rather than the leave-in-diagonals estimator of Chapter 2, as we no longer seek to annihilate the contribution from the non-stochastic terms with the leading term of the leave-out-diagonals double sum. The estimates $\check{\mathbf{\Psi}}_4$ and $\tilde{\mathbf{\Psi}}_4$ are obtained from $\mathbf{\Psi}_4$ by substituting $\check{\psi}_{\boldsymbol{r}}$ and $\tilde{\psi}_{\boldsymbol{r}}$ for $\psi_{\boldsymbol{r}}$. The BCV1 function is the version of BCV with $\check{\mathbf{\Psi}}_4$

$$\operatorname{BCV1}(\mathbf{H}) = n^{-1} R(K) |\mathbf{H}|^{-1/2} + \tfrac{1}{4} \mu_2(K)^2 (\operatorname{vech}^T \mathbf{H}) \check{\mathbf{\Psi}}_4 (\operatorname{vech} \mathbf{H}) \tag{3.9}$$

and the BCV2 function is the version with $\tilde{\mathbf{\Psi}}_4$

$$\operatorname{BCV2}(\mathbf{H}) = n^{-1} R(K) |\mathbf{H}|^{-1/2} + \tfrac{1}{4} \mu_2(K)^2 (\operatorname{vech}^T \mathbf{H}) \tilde{\mathbf{\Psi}}_4 (\operatorname{vech} \mathbf{H}). \tag{3.10}$$

The BCV selectors $\hat{\mathbf{H}}_{\mathrm{BCV}}$ are the minimisers of the appropriate BCV function. Sain et al. (1994) have conducted some research into diagonal BCV selectors. These authors computed the relative rates of convergence for the diagonal selector which we now replicate for the full selector.

The two estimators $\check{\psi}_{\boldsymbol{r}}$ and $\tilde{\psi}_{\boldsymbol{r}}$ are fairly similar to each other. If we use the normal kernel then we have $\phi_{\mathbf{H}}^{(\boldsymbol{r})} * \phi_{\mathbf{H}} = (-1)^{|\boldsymbol{r}|}\phi_{2\mathbf{H}}^{(\boldsymbol{r})}$ so the only difference is $\check{\psi}_{\boldsymbol{r}}$ uses $2\mathbf{H}$ and $\tilde{\psi}_{\boldsymbol{r}}$ uses $\mathbf{H}$. This difference does not affect the relative convergence rates as it does not affect the order of the asymptotic bias and variance. Thus we only need to find rates for BCV2 (the 2 will be dropped in the following calculations for clarity.) Lemma 9 contains the result for the asymptotic bias and Lemma 10, the asymptotic variance.

Lemma 9. *Assume A1 – A3 of the AMSE Lemma (Lemma 3). Then*

$$\operatorname{ABias}(\operatorname{vech} \hat{\mathbf{H}}_{\mathrm{BCV}}) = O(\mathbf{J}_{d'} n^{-2/(d+4)}) \operatorname{vech} \mathbf{H}_{\mathrm{AMISE}}.$$

Proof. We start with

$$(\mathrm{BCV} - \mathrm{AMISE})(\mathbf{H}) = \tfrac{1}{4}(\operatorname{vech}^T \mathbf{H})(\tilde{\mathbf{\Psi}}_4(\mathbf{H}) - \mathbf{\Psi}_4)(\operatorname{vech} \mathbf{H})[1 + o_p(1)]$$

then

$$\mathbb{E}(\mathrm{BCV} - \mathrm{AMISE})(\mathbf{H}) = \tfrac{1}{4}(\operatorname{vech}^T \mathbf{H})(\mathbb{E}\, \tilde{\mathbf{\Psi}}_4(\mathbf{H}) - \mathbf{\Psi}_4)(\operatorname{vech} \mathbf{H})[1 + o(1)].$$

Now, $\mathbb{E}\, \tilde{\mathbf{\Psi}}_4(\mathbf{H}) - \mathbf{\Psi}_4$ and is composed of elements of the type $\mathbb{E}\, \tilde{\psi}_{\boldsymbol{r}}(\mathbf{H}) - \psi_{\boldsymbol{r}}$. As

$$\mathbb{E}\, \tilde{\psi}_{\boldsymbol{r}}(\mathbf{H}) - \psi_{\boldsymbol{r}} = \tfrac{1}{2} \int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H} D^2 f(\boldsymbol{x})) f^{(\boldsymbol{r})}(\boldsymbol{x})\, d\boldsymbol{x}$$

thus $\mathbb{E}(\mathrm{BCV} - \mathrm{AMISE})(\mathbf{H}) = O(\|\operatorname{vech} \mathbf{H}\|^3)$ and

$$\mathbb{E}[D_{\mathbf{H}}(\mathrm{BCV} - \mathrm{AMISE})(\mathbf{H}_{\mathrm{AMISE}})] = O(\mathbf{J}_{d'} n^{-2/(d+4)}) \operatorname{vech} \mathbf{H}_{\mathrm{AMISE}}.$$

□

Lemma 10. *Assume A1 – A2 of the AMSE Lemma (Lemma 3), and that K is normal. Then*

$$\operatorname{AVar}(\operatorname{vech} \hat{\mathbf{H}}_{\mathrm{BCV}}) = O(\mathbf{J}_{d'} n^{-d/(d+4)})(\operatorname{vech} \mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T \mathbf{H}_{\mathrm{AMISE}}).$$

Proof. Let $\boldsymbol{y} = \operatorname{vech} \mathbf{H}$ and $\mathbf{A}(\boldsymbol{y}) = \tilde{\mathbf{\Psi}}_4(\mathbf{H})$. We have

$$\begin{aligned} d(\boldsymbol{y}^T \mathbf{A}(\boldsymbol{y})\boldsymbol{y}) &= d(\boldsymbol{y}^T \mathbf{A}(\boldsymbol{y}))\boldsymbol{y} + \boldsymbol{y}^T \mathbf{A}(\boldsymbol{y})\, d\boldsymbol{y} \\ &= [(d\boldsymbol{y}^T)\mathbf{A}(\boldsymbol{y}) + \boldsymbol{y}^T d\mathbf{A}(\boldsymbol{y})]\boldsymbol{y} + \boldsymbol{y}^T \mathbf{A}(\boldsymbol{y})\, d\boldsymbol{y} \\ &= 2\boldsymbol{y}^T \mathbf{A}(\boldsymbol{y})\, d\boldsymbol{y} + \operatorname{vec}^T(\boldsymbol{y}\boldsymbol{y}^T)\, d \operatorname{vec} \mathbf{A}(\boldsymbol{y}). \end{aligned}$$

Then using the first identification table of Magnus & Neudecker (1988, p. 176) the derivative is

$$\begin{aligned} D_{\boldsymbol{y}}(\boldsymbol{y}^T \mathbf{A}(\boldsymbol{y})\boldsymbol{y}) &= 2\mathbf{A}(\boldsymbol{y})\boldsymbol{y} + [D_{\boldsymbol{y}} \mathbf{A}(\boldsymbol{y})]^T \operatorname{vec}(\boldsymbol{y}\boldsymbol{y}^T) \\ &= 2\mathbf{A}(\boldsymbol{y})\boldsymbol{y} + [D_{\boldsymbol{y}} \mathbf{A}(\boldsymbol{y})]^T (\boldsymbol{y} \otimes \mathbf{I}_{d'})\boldsymbol{y} \end{aligned}$$

where $\otimes$ is the Kronecker (or tensor) product between two matrices. Using this, the derivative of BCV $-$ AMISE is

$$\begin{aligned}
&D_{\mathbf{H}}(\mathrm{BCV}-\mathrm{AMISE})(\mathbf{H})\\
&= D_{\mathbf{H}}[\tfrac{1}{4}(\mathrm{vech}^T\,\mathbf{H})(\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})-\boldsymbol{\Psi}_4)(\mathrm{vech}\,\mathbf{H})]\\
&= \tfrac{1}{2}(\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})-\boldsymbol{\Psi}_4)(\mathrm{vech}\,\mathbf{H}) + \tfrac{1}{4}[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]^T(\mathrm{vech}\,\mathbf{H}\otimes\mathbf{I}_{d'})(\mathrm{vech}\,\mathbf{H}).
\end{aligned}$$

Then the variance of $D_{\mathbf{H}}(\mathrm{BCV}-\mathrm{AMISE})(\mathbf{H})$ will be of the same rate as the minimum rate of $\mathrm{Var}[\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})(\mathrm{vech}\,\mathbf{H})]$ and $\mathrm{Var}\{[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]^T(\mathrm{vech}\,\mathbf{H}\otimes\mathbf{I}_{d'})\}$.

The first of these is

$$\begin{aligned}
\mathrm{Var}[\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})(\mathrm{vech}\,\mathbf{H})] &= \mathbb{E}[\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})(\mathrm{vech}\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H})\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]\\
&\quad - [\mathbb{E}\,\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})(\mathrm{vech}\,\mathbf{H})][(\mathrm{vech}^T\,\mathbf{H})\,\mathbb{E}\,\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})].
\end{aligned}$$

Now $\mathbb{E}[\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})] - [\mathbb{E}\,\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})][\mathbb{E}\,\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]$ contains elements of the type

$$\begin{aligned}
\mathbb{E}[\tilde{\psi}_{\boldsymbol{r}_1}(\mathbf{H})\tilde{\psi}_{\boldsymbol{r}_2}(\mathbf{H})] - [\mathbb{E}\,\tilde{\psi}_{\boldsymbol{r}_1}(\mathbf{H})][\mathbb{E}\,\tilde{\psi}_{\boldsymbol{r}_2}(\mathbf{H})] &= \mathrm{Cov}[\tilde{\psi}_{\boldsymbol{r}_1}(\mathbf{H}),\tilde{\psi}_{\boldsymbol{r}_2}(\mathbf{H})]\\
&= O(\min\{\mathrm{Var}\,\tilde{\psi}_{\boldsymbol{r}_1}(\mathbf{H}),\mathrm{Var}\,\tilde{\psi}_{\boldsymbol{r}_2}(\mathbf{H})\}).
\end{aligned}$$

We know that $\mathrm{Var}\,\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H}) = O(n^{-2}|\mathbf{H}|^{1/2}\|\mathrm{vech}\,\mathbf{H}\|^{-|\boldsymbol{r}|})$ if $n^{-2}|\mathbf{H}|^{-1/2}\,\|\mathrm{vech}\,\mathbf{H}\|^{-|\boldsymbol{r}|}\to 0$ as $n\to\infty$. This is true for $\mathbf{H} = O(\mathbf{J}_d n^{-2/(d+4)})$ and $|\boldsymbol{r}| = 4$. Thus we have

$$\mathrm{Var}[\tilde{\boldsymbol{\Psi}}_4(\mathbf{H}_{\mathrm{AMISE}})(\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}})] = O(\mathbf{J}_{d'}n^{-d/(d+4)})(\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}})(\mathrm{vech}^T\,\mathbf{H}_{\mathrm{AMISE}}). \tag{3.11}$$

The second term is

$$\begin{aligned}
&\mathrm{Var}\{[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]^T(\mathrm{vech}\,\mathbf{H}\otimes\mathbf{I}_{d'})(\mathrm{vech}\,\mathbf{H})\}\\
&= \mathbb{E}\{[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]^T(\mathrm{vech}\,\mathbf{H}\otimes\mathbf{I}_{d'})(\mathrm{vech}\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H}\otimes\mathbf{I}_{d'})D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})\}\\
&\quad - \mathbb{E}[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]^T(\mathrm{vech}\,\mathbf{H}\otimes\mathbf{I}_{d'})(\mathrm{vech}\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H})(\mathrm{vech}^T\,\mathbf{H}\otimes\mathbf{I}_{d'})\,\mathbb{E}[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})].
\end{aligned} \tag{3.12}$$

Finding the order of this variance is non-trivial and involves a long sequence of matrix calculus computations. The main component of the variance is

$$\mathbb{E}[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]^T[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})] - \mathbb{E}[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]^T\,\mathbb{E}[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H})]$$

(if we temporarily ignore the contribution from $\mathrm{vech}\,\mathbf{H}\otimes\mathbf{I}_{d'}$) and it contains blocks of elements of the type

$$\begin{aligned}
&\sum_{\boldsymbol{r}:|\boldsymbol{r}|=4}\mathbb{E}\{[D_{\mathbf{H}}\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H})][D_{\mathbf{H}}\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H})]^T\} - \mathbb{E}[D_{\mathbf{H}}\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H})]\,\mathbb{E}[D_{\mathbf{H}}\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H})]^T\\
&= \sum_{\boldsymbol{r}:|\boldsymbol{r}|=4}\mathrm{Var}\,D_{\mathbf{H}}\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H})\\
&= \sum_{\boldsymbol{r}:|\boldsymbol{r}|=4}\mathrm{Var}\left[n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n}D_{\mathbf{H}}\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{x}_i-\boldsymbol{X}_j)\right].
\end{aligned} \tag{3.13}$$

The derivative of $\phi_{\mathbf{H}}^{(\boldsymbol{r})}$ with respect to vech $\mathbf{H}$ is

$$\begin{aligned}
D_{\mathbf{H}}\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{x}) &= \frac{\partial^{|\boldsymbol{r}|}}{\partial x_1^{r_1}\dots\partial x_d^{r_d}} D_{\mathbf{H}}\phi_{\mathbf{H}}(\boldsymbol{x}) \\
&= \frac{\partial^{|\boldsymbol{r}|}}{\partial x_1^{r_1}\dots\partial x_d^{r_d}} \tfrac{1}{2}\phi_{\mathbf{H}}(\boldsymbol{x})\mathbf{D}_d^T \operatorname{vec}[\mathbf{H}^{-1}\boldsymbol{x}\boldsymbol{x}^T\mathbf{H}^{-1} - \mathbf{H}^{-1}] \\
&= \tfrac{1}{2}\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{x})\mathbf{D}_d^T \operatorname{vec}[\mathbf{H}^{-1}\boldsymbol{x}\boldsymbol{x}^T\mathbf{H}^{-1}] \\
&\quad + \tfrac{1}{2}\phi_{\mathbf{H}}(\boldsymbol{x})\mathbf{D}_d^T \operatorname{vec}\left[\mathbf{H}^{-1}\frac{\partial^{|\boldsymbol{r}|}}{\partial x_1^{r_1}\dots\partial x_d^{r_d}}(\boldsymbol{x}\boldsymbol{x}^T)\mathbf{H}^{-1}\right] - \tfrac{1}{2}\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{x})\mathbf{D}_d^T \operatorname{vec}\mathbf{H}^{-1}.
\end{aligned}$$

For $|\boldsymbol{r}| = 4$,

$$\sum_{i=1}^n \sum_{\substack{j=1\\ j\neq i}}^n \phi_{\mathbf{H}}(\boldsymbol{X}_i - \boldsymbol{X}_j)\frac{\partial^{|\boldsymbol{r}|}}{\partial x_1^{r_1}\dots\partial x_d^{r_d}}[(\boldsymbol{X}_i - \boldsymbol{X}_j)(\boldsymbol{X}_i - \boldsymbol{X}_j)^T] = \sum_{i=1}^n \sum_{\substack{j=1\\ j\neq i}}^n \phi_{\mathbf{H}}(\boldsymbol{X}_i - \boldsymbol{X}_j)\mathbf{C}_0$$

where

$$\mathbf{C}_0 = \begin{cases} 2\mathbf{E}_{kk} + 2\mathbf{E}_{\ell\ell} & \text{if } \boldsymbol{r} = 2\boldsymbol{e}_k + 2\boldsymbol{e}_\ell, k, \ell = 1, 2\dots, d \\ \mathbf{0} & \text{otherwise} \end{cases}$$

and $\mathbf{E}_{ij}$ is a $d'\times d'$ elementary matrix which has 1 as its (i,j)-th element and 0 elsewhere. So then

$$\begin{aligned}
&[n(n-1)]^{-1}\sum_{i=1}^n \sum_{\substack{j=1\\ j\neq i}}^n D_{\mathbf{H}}\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_i - \boldsymbol{X}_j) \\
&\quad = \tfrac{1}{2}\mathbf{D}_d^T(\mathbf{H}^{-1}\otimes\mathbf{H}^{-1})\operatorname{vec}\tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H}) + \tfrac{1}{2}\tilde{\psi}_{\mathbf{0}}(\mathbf{H})\mathbf{D}_d^T(\mathbf{H}^{-1}\otimes\mathbf{H}^{-1})\operatorname{vec}\mathbf{C}_0 \\
&\qquad - \tfrac{1}{2}\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H})\mathbf{D}_d^T \operatorname{vec}\mathbf{H}^{-1} \qquad (3.14)
\end{aligned}$$

using $\operatorname{vec}(\mathbf{ABC}) = (\mathbf{C}^T\otimes\mathbf{A})\operatorname{vec}\mathbf{B}$ and where

$$\operatorname{vec}\tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H}) = n^{-1}(n-1)^{-1}\sum_{i=1}^n \sum_{\substack{j=1\\ j\neq i}}^n \phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_i - \boldsymbol{X}_j)\operatorname{vec}[(\boldsymbol{X}_i - \boldsymbol{X}_j)(\boldsymbol{X}_i - \boldsymbol{X}_j)^T].$$

Now the order of the variance of the left hand side of Equation (3.14) is the minimum order of the three terms on the right hand side. Since we know that $\operatorname{Var}\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H}) = O(n^{-2}|\mathbf{H}|^{-1/2}\|\operatorname{vech}\mathbf{H}\|^{-|\boldsymbol{r}|})$ so the second term of the right hand side is

$$\begin{aligned}
&\operatorname{Var}[\tilde{\psi}_{\boldsymbol{r}}(\mathbf{H}_{\text{AMISE}})\mathbf{D}_d^T \operatorname{vec}\mathbf{H}_{\text{AMISE}}^{-1}] \\
&\quad = O(\mathbf{J}_{d'}n^{-2}|\mathbf{H}_{\text{AMISE}}|^{-1/2}\|\operatorname{vech}\mathbf{H}_{\text{AMISE}}\|^{-4})(\operatorname{vech}\mathbf{H}_{\text{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\text{AMISE}}) \\
&\quad = O(\mathbf{J}_{d'}n^{(-d+4)/(d+4)}) \qquad (3.15)
\end{aligned}$$

and the third term is

$$\begin{aligned}
&\operatorname{Var}[\tilde{\psi}_{\mathbf{0}}(\mathbf{H}_{\text{AMISE}})\mathbf{D}_d^T(\mathbf{H}_{\text{AMISE}}^{-1}\otimes\mathbf{H}_{\text{AMISE}}^{-1})\operatorname{vec}\mathbf{C}_0] \\
&\quad = O(\mathbf{J}_{d'}n^{-2}|\mathbf{H}_{\text{AMISE}}|^{-1/2})(\operatorname{vech}\mathbf{H}_{\text{AMISE}}^{-2})(\operatorname{vech}^T\mathbf{H}_{\text{AMISE}}^{-2}) \\
&\quad = O(\mathbf{J}_{d'}n^{-d/(d+4)}). \qquad (3.16)
\end{aligned}$$

What remains is the first term of the right hand side of Equation (3.14): as the summand of the double sum of $\operatorname{vec} \tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H})$ is a symmetric function so

$$\begin{aligned}\operatorname{Var} \operatorname{vec} \tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H}) &= 2n^{-2} \operatorname{Var} \phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \operatorname{vec}[(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T] \\ &\quad + 4n^{-1} \operatorname{Cov}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \operatorname{vec}[(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T], \\ &\quad \phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_2 - \boldsymbol{X}_3) \operatorname{vec}^T[(\boldsymbol{X}_2 - \boldsymbol{X}_3)(\boldsymbol{X}_2 - \boldsymbol{X}_3)^T]\}.\end{aligned}$$

The first term of $\operatorname{Var} \operatorname{vec} \tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H})$ is

$$\begin{aligned}&\operatorname{Var}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \operatorname{vec}[(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T]\} \\ &\quad = O(\mathbf{J}_{d^2} |\mathbf{H}|^{-1/2} \|\operatorname{vech} \mathbf{H}\|^{-|\boldsymbol{r}|})(\operatorname{vec} \mathbf{H})(\operatorname{vec}^T \mathbf{H}). \qquad (3.17)\end{aligned}$$

We obtain this expression by considering $\mathbb{E}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \operatorname{vec}[(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T]\}$ first

$$\begin{aligned}&\mathbb{E}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \operatorname{vec}[(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T]\} \\ &\quad = \int_{\mathbb{R}^{2d}} \phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{x} - \boldsymbol{y}) \operatorname{vec}[(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T] f(\boldsymbol{x}) f(\boldsymbol{y}) \, d\boldsymbol{x} d\boldsymbol{y} \\ &\quad = \int_{\mathbb{R}^{2d}} \phi_{\mathbf{H}}(\boldsymbol{x} - \boldsymbol{y}) \operatorname{vec}[(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T] f(\boldsymbol{x}) f^{(\boldsymbol{r})}(\boldsymbol{y}) \, d\boldsymbol{x} d\boldsymbol{y} \\ &\quad = \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}(\boldsymbol{w}) \operatorname{vec}(\mathbf{H}^{1/2} \boldsymbol{w} \boldsymbol{w}^T \mathbf{H}^{1/2}) f(\boldsymbol{y} + \mathbf{H}^{1/2} \boldsymbol{w}) \, d\boldsymbol{w} d\boldsymbol{y} \\ &\quad = \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}(\boldsymbol{w}) \operatorname{vec}(\mathbf{H}^{1/2} \boldsymbol{w} \boldsymbol{w}^T \mathbf{H}^{1/2}) [f(\boldsymbol{y}) + O(\|\operatorname{vech} \mathbf{H}\|)] \, d\boldsymbol{w} d\boldsymbol{y} \\ &\quad = \psi_{\boldsymbol{r}} \operatorname{vec} \mathbf{H} + O(\|\operatorname{vech} \mathbf{H}\|) \operatorname{vec} \mathbf{H};\end{aligned}$$

and next

$$\begin{aligned}&\mathbb{E}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1 - \boldsymbol{X}_2)^2 \operatorname{vec}[(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)^T] \operatorname{vec}^T[(\boldsymbol{X}_1 - \boldsymbol{X}_2)(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\} \\ &= \int_{\mathbb{R}^{2d}} \phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{x} - \boldsymbol{y})^2 \operatorname{vec}[(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T] \operatorname{vec}^T[(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T] f(\boldsymbol{x}) f(\boldsymbol{y}) \, d\boldsymbol{x} d\boldsymbol{y} \\ &= \int_{\mathbb{R}^{2d}} [|\mathbf{H}|^{-1/2} \phi_{\mathbf{I}}^{(\boldsymbol{r})}(\mathbf{H}^{-1/2}(\boldsymbol{x} - \boldsymbol{y})) O(\mathbf{J}_{d^2} \|\operatorname{vech} \mathbf{H}\|^{-|\boldsymbol{r}|/2})]^2 \operatorname{vec}[(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T] \\ &\quad \times \operatorname{vec}^T[(\boldsymbol{x} - \boldsymbol{y})(\boldsymbol{x} - \boldsymbol{y})^T] f(\boldsymbol{x}) f(\boldsymbol{y}) \, d\boldsymbol{x} d\boldsymbol{y} \\ &= O(\mathbf{J}_{d^2} |\mathbf{H}|^{-1/2} \|\operatorname{vech} \mathbf{H}\|^{-|\boldsymbol{r}|}) \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}^{(\boldsymbol{r})}(\boldsymbol{w})^2 \operatorname{vec}(\mathbf{H}^{1/2} \boldsymbol{w} \boldsymbol{w}^T \mathbf{H}^{1/2}) \\ &\quad \times \operatorname{vec}^T(\mathbf{H}^{1/2} \boldsymbol{w} \boldsymbol{w}^T \mathbf{H}^{1/2}) f(\boldsymbol{y} + \mathbf{H}^{1/2} \boldsymbol{w}) f(\boldsymbol{y}) \, d\boldsymbol{w} d\boldsymbol{y} \\ &= O(\mathbf{J}_{d^2} |\mathbf{H}|^{-1/2} \|\operatorname{vech} \mathbf{H}\|^{-|\boldsymbol{r}|}) \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}^{(\boldsymbol{r})}(\boldsymbol{w})^2 (\mathbf{H}^{1/2} \otimes \mathbf{H}^{1/2}) \operatorname{vec}(\boldsymbol{w} \boldsymbol{w}^T) \operatorname{vec}^T(\boldsymbol{w} \boldsymbol{w}^T) \\ &\quad \times (\mathbf{H}^{1/2} \otimes \mathbf{H}^{1/2})[f(\boldsymbol{y}) + o(1)] f(\boldsymbol{y}) \, d\boldsymbol{w} d\boldsymbol{y} \\ &= O(\mathbf{J}_{d^2} |\mathbf{H}|^{-1/2} \|\operatorname{vech} \mathbf{H}\|^{-|\boldsymbol{r}|})(\operatorname{vec} \mathbf{H})(\operatorname{vec}^T \mathbf{H}).\end{aligned}$$

Combining these two previous expressions gives Equation (3.17) as stated.

The second term of $\operatorname{Var}\operatorname{vec}\tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H})$ is

$$\begin{aligned}&\operatorname{Cov}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1-\boldsymbol{X}_2)\operatorname{vec}[(\boldsymbol{X}_1-\boldsymbol{X}_2)(\boldsymbol{X}_1-\boldsymbol{X}_2)^T],\\&\quad\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_2-\boldsymbol{X}_3)\operatorname{vec}^T[(\boldsymbol{X}_2-\boldsymbol{X}_3)(\boldsymbol{X}_2-\boldsymbol{X}_3)^T]\}=O(\mathbf{J}_{d^2})(\operatorname{vec}\mathbf{H})(\operatorname{vec}^T\mathbf{H}).\end{aligned}\tag{3.18}$$

This is because

$$\begin{aligned}&\mathbb{E}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1-\boldsymbol{X}_2)\operatorname{vec}[(\boldsymbol{X}_1-\boldsymbol{X}_2)(\boldsymbol{X}_1-\boldsymbol{X}_2)^T]\\&\times\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_2-\boldsymbol{X}_3)\operatorname{vec}^T[(\boldsymbol{X}_2-\boldsymbol{X}_3)(\boldsymbol{X}_2-\boldsymbol{X}_3)^T]\}\\&\quad=\int_{\mathbb{R}^{3d}}\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{x}-\boldsymbol{y})\operatorname{vec}[(\boldsymbol{x}-\boldsymbol{y})(\boldsymbol{x}-\boldsymbol{y})^T]\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{y}-\boldsymbol{z})\operatorname{vec}^T[(\boldsymbol{y}-\boldsymbol{z})(\boldsymbol{y}-\boldsymbol{z})^T]\\&\qquad\times f(\boldsymbol{x})f(\boldsymbol{y})f(\boldsymbol{z})\,d\boldsymbol{x}d\boldsymbol{y}d\boldsymbol{z}\\&\quad=\int_{\mathbb{R}^{3d}}\phi_{\mathbf{H}}(\boldsymbol{x}-\boldsymbol{y})\operatorname{vec}[(\boldsymbol{x}-\boldsymbol{y})(\boldsymbol{x}-\boldsymbol{y})^T]\phi_{\mathbf{H}}(\boldsymbol{y}-\boldsymbol{z})\operatorname{vec}^T[(\boldsymbol{y}-\boldsymbol{z})(\boldsymbol{y}-\boldsymbol{z})^T]\\&\qquad\times f^{(\boldsymbol{r})}(\boldsymbol{x})f^{(\boldsymbol{r})}(\boldsymbol{y})f(\boldsymbol{z})\,d\boldsymbol{x}d\boldsymbol{y}d\boldsymbol{z}\\&\quad=\int_{\mathbb{R}^{3d}}\phi_{\mathbf{I}}(\boldsymbol{v})\phi_{\mathbf{I}}(\boldsymbol{w})\operatorname{vec}(\mathbf{H}^{1/2}\boldsymbol{v}\boldsymbol{v}^T\mathbf{H}^{1/2})\operatorname{vec}^T(\mathbf{H}^{1/2}\boldsymbol{w}\boldsymbol{w}^T\mathbf{H}^{1/2})\\&\qquad\times f^{(\boldsymbol{r})}(\boldsymbol{y}+\mathbf{H}^{1/2}\boldsymbol{w})f^{(\boldsymbol{r})}(\boldsymbol{y})f(\boldsymbol{y}-\mathbf{H}^{1/2}\boldsymbol{w})\,d\boldsymbol{v}d\boldsymbol{w}d\boldsymbol{y}\\&\quad=O(\mathbf{J}_{d^2})(\operatorname{vec}\mathbf{H})(\operatorname{vec}^T\mathbf{H})\end{aligned}$$

is the same order as the product of $\mathbb{E}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_1-\boldsymbol{X}_2)\operatorname{vec}[(\boldsymbol{X}_1-\boldsymbol{X}_2)(\boldsymbol{X}_1-\boldsymbol{X}_2)^T]\}$ and $\mathbb{E}\{\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_2-\boldsymbol{X}_3)\operatorname{vec}^T[(\boldsymbol{X}_2-\boldsymbol{X}_3)(\boldsymbol{X}_2-\boldsymbol{X}_3)^T]\}$.

The expression for the order of $\operatorname{Var}\operatorname{vec}\tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H})$ is a result of combining Equations (3.17) and (3.18)

$$\begin{aligned}&\operatorname{Var}[\mathbf{D}_d^T(\mathbf{H}_{\mathrm{AMISE}}^{-1}\otimes\mathbf{H}_{\mathrm{AMISE}}^{-1})\operatorname{vec}\tilde{\boldsymbol{\psi}}_{\boldsymbol{r}}^{[2]}(\mathbf{H}_{\mathrm{AMISE}})]\\&\quad=O(\mathbf{J}_{d'}n^{-2}|\mathbf{H}_{\mathrm{AMISE}}|^{-1/2}\|\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}\|^{-4})(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}^{-2})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}}^{-2})\\&\qquad+O(\mathbf{J}_{d'}n^{-1})(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}})\\&\quad=O(\mathbf{J}_{d'}n^{(-d+4)/(d+4)}).\end{aligned}\tag{3.19}$$

Equations (3.15), (3.16) and (3.19) combine to give the variance of Equation (3.14):

$$\operatorname{Var}\left[n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n}D_{\mathbf{H}}\phi_{\mathbf{H}}^{(\boldsymbol{r})}(\boldsymbol{X}_i-\boldsymbol{X}_j)\right]=O(\mathbf{J}_{d'}n^{(-d+4)/(d+4)}).$$

This implies that expressions of the type in Equation (3.13) are of the same order, which in turn implies that Equation (3.12) becomes

$$\begin{aligned}&\operatorname{Var}\{[D_{\mathbf{H}}\tilde{\boldsymbol{\Psi}}_4(\mathbf{H}_{\mathrm{AMISE}})]^T(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}\otimes\mathbf{I}_{d'})(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})\}\\&\quad=O(\mathbf{J}_{d'}n^{(-d+4)/(d+4)})[(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}})]^2\\&\quad=O(\mathbf{J}_{d'}n^{-d/(d+4)})(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}}).\end{aligned}$$

This is the same order as $\mathrm{Var}[\tilde{\boldsymbol{\Psi}}_4(\mathbf{H}_{\mathrm{AMISE}})(\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}})]$, Equation (3.11). The order of $\mathrm{Var}[D_{\mathbf{H}}(\mathrm{BCV}-\mathrm{AMISE})(\mathbf{H})]$ is the minimum order of Equations (3.11) and (3.12) i.e. $\mathrm{Var}[D_{\mathbf{H}}(\mathrm{BCV}-\mathrm{AMISE})(\mathbf{H})] = O(\mathbf{J}_{d'}n^{-d/(d+4)})(\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}})(\mathrm{vech}^T\,\mathbf{H}_{\mathrm{AMISE}})$. □

The relative rate of convergence of the BCV selectors is obtained by combining the AMSE Lemma with Lemmas 9 and 10 to give Theorem 3.

Theorem 3. *Under the conditions of Lemmas 9 and 10 the relative rate of convergence of* $\hat{\mathbf{H}}_{\mathrm{BCV}}$ *to* $\mathbf{H}_{\mathrm{AMISE}}$ *is* $n^{-\min(d,4)/(2d+8)}$.

This rate is identical to the rate of the LSCV selector. Sain et al. (1994) give the rate for the BCV selector to be $n^{-d/(2d+8)}$. This seems incorrect for $d > 4$ as the squared bias term dominates the variance term in these dimensions. In particular, their claim that the BCV convergence rate tends to $n^{-1/2}$ as d increases (which implies that its performance increases as d increases) appears to be invalid. The proof of Sain et al. does not keep proper track of second order bias terms which should lead to an additional term of order h^5 in their Equation (15).

3.4 Smoothed cross validation

Smoothed cross validation (SCV) can be thought of as a hybrid of LSCV and BCV. The SCV criterion takes the asymptotic integrated variance but attempts to estimate the integrated squared bias exactly rather than using its asymptotic form:

$$\begin{aligned}\mathrm{SCV}(\mathbf{H}) = n^{-1}R(K)|\mathbf{H}|^{-1/2} + n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(K_{\mathbf{H}} * K_{\mathbf{H}} * L_{\mathbf{G}} * L_{\mathbf{G}} - 2K_{\mathbf{H}} * L_{\mathbf{G}} * L_{\mathbf{G}} \\ + L_{\mathbf{G}} * L_{\mathbf{G}})(\boldsymbol{X}_i - \boldsymbol{X}_j)\end{aligned}$$

where $L_{\mathbf{G}}$ is the pilot kernel with pilot bandwidth matrix $\mathbf{G}$. The SCV selector $\hat{\mathbf{H}}_{\mathrm{SCV}}$ is the minimiser of $\mathrm{SCV}(\mathbf{H})$. If there are no replications in the data, then

$$\mathrm{LSCV}(\mathbf{H}) = n^{-1}R(K)|\mathbf{H}|^{-1/2} + n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n}(K_{\mathbf{H}} * K_{\mathbf{H}} - 2K_{\mathbf{H}})(\boldsymbol{X}_i - \boldsymbol{X}_j)$$

which is $\mathrm{SCV}(\mathbf{H})$ with $\mathbf{G} = \mathbf{0}$ (since $L_{\mathbf{0}}$ can be thought of as the Dirac delta function). Equivalently we can think of SCV as pre-smoothing the data $\boldsymbol{X}_i$ with $L_{\mathbf{G}}$ or the data differences $\boldsymbol{X}_i - \boldsymbol{X}_j$ with $L_{\mathbf{G}} * L_{\mathbf{G}}$ before applying the LSCV. If $K = L = \phi$ then the SCV has a simpler form:

$$\mathrm{SCV}(\mathbf{H}) = n^{-1}|\mathbf{H}|^{-1/2}(4\pi)^{-d/2} + n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_i - \boldsymbol{X}_j). \quad (3.20)$$

This form will be used to simplify the calculations in Section 3.4.1.

The asymptotic equivalence between SCV and the smoothed bootstrap, mentioned in Section 1.3, carries over to the multivariate case. Let $\boldsymbol{X}_1^*, \boldsymbol{X}_2^*, \ldots, \boldsymbol{X}_n^*$ be a bootstrap sample taken from the pilot kernel density estimate $\hat{f}_P(\boldsymbol{x}; \mathbf{G}) = n^{-1} \sum_{i=1}^n L_{\mathbf{G}}(\boldsymbol{x} - \boldsymbol{X}_i)$. Let the bootstrap kernel density estimate be

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

and $\mathbb{E}^*$ the expected value with respect to the bootstrap density $\hat{f}_P(\boldsymbol{x}; \mathbf{G})$ then

$$\mathbb{E}^* \hat{f}^*(\boldsymbol{x}; \mathbf{H}) = \mathbb{E}^* K_{\mathbf{H}}(\boldsymbol{x} - \boldsymbol{X}^*) = \int_{\mathbb{R}^d} K_{\mathbf{H}}(\boldsymbol{x} - \boldsymbol{y}) \hat{f}_P(\boldsymbol{y}; \mathbf{G}) \, d\boldsymbol{y} = (K_{\mathbf{H}} * \hat{f}_P(\cdot; \mathbf{G}))(\boldsymbol{x})$$

and the smoothed bootstrap bias is

$$\operatorname{Bias}^* \hat{f}^*(\boldsymbol{x}; \mathbf{H}) = \mathbb{E}^* \hat{f}^*(\boldsymbol{x}; \mathbf{H}) - \hat{f}_P(\boldsymbol{x}; \mathbf{G}) = (K_{\mathbf{H}} * f_P(\cdot; \mathbf{G}))(\boldsymbol{x}) - \hat{f}_P(\boldsymbol{x}; \mathbf{G}).$$

Since $\operatorname{Bias} \hat{f}(\boldsymbol{x}; \mathbf{H}) = (K_{\mathbf{H}} * f)(\boldsymbol{x}) - f(\boldsymbol{x})$ so the smoothed bootstrap bias is obtained when f is replaced by the bootstrap density $\hat{f}_P$ in the usual bias expression. Let Var^* be the bootstrap variance then the bootstrap MISE is

$$\begin{aligned}
\operatorname{MISE}^* \hat{f}^*(\cdot; \mathbf{H}) &= \int_{\mathbb{R}^d} \operatorname{Var}^* \hat{f}^*(\boldsymbol{x}; \mathbf{H}) \, d\boldsymbol{x} + \int_{\mathbb{R}^d} [\operatorname{Bias}^* \hat{f}^*(\boldsymbol{x}; \mathbf{H})]^2 \, d\boldsymbol{x} \\
&= n^{-1} |\mathbf{H}|^{-1/2} R(K) + n^{-1} \int_{\mathbb{R}^d} (K_{\mathbf{H}} * \hat{f}_P(\cdot; \mathbf{G}))(\boldsymbol{x}) \, d\boldsymbol{x} \\
&\quad + \int_{\mathbb{R}^d} [(K_{\mathbf{H}} * \hat{f}_P(\cdot; \mathbf{G}))(\boldsymbol{x}) - \hat{f}_P(\boldsymbol{x}; \mathbf{G})]^2 \, d\boldsymbol{x} \\
&= \operatorname{SCV}(\mathbf{H}) + o(n^{-1} |\mathbf{H}|^{-1/2}).
\end{aligned}$$

3.4.1 Optimal pilot bandwidth selector

Now we have a similar problem to plug-in type selectors: how to select an optimal pilot bandwidth. Sain et al. (1994) set the pilot to be equal to the final bandwidth. This circumvents the need to select a separate pilot bandwidth but this is sub-optimal. Jones et al. (1991) look at the relative mean squared error (RMSE) of the univariate SCV selector. For a univariate selector $\hat{h}$, this is $\operatorname{RMSE}(\hat{h}) = \mathbb{E}[(\hat{h} - h_{\operatorname{AMISE}})/h_{\operatorname{AMISE}}]^2$. These authors then choose the pilot bandwidth which minimises this RMSE. We follow a similar process though instead we minimise the (A)MSE, keeping in mind that minimising the RMSE and the (A)MSE are equivalent since the denominator of the RMSE does not depend on the bandwidth selector.

We could generalise the univariate $\operatorname{MSE}(\hat{h}) = \mathbb{E}(\hat{h} - h_{\operatorname{AMISE}})^2$ in many ways. One such generalisation is

$$\operatorname{tr} \operatorname{MSE}(\operatorname{vech} \hat{\mathbf{H}}; \mathbf{G}) = \mathbb{E}[\operatorname{vech}^T (\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{AMISE}}) \operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{AMISE}})].$$

This exact MSE is difficult to compute so we use an asymptotic approximation and as in Chapter 2, we will use the parameterisation $g^2\mathbf{I}$ for the pilot bandwidth matrix $\mathbf{G}$ i.e. we wish to find

$$g_0 = \operatorname*{argmin}_{g>0} \operatorname{tr} \operatorname{AMSE}(\operatorname{vech} \hat{\mathbf{H}}_{\mathrm{SCV}}; g).$$

The actual value for g_0 is found in Theorem 4. Lemmas 12 and 13 are two preliminary results which lead to the theorem. Following the theorem is Lemma 14 which states that the theorem is still valid if the optimal pilot bandwidth g_0 is replaced by its (consistent) plug-in estimate.

Before we begin to evaluate asymptotic expressions for SCV selectors, we need a modified version of the AMSE Lemma which we call the AMSE$'$ Lemma. Since we are using an exact estimate of the integrated squared bias, the usual AMISE approximation is insufficient, we need a higher order expansion AMISE$'$

$$\operatorname{AMISE}'(\mathbf{H}) = \operatorname{AMISE}(\mathbf{H}) + \tfrac{1}{8} \int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H} D^2 f(\boldsymbol{x})) \operatorname{tr}(\mathbf{H}^2 (D^2)^2 f(\boldsymbol{x}))\, d\boldsymbol{x}$$

and an estimate of this is $\widehat{\operatorname{AMISE}}'$.

Lemma 11 (AMSE$'$). *Assume A1 – A3 from the AMSE Lemma (Lemma 3). Let* $\hat{\mathbf{H}} = \operatorname*{argmin}_{\mathbf{H}\in\mathcal{H}} \widehat{\operatorname{AMISE}}'$ *be a bandwidth selector then* $\operatorname{MSE}(\operatorname{vech} \hat{\mathbf{H}}) = [\mathbf{I}_{d'} + o(\mathbf{J}_{d'})] \operatorname{AMSE}'(\operatorname{vech} \hat{\mathbf{H}})$. *The higher order asymptotic MSE can be written as*

$$\operatorname{AMSE}'(\operatorname{vech} \hat{\mathbf{H}}) = \operatorname{AVar}'(\operatorname{vech} \hat{\mathbf{H}}) + [\operatorname{ABias}'(\operatorname{vech} \hat{\mathbf{H}})][\operatorname{ABias}'(\operatorname{vech} \hat{\mathbf{H}})]^T$$

in which

$$\begin{aligned}
\operatorname{ABias}'(\operatorname{vech} \hat{\mathbf{H}}) &= [D^2_{\mathbf{H}} \operatorname{AMISE}(\mathbf{H}_{\mathrm{AMISE}})]^{-1} \mathbb{E}[D_{\mathbf{H}}(\widehat{\operatorname{AMISE}}' - \operatorname{AMISE}')(\mathbf{H}_{\mathrm{AMISE}})] \\
\operatorname{AVar}'(\operatorname{vech} \hat{\mathbf{H}}) &= [D^2_{\mathbf{H}} \operatorname{AMISE}(\mathbf{H}_{\mathrm{AMISE}})]^{-1} \operatorname{Var}[D_{\mathbf{H}}(\widehat{\operatorname{AMISE}}' - \operatorname{AMISE}')(\mathbf{H}_{\mathrm{AMISE}})] \\
&\quad \times [D^2_{\mathbf{H}} \operatorname{AMISE}(\mathbf{H}_{\mathrm{AMISE}})]^{-1}.
\end{aligned}$$

Proof. We expand $D_{\mathbf{H}} \widehat{\operatorname{AMISE}}'$ as follows:

$$\begin{aligned}
D_{\mathbf{H}} \widehat{\operatorname{AMISE}}'(\hat{\mathbf{H}}) &= D_{\mathbf{H}}(\widehat{\operatorname{AMISE}}' - \operatorname{AMISE}')(\hat{\mathbf{H}}) + D_{\mathbf{H}} \operatorname{AMISE}'(\hat{\mathbf{H}}) \\
&= [\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})] D_{\mathbf{H}}(\widehat{\operatorname{AMISE}}' - \operatorname{AMISE}')(\mathbf{H}_{\mathrm{AMISE}}) \\
&\quad + \big\{ D_{\mathbf{H}} \operatorname{AMISE}'(\mathbf{H}_{\mathrm{AMISE}'}) + [\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})] D^2_{\mathbf{H}} \operatorname{AMISE}(\mathbf{H}_{\mathrm{AMISE}'}) \\
&\quad \times \operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\mathrm{AMISE}'}) \big\}.
\end{aligned}$$

We have $D_{\mathbf{H}} \widehat{\operatorname{AMISE}}'(\hat{\mathbf{H}}) = \mathbf{0}$ and $D_{\mathbf{H}} \operatorname{AMISE}'(\mathbf{H}_{\mathrm{AMISE}'}) = \mathbf{0}$. This implies that

$$\begin{aligned}
\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\mathrm{AMISE}'}) &= -[\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})][D^2_{\mathbf{H}} \operatorname{AMISE}(\mathbf{H}_{\mathrm{AMISE}'})]^{-1} \\
&\quad \times D_{\mathbf{H}}(\widehat{\operatorname{AMISE}}' - \operatorname{AMISE}')(\mathbf{H}_{\mathrm{AMISE}}).
\end{aligned}$$

To rewrite the right hand side, we note that

$$\begin{aligned} D^2_{\mathbf{H}}\mathrm{AMISE}'(\mathbf{H}_{\mathrm{AMISE}'}) &= [\mathbf{I}_{d'} + o(\mathbf{J}_{d'})]D^2_{\mathbf{H}}\mathrm{AMISE}'(\mathbf{H}_{\mathrm{AMISE}}) \\ &= [\mathbf{I}_{d'} + o(\mathbf{J}_{d'})][D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H}_{\mathrm{AMISE}}) + D^2_{\mathbf{H}}O(\|\mathbf{H}_{\mathrm{AMISE}}\|^3)] \\ &= O(\mathbf{J}_{d'})D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H}_{\mathrm{AMISE}}). \end{aligned}$$

For the left hand side, we have $\mathbf{H}_{\mathrm{AMISE}'} = [\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'}]\mathbf{H}_{\mathrm{AMISE}}$ so

$$\mathrm{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\mathrm{AMISE}'}) = [\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'}]\,\mathrm{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\mathrm{AMISE}}).$$

Putting all this together,

$$\begin{aligned} \mathrm{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\mathrm{AMISE}}) = &-[\mathbf{I}_{d'} + o_p(\mathbf{J}_{d'})][D^2_{\mathbf{H}}\mathrm{AMISE}(\mathbf{H}_{\mathrm{AMISE}})]^{-1} \\ &\times D_{\mathbf{H}}(\widehat{\mathrm{AMISE}}' - \mathrm{AMISE}')(\mathbf{H}_{\mathrm{AMISE}}). \end{aligned}$$

Taking expectations and variances respectively completes the proof. □

Lemma 12. *Assume A1 – A2 from the* AMSE′ *Lemma (Lemma 11). Also assume that*

(S1) f has bounded and continuous eighth order partial derivatives

(S2) each element of $\mathbf{\Theta}_6 = \int_{\mathbb{R}^d}(D^2)^3 f(\boldsymbol{x})f(\boldsymbol{x})\,d\boldsymbol{x}$ is finite

(S3) the sequence of pilot bandwidths $g = g_n$ satisfies $g^{-2}\mathbf{H} \to \mathbf{0}$ as $n \to \infty$

(S4) K and L are normal kernels

then

$$\begin{aligned} \mathrm{ABias}'(\mathrm{vech}\,\hat{\mathbf{H}}_{\mathrm{SCV}}; g) = &\, n^{-2/(d+4)}g^2\boldsymbol{C}_{\mu_1} + n^{-2/(d+4)}n^{-1}g^{-d-4}\boldsymbol{C}_{\mu_2} \\ &+ O(\mathbf{J}_{d'}(g^4 + n^{-1}g^{-d-6}))\,\mathrm{vech}\,\mathbf{H}_{\mathrm{AMISE}} \end{aligned}$$

where

$$\begin{aligned} \boldsymbol{C}_{\mu_1} &= \tfrac{1}{2}n^{2/(d+4)}\mathbf{D}_d^T\,\mathrm{vec}(\mathbf{\Theta}_6\mathbf{H}_{\mathrm{AMISE}}) \\ \boldsymbol{C}_{\mu_2} &= \tfrac{1}{8}(4\pi)^{-d/2}n^{2/(d+4)}[2\mathbf{D}_d^T\,\mathrm{vec}\,\mathbf{H}_{\mathrm{AMISE}} + (\mathrm{tr}\,\mathbf{H}_{\mathrm{AMISE}})\mathbf{D}_d^T\,\mathrm{vec}\,\mathbf{I}_d]. \end{aligned}$$

Proof. To find $\mathrm{ABias}'(\mathrm{vech}\,\hat{\mathbf{H}}_{\mathrm{SCV}}; g)$, we first find $\mathbb{E}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\mathbf{H}_{\mathrm{AMISE}})]$. As $K = L = \phi$, we know that

$$\mathrm{SCV}(\mathbf{H}) = n^{-1}|\mathbf{H}|^{-1/2}(4\pi)^{-d/2} + n^{-2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_i - \boldsymbol{X}_j).$$

If we remove the non-stochastic terms from the double sum:

$$\begin{aligned}\operatorname{SCV}(\mathbf{H}) &= n^{-1}(4\pi)^{-d/2}|\mathbf{H}|^{-1/2} + n^{-1}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\mathbf{0}) \\ &\quad + n^{-2}\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_i - \boldsymbol{X}_j).\end{aligned}$$

The expected value of this is

$$\mathbb{E}\operatorname{SCV}(\mathbf{H}) = n^{-1}[(4\pi)^{-d/2}|\mathbf{H}|^{-1/2} + C_1] + \mathbb{E}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2).$$

where $C_1 = (2\pi)^{-d/2}|2\mathbf{H}+2\mathbf{G}|^{-1/2} - 2(2\pi)^{-d/2}|\mathbf{H}+2\mathbf{G}|^{-1/2} + (2\pi)^{-d/2}|2\mathbf{G}|^{-1/2}$.

For $\mathbf{A} = a\mathbf{H} + b\mathbf{G}$,

$$\begin{aligned}\mathbb{E}\,\phi_{\mathbf{A}}(\boldsymbol{X}_1 - \boldsymbol{X}_2) &= \int_{\mathbb{R}^{2d}} \phi_{\mathbf{A}}(\boldsymbol{x}-\boldsymbol{y})f(\boldsymbol{x})f(\boldsymbol{y})\,d\boldsymbol{x}d\boldsymbol{y} \\ &= \int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}(\boldsymbol{w})f(\boldsymbol{y}+\mathbf{A}^{1/2}\boldsymbol{w})f(\boldsymbol{y})\,d\boldsymbol{w}d\boldsymbol{y}.\end{aligned}$$

The eighth order Taylor series expansion of $f(\boldsymbol{y}+\mathbf{A}^{1/2}\boldsymbol{w})$ is

$$\begin{aligned}f(\boldsymbol{y}+\mathbf{A}^{1/2}\boldsymbol{w}) &= f(\boldsymbol{y}) + \operatorname{tr}(\mathbf{A}^{1/2}D\boldsymbol{w}^T)f(\boldsymbol{y}) + \tfrac{1}{2!}\operatorname{tr}(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)f(\boldsymbol{y}) \\ &\quad + \tfrac{1}{3!}[\operatorname{tr}(\mathbf{A}^{1/2}D\boldsymbol{w}^T)\ \operatorname{tr}(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)]f(\boldsymbol{y}) + \tfrac{1}{4!}\operatorname{tr}^2(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)f(\boldsymbol{y}) \\ &\quad + \tfrac{1}{5!}[\operatorname{tr}(\mathbf{A}^{1/2}D\boldsymbol{w}^T)\ \operatorname{tr}^2(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)]f(\boldsymbol{y}) + \tfrac{1}{6!}\operatorname{tr}^2(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)f(\boldsymbol{y}) \\ &\quad + \tfrac{1}{7!}[\operatorname{tr}(\mathbf{A}^{1/2}D\boldsymbol{w}^T)\ \operatorname{tr}^3(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)]f(\boldsymbol{y}) + \tfrac{1}{8!}\operatorname{tr}^4(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)f(\boldsymbol{y}) \\ &\quad + o(\|\operatorname{vech}\mathbf{A}\|^4).\end{aligned}$$

For $i = 0, 1, 2, \ldots,$ let

$$\begin{aligned}m_{2i} = m_{2i}(\phi_{\mathbf{I}};\mathbf{A}) &= \int_{\mathbb{R}^d} \phi_{\mathbf{I}}(\boldsymbol{w})\operatorname{tr}^i(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)\,d\boldsymbol{w} \\ m_{2i+1} = m_{2i+1}(\phi_{\mathbf{I}};\mathbf{A}) &= \int_{\mathbb{R}^d} \phi_{\mathbf{I}}(\boldsymbol{w})\operatorname{tr}^i(\mathbf{A}D^2\boldsymbol{w}\boldsymbol{w}^T)\operatorname{tr}(\mathbf{A}^{1/2}D\boldsymbol{w}^T)\,d\boldsymbol{w}\end{aligned}$$

then $m_0 = 1, m_2 = \operatorname{tr}(\mathbf{A}D^2), m_4 = 3\operatorname{tr}(\mathbf{A}^2(D^2)^2), m_6 = 15\operatorname{tr}(\mathbf{A}^3(D^2)^3)$ and $m_8 = 105\operatorname{tr}(\mathbf{A}^4(D^2)^4)$; and $m_1 = m_3 = m_5 = m_7 = 0$. Thus

$$\begin{aligned}&\mathbb{E}\,\phi_{\mathbf{A}}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \\ &= \int_{\mathbb{R}^d}[m_0 f(\boldsymbol{y}) + \tfrac{1}{2}m_2 f(\boldsymbol{y}) + \tfrac{1}{4!}m_4 f(\boldsymbol{y}) + \tfrac{1}{6!}m_6 f(\boldsymbol{y}) + \tfrac{1}{8!}m_8 f(\boldsymbol{y})]f(\boldsymbol{y})\,d\boldsymbol{y} \\ &\quad + o(\|\operatorname{vech}\mathbf{A}\|^4) \\ &= \int_{\mathbb{R}^d}\big[f(\boldsymbol{y}) + \tfrac{1}{2}\operatorname{tr}(\mathbf{A}D^2 f(\boldsymbol{y})) + \tfrac{1}{8}\operatorname{tr}(\mathbf{A}^2(D^2)^2 f(\boldsymbol{y})) + \tfrac{5}{240}\operatorname{tr}(\mathbf{A}^3(D^2)^3 f(\boldsymbol{y})) \\ &\quad + \tfrac{1}{384}\operatorname{tr}(\mathbf{A}^4(D^2)^4 f(\boldsymbol{y}))\big]f(\boldsymbol{y})\,d\boldsymbol{y} + o(\|\operatorname{vech}\mathbf{A}\|^4).\end{aligned}$$

Now as

$$\begin{aligned}
\operatorname{tr}(2\mathbf{H} + 2\mathbf{G} - 2(\mathbf{H} + 2\mathbf{G}) + 2\mathbf{G}) &= 0 \\
\operatorname{tr}((2\mathbf{H} + 2\mathbf{G})^2 - 2(\mathbf{H} + 2\mathbf{G})^2 + (2\mathbf{G})^2) &= \operatorname{tr}(2\mathbf{H}^2) \\
\operatorname{tr}((2\mathbf{H} + 2\mathbf{G})^3 - 2(\mathbf{H} + 2\mathbf{G})^3 + (2\mathbf{G})^3) &= \operatorname{tr}(6\mathbf{H}^3 + 12\mathbf{H}^2\mathbf{G}) \\
\operatorname{tr}((2\mathbf{H} + 2\mathbf{G})^4 - 2(\mathbf{H} + 2\mathbf{G})^4 + (2\mathbf{G})^4) &= \operatorname{tr}(14\mathbf{H}^4 + 48\mathbf{H}^3\mathbf{G} + 48\mathbf{H}^2\mathbf{G}^2)
\end{aligned}$$

then

$$\begin{aligned}
&\mathbb{E}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2) \\
&= \tfrac{1}{4}\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^2(D^2)^2 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} + \tfrac{1}{4}\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^2\mathbf{G}(D^2)^3 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} \\
&\quad + \tfrac{1}{8}\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^3(D^2)^3 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} + O(\|\operatorname{vech}\mathbf{H}^2\mathbf{G}^2\|).
\end{aligned}$$

As

$$\begin{aligned}
\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^2(D^2)^2 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} &= \int_{\mathbb{R}^d} \operatorname{tr}^2(\mathbf{H}D^2 f(\boldsymbol{y}))\ d\boldsymbol{y} \\
\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^3(D^2)^3 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} &= \int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}D^2 f(\boldsymbol{y}))\operatorname{tr}(\mathbf{H}^2(D^2)^2 f(\boldsymbol{y}))\ d\boldsymbol{y}
\end{aligned}$$

then

$$\begin{aligned}
\mathbb{E}\operatorname{SCV}(\mathbf{H}) = n^{-1}C_1 + \operatorname{AMISE}'(\mathbf{H}) &+ \tfrac{1}{4}\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^2\mathbf{G}(D^2)^3 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} \\
&+ O(\|\operatorname{vech}\mathbf{H}^2\mathbf{G}^2\|)
\end{aligned}$$

or

$$\mathbb{E}[(\operatorname{SCV} - \operatorname{AMISE}')(\mathbf{H})] = n^{-1}C_1 + \tfrac{1}{4}\operatorname{tr}(\mathbf{H}^2\mathbf{G}\boldsymbol{\Theta}_6) + O(\|\operatorname{vech}\mathbf{H}^2\mathbf{G}^2\|)$$

where $\boldsymbol{\Theta}_6 = \int_{\mathbb{R}^d}(D^2)^3 f(\boldsymbol{y})f(\boldsymbol{y})\ d\boldsymbol{y}$. (Note that the subscript on $\boldsymbol{\Theta}_6$ indicates the order of the derivatives involved.)

We now have $\mathbb{E}(\operatorname{SCV} - \operatorname{AMISE}')(\mathbf{H})$. The next step is to find the derivative of this. The derivative of C_1 is

$$\begin{aligned}
D_{\mathbf{H}}C_1 = &-(2\pi)^{-d/2}|2\mathbf{H} + 2\mathbf{G}|^{-1/2}\mathbf{D}_d^T \operatorname{vec}(2\mathbf{H} + 2\mathbf{G})^{-1} \\
&+ (2\pi)^{-d/2}|\mathbf{H} + 2\mathbf{G}|^{-1/2}\mathbf{D}_d^T \operatorname{vec}(\mathbf{H} + 2\mathbf{G})^{-1}
\end{aligned}$$

as $D_{\mathbf{H}}|\mathbf{H}|^{-1/2} = -\frac{1}{2}|\mathbf{H}|^{-1/2}\mathbf{D}_d^T \operatorname{vec}\mathbf{H}^{-1}$. We will now expand these determinants and matrix inverses to simplify this derivative. The inverse of $\mathbf{I} + \mathbf{A}$ can be expanded as

$$(\mathbf{I} + \mathbf{A})^{-1} = \mathbf{I} - \mathbf{A} + O(\|\operatorname{vech}\mathbf{A}\|^2)$$

Furthermore, let every element of $\mathbf{G}^{-1}\mathbf{H} \to 0$ as $n \to \infty$ or equivalently for $\mathbf{G} = g^2\mathbf{I}$, $g^{-2}\operatorname{tr}\mathbf{H} \to 0$ as $n \to \infty$ then

$$\begin{aligned}(a\mathbf{H} + b\mathbf{G})^{-1} &= (a\mathbf{H} + bg^2\mathbf{I})^{-1} \\ &= [bg^2(b^{-1}g^{-2}a\mathbf{H} + \mathbf{I})]^{-1} \\ &= b^{-1}g^{-2}[\mathbf{I} - b^{-1}g^{-2}a\mathbf{H} + O(g^{-4}\|\operatorname{vech}\mathbf{H}\|^2)] \\ &= b^{-1}g^{-2}\mathbf{I} - ab^{-2}g^{-4}\mathbf{H} + o(g^{-6}\|\operatorname{vech}\mathbf{H}\|^2).\end{aligned}$$

The determinant can be expanded using a result from Miller (1987, p.7, 14 – 15)

$$|\mathbf{I} + \mathbf{A}| = 1 + \sum_{i=1}^{\operatorname{rank}(\mathbf{A})} \sigma_i$$

where $\sigma_i = i^{-1}\sum_{j=1}^{i}(-1)^{j+1}\sigma_{i-j}\operatorname{tr}\mathbf{A}^j$ and $\sigma_0 = 1$. Then $|\mathbf{I}+\mathbf{A}| = 1+\operatorname{tr}\mathbf{A}+O(\|\operatorname{vech}\mathbf{A}\|^2)$ which means that

$$|\mathbf{I} + \mathbf{A}|^{-1/2} = 1 - \tfrac{1}{2}\operatorname{tr}\mathbf{A} + O(\|\operatorname{vech}\mathbf{A}\|^2)$$

and so

$$\begin{aligned}|a\mathbf{H} + b\mathbf{G}|^{-1/2} &= |a\mathbf{H} + bg^2\mathbf{I}|^{-1/2} \\ &= |bg^2(ab^{-1}g^{-2}\mathbf{H} + \mathbf{I})|^{-1/2} \\ &= b^{-d/2}g^{-d}[1 - \tfrac{1}{2}ab^{-1}g^{-2}\operatorname{tr}\mathbf{H} + O(g^{-4}\|\operatorname{vech}\mathbf{H}\|^2)] \\ &= b^{-d/2}g^{-d} - \tfrac{1}{2}ab^{-(d+2)/2}g^{-d-2}\operatorname{tr}\mathbf{H} + O(g^{-d-4}\|\operatorname{vech}\mathbf{H}\|^2).\end{aligned}$$

Combining these two expansions we have

$$\begin{aligned}&|a\mathbf{H} + b\mathbf{G}|^{-1/2}(a\mathbf{H} + b\mathbf{G})^{-1} \\ &= [b^{-d/2}g^{-d} - \tfrac{1}{2}ab^{-(d+2)/2}g^{-d-2}\operatorname{tr}\mathbf{H} + O(g^{-d-4}\|\operatorname{vech}\mathbf{H}\|^2)] \\ &\quad \times [b^{-1}g^{-2}\mathbf{I} - ab^{-2}g^{-4}\mathbf{H} + O(g^{-d-4}\|\operatorname{vech}\mathbf{H}\|^2)] \\ &= b^{-(d+2)/2}g^{-d-2}\mathbf{I} - ab^{-(d+4)/2}g^{-d-4}\mathbf{H} - \tfrac{1}{2}ab^{-(d+4)/2}g^{-d-4}(\operatorname{tr}\mathbf{H})\mathbf{I} \\ &\quad + O(g^{-d-6}\|\operatorname{vech}\mathbf{H}\|^2) \\ &= b^{-(d+2)/2}g^{-d-2}\mathbf{I} - \tfrac{1}{2}ab^{-(d+4)/2}g^{-d-4}[2\mathbf{H} + (\operatorname{tr}\mathbf{H})\mathbf{I}] + O(g^{-d-6}\|\operatorname{vech}\mathbf{H}\|^2).\end{aligned}$$

The derivative of C_1 becomes

$$\begin{aligned}D_{\mathbf{H}}C_1 &= -(4\pi)^{-d/2}[\tfrac{1}{2}g^{-d-2}\mathbf{D}_d^T\operatorname{vec}\mathbf{I}_d - \tfrac{1}{4}g^{-d-4}(2\mathbf{D}_d^T\operatorname{vec}\mathbf{H} + (\operatorname{tr}\mathbf{H})\mathbf{D}_d^T\operatorname{vec}\mathbf{I}_d)] \\ &\quad + (4\pi)^{-d/2}[\tfrac{1}{2}g^{-d-2}\mathbf{D}_d^T\operatorname{vec}\mathbf{I}_d - \tfrac{1}{8}g^{-d-4}(2\mathbf{D}_d^T\operatorname{vec}\mathbf{H} + (\operatorname{tr}\mathbf{H})\mathbf{D}_d^T\operatorname{vec}\mathbf{I}_d)] \\ &\quad + O(g^{-d-6}\|\operatorname{vech}\mathbf{H}\|^2) \\ &= \tfrac{1}{8}(4\pi)^{-d/2}g^{-d-4}[2\mathbf{D}_d^T\operatorname{vec}\mathbf{H} + (\operatorname{tr}\mathbf{H})\mathbf{D}_d^T\operatorname{vec}\mathbf{I}_d] + O(g^{-d-6}\|\operatorname{vech}\mathbf{H}\|^2).\end{aligned}$$

The derivative of $\frac{1}{4}g^2 \operatorname{tr}(\mathbf{H}^2\mathbf{\Theta}_6) + O(g^4\|\operatorname{vech}\mathbf{H}\|^2)$ is $\frac{1}{2}g^2\mathbf{D}_d^T \operatorname{vec}(\mathbf{\Theta}_6\mathbf{H}) + O(g^4 \operatorname{vech}\mathbf{H})$. Combining these two derivatives and then interchanging the expectation and derivative operators, we have

$$\begin{aligned}
&\mathbb{E}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\mathbf{H}_{\mathrm{AMISE}})] \\
&\quad = \tfrac{1}{2}g^2\mathbf{D}_d^T \operatorname{vec}(\mathbf{\Theta}_6\mathbf{H}_{\mathrm{AMISE}}) + \tfrac{1}{8}(4\pi)^{-d/2}n^{-1}g^{-d-4}[2\mathbf{D}_d^T \operatorname{vec}\mathbf{H}_{\mathrm{AMISE}} \\
&\qquad + (\operatorname{tr}\mathbf{H}_{\mathrm{AMISE}})\mathbf{D}_d^T \operatorname{vec}\mathbf{I}_d] + o(g^2 + n^{-1}g^{-d-4}) \operatorname{vech}\mathbf{H}_{\mathrm{AMISE}}.
\end{aligned}$$

As $D_{\mathbf{H}}^2\mathrm{AMISE}(\mathbf{H}_{\mathrm{AMISE}}) = O(\mathbf{J}_{d'})$, the result for ABias$'$ follows immediately. □

Lemma 13. *Assume A1 – A4 from the* AMSE$'$ *Lemma (Lemma 11) and S1 – S4 from Lemma 12. Then*

$$\mathrm{AVar}'(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{SCV}}; g) = O(\mathbf{J}_{d'}(n^{-2}g^{-d-8} + n^{-1}))(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T\mathbf{H}_{\mathrm{AMISE}}).$$

Proof. To find $\mathrm{AVar}'(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{SCV}}; g)$, we first find $\mathrm{Var}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\mathbf{H}_{\mathrm{AMISE}})]$:

$$\begin{aligned}
&\mathrm{Var}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\mathbf{H})] \\
&\quad = \mathrm{Var}[D_{\mathbf{H}}\mathrm{SCV}(\mathbf{H})] \\
&\quad = n^{-4}\operatorname{Var}\left[\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n} D_{\mathbf{H}}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_i - \boldsymbol{X}_j)\right] \\
&\quad = n^{-4}\operatorname{Var}\left[\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n} (\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_i - \boldsymbol{X}_j)\right]
\end{aligned}$$

where $\boldsymbol{\varphi}_{\mathbf{A}}(\cdot)$ was defined in Equation (3.4). As $\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}}$ is a symmetric function, the variance simplifies to

$$\begin{aligned}
&\mathrm{Var}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\mathbf{H})] \\
&= 2n^{-2}\operatorname{Var}[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)] \\
&\quad + 4n^{-1}\operatorname{Cov}[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2), (\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)]. \quad (3.21)
\end{aligned}$$

The first term of $\mathrm{Var}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\mathbf{H})]$ is

$$\begin{aligned}
&\operatorname{Var}[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)] \\
&= \mathbb{E}\left\{[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\,[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]^T\right\} \\
&\quad - [\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\,[\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]^T.
\end{aligned}$$

From Lemma 12,

$$\begin{aligned}
&\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2) \\
&\quad = D_{\mathbf{H}}[\mathbb{E}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)] \\
&\quad = D_{\mathbf{H}}\left[\tfrac{1}{4}\int_{\mathbb{R}^d} \operatorname{tr}(\mathbf{H}^2(D^2)^2 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} + o(\|\operatorname{vech}\mathbf{H}\|^2)\right] \\
&\quad = \tfrac{1}{2}\int_{\mathbb{R}^d} \mathbf{D}_d^T \operatorname{vec}(\mathbf{H}(D^2)^2 f(\boldsymbol{y}))f(\boldsymbol{y})\ d\boldsymbol{y} + o(\operatorname{vech}\mathbf{H}).
\end{aligned}$$

To further simplify this expression, we expand $\phi_{a\mathbf{H}+b\mathbf{G}}$ about $\phi_{b\mathbf{G}}$:

$$\begin{aligned}
\phi_{a\mathbf{H}+b\mathbf{G}}(\boldsymbol{x}) &= (2\pi)^{-d/2}|a\mathbf{H} + b\mathbf{G}|^{-1/2}\exp\left[-\tfrac{1}{2}\boldsymbol{x}^T(a\mathbf{H} + b\mathbf{G})^{-1}\boldsymbol{x}\right] \\
&= (2\pi)^{-d/2}|b\mathbf{G}|^{-1/2}[1 + O(\|\operatorname{vech}\mathbf{G}^{-1}\mathbf{H}\|)] \\
&\quad \times \exp\left\{-\tfrac{1}{2}\boldsymbol{x}^T(b\mathbf{G})^{-1}\boldsymbol{x}[1 + O(\|\operatorname{vech}\mathbf{G}^{-1}\mathbf{H}\|)]\right\} \\
&= \phi_{b\mathbf{G}}(\boldsymbol{x})[1 + O(\|\operatorname{vech}\mathbf{G}^{-1}\mathbf{H}\|)]
\end{aligned}$$

and then

$$\begin{aligned}
\boldsymbol{\varphi}_{a\mathbf{H}+b\mathbf{G}}(\boldsymbol{x}) = \phi_{b\mathbf{G}}(\boldsymbol{x})\mathbf{D}_d^T \operatorname{vec}[&(b\mathbf{G})^{-1}\boldsymbol{x}\boldsymbol{x}^T(b\mathbf{G})^{-1} - (b\mathbf{G})^{-1} - (b\mathbf{G})^{-2}\boldsymbol{x}\boldsymbol{x}^T(a\mathbf{H})(b\mathbf{G})^{-1} \\
&+ (b\mathbf{G})^{-1}\boldsymbol{x}\boldsymbol{x}^T(a\mathbf{H})(b\mathbf{G})^{-2} + (b\mathbf{G})^{-1}(a\mathbf{H})(b\mathbf{G})^{-1} + O(\operatorname{vech}\mathbf{G}^{-3}\mathbf{H}^2)]
\end{aligned}$$

which means that

$$\begin{aligned}
&(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{x}) \\
&\quad = -\phi_{2\mathbf{G}}(\boldsymbol{x})\mathbf{D}_d^T \operatorname{vec}[\tfrac{1}{8}\mathbf{G}^{-2}\boldsymbol{x}\boldsymbol{x}^T\mathbf{H}\mathbf{G}^{-1} + \tfrac{1}{8}\mathbf{G}^{-1}\boldsymbol{x}\boldsymbol{x}^T\mathbf{H}\mathbf{G}^{-2} - \tfrac{1}{4}\mathbf{G}^{-1}\mathbf{H}\mathbf{G}^{-1}] \\
&\qquad \times [1 + O(\|\operatorname{vech}\mathbf{G}^{-3}\mathbf{H}^2\|)].
\end{aligned}$$

As

$$\begin{aligned}
&\mathbb{E}\left\{[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)][(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]^T\right\} \\
&= |2\mathbf{G}|^{-1/2}\int_{\mathbb{R}^{2d}} \phi_{\mathbf{I}}(\boldsymbol{w})^2\mathbf{D}_d^T \operatorname{vec}(\tfrac{1}{4}g^{-4}\mathbf{H} - \tfrac{1}{2}g^{-4}\boldsymbol{w}\boldsymbol{w}^T\mathbf{H}) \operatorname{vec}^T(\tfrac{1}{4}g^{-4}\mathbf{H} - \tfrac{1}{2}g^{-4}\boldsymbol{w}\boldsymbol{w}^T\mathbf{H}) \\
&\quad \times \mathbf{D}_d[f(\boldsymbol{y})^2 + O(g^2)]\ d\boldsymbol{w}d\boldsymbol{y} \\
&= 2^{-d/2}g^{-d-8}[R(f) + O(g^2)]\int_{\mathbb{R}^d} \phi_{\mathbf{I}}(\boldsymbol{w})^2\mathbf{D}_d^T[\tfrac{1}{4}\operatorname{vec}\mathbf{H} - \tfrac{1}{2}(\mathbf{I}\otimes\mathbf{H})\operatorname{vec}(\boldsymbol{w}\boldsymbol{w}^T)] \\
&\quad \times [\tfrac{1}{4}\operatorname{vec}^T\mathbf{H} - \tfrac{1}{2}(\mathbf{I}\otimes\mathbf{H})\operatorname{vec}^T(\boldsymbol{w}\boldsymbol{w}^T)]\mathbf{D}_d\ d\boldsymbol{w} \\
&= O(\mathbf{J})_{d'}g^{-d-8})(\operatorname{vech}\mathbf{H})(\operatorname{vech}^T\mathbf{H})
\end{aligned}$$

and $\mathbb{E}\left\{[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\right\} = O(\mathbf{I}_{d'})\operatorname{vech}\mathbf{H}$ then

$$\operatorname{Var}[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)] = O(\mathbf{J}_{d'}g^{-d-8})(\operatorname{vech}\mathbf{H})(\operatorname{vech}^T\mathbf{H}). \tag{3.22}$$

We now turn our attention to the second term of $\operatorname{Var}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\mathbf{H})]$:

$$\begin{aligned}
&\operatorname{Cov}[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2), (\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)] \\
&\quad = \mathbb{E}\left\{[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\,[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)]^T\right\} \\
&\qquad - [\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)]\,[\mathbb{E}(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)]^T
\end{aligned}$$

We already have values for the second part of this expression. For the first part, we can follow a similar procedure in Lemma 12 to find that

$$\mathbb{E}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X} - \boldsymbol{y}) = \tfrac{1}{4}\operatorname{tr}(\mathbf{H}^2(D^2)^2 f(\boldsymbol{y}))[1 + o(1)]$$

and so

$$D_{\mathbf{H}}\,\mathbb{E}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X} - \boldsymbol{y}) = \tfrac{1}{2}\mathbf{D}_d^T \operatorname{vec}(\mathbf{H}(D^2)^2 f(\boldsymbol{y}))[\mathbf{I}_{d'} + o(\mathbf{I}_{d'})].$$

Then, swapping the order of expectation and differentiation,

$$\begin{aligned}
&\mathbb{E}\left\{[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2)][(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)]^T\right\} \\
&= \int_{\mathbb{R}^{3d}} D_{\mathbf{H}}(\phi_{2\mathbf{H}+2\mathbf{G}} - \phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{x} - \boldsymbol{y}) \\
&\quad \times [D_{\mathbf{H}}(\phi_{2\mathbf{H}+2\mathbf{G}} - \phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{y} - \boldsymbol{z})]^T f(\boldsymbol{x}) f(\boldsymbol{y}) f(\boldsymbol{z})\; d\boldsymbol{x} d\boldsymbol{y} d\boldsymbol{z} \\
&= \int_{\mathbb{R}^d} D_{\mathbf{H}}\,\mathbb{E}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{y}) \\
&\quad \times [D_{\mathbf{H}}\,\mathbb{E}(\phi_{2\mathbf{H}+2\mathbf{G}} - 2\phi_{\mathbf{H}+2\mathbf{G}} + \phi_{2\mathbf{G}})(\boldsymbol{y} - \boldsymbol{X}_3)] f(\boldsymbol{y})\; d\boldsymbol{y} \\
&= \tfrac{1}{4}\int_{\mathbb{R}^d} \mathbf{D}_d^T \operatorname{vec}(\mathbf{H}(D^2)^2 f(\boldsymbol{y}))\operatorname{vec}^T(\mathbf{H}(D^2)^2 f(\boldsymbol{y}))\mathbf{D}_d f(\boldsymbol{y}) d\boldsymbol{y}\; [\mathbf{I}_{d'} + o(\mathbf{I}_{d'})] \\
&= O(\mathbf{J}_{d'})(\operatorname{vec}\mathbf{H})(\operatorname{vec}^T \mathbf{H}).
\end{aligned}$$

Thus

$$\begin{aligned}
&\operatorname{Cov}[(\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_1 - \boldsymbol{X}_2), (\boldsymbol{\varphi}_{2\mathbf{H}+2\mathbf{G}} - \boldsymbol{\varphi}_{\mathbf{H}+2\mathbf{G}})(\boldsymbol{X}_2 - \boldsymbol{X}_3)] \\
&\quad = O(\mathbf{J}_{d'})(\operatorname{vec}\mathbf{H})(\operatorname{vec}^T \mathbf{H}).
\end{aligned} \tag{3.23}$$

If we substitute Equations (3.22) and (3.23) into Equation (3.21):

$$\begin{aligned}
&\operatorname{Var}[D_{\mathbf{H}}(\mathrm{SCV} - \mathrm{AMISE}')(\hat{\mathbf{H}}_{\mathrm{SCV}}; g)] \\
&\quad = O(\mathbf{J}_{d'}(n^{-2}g^{-d-8} + n^{-1}))(\operatorname{vech}\mathbf{H}_{\mathrm{AMISE}})(\operatorname{vech}^T \mathbf{H}_{\mathrm{AMISE}}).
\end{aligned}$$

□

We are in a position now to state the main theoretical result of this section; that is to find an explicit expression for $g_0 = \operatorname*{argmin}_{g>0} \operatorname{tr}\mathrm{AMSE}'(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{SCV}}; g)$.

Theorem 4. *Under the conditions of Lemmas 12 and 13, the pilot bandwidth which minimises the trace of* $\operatorname{AMSE}'(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{SCV}}; g)$ *for* $d > 1$ *is*

$$g_0 = \left\{ \frac{2(d+4)\boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_2}}{\big[-(d+2)\boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_1} + C_{\mu_0}^{1/2} \big] n} \right\}^{1/(d+6)}$$

where

$$\begin{aligned}
C_{\mu_0} &= (d+2)^2 (\boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_1})^2 + 8(d+4)(\boldsymbol{C}_{\mu_1}^T \boldsymbol{C}_{\mu_1})(\boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_2}) \\
\boldsymbol{C}_{\mu_1} &= \tfrac{1}{2} n^{2/(d+4)} \mathbf{D}_d^T \operatorname{vec}(\boldsymbol{\Theta}_6 \mathbf{H}_{\mathrm{AMISE}}) \\
\boldsymbol{C}_{\mu_2} &= \tfrac{1}{8} (4\pi)^{-d/2} n^{2/(d+4)} [2\mathbf{D}_d^T \operatorname{vec} \mathbf{H}_{\mathrm{AMISE}} + (\operatorname{tr} \mathbf{H}_{\mathrm{AMISE}}) \mathbf{D}_d^T \operatorname{vec} \mathbf{I}_d].
\end{aligned}$$

Note that the expressions $C_{\mu_0}, \boldsymbol{C}_{\mu_1}$ *and* $\boldsymbol{C}_{\mu_2}$ *are constant with respect to* n*.*

Proof. To find g_0 we need to minimise $\operatorname{tr} \operatorname{AMSE}'(\operatorname{vec}\hat{\mathbf{H}}; g)$. From Lemma 12,

$$\begin{aligned}
&[\operatorname{ABias}'(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{SCV}}; g)]^T [\operatorname{ABias}'(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{SCV}}; g)] \\
&\quad = n^{-2/(d+4)} (g^2 \boldsymbol{C}_{\mu_1} + n^{-1} g^{-d-4} \boldsymbol{C}_{\mu_2})^T n^{-2/(d+4)} (g^2 \boldsymbol{C}_{\mu_1} + n^{-1} g^{-d-4} \boldsymbol{C}_{\mu_2}) \\
&\quad = n^{-4/(d+4)} [g^4 \boldsymbol{C}_{\mu_1}^T \boldsymbol{C}_{\mu_1} + 2n^{-1} g^{-d-2} \boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_1} + n^{-2} g^{-2d-8} \boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_2}]. \qquad (3.24)
\end{aligned}$$

From Lemma 13,

$$\operatorname{AVar}'(\operatorname{vech}\hat{\mathbf{H}}_{\mathrm{SCV}}; g) = O(n^{-2} g^{-d-8}) \|\operatorname{vech} \mathbf{H}_{\mathrm{AMISE}}\|^2 = O(n^{-4/(d+4)} n^{-2} g^{-d-8}).$$

Since the variance is asymptotically negligible compared to the squared bias which is order $n^{-4/(d+4)} n^{-2} g^{-2d-8}$, we can attempt to annihilate this squared bias, much like Jones & Kappenman (1992). The discriminant of the quadratic in Equation (3.24) is $4(\boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_1})^2 - 4(\boldsymbol{C}_{\mu_1}^T \boldsymbol{C}_{\mu_1})(\boldsymbol{C}_{\mu_2}^T \boldsymbol{C}_{\mu_2})$. Let $\boldsymbol{a} = (a_1, a_2, \ldots, a_d), \boldsymbol{b} = (b_1, b_2, \ldots, b_d)$ then

$$\begin{aligned}
(\boldsymbol{a}^T \boldsymbol{b})^2 - (\boldsymbol{a}^T \boldsymbol{a})(\boldsymbol{b}^T \boldsymbol{b}) &= \sum_{i=1}^d \sum_{j=1}^d a_i b_i a_j b_j - \sum_{i=1}^d \sum_{j=1}^d a_i^2 b_i^2 \\
&= \sum_{i=1}^d a_i^2 b_i^2 + \sum_{i=1}^d \sum_{\substack{j=1 \\ j \neq i}}^d a_i b_i a_j b_j - \sum_{i=1}^d a_i^2 b_i^2 - \sum_{i=1}^d \sum_{\substack{j=1 \\ j \neq i}}^d a_i^2 b_j^2 \\
&= \sum_{i=1}^d \sum_{j>i} (-a_i^2 b_j^2 + 2a_i b_i a_j b_j - a_j^2 b_i^2) \\
&= -\sum_{i=1}^d \sum_{j>i} (a_i b_j - a_j b_i)^2 \\
&\leq 0
\end{aligned}$$

with equality holding iff $a_i = a$ for all i and $b_j = b$ for all j. Thus equality holds in general only for $d = 1$ and so for the multivariate case, the discriminant is negative

(with probability 1), and we can only minimise this squared bias rather than annihilating its leading terms. Differentiating Equation (3.24) with respect to g, dividing by $2g^3$, multiplying by $n^{4/(d+4)}$ and setting to zero we have

$$2\boldsymbol{C}_{\mu_1}^T\boldsymbol{C}_{\mu_1} - (d+2)n^{-1}g^{-d-6}\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_1} - (d+4)n^{-2}g^{-2d-12}\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_2} = 0$$

which is a quadratic in $n^{-1}g^{-d-6}$ and has solution

$$g_0 = \left\{ \frac{2(d+4)\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_2}}{\big[-(d+2)\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_1} + C_{\mu_0}^{1/2}\big]n} \right\}^{1/(d+6)}$$

where $C_{\mu_0} = (d+2)^2(\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_1})^2 + 8(d+4)(\boldsymbol{C}_{\mu_1}^T\boldsymbol{C}_{\mu_1})(\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_2})$. This value of g is real-valued as $\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_1} < 0$ as shown by the following. The (i,j) element of $\boldsymbol{\Theta}_6$ is

$$[\boldsymbol{\Theta}_6]_{ij} = \sum_{k=1}^{d}\sum_{\ell=1}^{d} \psi_{\boldsymbol{e}_i+2\boldsymbol{e}_k+2\boldsymbol{e}_\ell+\boldsymbol{e}_j}.$$

The elements on the main diagonal of $\boldsymbol{\Theta}_6$ are

$$[\boldsymbol{\Theta}_6]_{ii} = \sum_{k=1}^{d}\sum_{\ell=1}^{d} \psi_{2\boldsymbol{e}_i+2\boldsymbol{e}_k+2\boldsymbol{e}_\ell} = -\sum_{k=1}^{d}\sum_{\ell=1}^{d} R(f^{(\boldsymbol{e}_i+\boldsymbol{e}_k+\boldsymbol{e}_\ell)}) < 0$$

and so $\operatorname{tr}\boldsymbol{\Theta}_6 < 0$. For a quadratic form, $\operatorname{sgn}(\boldsymbol{y}^T\mathbf{A}\boldsymbol{y}) = \operatorname{sgn}(\operatorname{tr}\mathbf{A})$, so

$$\operatorname{sgn}(\boldsymbol{C}_{\mu_2}^T\boldsymbol{C}_{\mu_1}) = \operatorname{sgn}(\operatorname{tr}(\mathbf{I}_d \otimes \boldsymbol{\Theta}_6)) = \operatorname{sgn}((\operatorname{tr}\mathbf{I}_d)(\operatorname{tr}\boldsymbol{\Theta}_6)) = -1.$$

□

The relative rate of convergence of the SCV selector is an immediate consequence of Theorem 4 and the AMSE′ Lemma i.e. if $\operatorname{tr}\operatorname{MSE}(\operatorname{vech}\hat{\mathbf{H}}) = O(n^{-2\alpha}\|\operatorname{vech}\mathbf{H}_{\text{AMISE}}\|^2)$ then $\hat{\mathbf{H}}$ has relative rate of convergence to $\mathbf{H}_{\text{AMISE}}$ of $n^{-\alpha}$.

Theorem 5. *Under the conditions of Lemmas 12 and 13, for $d > 1$ the relative rate of convergence of $\hat{\mathbf{H}}_{\text{SCV}}$ to $\mathbf{H}_{\text{AMISE}}$ is $n^{-2/(d+6)}$.*

Proof. From Theorem 4, the optimal rate of the trace of the AMSE′ is

$$\begin{aligned}
\operatorname{tr}\operatorname{AMSE}'(\operatorname{vech}\hat{\mathbf{H}}_{\text{SCV}}; g_0) &= O(n^{-2}g_0^{-d-8}\|\operatorname{vech}\mathbf{H}_{\text{AMISE}}\|^2 + g_0^4\|\operatorname{vech}\mathbf{H}_{\text{AMISE}}\|^2) \\
&= O((n^{-(d+4)/(d+6)} + n^{-4/(d+6)})\|\operatorname{vech}\mathbf{H}_{\text{AMISE}}\|^2) \\
&= O(n^{-4/(d+6)}\|\operatorname{vech}\mathbf{H}_{\text{AMISE}}\|^2)
\end{aligned}$$

as $g_0 = O(n^{-1/(d+6)})$. The rate of convergence is thus $n^{-2/(d+6)}$. □

This is the same rate as the plug-in selector with a SAMSE pilot bandwidth from Section 2.2.2, which is not unexpected as both of these selectors use a single pilot bandwidth except for the univariate SCV selector whose construction is slightly different and so has a different convergence rate, as shown in Jones et al. (1991). The SCV rate is split into two cases because for $d = 1$, where bias annihilation is possible and for $d > 1$, where only bias minimisation is possible. We have now determined the convergence rates for all the fixed full bandwidth selectors we will consider. The results are summarised in Table 3.2. This table shows that for all the selectors, the performance decreases with increasing dimension. The AMSE plug-in selectors are always the fastest. For $d \leq 3$, the BCV and LSCV selectors are slower than the SCV and SAMSE plug-in selectors. This swaps over for $d > 3$. Also important to note is that the discrepancy between $\mathbf{H}_{\text{AMISE}}$ and $\mathbf{H}_{\text{MISE}}$ is dominated by the rate for any selector, except for the AMSE plug-in, LSCV and BCV selectors for $d > 3$. This means, apart from these exceptions, the convergence rates to $\mathbf{H}_{\text{AMISE}}$ and to $\mathbf{H}_{\text{MISE}}$ are the same.

	Convergence rate to $\mathbf{H}_{\text{AMISE}}$						
Selector	d	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
$\hat{\mathbf{H}}_{\text{PI,AMSE}}$ (diagonal)	$n^{-\min(8,d+4)/(2d+12)}$	$n^{-5/14}$	$n^{-3/8}$	$n^{-7/18}$	$n^{-2/5}$	$n^{-4/11}$	$n^{-1/3}$
$\hat{\mathbf{H}}_{\text{PI,AMSE}}$	$n^{-4/(d+12)}$	$n^{-4/13}$	$n^{-2/7}$	$n^{-4/15}$	$n^{-1/4}$	$n^{-4/17}$	$n^{-2/9}$
$\hat{\mathbf{H}}_{\text{PI,SAMSE}}$	$n^{-2/(d+6)}$	$n^{-2/7}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$	$n^{-2/11}$	$n^{-1/6}$
$\hat{\mathbf{H}}_{\text{LSCV}}$	$n^{-\min(d,4)/(2d+8)}$	$n^{-1/10}$	$n^{-1/6}$	$n^{-3/14}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$
$\hat{\mathbf{H}}_{\text{BCV1}}, \hat{\mathbf{H}}_{\text{BCV2}}$	$n^{-\min(d,4)/(2d+8)}$	$n^{-1/10}$	$n^{-1/6}$	$n^{-3/14}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$
$\hat{\mathbf{H}}_{\text{SCV}}$	$\begin{cases} n^{-5/14} & d=1 \\ n^{-2/(d+6)} & d>1 \end{cases}$	$n^{-5/14}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$	$n^{-2/11}$	$n^{-1/6}$
$\mathbf{H}_{\text{AMISE}} - \mathbf{H}_{\text{MISE}}$	$n^{-2/(d+4)}$	$n^{-2/5}$	$n^{-1/3}$	$n^{-2/7}$	$n^{-1/4}$	$n^{-2/9}$	$n^{-1/5}$

Table 3.2: Comparison of convergence rates – all selectors

3.4.2 Estimating the optimal pilot bandwidth

To apply Theorem 4 (i.e. to estimate g_0), we need to estimate $\boldsymbol{C}_{\mu_1}$ and $\boldsymbol{C}_{\mu_2}$. We will use SAMSE plug-in methods from Chapter 2. The $\psi_{\boldsymbol{r}}$ functionals can be used to derive an explicit expression for $\boldsymbol{\Theta}_6$. For the bivariate case,

$$\boldsymbol{\Theta}_6 = \begin{bmatrix} \psi_{60} + 2\psi_{42} + \psi_{24} & \psi_{51} + 2\psi_{33} + \psi_{15} \\ \psi_{51} + 2\psi_{33} + \psi_{15} & \psi_{42} + 2\psi_{24} + \psi_{06} \end{bmatrix}.$$

The plug-in estimator $\hat{\boldsymbol{\Theta}}_6(g'_{6,\text{SAMSE}})$ is constructed by replacing $\psi_{\boldsymbol{r}}$ with $\hat{\psi}_{\boldsymbol{r}}(g'_{6,\text{SAMSE}})$. Since we now are estimating g_0 we introduce estimation error. The following lemma states that $\hat{g}_0$ is relatively consistent for g_0. Since the SCV rate from Theorem 5 comes about due to a bias minimisation computation then the consistency of $\hat{g}_0$ guarantees that this rate remains valid when g_0 is replaced by its estimate.

Lemma 14. *Let $\hat{\boldsymbol{C}}_{\mu_1}, \hat{\boldsymbol{C}}_{\mu_2}$ be plug-in estimators of $\boldsymbol{C}_{\mu_1}, \boldsymbol{C}_{\mu_2}$ i.e.*

$$\begin{aligned}\hat{\boldsymbol{C}}_{\mu_1} &= \tfrac{1}{2}\mathbf{D}_d^T \operatorname{vec}(\hat{\boldsymbol{\Theta}}_6 \hat{\mathbf{H}}_{\mathrm{PI}}) \\ \hat{\boldsymbol{C}}_{\mu_2} &= \tfrac{3}{8}(4\pi)^{-d/2}[2\mathbf{D}_d^T \operatorname{vec} \hat{\mathbf{H}}_{\mathrm{PI}} + (\operatorname{tr} \hat{\mathbf{H}}_{\mathrm{PI}})\mathbf{D}_d^T \operatorname{vec} \mathbf{I}_d]\end{aligned}$$

where $\hat{\boldsymbol{\Theta}}_6$ and $\hat{\mathbf{H}}_{\mathrm{PI}}$ are SAMSE plug-in estimates of $\boldsymbol{\Theta}_6$ and $\mathbf{H}_{\mathrm{AMISE}}$. Let $\hat{g}_0$ be constructed by replacing $\boldsymbol{C}_{\mu_1}$ and $\boldsymbol{C}_{\mu_2}$ in g_0 by their plug-in estimates. Assume S1 – S4 from Lemma 12 then the relative rate of convergence of $\hat{g}_0$ to g_0 is $n^{-2/(d+8)}$.

Proof. Similar to the proof for Lemma 6, we start with

$$\frac{\hat{g}_0 - g_0}{g_0} = O_p(\hat{g}_0^{d+6} - g_0^{d+6})O(n)$$

since $g_0 = O(n^{-1/(d+6)})$. We now examine $\hat{g}_0^{d+6} - g_0^{d+6}$:

$$\hat{g}_0^{d+6} - g_0^{d+6} = O_p\left(\frac{\hat{\boldsymbol{C}}_{\mu_1}^T \hat{\boldsymbol{C}}_{\mu_2}}{\hat{\boldsymbol{C}}_{\mu_1}^T \hat{\boldsymbol{C}}_{\mu_1} n} - \frac{\boldsymbol{C}_{\mu_1}^T \boldsymbol{C}_{\mu_2}}{\boldsymbol{C}_{\mu_1}^T \boldsymbol{C}_{\mu_1} n}\right) = O_p\left(\frac{\hat{\boldsymbol{C}}_{\mu_1}^T (\hat{\boldsymbol{C}}_{\mu_2} - \hat{\boldsymbol{C}}_{\mu_1})}{\hat{\boldsymbol{C}}_{\mu_1}^T \hat{\boldsymbol{C}}_{\mu_1} n}\right).$$

From Section 2.3, using a sixth order SAMSE pilot $g'_{6,\mathrm{SAMSE}}$, then we know $\hat{\boldsymbol{\Theta}}_6 - \boldsymbol{\Theta}_6 = O_p(\mathbf{J}_d n^{-2/(d+8)})$ and $\hat{\boldsymbol{\Theta}}_6 = O_p(\mathbf{J}_d)$. The discrepancy between $\hat{\boldsymbol{C}}_{\mu_1}$ and $\boldsymbol{C}_{\mu_1}$ is

$$\begin{aligned}\hat{\boldsymbol{C}}_{\mu_1} - \boldsymbol{C}_{\mu_1} &= \tfrac{1}{2} n^{2/(d+4)} \mathbf{D}_d^T \operatorname{vec}(\hat{\boldsymbol{\Theta}}_6 \hat{\mathbf{H}}_{\mathrm{PI}} - \boldsymbol{\Theta}_6 \mathbf{H}_{\mathrm{AMISE}}) \\ &= \tfrac{1}{2} n^{2/(d+4)} \mathbf{D}_d^T \operatorname{vec}[(\hat{\boldsymbol{\Theta}}_6 - \boldsymbol{\Theta}_6)\mathbf{H}_{\mathrm{AMISE}} + \hat{\boldsymbol{\Theta}}_6(\hat{\mathbf{H}}_{\mathrm{PI}} - \mathbf{H}_{\mathrm{AMISE}})] \\ &= O_p((n^{-2/(d+8)} + n^{-2/(d+6)}) \operatorname{vech} \mathbf{J}_d) \\ &= O_p(n^{-2/(d+8)} \operatorname{vech} \mathbf{J}_d).\end{aligned}$$

The discrepancy between $\hat{\boldsymbol{C}}_{\mu_2}$ and $\boldsymbol{C}_{\mu_2}$ is

$$\begin{aligned}\hat{\boldsymbol{C}}_{\mu_2} - \boldsymbol{C}_{\mu_2} &= \tfrac{1}{8}(4\pi)^{-d/2}[2\mathbf{D}_d^T \operatorname{vec}(\hat{\mathbf{H}}_{\mathrm{PI}} - \mathbf{H}_{\mathrm{AMISE}}) + \operatorname{tr}(\hat{\mathbf{H}}_{\mathrm{PI}} - \mathbf{H}_{\mathrm{AMISE}})\mathbf{D}_d^T \operatorname{vec} \mathbf{I}_d] \\ &= O_p(n^{-2/(d+6)} \operatorname{vech} \mathbf{J}_d)\end{aligned}$$

which is dominated by $\hat{\boldsymbol{C}}_{\mu_1} - \boldsymbol{C}_{\mu_1}$. Moreover

$$\hat{\boldsymbol{C}}_{\mu_1}^T (\hat{\boldsymbol{C}}_{\mu_2} - \hat{\boldsymbol{C}}_{\mu_1}) = O_p(\hat{\boldsymbol{C}}_{\mu_1}^T (\hat{\boldsymbol{C}}_{\mu_2} - \boldsymbol{C}_{\mu_2} + \boldsymbol{C}_{\mu_1} - \hat{\boldsymbol{C}}_{\mu_1})) = O_p(n^{-2/(d+8)})$$

thus $\hat{g}_0^{d+6} - g_0^{d+6} = O_p(n^{-2/(d+8)} n^{-1})O(n) = O_p(n^{-2/(d+8)})$. □

This relative rate of convergence for the SCV pilot and its estimate is the same as for the SAMSE pilot and its estimate, as given in Lemma 6.

3.5 Practical performance of cross validation selectors

We state explicitly the algorithms we use for the various cross validation bandwidth selectors. These are then implemented in a simulation study and real data analysis.

3.5.1 Algorithms for cross validation bandwidth selectors

The algorithms for LSCV and BCV selectors are straightforward - all that is required is the numerically minimise the appropriate criterion. The SCV selector is more complex, as we need to select a pilot bandwidth using plug-in techniques. The SCV selector also requires the data to be pre-transformed (as described in Section 2.2.3) which yields a bandwidth matrix that is back-transformed to the original data scale.

Algorithm for LSCV bandwidth matrix selectors

1. Numerically minimise Equation (3.1) LSCV($\mathbf{H}$).

Algorithm for BCV bandwidth matrix selectors

1. Numerically minimise equation
   (a) Equation (3.9) BCV1($\mathbf{H}$) or
   (b) Equation (3.10) BCV2($\mathbf{H}$).

Algorithm for m-stage SCV bandwidth matrix selectors

1. Set $j_{\max} = 2m + 4$. Obtain normal reference estimates $\hat{\psi}_{\boldsymbol{r}}^{\mathrm{NR}}$ for $|\boldsymbol{r}| = j_{\max}$. Plug these estimates into the SAMSE pilot bandwidth $g'_{j_{\max}-2,\mathrm{SAMSE}}$.
2. For $j = j_{\max} - 2, j_{\max} - 4, \ldots, 6$:
   (a) Calculate kernel estimates of $\psi_{\boldsymbol{r}}$ functionals of order $j = |\boldsymbol{r}|$ using plug-in estimate of $g'_{j,\mathrm{SAMSE}}$.
   (b) Substitute $\hat{\psi}_{\boldsymbol{r}}$ estimates into Equation (2.10) to give plug-in estimate of pilot $g'_{j-2,\mathrm{SAMSE}}$.
3. Employ $g'_{6,\mathrm{SAMSE}}$ to produce kernel estimate $\hat{\boldsymbol{\Theta}}_6$.
4. Employ $g'_{4,\mathrm{SAMSE}}$ to produce kernel estimate $\hat{\boldsymbol{\Psi}}_4$. Plug this estimate into Equation (1.5) to give PI($\mathbf{H}$).
5. Numerically minimise PI($\mathbf{H}$) to obtain required plug-in bandwidth matrix $\hat{\mathbf{H}}_{\mathrm{PI}}$.
6. Use $\hat{\mathbf{H}}_{\mathrm{PI}}$ and $\hat{\boldsymbol{\Theta}}_6$ to form estimate $\hat{g}_0$ from Theorem 4.
7. Substitute $\hat{g}_0$ into Equation (3.20) to form SCV($\mathbf{H}$) and numerically minimise.

3.5.2 Simulation results for normal mixture densities

We perform a simulation study, similar to that of Section 2.5, to look at the performance of the following selectors:

- Sain et al. (1994)'s diagonal LSCV and diagonal BCV2 bandwidth matrix selectors (labelled DL and DB2 respectively)
- full LSCV bandwidth matrix selector (labelled L)
- full BCV1 and BCV2 bandwidth matrix selectors (labelled B1 and B2)
- full 1-stage SCV bandwidth matrix selector of Section 3.4 with pre-scaling and pre-sphering (labelled SC and SC*).

Each selector is run for two sample sizes, $n = 100$ and $n = 1000$, both for 400 trials (except for the B1, B2 and DB2 selectors which were run only for 100 trials for the larger sample size because they proved to be extremely computationally expensive). We employ a quasi-Newton (variable metric) method of numerical minimisation for the L and SC and SC* selectors. We use a constrained version for the B1 and B2 selectors. In the simulation study reported in Section 3.5, we did not encounter any significant computational difficulties for the L, DL, SC and SC* implementations. However, the implementation for the B1 and B2 selectors is extremely time consuming. Moreover, the constrained optimisation algorithm for B1, B2 and DB2 sometimes did not converge properly. The percentage rates for this non-convergence are contained in Table 3.3. For more details about the computer implementation in the `ks` library, see Appendix C.

		Target density					
Selector		A	B	C	D	E	F
B1	$n = 100$	0.0	0.0	1.0	0.0	2.0	0.0
	$n = 1000$	1.0	0.0	2.0	1.0	1.0	3.0
B2	$n = 100$	0.0	0.0	0.0	0.0	0.0	0.0
	$n = 1000$	0.0	0.0	0.0	0.0	1.0	0.0
DB2	$n = 100$	0.0	0.0	0.0	7.0	0.0	0.0
	$n = 1000$	0.0	0.0	11.0	8.0	0.0	1.0

Table 3.3: Percentage rates of non-convergence for biased cross validation selectors

In this section, we present the box plots of the log(ISE) in Figure 3.1 for $n = 100$ and in Figure 3.2 for $n = 1000$. (In Appendix B, Table B.5 contains the bandwidth matrix that attains the median ISE and Table B.6 contains the means and standard deviations of the ISE.) Like the results for the plug-in selectors, there is no uniformly best selector, the performance of a selector depends heavily on the shape of the target density. Overall, the median of the log (ISE) values is somewhat constant across all cross validation selectors, for

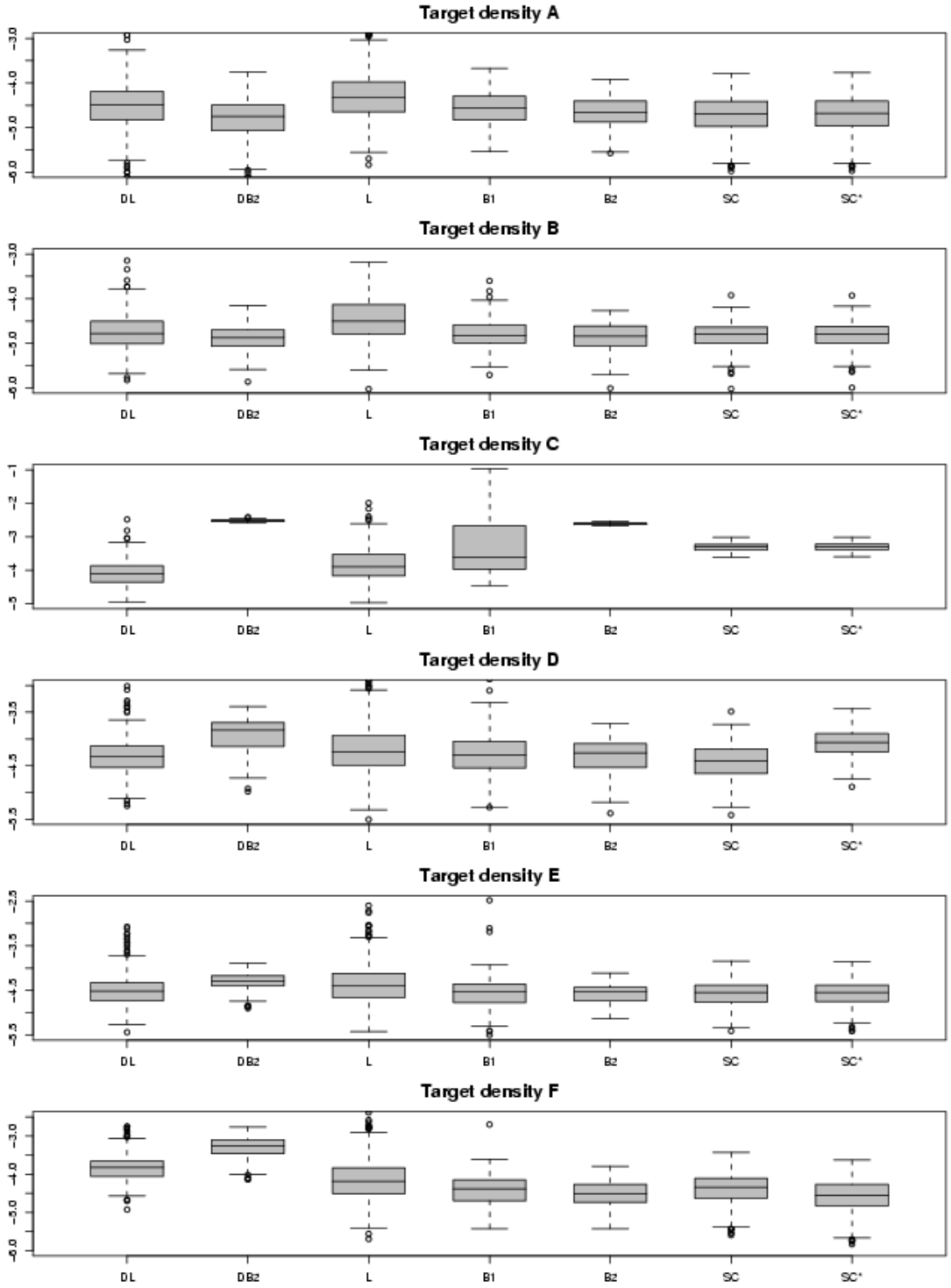

Figure 3.1: Box plots of log(ISE) for cross validation selectors, sample size $n = 100$.

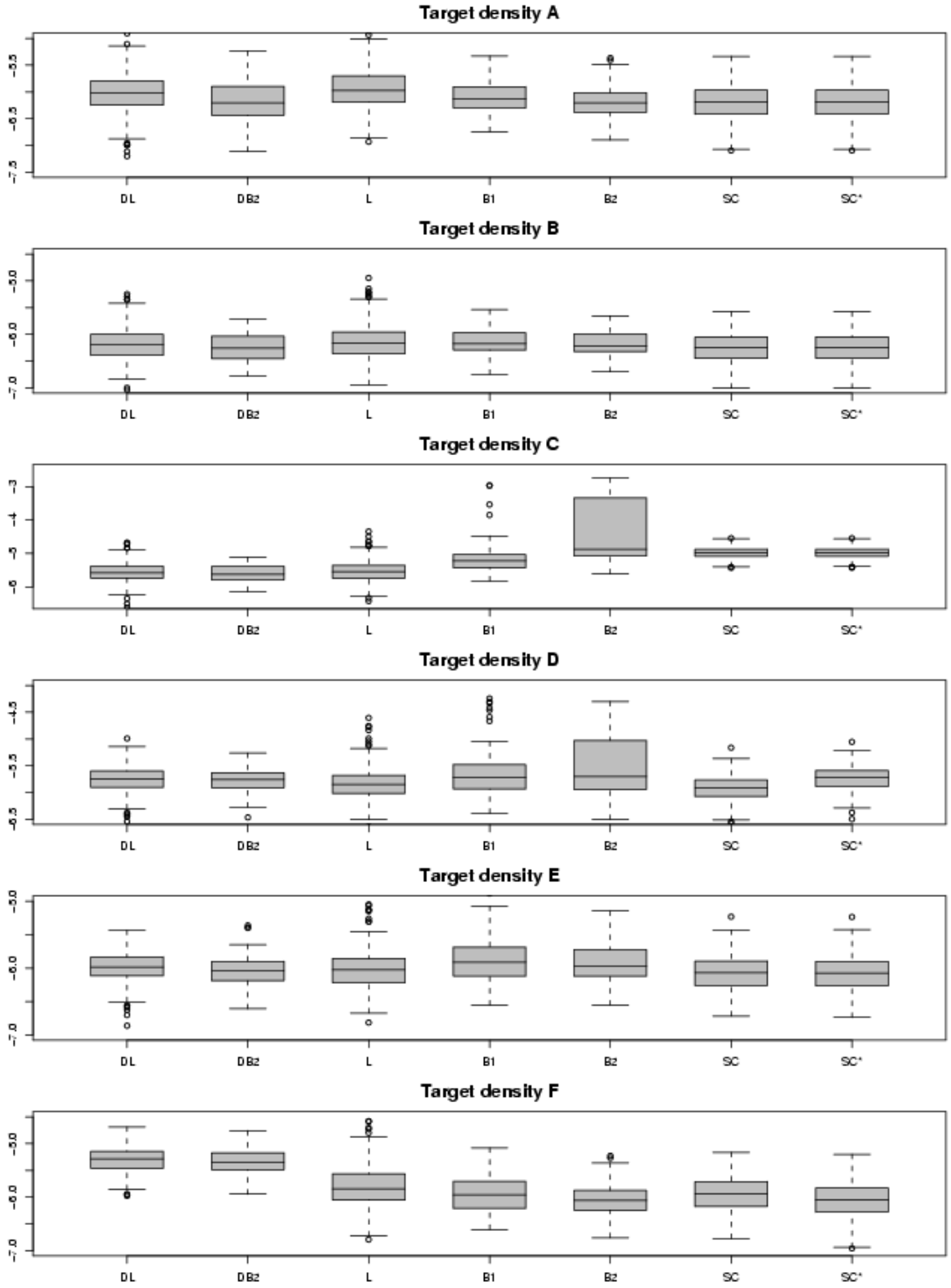

Figure 3.2: Box plots of log(ISE) for cross validation selectors, sample size $n = 1000$

a given test density except for density C. What varies more is the spread of the log(ISE). Looking at the box plots, the wide variability of L and DL selectors, as noted by various researchers in the past, again is shown here.

For target densities A and B, all the cross validation selectors have similar performance with perhaps a slight advantage to the SC and SC* selectors. For density C, the non-asymptotic nature of DL and L give them their better performance. The widely separated modes of this density tends to increase the bias of the other asymptotic selectors, with DB2 and B2 being particularly adversely affected. For the remaining densities D, E and F, the SC and SC* selectors perform the best overall. The structure of these latter densities is more intricate: it appears that using an independent pilot bandwidth assists in extracting more structure. The difference between pre-scaling and pre-sphering, i.e. between SC and SC*, is small. Only for density D can we see an advantage for pre-scaling. This density has two components, one with correlation zero and the other 0.7, which, when put together, have an overall correlation of about -0.58 so pre-sphering corrupts important structure of the data. This effect was similarly observed for plug-in selectors in Section 3.5.

It is important to note is that the diagonal selectors DL and DB2 from Sain et al. (1994) have good performance when compared to the full selectors when $\mathbf{H}_{\mathrm{MISE}}$ itself is a diagonal matrix (i.e. target densities A, B and C). Whereas for target densities D, E and F where $\mathbf{H}_{\mathrm{MISE}}$ is non-diagonal, these DL and DB2 selectors fare less well. From the simulation study in Sain et al. (1994), they recommend the DB2 selector over the SC selector. However this was because their implementation of SC was sub-optimal since it did not use an independent pilot bandwidth (it was set to be equal to the final bandwidth). From our simulation study, we see that the SC selector with an appropriately chosen pilot can have better performance than DB2.

3.5.3 Results for real data

We again turn our attention to the 'Old Faithful' geyser data to test the efficacy of the cross validation selectors on a real data set. The estimates of the bandwidth selectors are in Table 3.4. The contour plots for the corresponding kernel density estimates are in Figure 3.3. From the previous chapter, we saw that the pre-sphered full bandwidth selectors were better at capturing the structure of the data as they produced smoother, oblique contours that were aligned to the dataset rather than to the co-ordinate axes. Here, the L, DB2 and SC selectors produce contours, for the mode in the lower left, that are aligned to the axes, and for the main mode, contours that are wobbly. This wobbliness is more apparent for the B1 estimate. The DL selector did not converge for this data. This leaves B2 and SC* to give density estimates with noticeably oblique and smooth contours, though the B2 estimate is perhaps oversmoothed.

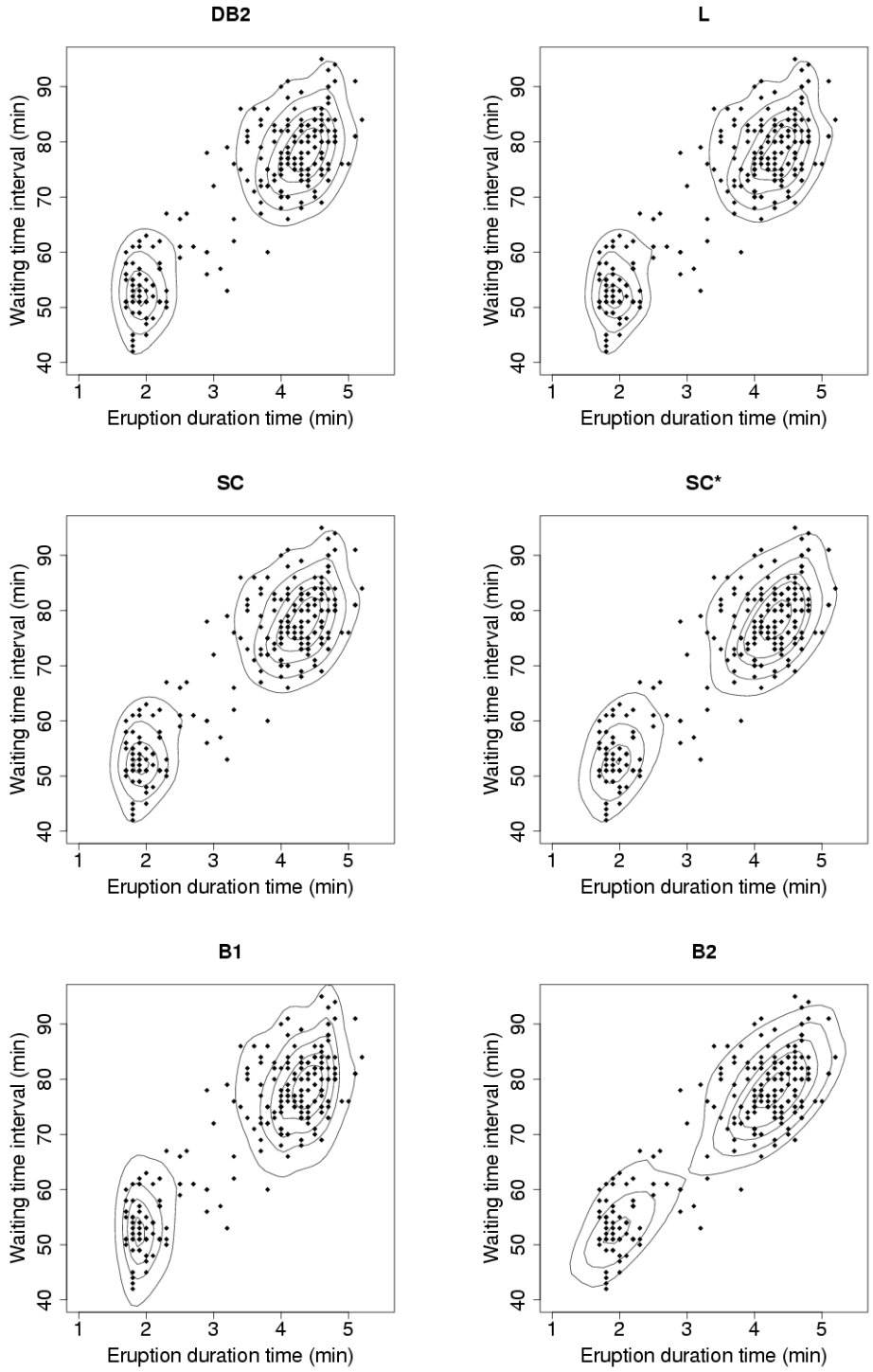

Figure 3.3: 'Old Faithful' geyser data contour plots - cross validation selectors

DB2	L	B1	B2	SC	SC*
$\begin{bmatrix} 0.0320 & 0 \\ 0 & 11.80 \end{bmatrix}$	$\begin{bmatrix} 0.0282 & 0.0295 \\ 0.0295 & 6.6000 \end{bmatrix}$	$\begin{bmatrix} 0.0156 & 0.0012 \\ 0.0012 & 24.989 \end{bmatrix}$	$\begin{bmatrix} 0.1849 & 1.9151 \\ 1.9151 & 25.778 \end{bmatrix}$	$\begin{bmatrix} 0.0365 & 0.1069 \\ 0.1069 & 8.9714 \end{bmatrix}$	$\begin{bmatrix} 0.0704 & 0.6197 \\ 0.6197 & 14.182 \end{bmatrix}$

Table 3.4: Cross validation bandwidth matrices for 'Old Faithful' geyser data

The other data set we analysed previously is the child mortality-life expectancy data. The cross validation selectors for this data set are in Table 3.5. This time B1 and B2 gave the same selector - the contours in Figure 3.4 are too circular-ish whereas most of the data mass is aligned at angle to the co-ordinate axes. This is a result from the orientation of the B1 and B2 selectors: they have positive correlation whilst the data have negative correlation. The L selector gives contours that are smoother and more oblique than for B1 and B2, though there is still evidence of undersmoothing. SC and SC* have smoother contours still (SC* maybe is oversmoothed) and are unimodal, unlike the L estimate which has a small mode in the right hand corner. The DB2 and DL estimates are strongly bimodal which we believe is an artifact from using kernels that are oriented parallel to the axes. At the 'narrow' part of the data set, around under-5 mortality of 100 and life expectancy of 60, the lack of smoothing in the oblique direction results in a lower density estimate here, creating a trough and the appearance of two modes. Taking this into account, the SC and SC* selectors probably best balance the trade-off between the demands of smoothness with structure recovery in this case.

DL	DB2	L	B1, B2	SC	SC*
$\begin{bmatrix} 670.52 & 0 \\ 0 & 9.979 \end{bmatrix}$	$\begin{bmatrix} 1072.8 & 0 \\ 0 & 9.298 \end{bmatrix}$	$\begin{bmatrix} 388.2 & -83.34 \\ -83.34 & 25.13 \end{bmatrix}$	$\begin{bmatrix} 1087.1 & 135.3 \\ 135.3 & 23.59 \end{bmatrix}$	$\begin{bmatrix} 694.1 & -73.07 \\ -73.07 & 17.50 \end{bmatrix}$	$\begin{bmatrix} 1322 & -191.8 \\ -191.8 & 34.99 \end{bmatrix}$

Table 3.5: Cross validation bandwidth matrices for child mortality-life expectancy data

In the above analysis, we suggest that the bimodality produced by the DL and DB2 selectors on the UNICEF data may be an artifice. We now present some evidence to justify this statement. The UNICEF data has two large, circular-ish regions connected with a narrow, angled region. A target density with a 'dumbbell' shape, as shown in Figure 3.5, approximates the shape of this data. The formula for this density is

$$\tfrac{4}{11} N\left(\begin{bmatrix} -2 \\ 2 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) + \tfrac{3}{11} N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.8 & -0.72 \\ -0.72 & 0.8 \end{bmatrix} \right) + \tfrac{4}{11} N\left(\begin{bmatrix} 2 \\ -2 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right)$$

Most important is that this density is *unimodal*, with the mode located at the 'bridge' that connects the two flatter 'discs'. We will show that using a diagonal bandwidth matrix with data drawn from this density produces bimodality whereas a full bandwidth matrix does not.

We compute the DL, DB2, SC and SC* selectors for a random sample of size 200 from this density. The results are in Table 3.6. Their corresponding density estimates are in Figure 3.6. For the L density estimate, there is insufficient smoothing overall, producing

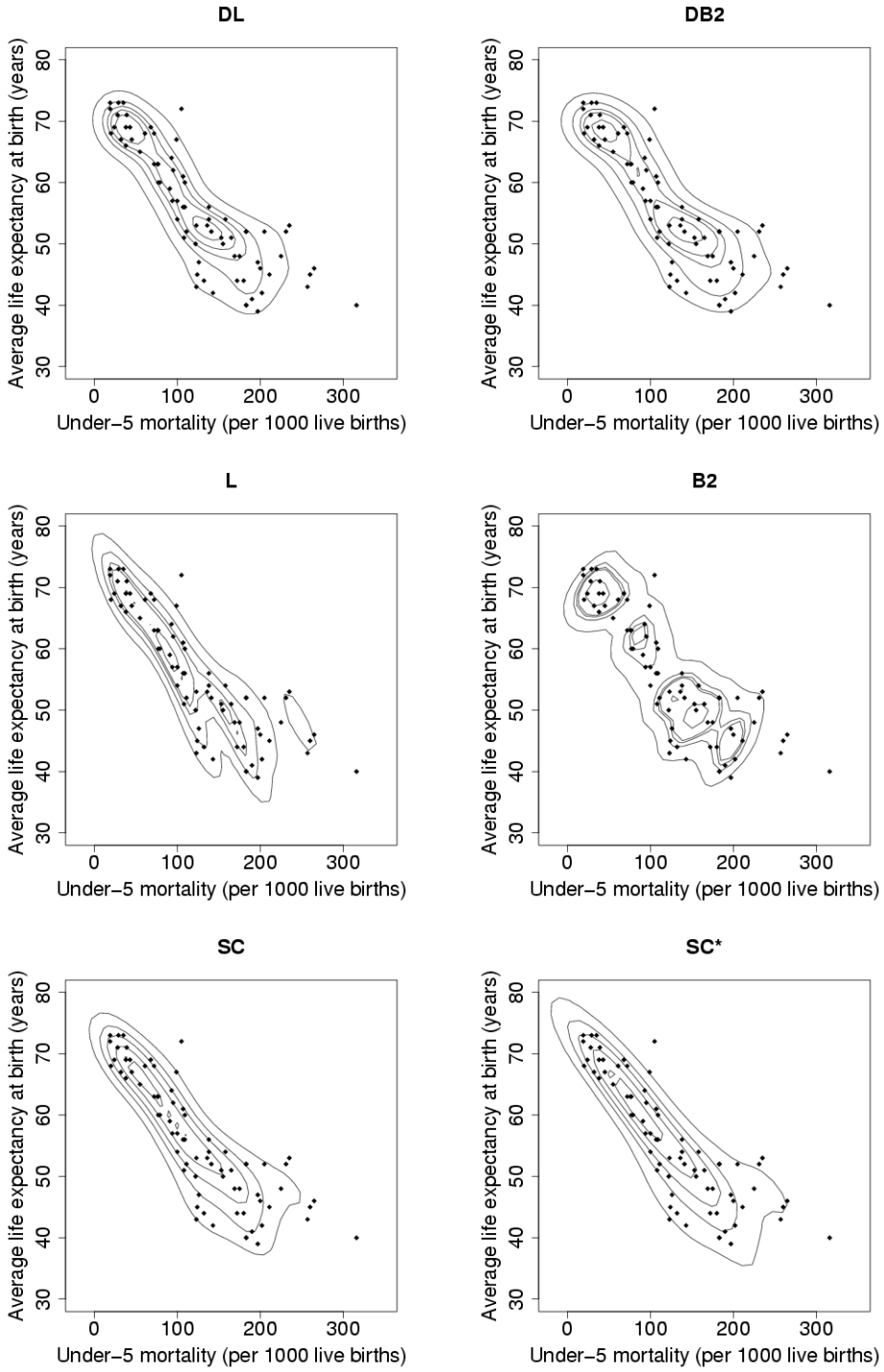

Figure 3.4: Child mortality-life expectancy contour plots - cross validation selectors

a noisy estimate with many spurious modes. We can see that DL density estimate, in the central part, which is narrower and at an angle, there is insufficient smoothing in the direction of this angle. This leads to lower heights of the density estimate here than in the flatter, circular ends and thus to a bimodal artifice. The SC and SC* density estimates, with full bandwidth matrices, are able to appropriately smooth the central, angled region and thus reproduce the unimodality of the target density (though the SC estimate's mode is off-centre whereas the SC* estimate's mode is centred). So the SC* selector most accurately reconstructs the 'dumbbell' density shape from the data.

DL	DB2	SC	SC*
$\begin{bmatrix} 0.1529 & 0 \\ 0 & 0.1305 \end{bmatrix}$	$\begin{bmatrix} 0.4477 & 0 \\ 0 & 0.5612 \end{bmatrix}$	$\begin{bmatrix} 0.3331 & -0.1245 \\ -0.1245 & 0.2891 \end{bmatrix}$	$\begin{bmatrix} 0.5646 & -0.4043 \\ -0.4043 & 0.4934 \end{bmatrix}$

Table 3.6: Cross validation bandwidth matrices for 'dumbbell' density

3.6 Conclusion

Cross validation bandwidth selectors have already been demonstrated to be useful and in the one dimensional case and for diagonal bandwidth matrices in the multidimensional case. In this chapter, we have generalised cross validation selectors to full, unconstrained bandwidth matrices. Their asymptotic properties, including their relative rates of convergence were derived. These were supplemented by a simulation study of their finite sample properties. From the consideration of these theoretical and practical properties, the SCV selectors, with either pre-sphering or pre-scaling, appear to be the best performing cross validation selector.

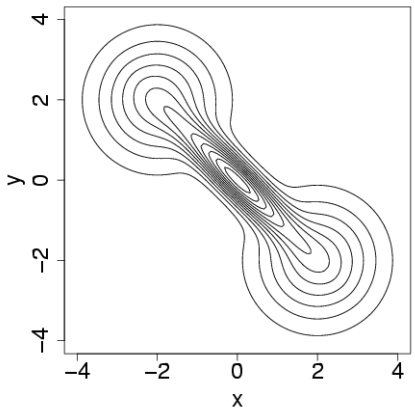

Figure 3.5: Contour plot for 'dumbbell' density

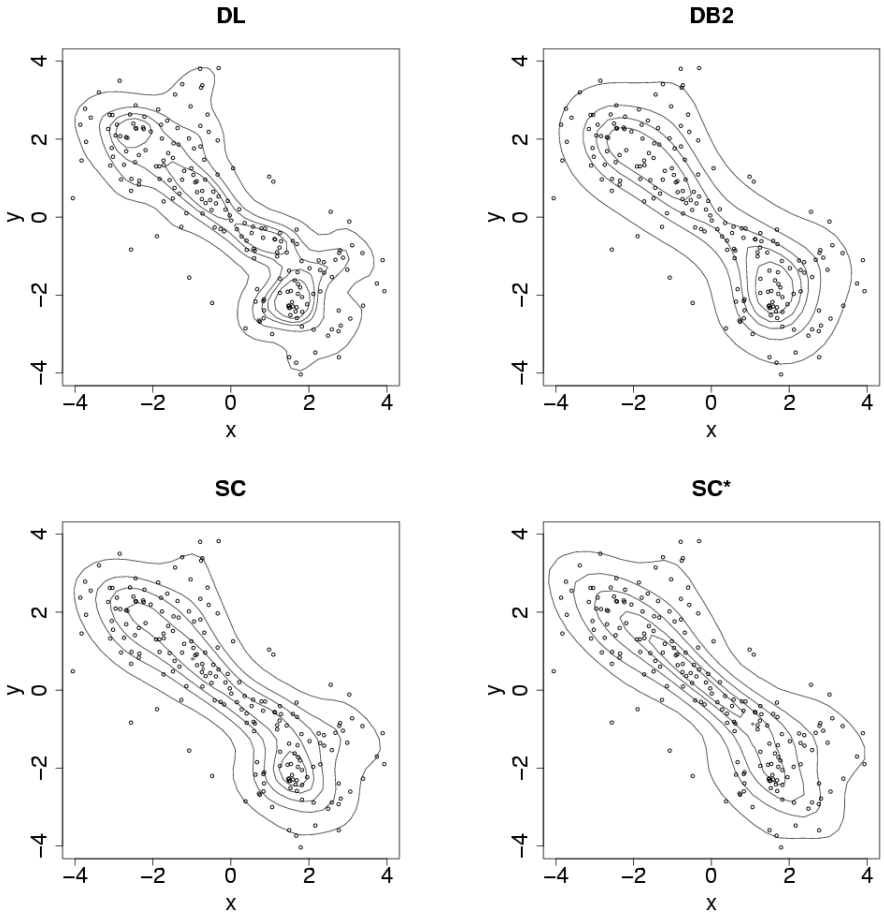

Figure 3.6: Contour plot for 'dumbbell' density estimates

Chapter 4

Partitioned bandwidth selectors

4.1 Introduction

Variable bandwidth selectors are a generalisation of fixed bandwidth selectors, as we saw in Section 1.3.3. Most of the research in variable bandwidth selectors, like fixed bandwidth selectors, has focused on the univariate case. In this chapter we explore multivariate variable bandwidth selectors of the type exemplified by Sain (2002). This selector is a sample point selector with two main features: (a) the sample space is partitioned and then (b) within each partition, an optimal bandwidth matrix is selected. The important assumption is that the bandwidth matrix function $\mathbf{\Omega}(\cdot)$ and the partition $\mathcal{P}$ of the sample space are both *non-random*, in an analogous way to how we assume a non-random bandwidth matrix $\mathbf{H}$ in fixed bandwidth kernel density estimation but is in practice determined from the data. The partitioned kernel density estimate is defined by

$$\hat{f}_{\mathrm{PT}}(\boldsymbol{x}; \mathbf{\Omega}, \mathcal{P}) = n^{-1} \sum_{i=1}^{n} K_{\mathbf{\Omega}(\boldsymbol{X}_i)}(\boldsymbol{x} - \boldsymbol{X}_i). \tag{4.1}$$

For our random sample $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$, the bandwidth matrix associated with $\boldsymbol{X}_i$ is $\mathbf{\Omega}(\boldsymbol{X}_i)$. Our hope is that the extra flexibility of having different bandwidths in different parts of the sample space will give us better performance than using a single bandwidth fixed over all the sample space. Our task is more complicated as we need to select a partition *and* a bandwidth matrix function.

The task of selecting a bandwidth matrix function of arbitrary form appears to be daunting. To simplify the problem, we restrict $\mathbf{\Omega}(\cdot)$ to be a piecewise constant function over $\mathcal{P} = \{P_1, P_2, \ldots, P_\nu\}$ i.e. we associate a fixed bandwidth matrix $\mathbf{H}_j$ with class $P_j, j = 1, 2, \ldots, \nu$. If the data points are in the same partition class P_j then they are associated with the same bandwidth matrix $\mathbf{H}_j$. Figure 4.1 displays an example of a data set that would benefit from having different bandwidth matrices in each partition class. The sample space is the large rectangle, partitioned into 3 classes. For example, all the data points in P_1 are associated with $\mathbf{H}_1$ (denoted as $P_1 \leftrightarrow \mathbf{H}_1$ in the figure) and so

on. The bandwidth matrices follow the local orientation of the data points within each partition class, rather over the whole sample space.

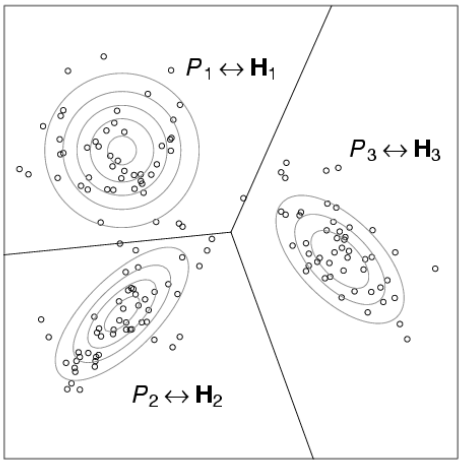

Figure 4.1: Partition of sample space with data points and associated bandwidth matrices

To select this piecewise constant bandwidth matrix function, we will draw upon the properties of fixed bandwidth matrices from the previous two chapters. Before we do this, we write down the various error criteria expressions for this partitioned kernel density estimator in Section 4.2. Since a pre-specified form of the partition is not required to proceed with the theoretical development of partitioned bandwidth selectors, we look at bandwidth selection first in Section 4.3. After this, we then examine two partition selection methods in Section 4.4. We put the theoretical results from the previous two sections into practice in Section 4.5 which contains a simulation study and real data analysis.

4.2 Error criteria

For fixed kernel density estimators, we have used the MISE criterion throughout this thesis for both its mathematical tractability and widespread use. In the fixed case, we consider the MISE to be a function of the bandwidth $\mathbf{H}$; here we consider the MISE to be a functional of $\boldsymbol{\Omega}$, defined by

$$\operatorname{MISE}(\boldsymbol{\Omega}) \equiv \operatorname{MISE} \hat{f}_{\mathrm{PT}}(\cdot; \boldsymbol{\Omega}) = \mathbb{E} \int_{\mathbb{R}^d} [\hat{f}_{\mathrm{PT}}(\boldsymbol{x}; \boldsymbol{\Omega}) - f(\boldsymbol{x})]^2 \, d\boldsymbol{x}. \tag{4.2}$$

We stop explicitly denoting the dependence of $\hat{f}_{\mathrm{PT}}$ on the partition $\mathcal{P}$ since it is now implicit in the specification of $\boldsymbol{\Omega}$.

As is usual, the first step towards writing down a more explicit expression for the MISE is to first compute the expected value and variance of the partitioned estimator.

The expected value is

$$\begin{aligned}\mathbb{E}\,\hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) &= \mathbb{E}\,K_{\boldsymbol{\Omega}(\boldsymbol{X})}(\boldsymbol{x}-\boldsymbol{X}) \\ &= \int_{\mathbb{R}^d} K_{\boldsymbol{\Omega}(\boldsymbol{y})}(\boldsymbol{x}-\boldsymbol{y})f(\boldsymbol{y})\,d\boldsymbol{y} \\ &= \sum_{j=1}^{\nu}\int_{\mathbb{R}^d} K_{\mathbf{H}_j}(\boldsymbol{x}-\boldsymbol{y})f(\boldsymbol{y})\mathbf{1}\{\boldsymbol{y}\in P_j\}\,d\boldsymbol{y} \\ &= \sum_{j=1}^{\nu}(K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x})\end{aligned}$$

where $f_{P_j}(\boldsymbol{x}) = f(\boldsymbol{x})\mathbf{1}\{\boldsymbol{x}\in P_j\}$ is the density f restricted to P_j. The variance is

$$\operatorname{Var}\hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) = n^{-2}\sum_{i=1}^{n}\operatorname{Var}K_{\boldsymbol{\Omega}(\boldsymbol{X}_i)}(\boldsymbol{x}-\boldsymbol{X}_i) = n^{-1}\operatorname{Var}K_{\boldsymbol{\Omega}(\boldsymbol{X})}(\boldsymbol{x}-\boldsymbol{X}).$$

In a similar calculation for $\mathbb{E}\,K_{\boldsymbol{\Omega}(\boldsymbol{X})}(\boldsymbol{x}-\boldsymbol{X})$,

$$\mathbb{E}\,K_{\boldsymbol{\Omega}(\boldsymbol{X})}(\boldsymbol{x}-\boldsymbol{X})^2 = \sum_{j=1}^{\nu}(K_{\mathbf{H}_j}^2 * f_{P_j})(\boldsymbol{x})$$

then

$$\operatorname{Var}\hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) = n^{-1}\sum_{j=1}^{\nu}(K_{\mathbf{H}_j}^2 * f_{P_j})(\boldsymbol{x}) - n^{-1}\Bigg[\sum_{j=1}^{\nu}(K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x})\Bigg]^2$$

which gives the MSE to be

$$\begin{aligned}&\operatorname{MSE}\,\hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) \\ &= n^{-1}\sum_{j=1}^{\nu}(K_{\mathbf{H}_j}^2 * f_{P_j})(\boldsymbol{x}) - n^{-1}\Bigg[\sum_{j=1}^{\nu}(K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x})\Bigg]^2 + \Bigg[\sum_{j=1}^{\nu}(K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x}) - f(\boldsymbol{x})\Bigg]^2 \\ &= n^{-1}\sum_{j=1}^{\nu}(K_{\mathbf{H}_j}^2 * f_{P_j})(\boldsymbol{x}) + (1-n^{-1})\sum_{j=1}^{\nu}\sum_{j'=1}^{\nu}(K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x})(K_{\mathbf{H}_{j'}} * f_{P_{j'}})(\boldsymbol{x}) \\ &\quad - 2\sum_{j=1}^{\nu}(K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x})f(\boldsymbol{x}) + f(\boldsymbol{x})^2.\end{aligned}$$

This can then be integrated to yield a corresponding MISE expression. This expression can be simplified a little if we note that the integral of the first term of the MSE is

$$\begin{aligned}n^{-1}\sum_{j=1}^{\nu}\int_{\mathbb{R}^d}(K_{\mathbf{H}_j}^2 * f_{P_j})(\boldsymbol{x})\,d\boldsymbol{x} &= n^{-1}\sum_{j=1}^{\nu}\int_{\mathbb{R}^{2d}} K_{\mathbf{H}_j}^2(\boldsymbol{x}-\boldsymbol{y})f_{P_j}(\boldsymbol{y})\,d\boldsymbol{x}d\boldsymbol{y} \\ &= n^{-1}\sum_{j=1}^{\nu}\int_{\mathbb{R}^d}\Bigg[|\mathbf{H}_j|^{-1/2}\int_{\mathbb{R}^d}K(\boldsymbol{w})^2\,d\boldsymbol{w}\Bigg]f_{P_j}(\boldsymbol{y})\,d\boldsymbol{y} \\ &= n^{-1}R(K)\sum_{j=1}^{\nu}|\mathbf{H}_j|^{-1/2}\int_{\mathbb{R}^d}f_{P_j}(\boldsymbol{y})\,d\boldsymbol{y}.\end{aligned}$$

The MISE is thus

$$\begin{aligned}
&\operatorname{MISE} \hat{f}_{\mathrm{PT}}(\cdot; \boldsymbol{\Omega}) \\
&\quad = n^{-1} R(K) \sum_{j=1}^{\nu} \pi_j |\mathbf{H}_j|^{-1/2} + (1 - n^{-1}) \sum_{j=1}^{\nu} \sum_{j'=1}^{\nu} \int_{\mathbb{R}^d} (K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x}) (K_{\mathbf{H}_{j'}} * f_{P_{j'}})(\boldsymbol{x}) \, d\boldsymbol{x} \\
&\qquad - 2 \sum_{j=1}^{\nu} \int_{\mathbb{R}^d} (K_{\mathbf{H}_j} * f_{P_j})(\boldsymbol{x}) f(\boldsymbol{x}) \, d\boldsymbol{x} + R(f).
\end{aligned}$$

where $\pi_j = \int_{\mathbb{R}^d} f_{P_j}(\boldsymbol{x}) \, d\boldsymbol{x} = \int_{P_j} f(\boldsymbol{x}) \, d\boldsymbol{x}$ is the probability mass of f in P_j. The integrals of this MISE do not have closed forms so we will work towards a tractable asymptotic expression.

To progress further, we need an extra condition on the structure of the partition. We assume that the classes $P_1, P_2, \ldots, P_\nu$ are open sets and that the boundaries of these classes ∂P has measure zero; and that $\{P_1, P_2, \ldots, P_\nu, \partial P\}$ form a partition of the sample space i.e. $P_i \cap P_j = \emptyset, P_i \cap \partial P = \emptyset$ for all i, j and $\bigcup_{i=1}^{\nu} P_i \cup \partial P$ is the sample space. For the moment, suppose that $K(\cdot - \boldsymbol{x})$ has compact support, denoted by $\operatorname{supp}(K, \boldsymbol{x})$. Let $\boldsymbol{x}$ be an interior point in P_j, then there exists $\varepsilon > 0$ such that $B(\boldsymbol{x}, \varepsilon) \subset P_j$, where $B(\boldsymbol{x}, \varepsilon)$ is the open ball, centred at $\boldsymbol{x}$ and with radius ε. For all $\varepsilon > 0$, there exists $\mathbf{H}$ such that $\operatorname{supp}(K_{\mathbf{H}}, \boldsymbol{x}) \subset B(\boldsymbol{x}, \varepsilon)$. Taking these together, we have for all $\varepsilon > 0$ we can find $\mathbf{H}$ where $\operatorname{supp}(K_{\mathbf{H}}, \boldsymbol{x}) \subset P_j$ since P_j is an open set. So for small enough $\mathbf{H}$, we can say that the contribution of the kernel centred at the point $\boldsymbol{x}$ lies entirely within a single partition class P_j. Hence an integral over P_j can be reduced to an integral over $\operatorname{supp}(K, \boldsymbol{x})$. Using this asymptotic argument, we can simplify the expected value

$$\begin{aligned}
\mathbb{E} \, \hat{f}_{\mathrm{PT}}(\boldsymbol{x}; \boldsymbol{\Omega}) &= \sum_{j=1}^{\nu} \int_{P_j} K_{\mathbf{H}_j}(\boldsymbol{y} - \boldsymbol{x}) f(\boldsymbol{y}) \mathbf{1}\{\boldsymbol{y} \in P_j\} \, d\boldsymbol{y} \\
&= \sum_{j=1}^{\nu} \int_{\operatorname{supp}(K_{\mathbf{H}_j}, \boldsymbol{x})} K_{\mathbf{H}_j}(\boldsymbol{y} - \boldsymbol{x}) f(\boldsymbol{y}) \mathbf{1}\{\boldsymbol{y} \in P_j\} [1 + o(1)] \, d\boldsymbol{y} \\
&= \sum_{j=1}^{\nu} \int_{\operatorname{supp}(K, \mathbf{0})} K(\boldsymbol{w}) f(\boldsymbol{x} + \mathbf{H}_j^{1/2} \boldsymbol{w}) \mathbf{1}\{\boldsymbol{x} + \mathbf{H}_j^{1/2} \boldsymbol{w} \in P_j\} [1 + o(1)] \, d\boldsymbol{w} \\
&= \sum_{j=1}^{\nu} \int_{\operatorname{supp}(K, \mathbf{0})} K(\boldsymbol{w}) [f(\boldsymbol{x}) - \boldsymbol{w}^T \mathbf{H}_j^{1/2} Df(\boldsymbol{x}) + \tfrac{1}{2} \boldsymbol{w}^T \mathbf{H}_j^{1/2} D^2 f(\boldsymbol{x}) \mathbf{H}_j^{1/2} \boldsymbol{w} \\
&\quad + o(\|\operatorname{vech} \mathbf{H}_j\|)] \mathbf{1}\{\boldsymbol{x} \in P_j\} [1 + o(1)] \, d\boldsymbol{w} \\
&= f(\boldsymbol{x}) + \tfrac{1}{2} \mu_2(K) \sum_{j=1}^{\nu} \operatorname{tr}(\mathbf{H}_j D^2 f(\boldsymbol{x})) \mathbf{1}\{\boldsymbol{x} \in P_j\} + o(\|\operatorname{vech} \mathbf{H}_{\max}\|)
\end{aligned}$$

where $\mathbf{H}_{\max}$ is the bandwidth matrix which attains the maximum of $\{\|\operatorname{vech} \mathbf{H}_j\| : j = 1, 2, \ldots, \nu\}$. This then leads to the bias expression

$$\operatorname{Bias} \hat{f}_{\mathrm{PT}}(\boldsymbol{x}; \boldsymbol{\Omega}) = \tfrac{1}{2} \mu_2(K) \sum_{j=1}^{\nu} \operatorname{tr}(\mathbf{H}_j D^2 f(\boldsymbol{x})) \mathbf{1}\{\boldsymbol{x} \in P_j\} + o(\|\operatorname{vech} \mathbf{H}_{\max}\|).$$

The squared bias has a simple form:

$$\begin{aligned}\operatorname{Bias}^2 \hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) &= \tfrac{1}{4}\mu_2(K)^2 \sum_{j=1}^{\nu}\sum_{j'=1}^{\nu} \operatorname{tr}(\mathbf{H}_j D^2 f(\boldsymbol{x}))\mathbf{1}\{\boldsymbol{x}\in P_j\}\operatorname{tr}(\mathbf{H}_j D^2 f(\boldsymbol{x}))\mathbf{1}\{\boldsymbol{x}\in P_{j'}\} \\ &= \tfrac{1}{4}\mu_2(K)^2 \sum_{j=1}^{\nu} \operatorname{tr}^2(\mathbf{H}_j D^2 f(\boldsymbol{x}))\mathbf{1}\{\boldsymbol{x}\in P_j\}\end{aligned}$$

since $\boldsymbol{x}$ is an interior point and cannot belong to two partition classes simultaneously.

We simplify the variance in a similar manner:

$$\begin{aligned}\mathbb{E}\, K_{\boldsymbol{\Omega}(\boldsymbol{X})}(\boldsymbol{x}-\boldsymbol{X})^2 &= \int_{\mathbb{R}^d} K_{\boldsymbol{\Omega}(\boldsymbol{y})}(\boldsymbol{x}-\boldsymbol{y})^2 f(\boldsymbol{y})\, d\boldsymbol{y} \\ &= \sum_{j=1}^{\nu}\int_{P_j} K_{\boldsymbol{\Omega}(\boldsymbol{y})}(\boldsymbol{x}-\boldsymbol{y})^2 f(\boldsymbol{y})\mathbf{1}\{\boldsymbol{y}\in P_j\}\, d\boldsymbol{y} \\ &= \sum_{j=1}^{\nu}|\mathbf{H}_j|^{-1/2}\int_{\operatorname{supp}(K,\mathbf{0})} K(\boldsymbol{w})^2 f(\boldsymbol{x}+\mathbf{H}_j^{1/2}\boldsymbol{w})\mathbf{1}\{\boldsymbol{x}+\mathbf{H}_j^{1/2}\boldsymbol{w}\in P_j\}\, d\boldsymbol{w} \\ &= \sum_{j=1}^{\nu}|\mathbf{H}_j|^{-1/2}\int_{\operatorname{supp}(K,\mathbf{0})} K(\boldsymbol{w})^2 [f(\boldsymbol{x})+o(1)]\mathbf{1}\{\boldsymbol{x}\in P_j\}[1+o(1)]\, d\boldsymbol{w} \\ &= R(K)\sum_{j=1}^{\nu}|\mathbf{H}_j|^{-1/2} f(\boldsymbol{x})\mathbf{1}\{\boldsymbol{x}\in P_j\} + o(|\mathbf{H}_{\min}|^{-1/2})\end{aligned}$$

where $\mathbf{H}_{\min}$ is defined in an analogous way to $\mathbf{H}_{\max}$. This dominates $[\mathbb{E}\, K_{\boldsymbol{\Omega}(\boldsymbol{X})}(\boldsymbol{x}-\boldsymbol{X})]^2 = f(\boldsymbol{x})^2 + o(1)$ so

$$\operatorname{Var}\hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) = n^{-1}R(K)\sum_{j=1}^{\nu}|\mathbf{H}_j|^{-1/2} f(\boldsymbol{x})\mathbf{1}\{\boldsymbol{x}\in P_j\} + o(n^{-1}|\mathbf{H}_{\min}|^{-1/2}).$$

If we combine these to form the AMSE

$$\operatorname{AMSE}\hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) = n^{-1}R(K)\sum_{j=1}^{\nu}|\mathbf{H}_j|^{-1/2} f_{P_j}(\boldsymbol{x}) + \tfrac{1}{4}\mu_2(K)^2\sum_{j=1}^{\nu}\operatorname{tr}^2(\mathbf{H}_j D^2 f_{P_j}(\boldsymbol{x}))$$

The AMSE is valid for points that are in the interior of the partition classes and if $n^{-1}|\mathbf{H}_{\min}|^{-1/2}\to 0$ and every element of $\mathbf{H}_{\max}\to 0$ as $n\to\infty$. As the boundary points altogether have measure zero, we can effectively ignore them when integrating to form the AMISE

$$\operatorname{AMISE}\hat{f}_{\mathrm{PT}}(\cdot;\boldsymbol{\Omega}) = n^{-1}R(K)\sum_{j=1}^{\nu}\pi_j|\mathbf{H}_j|^{-1/2} + \tfrac{1}{4}\mu_2(K)^2\sum_{j=1}^{\nu}(\operatorname{vech}^T\mathbf{H}_j)\boldsymbol{\Psi}_{4,P_j}(\operatorname{vech}\mathbf{H}_j)$$

where

$$\begin{aligned}\boldsymbol{\Psi}_{4,P_j} &= \int_{\mathbb{R}^d}\operatorname{vech}(2D^2 f_{P_j}(\boldsymbol{x}) - \operatorname{dg} D^2 f_{P_j}(\boldsymbol{x}))\operatorname{vech}^T(2D^2 f_{P_j}(\boldsymbol{x}) - \operatorname{dg} D^2 f_{P_j}(\boldsymbol{x}))\, d\boldsymbol{x} \\ &= \int_{P_j}\operatorname{vech}(2D^2 f(\boldsymbol{x}) - \operatorname{dg} D^2 f(\boldsymbol{x}))\operatorname{vech}^T(2D^2 f(\boldsymbol{x}) - \operatorname{dg} D^2 f(\boldsymbol{x}))\, d\boldsymbol{x}.\end{aligned}$$

For this AMISE expression we have assumed that the kernel K has compact support. This is true for many common kernels (e.g. Epanechnikov, biweight, triangle) though not the normal kernel. Fortunately the normal kernel has an 'effective' compact support i.e. the probability mass outside this effective support is 'close enough' to zero that it can be ignored for practical purposes. We could consider compact supports that are hyperspheres which are 'natural' when dealing with spherically symmetric kernels. However we examine compact supports which are hypercubes since these will aid our computer implementation of kernel density estimators over hypergrids.

We know that the standard multivariate normal density $\phi_{\mathbf{I}}$ is the product of d univariate standard normal densities. From the univariate standard normal density, we can obtain the upper and lower $\alpha/2$ quantiles $z_{\alpha/2}$ and $-z_{\alpha/2}$ easily. So we can treat the d-dimensional hypercube $[-z_{\alpha/2}, z_{\alpha/2}]^d$ as an effective support since

$$\int_{[-z_{\alpha/2},z_{\alpha/2}]^d} \phi_{\mathbf{I}}(\boldsymbol{x})\ d\boldsymbol{x} = \prod_{i=1}^{d} \int_{-z_{\alpha/2}}^{z_{\alpha/2}} \phi(x_i)\ dx_i = (1-\alpha)^d.$$

For example, for $z_{\alpha/2} = 3.7$ where $\alpha = 0.0002156$, the bivariate normal kernel has only about 0.04% of its probability mass outside $[-3.7, 3.7]^2$. It is possible to effectively restrict the support of the normal kernel because it has fast (i.e. exponentially) decaying tails.

The MISE and AMISE expressions remain unknown in practice since their values depend on the target density f. So MISE- and AMISE-optimal $\boldsymbol{\Omega}$ are still unattainable. In the next section, we look at data-based bandwidth selection. In the section after that, we look at data-based partition selection.

4.3 Bandwidth selection

The problem we tackle in this section is the bandwidth selection. In the ideal case, we are aiming for a MISE-optimal bandwidth function

$$\boldsymbol{\Omega}_{\mathrm{MISE}} = \operatorname*{argmin}_{\boldsymbol{\Omega}} \mathrm{MISE}\,(\boldsymbol{\Omega}).$$

We can similarly define an AMISE-optimal $\boldsymbol{\Omega}_{\mathrm{AMISE}}$. We use the fixed bandwidth selectors from the previous chapters as a base to construct our partitioned bandwidth selectors.

The partitioned LSCV is a straightforward extension of the fixed bandwidth case:

$$\mathrm{LSCV}(\boldsymbol{\Omega}) = R(\hat{f}_{\mathrm{PT}}(\cdot; \boldsymbol{\Omega})) - 2n^{-1} \sum_{i=1}^{n} \hat{f}_{\mathrm{PT},-i}(\boldsymbol{X}_i; \boldsymbol{\Omega}) \tag{4.3}$$

where

$$\hat{f}_{\mathrm{PT},-i}(\boldsymbol{X}_i; \boldsymbol{\Omega}) = (n-1)^{-1} \sum_{\substack{i'=1 \\ i' \neq i}}^{n} K_{\boldsymbol{\Omega}(\boldsymbol{X}_{i'})}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}).$$

The unbiasedness property of the fixed bandwidth LSCV carries over. The MISE is

$$\operatorname{MISE}(\boldsymbol{\Omega}) = \mathbb{E}\, R(\hat{f}_{\mathrm{PT}}(\cdot;\boldsymbol{\Omega})) - 2\int_{\mathbb{R}^d} \mathbb{E}\, \hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) f(\boldsymbol{x})\, d\boldsymbol{x} + R(f).$$

We have $\mathbb{E}\operatorname{LSCV}(\boldsymbol{\Omega}) = \operatorname{MISE}(\boldsymbol{\Omega}) - R(f)$ as

$$\begin{aligned}
\mathbb{E}\left[n^{-1}\sum_{i=1}^{n} \hat{f}_{-i}(\boldsymbol{X}_i;\boldsymbol{\Omega})\right] &= n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{i'=1\\ i'\neq i}}^{n} K_{\boldsymbol{\Omega}(\boldsymbol{X}_{i'})}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}) \\
&= \mathbb{E}\, K_{\boldsymbol{\Omega}(\boldsymbol{X}_1)}(\boldsymbol{X}_1 - \boldsymbol{X}_2) \\
&= \int_{\mathbb{R}^{2d}} K_{\boldsymbol{\Omega}(\boldsymbol{x})}(\boldsymbol{x}-\boldsymbol{y}) f(\boldsymbol{x}) f(\boldsymbol{y})\, d\boldsymbol{x} d\boldsymbol{y} \\
&= \int_{\mathbb{R}^d} \left[\int_{\mathbb{R}^d} K_{\boldsymbol{\Omega}(\boldsymbol{x})}(\boldsymbol{x}-\boldsymbol{y}) f(\boldsymbol{y})\, d\boldsymbol{y}\right] f(\boldsymbol{x})\, d\boldsymbol{x} \\
&= \int_{\mathbb{R}^d} \mathbb{E}\, \hat{f}_{\mathrm{PT}}(\boldsymbol{x};\boldsymbol{\Omega}) f(\boldsymbol{x})\, d\boldsymbol{x}.
\end{aligned}$$

The LSCV can be rewritten as

$$\begin{aligned}
\operatorname{LSCV}(\boldsymbol{\Omega}) = n^{-2}\sum_{i=1}^{n}\sum_{i'=1}^{n} (K_{\boldsymbol{\Omega}(\boldsymbol{X}_i)} * K_{\boldsymbol{\Omega}(\boldsymbol{X}_{i'})})(\boldsymbol{X}_i - \boldsymbol{X}_{i'}) \\
- 2[n(n-1)]^{-1}\sum_{i=1}^{n}\sum_{\substack{i'=1\\ i'\neq i}}^{n} K_{\boldsymbol{\Omega}(\boldsymbol{X}_{i'})}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}). \qquad (4.4)
\end{aligned}$$

This further simplifies for normal kernels to

$$\begin{aligned}
&\operatorname{LSCV}(\boldsymbol{\Omega}) \\
&= n^{-2}\sum_{i=1}^{n}\sum_{i'=1}^{n} \phi_{\boldsymbol{\Omega}(\boldsymbol{X}_i)+\boldsymbol{\Omega}(\boldsymbol{X}_{i'})}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}) - 2[n(n-1)]^{-1}\sum_{i=1}^{n}\sum_{\substack{i'=1\\ i'\neq i}}^{n} \phi_{\boldsymbol{\Omega}(\boldsymbol{X}_{i'})}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}).
\end{aligned}$$

The LSCV selector $\hat{\boldsymbol{\Omega}}_{\mathrm{LSCV}}$ is the minimiser of $\operatorname{LSCV}(\boldsymbol{\Omega})$. Another simplification can be obtained if we use the $h^2\mathbf{I}$ type parameterisation, as in Sain (2002) i.e. we have $\boldsymbol{\Omega}(\boldsymbol{X}_i) = \omega(\boldsymbol{X}_i)^2\mathbf{I}$ where $\omega(\boldsymbol{X}_i) = h_j$ if $\boldsymbol{X}_i$ belongs to class j. This is done in an attempt to reduce the complexity (from $\frac{1}{2}d(d+1)\nu$ to ν bandwidths) and increase the stability of bandwidth selection:

$$\begin{aligned}
\operatorname{LSCV}(\omega) = n^{-2}\sum_{i=1}^{n}\sum_{i'=1}^{n} (K_{\omega(\boldsymbol{X}_i)^2\mathbf{I}} * K_{\omega(\boldsymbol{X}_{i'})^2\mathbf{I}})(\boldsymbol{X}_i - \boldsymbol{X}_{i'}) \\
- 2[n(n-1)]^{-1}\sum_{i=1}^{n}\sum_{\substack{i'=1\\ i'\neq i}}^{n} K_{\omega(\boldsymbol{X}_{i'})^2\mathbf{I}}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}). \qquad (4.5)
\end{aligned}$$

In the above calculations for LSCV, we do not use the special the structure we impose on $\boldsymbol{\Omega}$ (i.e. piecewise constancy) to write down $\operatorname{LSCV}(\boldsymbol{\Omega})$. So this expression is valid

for a general bandwidth matrix function $\boldsymbol{\Omega}$. For the Abramson selector we use $\boldsymbol{\Omega}(\boldsymbol{X}_i) = h^2 f(\boldsymbol{X}_i)^{-1}\mathbf{I}$. This parameterisation appears to be somewhat restrictive, given the evidence of the previous results for full fixed bandwidth matrices. However it is mitigated by the fact that these variable bandwidth matrices take into account the locally varying number of data points (as measured by the height of the density function) which is ignored by fixed bandwidths. We denote its least squares cross validation as

$$\begin{aligned}\mathrm{LSCV}'(h) = n^{-2}\sum_{i=1}^{n}\sum_{i'=1}^{n}(K_{h^2 f(\boldsymbol{X}_i)^{-1}\mathbf{I}} * K_{h^2 f(\boldsymbol{X}_{i'})^{-1}\mathbf{I}})(\boldsymbol{X}_i - \boldsymbol{X}_{i'}) \\ - 2n^{-1}(n-1)^{-1}\sum_{i=1}^{n}\sum_{\substack{i'=1\\ i'\neq i}}^{n} K_{h^2 f(\boldsymbol{X}_{i'})^{-1}\mathbf{I}}(\boldsymbol{X}_i - \boldsymbol{X}_{i'}). \qquad (4.6)\end{aligned}$$

Before we can minimise this in practice, we estimate f with a pilot estimate $\tilde{f}_P(\cdot;\mathbf{G})$. The minimiser of this then is $\hat{h}_{\mathrm{LSCV}'}$.

4.4 Partition selection

The approach to partition selection taken by Sain (2002) is based on a pilot kernel density estimate. A pilot kernel density estimate is computed from the data and its sample modes extracted. The data points are then associated with closest sample mode. For the data set, labelled 1–15, in Figure 4.2, we construct a normal reference pilot kernel density estimate and extract its sample modes. There are three of them and they are denoted by the solid triangles. The resulting partition of the data set is then $\{1,2,5,9,11,12\},\{3,4,7,10,13\},\{6,8,14,15\}$.

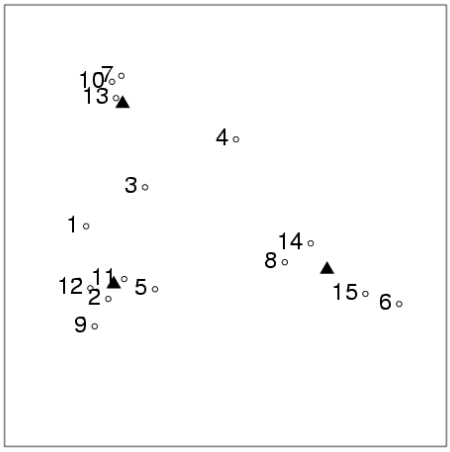

Figure 4.2: Partition based on sample mode allocation

Our approach to selecting the partition is via multivariate clustering. There are many clustering algorithms available as thoroughly described in the monographs by Everitt

(1993) and Gordon (1999). We focus on hierarchical clustering algorithms. These are based on a constructing a whole family of relationships between the data points, based on their *dissimilarity* $d(C_j, C_{j'})$ which is, as its name suggests, a measure of how far apart clusters C_j and $C_{j'}$ are.

1. We start with the data $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ placed into n singleton clusters $C_1, \ldots, C_n$.
2. Compute the dissimilarities for each pair of distinct clusters $d(C_j, C_{j'})$.
3. Fuse together the clusters which have the smallest dissimilarity into a single cluster – there is now one less cluster.
4. Repeat steps 2 – 3 until there is one cluster containing all data points.

From this algorithm we see that we build clusters with increasingly more members so this type of hierarchical clustering are known as agglomerative.

There are many way of measuring the dissimilarity between two clusters including this list given by Gordon (1999, p. 79): single linkage, complete linkage, group average linkage, weighted average linkage, mean dissimilarity, sum of squares, incremental sum of squares, centroid, median. The one that we will use is the group average linkage where

$$d(C_j, C_{j'}) = n_j^{-1} n_{j'}^{-1} \sum_{\boldsymbol{X}_i \in C_j} \sum_{\boldsymbol{X}_{i'} \in C_{j'}} (\boldsymbol{X}_i - \boldsymbol{X}_{i'})^T (\boldsymbol{X}_i - \boldsymbol{X}_{i'})$$

where n_j is the number of data points in C_j. Here we are using the L_2 or Euclidean distance. There are many ways of measuring the dissimilarity between two points - we choose the Euclidean distance as it is the most mathematically tractable. Others include the city block (or Manhattan), Canberra and Minkowski distances, see Gordon (1999, Section 2.2.3). There is a vast literature on the most appropriate choice of dissimilarity and there is not always consensus because the most appropriate choice is dependent on the structure of the data sample. For a summary discussion, consult Everitt (1993, Section 4.4) or Gordon (1999, Section 4.3). We have chosen to use the group average link as, following the conclusions of the above authors, it is not affected by chaining (the tendency to create long sequences of points fused into a cluster even if the end points are far apart), does not impose spherical clusters and is a compromise between the extremes of single and complete linkage.

The hierarchical clustering structure can be represented by a dendogram. A dendogram is an upside-down tree with the root node being the cluster containing all points, splitting as each cluster is divided, until the leaves are the singleton clusters. We illustrate this with a small data example in Figure 4.3. On the right is the data set of 15 points from Figure 4.2. The corresponding dendogram is given on the right. The dendogram gives us

an easy visual device to describe the clusters. For example, we wish to find 3 clusters in this data set: to do this we simply cut the dendogram so that a horizontal line intersects exactly three branches. The cluster memberships can then be read off the dendogram i.e. $\{1, 2, 3, 5, 9, 11, 12\}, \{4, 7, 10, 13\}, \{6, 8, 14, 15\}$.

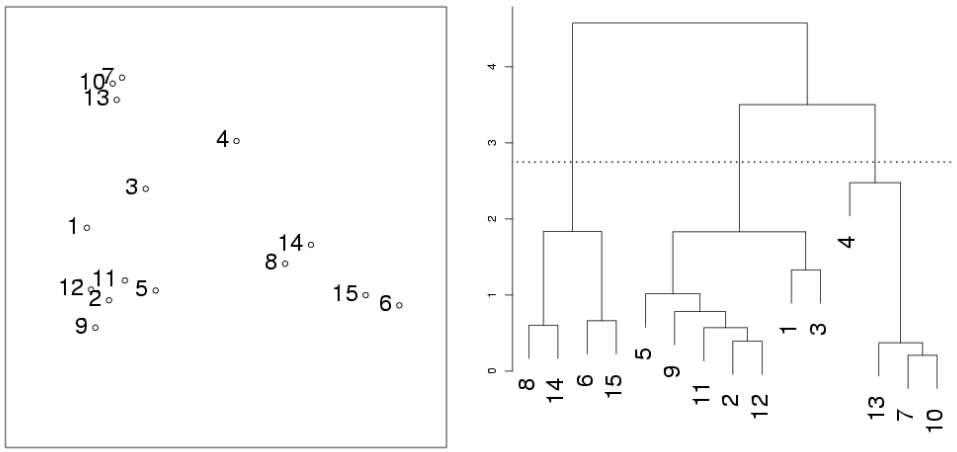

Figure 4.3: Example of dendogram

Deciding the number of clusters in the data set is crucial next step. There are many *stopping rules* to decide this. Milligan & Cooper (1985) conducts an extensive study of 30 stopping rules. One method that these authors recommend is from Duda & Hart (1973, Section 6.12). The advantage of this method is that it can decide whether to divide the whole data set into two clusters. Some of the other methods recommended in Milligan & Cooper (1985) are not designed to do this (i.e. these assume the existence of at least two clusters). This stopping rule is based on finding significant changes in the value of the within-clusters sum of squares, for ν clusters,

$$W(\nu) = \sum_{i=1}^{n} (\boldsymbol{X}_i - \bar{\boldsymbol{X}}_{\alpha(\boldsymbol{X}_i)})^T (\boldsymbol{X}_i - \bar{\boldsymbol{X}}_{\alpha(\boldsymbol{X}_i)}) = \sum_{j=1}^{\nu} \sum_{\boldsymbol{X}_i \in C_j} (\boldsymbol{X}_i - \bar{\boldsymbol{X}}_j)^T (\boldsymbol{X}_i - \bar{\boldsymbol{X}}_j)$$

where $\alpha(\boldsymbol{X}_i) = j$ when $\boldsymbol{X}_i$ belongs to C_j and $\bar{\boldsymbol{X}}_j = n_j^{-1} \sum_{\boldsymbol{X}_i \in C_j} \boldsymbol{X}_i$. Assuming that the data $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ are drawn from a d-variate normal density with mean $\boldsymbol{\mu}$ and variance $\sigma^2 \mathbf{I}_d$, we will use the following hypothesis test

H_0: Population distribution is $N(\boldsymbol{\mu}, \sigma^2 \mathbf{I}_d)$ i.e. there is one cluster

H_1: Not H_0 i.e. there are (at least) two clusters

using the test statistic

$$W = W(2)/W(1).$$

The exact sampling distribution for W is unknown though Duda & Hart (1973) derive the following approximate results.

Under the null hypothesis

$$W(1) \dot{\sim} N(dn\sigma^2, 2dn\sigma^4).$$

We then divide these n data points into two clusters (which is spurious under the null hypothesis) by dividing them with a hyperplane containing the sample mean $\bar{\boldsymbol{X}}$ to obtain

$$W(2) \dot{\sim} N\left(dn\sigma^2 - \frac{2n\sigma^2}{\pi}, 2dn\sigma^4 - \frac{16dn\sigma^4}{\pi^2}\right).$$

This means that W is a ratio of two normal random variables. Duda & Hart then use a normal approximation of W for tractability reasons. Let $Y_1 \sim N(\mu_1, \sigma_1^2)$ and $Y_2 \sim N(\mu_2, \sigma_2^2)$ be univariate normal random variables then

$$\frac{Y_2}{Y_1} \dot{\sim} N\left(\frac{\mu_2}{\mu_1}, \frac{\sigma_2^2}{\mu_1^2}\right)$$

if $\sigma_1^2/\mu_1^2 \to 0$ as $n \to \infty$. The approximate sampling distribution for W is

$$W \dot{\sim} N\left(1 - \frac{2\pi}{d}, \frac{2}{dn_j} - \frac{16}{d^2 n_j \pi}\right)$$

as $2nd\sigma^4/(n^2d^2\sigma^4) = 2/(nd) \to 0$ as $n \to \infty$. We know that $W(2)$ is no larger than $W(1)$ under $\mathbf{H}_0$ so we use a one-sided test. We will reject the null hypothesis at significance level α when

$$W > 1 - \frac{2\pi}{d} + z_{1-\alpha}\sqrt{\frac{2}{dn_j} - \frac{16}{d^2 n_j \pi}}$$

where $z_{1-\alpha}$ is the $(1-\alpha)$-quantile of the standard normal distribution.

This test can be conducted in series. We start with the one cluster containing all n data points, $\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$, and calculate $W(1)$ from these points. We cut the dendogram at two clusters and calculate $W(2)$ and apply this test. If the null hypothesis is accepted then we conclude that there is only one cluster in the data. Otherwise if it is rejected, and we conclude that we have (at least) two clusters. As a result of the hierarchical structure of the clustering, it follows that one of these two clusters remains intact and the other divides into two clusters. We apply the test to the two daughter clusters and the mother cluster and so on, till no more clusters are statistically significant. The value of the individual level of significance α then does not correspond to a combined level of significance since the series of tests are related. Milligan & Cooper (1985) in their simulations trials use a heuristically chosen $z_\alpha = 3.20$ which corresponds to $\alpha = 0.0006871$ whereas we use $\alpha = 0.001$ in our simulation study in Section 4.5. (We also tried $\alpha = 0.01, 0.05$ but these give spurious clusters more often than with $\alpha = 0.001$.)

We have now a method of deciding on the the most appropriate clustering/partition of our data. With this partition, we can then compute LSCV($\boldsymbol{\Omega}$), Equation (4.4), and then find the resulting minimiser bandwidth matrices. We call these *pre-clustered* bandwidth matrices.

4.5 Practical performance for variable bandwidth matrix selectors

The algorithms for pre-clustered bandwidth selectors are similar to their fixed bandwidth counterparts. The main difference is that the data are pre-transformed *then* pre-clustered, ensuring that the pre-clustering is scale independent. This is followed by the numerical optimisation of the appropriate criterion and back-transforming to the original data scale.

Along with the algorithm for the pre-clustered LSCV selector, we describe the algorithm of the Abramson (1982) selector, in the implementation provided by Silverman (1986, Section 5.3), as a benchmark. The description below of Sain (2002)'s selector is slightly different to the one the author uses. Instead of using the exact form of LSCV($\boldsymbol{\Omega}$) as we do, he relies on a binned form. Binning consists dividing the data set into *bins* and then counting the number of data points that fall into these bins. This is a similar procedure for constructing a (multivariate) histogram though here we are not restricted to using hyperrectangular bins. These bins counts can then be used to compute the LSCV. The advantage of binning is that its complexity depends on the number of bins rather than the number of data points which makes it useful in large sample computations. For more details on binning, see Wand & Jones (1995, Appendix D). However for consistency for comparison, we implement it here in its exact form.

4.5.1 Algorithms for variable bandwidth matrix selectors

Algorithm for Abramson LSCV bandwidth matrix selector

1. Compute a pilot density estimate $\tilde{f}(\cdot; \hat{\mathbf{G}}^{\mathrm{NR}})$ with the normal reference selector
$$\hat{\mathbf{G}}^{\mathrm{NR}} = \left[\frac{4}{(d+2)n}\right]^{-2/(d+4)} \mathbf{S}.$$
2. Substitute $\tilde{f}(\cdot; \hat{\mathbf{G}}^{\mathrm{NR}})$ into LSCV$'(h)$, Equation (4.6), and numerically minimise over h to obtain $\hat{h}_{\mathrm{LSCV}'}$.
3. The bandwidth matrices are given by $\hat{\boldsymbol{\Omega}}(\boldsymbol{X}_i) = \hat{h}^2_{\mathrm{LSCV}'} \tilde{f}(\boldsymbol{X}_i; \hat{\mathbf{G}}^{\mathrm{NR}})^{-1} \mathbf{I}$. Note that there are n of these.

Algorithm for Sain partitioned LSCV bandwidth matrix selector

1. Pre-scale the data. Compute a pilot density estimate $\tilde{f}(\cdot; \hat{\mathbf{G}}^{\mathrm{NR}})$ with the normal reference selector
$$\hat{\mathbf{G}}^{\mathrm{NR}} = \left[\frac{4}{(d+2)n}\right]^{-2/(d+4)} \mathbf{S}^*_{\mathcal{D}}.$$
where $\mathbf{S}^*_{\mathcal{D}}$ is the variance of the pre-scaled data

2. Identify the modes of $\tilde{f}$. Associate data points to the nearest mode. This induces a partition of the data $\mathcal{P} = \{P_1, P_2, \ldots, P_\nu\}$ where ν is the number of sample modes.

3. Numerically minimise Equation (4.5), LSCV(ω), with respect to ω to obtain $\hat{\omega} \equiv \{\hat{h}_{\mathrm{LSCV},1}, \ldots, \hat{h}_{\mathrm{LSCV},\nu}\}$. Note that there are ν bandwidths. Back-transform to the original data scale i.e. $\hat{\boldsymbol{\Omega}}(\boldsymbol{X}_i) = \hat{h}^2_{\mathrm{LSCV},j}\mathbf{S}_{\mathcal{D}}$, $\boldsymbol{X}_i \in P_j$.

Algorithm for pre-clustered LSCV bandwidth matrix selectors

1. Pre-cluster the data. This involves choosing a metric, a dissimilarity, a stopping rule and a significance level. This clustering then induces a partition of the data $\mathcal{P} = \{P_1, P_2, \ldots, P_\nu\}$ where ν is the number of clusters.

2. Numerically minimise Equation (4.4), LSCV($\boldsymbol{\Omega}$), over $\boldsymbol{\Omega}$ to obtain $\hat{\boldsymbol{\Omega}} \equiv \{\hat{\mathbf{H}}_{\mathrm{LSCV},1}, \ldots, \hat{\mathbf{H}}_{\mathrm{LSCV},\nu}\}$. Note that there are ν bandwidth matrices.

4.5.2 Simulation results for mixture densities

We perform a simulation study, similar to those of Section 2.5.2 and 3.5.2 except that we replace densities C and F with two new mixture densities. Density A is a base case as before. Density B has two modes which are not widely separated. Density D was noted as providing a challenge to fixed bandwidth selectors, in the previous chapters. Its modes have differing orientations with a small gap separating them. Density E is a trimodal, kurtotic density. Density G is a normal mixture with widely separated modes with components perpendicular to each other. This density is a sort of benchmark density where we expect that the pre-clustered selector should perform well. Density H is similar to density G except that it is a t-mixture. We use a t-mixture to show that pre-clustered selectors do not rely on the normal mixture structure. In Table 4.1, a multivariate t distribution with location parameter $\boldsymbol{\mu}$, scale parameter $\boldsymbol{\Sigma}$ and df degrees of freedom has density

$$t(\boldsymbol{\mu}, \boldsymbol{\Sigma}, df) = \frac{\Gamma((df+d)/2)}{(df\pi)^{d/2}\Gamma(df/2)|\boldsymbol{\Sigma}|^{1/2}}\left[1 + \frac{1}{df}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right]^{-(d+df)/2}.$$

The contour plots for these target densities are in Figure 4.4.

We look at the performance of the following selectors:

- fixed 2-stage SAMSE plug-in selector with pre-scaling (labelled S2)
- fixed LSCV bandwidth matrix selector (labelled L)
- fixed 1-stage SCV bandwidth matrix selector with pre-scaling (labelled SC)
- Abramson (1982) bandwidth matrix selector (labelled AL)

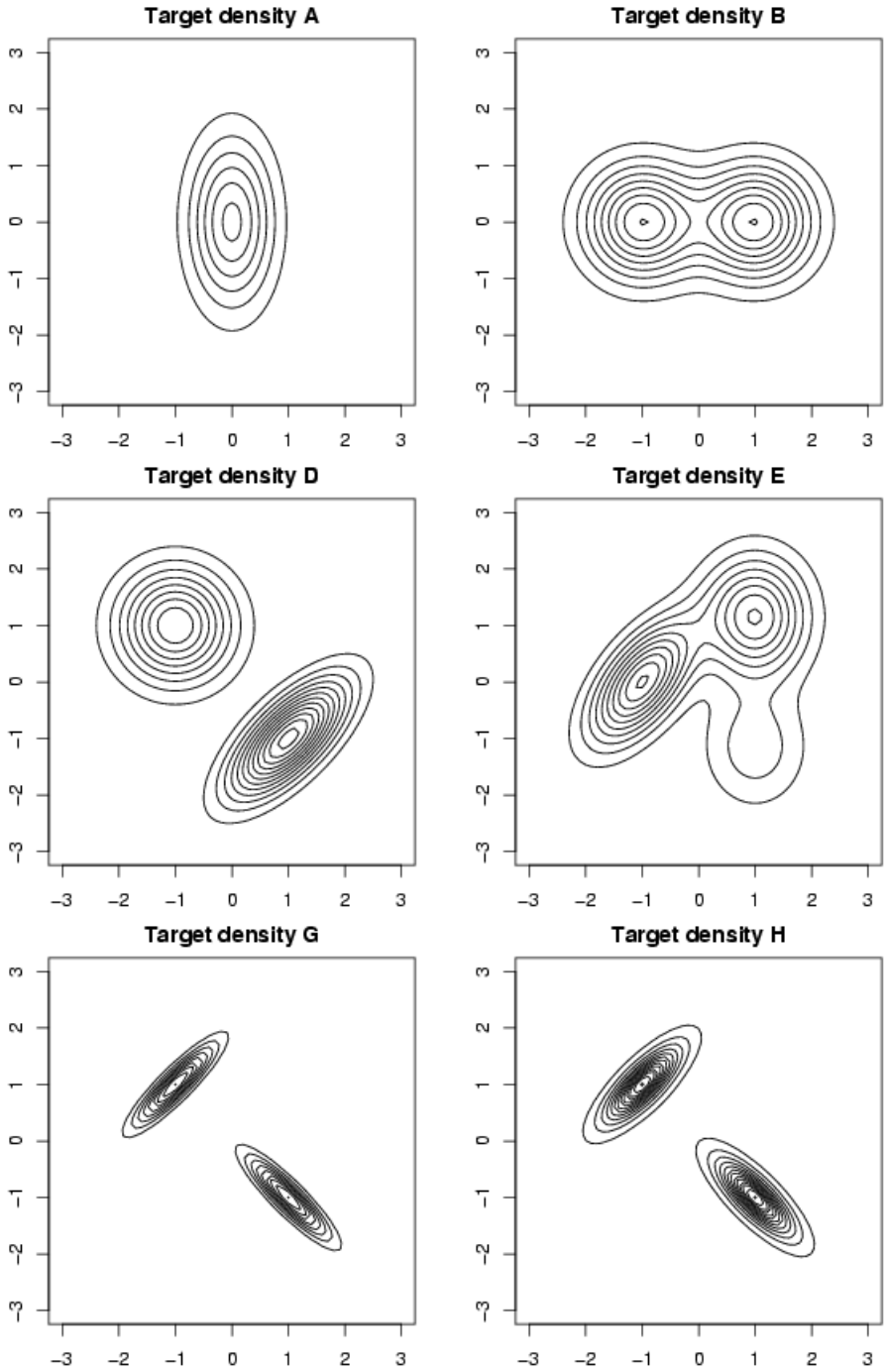

Figure 4.4: Contour plots for target densities A, B, D, E, G & H

Target density	Formula
A	$N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0.25 & 0 \\ 0 & 1 \end{bmatrix} \right)$
B	$\frac{1}{2} N\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & 0 \\ 0 & \frac{4}{9} \end{bmatrix} \right) + \frac{1}{2} N\left(\begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & 0 \\ 0 & \frac{4}{9} \end{bmatrix} \right)$
D	$\frac{1}{2} N\left(\begin{bmatrix} 1 \\ -1 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & \frac{14}{45} \\ \frac{14}{45} & \frac{4}{9} \end{bmatrix} \right) + \frac{1}{2} N\left(\begin{bmatrix} -1 \\ 1 \end{bmatrix}, \begin{bmatrix} \frac{4}{9} & 0 \\ 0 & \frac{4}{9} \end{bmatrix} \right)$
E	$\frac{3}{7} N\left(\begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{9}{25} & \frac{63}{250} \\ \frac{63}{250} & \frac{49}{100} \end{bmatrix} \right) + \frac{3}{7} N\left(\begin{bmatrix} 1 \\ \frac{2}{\sqrt{3}} \end{bmatrix}, \begin{bmatrix} \frac{9}{25} & 0 \\ 0 & \frac{49}{100} \end{bmatrix} \right) + \frac{1}{7} N\left(\begin{bmatrix} 1 \\ -\frac{2}{\sqrt{3}} \end{bmatrix}, \begin{bmatrix} \frac{9}{25} & 0 \\ 0 & \frac{49}{100} \end{bmatrix} \right)$
G	$\frac{1}{2} N\left([-1,1], \begin{bmatrix} \frac{1}{5} & \frac{4}{25} \\ \frac{4}{25} & \frac{1}{5} \end{bmatrix} \right) + \frac{1}{2} N\left([1,-1], \begin{bmatrix} \frac{1}{5} & \frac{4}{25} \\ \frac{4}{25} & \frac{1}{5} \end{bmatrix} \right)$
H	$\frac{1}{2} t\left([-1,1], \begin{bmatrix} \frac{1}{5} & \frac{9}{50} \\ \frac{9}{50} & \frac{1}{5} \end{bmatrix}, 4 \right) + \frac{1}{2} t\left([1,-1], \begin{bmatrix} \frac{1}{5} & -\frac{9}{50} \\ -\frac{9}{50} & \frac{1}{5} \end{bmatrix}, 4 \right)$

Table 4.1: Formulas for target densities A, B, D, E, G & H

- Sain (2002) bandwidth matrix selector with $h^2\mathbf{I}$ parameterisation (labelled SL)
- pre-clustered LSCV bandwidth matrix selector with Euclidean metric with average linkage, the Duda & Hart stopping rule with a significance level of 0.001 (labelled PL)

The labels for the variable bandwidth selectors end in 'L' to denote their dependence on the LSCV. The S2 and SC selectors can be considered to be amongst the best of the fixed selectors from the preceding chapters. We do not use pre-sphering since we have widely separated modes and we know that pre-sphering is not appropriate for these cases as a prelude to clustering and to bandwidth selection. We include the fixed L selector for comparison to its pre-clustered version. The AL and SL selectors serve as benchmarks for variable bandwidth performance. We run 400 trials for each sample size, target density and bandwidth selector combination (except for $n = 1000$, the SL selectors, due to their extremely heavy computational burden, are run for 100 trials).

Before we look at the ISE performance, we examine the performance of the pre-clustering in Table 4.2. The pre-clustering determines the number of bandwidth matrices $\hat{\nu}$ that we use to smooth the data. Our choice of clustering using average linkage with the Duda & Hart stopping rule at 0.001 significance level, performs quite well for the normal mixture densities A, D and G. It does less well for densities B and E whose clusters are not as well separated. It also does less well for density H. The heavy tails of the t-mixtures makes it more difficult to distinguish between clusters, especially for $n = 1000$. (So in this case where we can only find one cluster, we reproduce the fixed L selector.) This may be improved by using other dissimilarity measures, stopping rules and metrics.

We now present the box plots of the log(ISE) in Figure 4.5 for $n = 100$ and in Figure 4.6 for $n = 1000$. (In Appendix B, Table B.7 contains the means and standard deviations of

		$n = 100$				$n = 1000$			
Target density	ν	$\hat{\nu} = 1$	$\hat{\nu} = 2$	$\hat{\nu} = 3$	$\hat{\nu} \geq 4$	$\hat{\nu} = 1$	$\hat{\nu} = 2$	$\hat{\nu} = 3$	$\hat{\nu} \geq 4$
A	1	84.00	15.00	1.00	0.00	97.25	2.25	0.50	0.00
B	2	21.25	78.50	0.25	0.00	55.75	44.00	0.25	0.00
D	2	3.00	96.50	0.50	0.00	10.50	89.25	0.25	0.00
E	3	69.50	11.00	19.00	0.50	59.00	1.50	38.50	1.00
G	2	0.00	95.50	3.25	0.00	0.25	97.50	0.25	2.00
H	2	42.50	57.00	0.05	0.00	93.25	6.75	0.00	0.00

Table 4.2: Percentages for the estimated number of clusters ($\hat{\nu}$) compared to true number of clusters (ν)

the ISE.) The results are mixed: it is surprisingly difficult to improve over the AL selector. This selector is better than all the fixed selectors for all sample sizes and target densities except for density D where it concedes some performance to the S2 and SC selectors. The AL selector outperforms the SL selector in *all* cases presented here. For our PL selector, the comparison is somewhat patchy since it clearly has the lowest median log(ISE) values only for density G. For density H, its performance is a little worse than the AL selector though both are markedly better than the SL and fixed selectors. For the other target densities, A, B, D and E, the PL selector is worse than the two other variable selectors and the fixed selectors S2 and SC, though it has similar performance as the L selector. This suggests that the PL selector best handles target densities that have tight, compact, well separated clusters. By visual inspection of the structure of the densities G and H, it is easy to ascertain that the most appropriate smoothing is to individually smooth each data cluster. Thus the value of the PL selector lies in its ability to perform this differential smoothing automatically.

4.5.3 Results for real data

We analyse the 'Old Faithful' geyser data again, with the variable bandwidth selectors, comparing them to the fixed plug-in and smoothed cross validation selectors. In Figure 4.7, the S2, L and SL estimates have wavy contours for the upper right mode. The PL estimate is able to apply different amounts of smoothing in different areas: the result of clustering for the PL selector divides the data into a lower left cluster (denoted by the triangles) and a upper right cluster (denoted by the circles). For the upper right mode, its contours are both inclined and smooth (like the SC and AL estimates). Moreover, for the lower left mode, its contours this time are still smooth though now aligned to the co-ordinate axes, thus illustrating the flexibility of the PL bandwidths. The AL estimate is similar to the PL one, in that it is able to reproduce the direction and degree of this smoothing. The SL pilot kernel density estimate divides the data into three groups, denoted by circles, triangles and crosses. Though this time the partition is such that the restricted local bandwidth

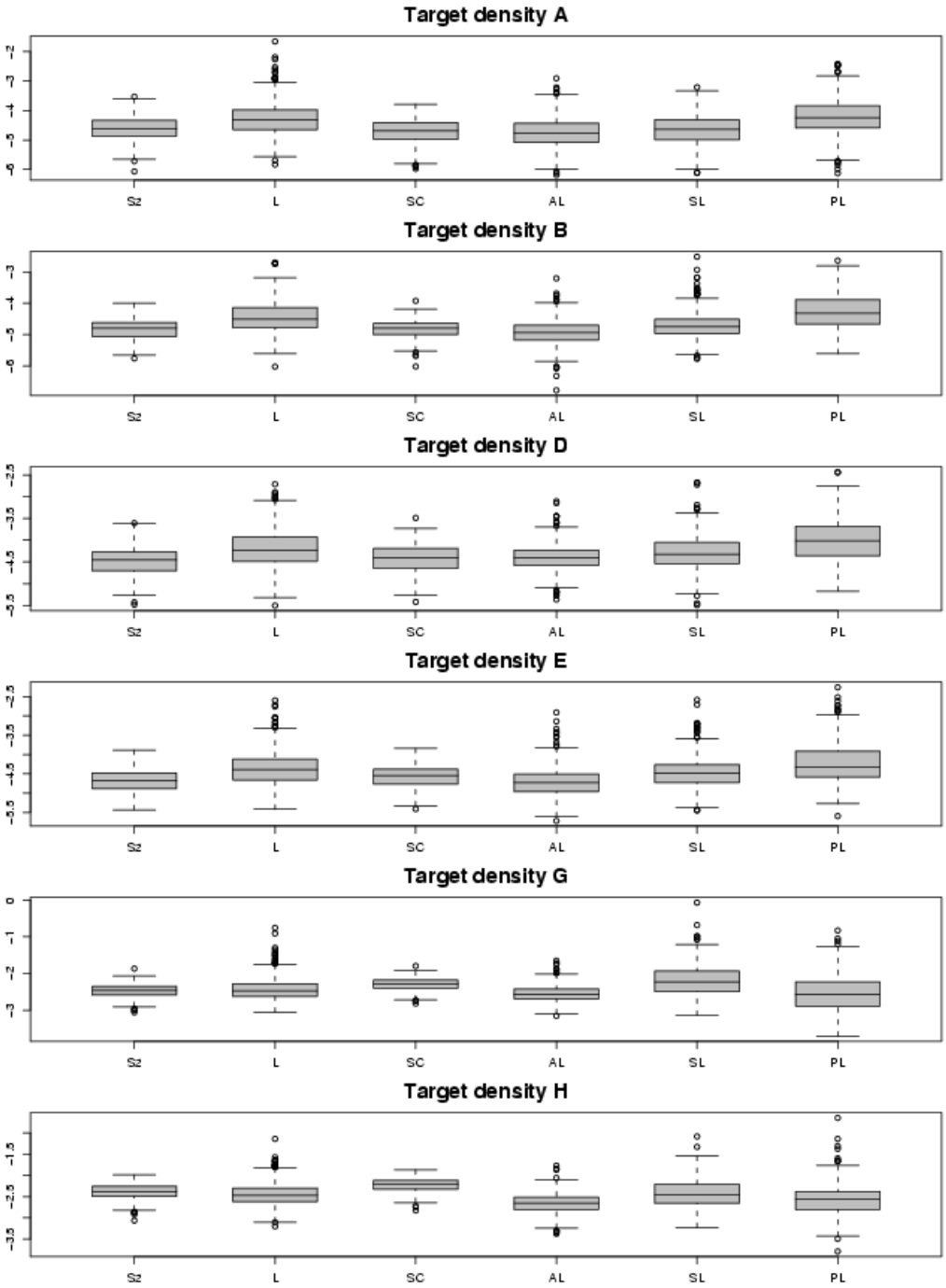

Figure 4.5: Box plots of log(ISE) for fixed and variable selectors, sample size $n = 100$

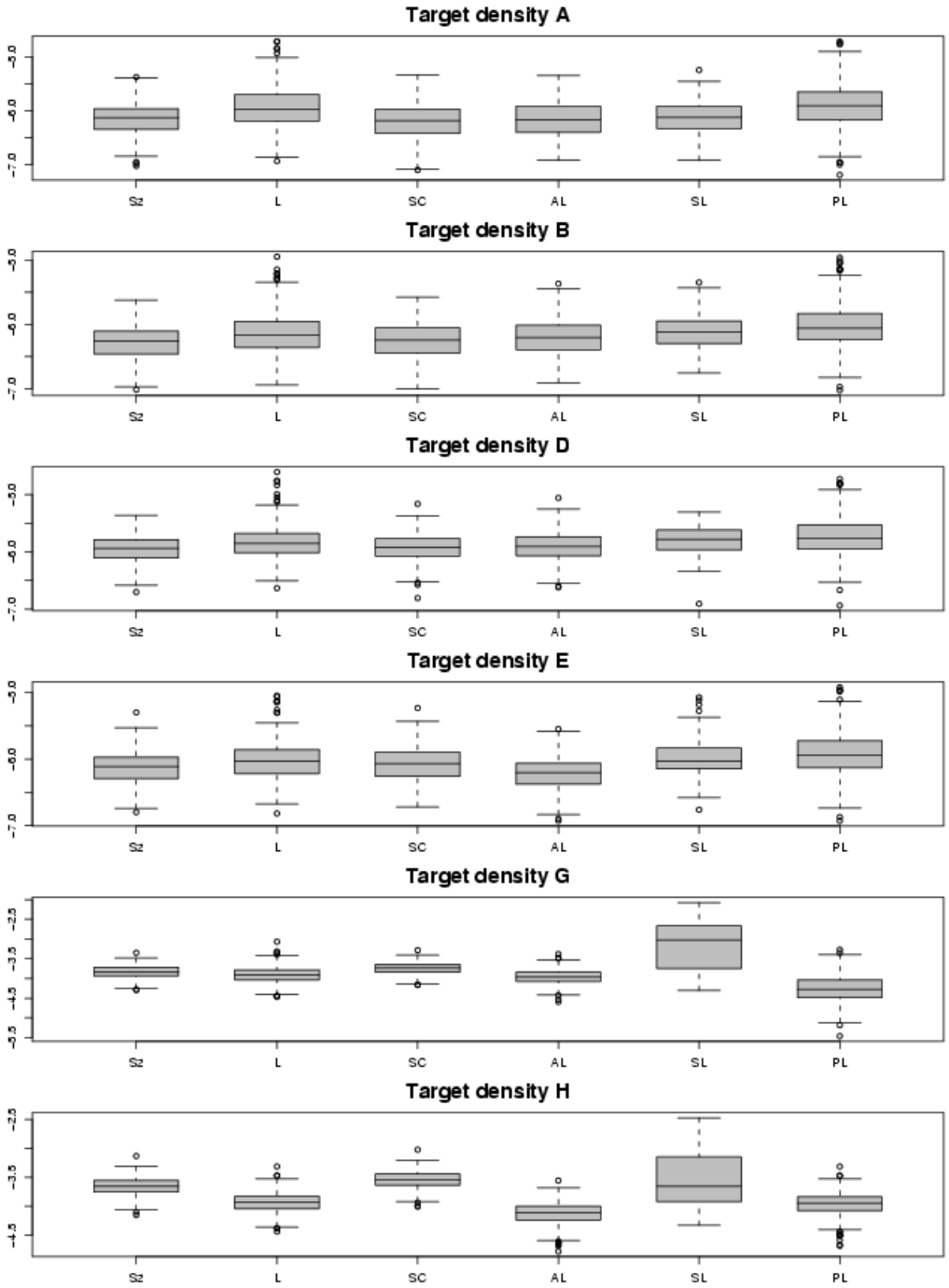

Figure 4.6: Box plots of log(ISE) for fixed and variable selectors, sample size $n = 1000$

matrices are not able to produce appropriate degrees and directions of smoothing. So it does not have the same smoothness of the AL and PL estimates.

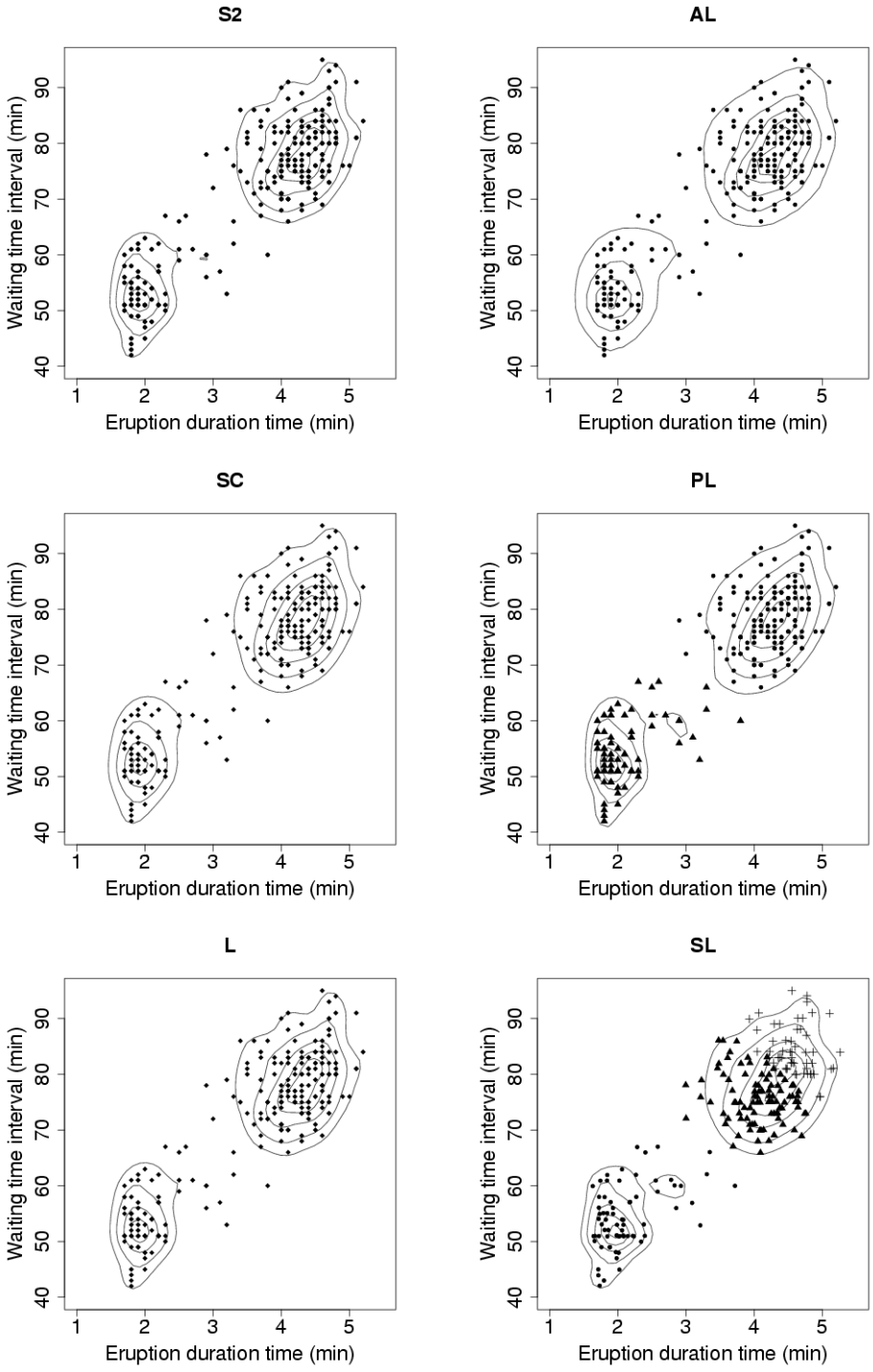

Figure 4.7: 'Old Faithful' geyser data contour plots - fixed and variable selectors – for PL and SL, the different data groups are denoted by circles, triangles and pluses

For the UNICEF data, the density estimates are in Figure 4.8. The PL selector gives

rise to an estimate that appears to be undersmoothed in the lower right half (denoted by the circles) with an overall bimodality, much like the L estimate. The SL pilot estimate also divides the dataset into two classes, producing the bimodality again but with smoother contours. The AL and S2 estimates give contours that are similar to the SL estimate. The SC selector (i.e. a fixed bandwidth selector) gives an estimate that is unimodal. From this example, we see that variable bandwidth selectors can be difficult to calibrate and that fixed bandwidth selectors can be useful even if there is clustered structure in the data set.

4.6 Conclusion

The implementation of a pre-clustered bandwidth selector has been examined here. There are many factors that could affect the performance, e.g. choice of distance function, choice of clustering criterion, choice of stopping rule, that have not been explored fully to search for optimality. However we have demonstrated that the pre-clustered kernel density estimate can extract more structure from the data in certain situations. Our caveat is that the performance of variable bandwidth selectors for finite samples is not always assured to be better than fixed selectors.

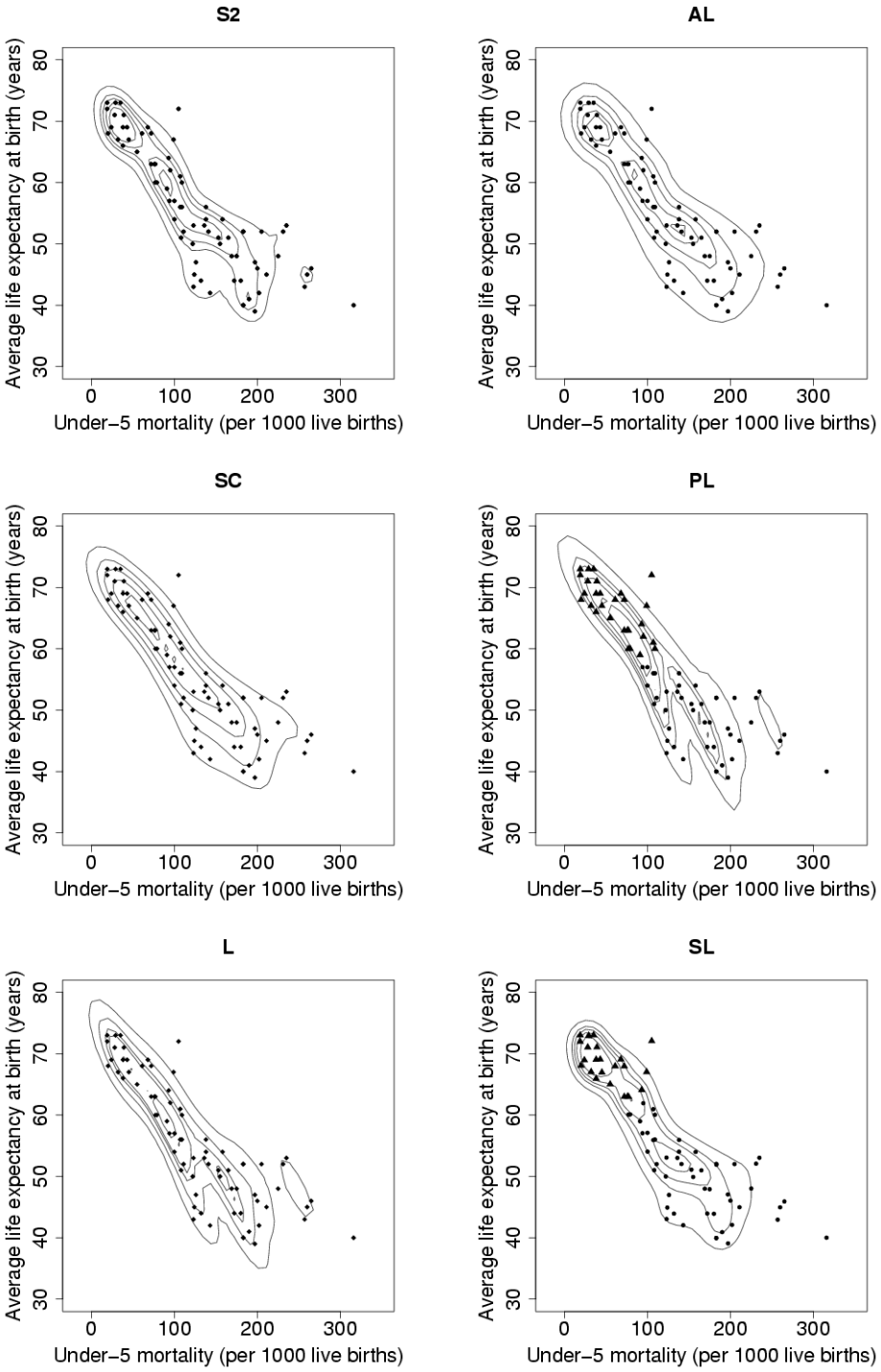

Figure 4.8: Child mortality data contour plots – fixed and variable selectors – for PL and SL, the different data groups are denoted by circles and triangles

Chapter 5

Kernel discriminant analysis

5.1 Introduction

In the previous chapters, we have seen that kernel density estimation is useful and important in its own right, especially for exploratory data analysis. In this chapter, we demonstrate the utility of kernel density estimators as applied to discriminant analysis. Suppose we have a set of ν *populations* or *groups* that correspond to density functions $f_1, f_2, \ldots, f_\nu$. Our aim is to assign all points $\boldsymbol{x}$ from the sample space to one of these groups or densities. We compare the weighted heights of the density functions to obtain the Bayes discriminant rule

$$\boldsymbol{x} \text{ is allocated to group } j_0 \text{ if } j_0 = \underset{j \in \{1,\ldots,\nu\}}{\operatorname{argmax}} \pi_j f_j(\boldsymbol{x}) \tag{5.1}$$

where π_j is the prior probability of drawing from density f_j. If we enumerate for all $\boldsymbol{x}$ from the sample space, we produce a partition $\mathcal{P} = \{P_1, P_2, \ldots, P_\nu\}$ of the sample space using

$$\boldsymbol{x} \in P_j \text{ if } \boldsymbol{x} \text{ is allocated to group } j.$$

The discriminant rule, Equation (5.1), contains the unknown density functions and the (possibly) unknown prior probabilities. Once we collect some data, we can modify this abstract rule into a practical one. We collect *training data* $\boldsymbol{\mathcal{X}}_j = \{\boldsymbol{X}_{j1}, \boldsymbol{X}_{j2}, \ldots, \boldsymbol{X}_{jn_j}\}$, drawn from f_j, for $j = 1, 2, \ldots, \nu$. (The sample sizes n_j are known and non-random.) A priori there is a class structure in the population since we know which data points are drawn from which density function. From these training data, we can construct a practical discriminant rule and subsequent partition. Using this discriminant rule/partition, we classify the *test data* $\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_m$, drawn from $f = \sum_{j=1}^{\nu} \pi_j f_j$. This time, we do not know which populations generated which data points.

An illustration of partitioning and discriminating using this Bayes discriminant rule into three groups is given in Figure 5.4. There are three training sets, each of size 10, denoted by the pluses, diamonds and triangles on the left diagram. The prior probabilities

are equal to 1/3. The three (normal) density functions (not shown) are compared according to Equation (5.1) and this yields the partition on the right: white – pluses, dark grey – diamonds and light grey – triangles. The circles are the 30 test data points that we are attempting to classify.

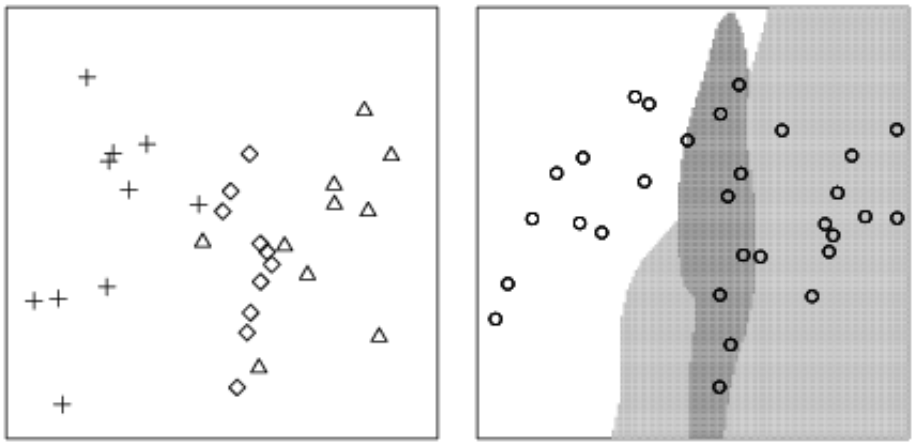

Figure 5.1: Partition and discrimination from discriminant analysis: plus – white, circle – dark grey, triangle – light grey, circles are test data points

The usual approach (and the one used in the above example) is to estimate these density functions (and prior probabilities if needed) and substitute into the discriminant rule. The usual parametric approaches are the well-known and widely used linear and quadratic discriminant techniques. However these suffer from the restrictive assumption of normality. With non-parametric discriminant analysis we relax this assumption and thus are able to tackle more complex cases. We will focus on kernel methods for discriminant analysis. The monographs Silverman (1986, Chapter 6), Scott (1992, Chapter 9) and Simonoff (1996, Chapter 7) contain summaries of kernel discriminant analysis while Hand (1982) contains more detailed and lengthy expositions on this subject.

The structure of this chapter is as follows. In Section 5.2 there is a theoretical exposition of parametric and non-parametric discriminant analysers. The practical performance of kernel discriminant analysers are compared with their linear and quadratic counterparts in Section 5.3 with a simulation study and real data.

5.2 Parametric and non-parametric discriminant analysis

The two parametric methods that we describe in more detail here, linear and quadratic discriminant analysis, are among the most commonly used. Their ease of computation is a result from some underlying normality assumptions: (a) for linear discriminants, we assume that the densities f_j are normal with different mean vectors $\boldsymbol{\mu}_j$ and with common

variance matrix $\mathbf{\Sigma}$ and (b) for quadratic discriminants, we have that the densities are normal with different means $\boldsymbol{\mu}_j$ and different variances $\mathbf{\Sigma}_j$.

For linear discriminant analysis, the key assumption is $f_j \sim N(\boldsymbol{\mu}_j, \mathbf{\Sigma})$. The discriminant rule, Equation (5.1), reduces to (after taking logarithms of f_j)

$$\boldsymbol{x} \text{ is allocated to group } j_0 \text{ if } j_0 = \underset{j\in\{1,\ldots,\nu\}}{\operatorname{argmax}} \ \log(\pi_j) - \tfrac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_j)^T\mathbf{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_j). \tag{5.2}$$

From this equation, we can see that resulting partition is obtained by intersections of ellipsoids with different centres and with the same orientation. This yields partition boundaries that are hyperplanes. For our example data from Figure 5.1, we apply the linear discriminant rule to obtain the partition in Figure 5.2, using the sample mean $\bar{\boldsymbol{X}}_j$ as estimate of $\boldsymbol{\mu}_j$ and $\mathbf{S} = (n-\nu)^{-1}\sum_{j=1}^{\nu} n_j\mathbf{S}_j$ for $\mathbf{\Sigma}$, where $\mathbf{S}_j$ is the sample variance.

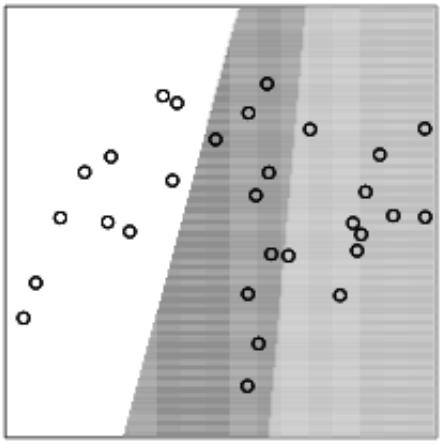

Figure 5.2: Partition from linear discriminant analysis

For quadratic discriminant analysis, we relax the assumption of common variance of linear discriminant analysis i.e. we have $f_j \sim N(\boldsymbol{\mu}_j, \mathbf{\Sigma}_j)$. Equation (5.1) becomes (after taking logarithms of f_j)

$$\boldsymbol{x} \text{ is allocated to group } j_0 \text{ if } j_0 = \underset{j\in\{1,\ldots,\nu\}}{\operatorname{argmax}} \ \log(\pi_j) - \tfrac{1}{2}\log|\mathbf{\Sigma}_j| - \tfrac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_j)^T\mathbf{\Sigma}_j^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_j). \tag{5.3}$$

This discriminant rule yields a partition defined by intersections of ellipsoids with differing centres and orientations. The boundaries are thus piecewise paraboloidal curves, as is illustrated in Figure 5.3, obtained by replacing the means and variances with their sample statistics.

To use the parametric discriminant rules, we replace the unknown parameters with their usual sample estimates. To generalise these parametric methods to a non-parametric one is straightforward. Instead of assuming a normal (or any other parametric) form for the densities, we simply estimate the densities non-parametrically. In our case, we use

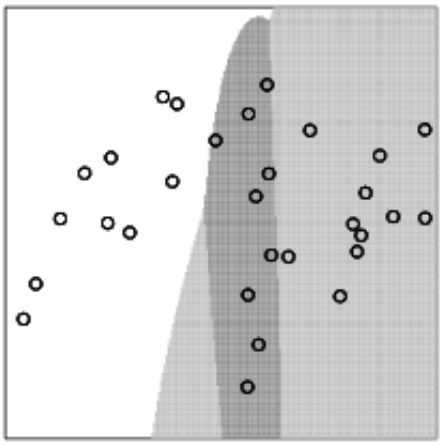

Figure 5.3: Partition from quadratic discriminant analysis

kernel density estimators constructed from the training data. The kernel discriminant rule (KDR) is

$$\text{KDR} : \boldsymbol{x} \text{ is allocated to group } j_0 \text{ if } j_0 = \underset{j\in\{1,\ldots,\nu\}}{\operatorname{argmax}} \hat{\pi}_j \hat{f}_j(\boldsymbol{x}; \mathbf{H}_j) \tag{5.4}$$

where $\hat{f}_j(\boldsymbol{x}; \mathbf{H}_j)$ is the kernel density estimate corresponding to the j-th group. To illustrate its implementation, the resulting partition is in Figure 5.4 where we have used plug-in bandwidth selectors for $\mathbf{H}_j$.

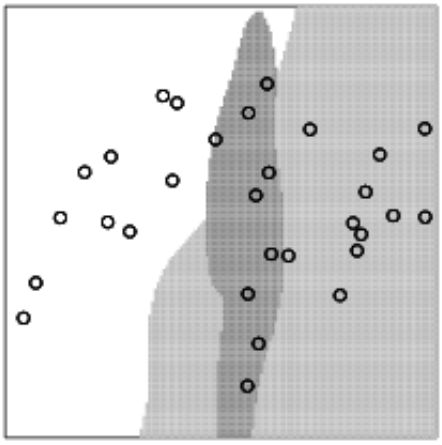

Figure 5.4: Partition from kernel discriminant analysis

Now that we are using kernel density estimators for discriminant analysis, selection of appropriate bandwidths is crucial. Hand (1982) contains discussion on this question. On one hand, we can attempt to find optimal bandwidths for optimal individual kernel density estimates. On the other hand, we could find optimal bandwidths which directly optimise the *misclassification rate* or MR, as Hall & Wand (1988) attempt for the two

class problem. This rate is the proportion of points that are assigned to an incorrect group based on a discriminant rule. Then we have

$$\begin{aligned} 1 - \text{MR} &= \mathbb{P}(\boldsymbol{Y} \text{ is classified correctly}) \\ &= \mathbb{E}_{\boldsymbol{Y}}[\mathbf{1}\{\boldsymbol{Y} \text{ is classified correctly}\}] \\ &= \mathbb{E}_{\boldsymbol{\mathcal{X}}}\left[\mathbb{E}_{\boldsymbol{Y}}[\mathbf{1}\{\boldsymbol{Y} \text{ is classified correctly}\}] \,|\, \boldsymbol{\mathcal{X}}_1, \boldsymbol{\mathcal{X}}_2, \ldots, \boldsymbol{\mathcal{X}}_\nu\right] \end{aligned}$$

where $\mathbb{E}_{\boldsymbol{Y}}$ is expectation with respect to $\boldsymbol{Y}$ or $\sum_{j=1}^{\nu} \pi_j f_j$, and $\mathbb{E}_{\boldsymbol{\mathcal{X}}}$ is expectation with respect to $\boldsymbol{\mathcal{X}}_1, \boldsymbol{\mathcal{X}}_2, \ldots, \boldsymbol{\mathcal{X}}_\nu$ or $\pi_1 f_1, \pi_2 f_2, \ldots, \pi_\nu f_\nu$. Hand recommends the former approach for three reasons. First, accurate estimates of the individual density functions are useful in their own right; second, accurate density estimates can be used in other, more complex discriminant problems which look at measures other than the misclassification rate; and third, direct optimisation with respect to a misclassification rate poses many difficult mathematical obstacles.

Whilst we will not use the misclassification rate to select bandwidths, we will still use it as our performance measure of a discriminant rule. So we need to estimate it. The most appropriate estimate depends on whether we have test data or not. If we do, as is the usual case for simulated data, then a simple estimate is obtained by counting the number of $\boldsymbol{Y}_j$ that are assigned to an incorrect group, divided by the total number of data points m. On the other hand, if we do not have test data, as is the usual case for real data, then we use the cross validation estimate of MR, as recommended by Silverman (1986) and Hand (1982). This involves leaving out each $\boldsymbol{X}_{ji}$, constructing a corresponding leave-one-out density estimate and subsequent discriminant rule. We then compare the label assigned to $\boldsymbol{X}_{ji}$ based on the leave-one-out discriminant rule to its correct group label. These counts are then summed and divided by n.

5.3 Practical performance of kernel discriminant analysis

The algorithm for kernel discriminant analysis is given below. The algorithms for linear and quadratic discriminant analysis are similar except that any kernel methods are replaced by the appropriate parametric methods. We put these algorithms into practice with both simulated and real data.

Algorithm for kernel discriminant analysis

1. For each training sample $\boldsymbol{\mathcal{X}}_j = \{\boldsymbol{X}_{j1}, \boldsymbol{X}_{j2}, \ldots, \boldsymbol{X}_{jn_j}\}, j = 1, 2, \ldots, \nu$, compute a kernel density estimate

$$\hat{f}(\boldsymbol{x}; \mathbf{H}_j) = n_j^{-1} \sum_{i=1}^{n_j} K_{\mathbf{H}_j}(\boldsymbol{x} - \boldsymbol{X}_{ji}).$$

We can use any sensible bandwidth selector $\mathbf{H}_j$.

2. If prior probabilities are available then use these. Otherwise estimate using them using the training sample proportions $\hat{\pi}_j = n_j/n$.

3. (a) Allocate test data points $\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_m$ according to KDR/Equation (5.4) or

   (b) Allocate all points $\boldsymbol{x}$ from the sample space according to KDR/Equation (5.4).

4. (a) If we have test data then the estimate of the misclassification rate is

$$\widehat{\mathrm{MR}} = 1 - m^{-1} \sum_{k=1}^{\nu} \mathbf{1}\{\boldsymbol{Y}_k \text{ is classified correctly using KDR}\}.$$

   (b) If we do not have test data the cross validation estimate of the misclassification rate is

$$\widehat{\mathrm{MR}}_{\mathrm{CV}} = 1 - n^{-1} \sum_{j=1}^{\nu} \sum_{i=1}^{n_j} \mathbf{1}\{\boldsymbol{X}_{ji} \text{ is classified correctly using KDR}_{-ji}\}$$

   where KDR_{-ji} is similar to KDR except that $\hat{\pi}_j$ and $\hat{f}_j(\cdot; \mathbf{H}_j)$ are replaced by their leave-one-out estimates obtained by removing $\boldsymbol{X}_{ji}$ i.e. $\hat{\pi}_{j,-i} = (n_j - 1)/n$ and

$$\hat{f}_{j,-i}(\boldsymbol{x}; \mathbf{H}_{j,-i}) = (n_j - 1)^{-1} \sum_{\substack{i'=1 \\ i' \neq i}}^{n_j} K_{\mathbf{H}_{j,-i}}(\boldsymbol{x} - \boldsymbol{X}_{j,i'}).$$

   That is, we repeat step 3 to classify all $\boldsymbol{X}_{ji}$ using KDR_{-ji}.

5.3.1 Simulation results for normal mixture densities

We conduct a similar comparison to the simulation studies contained in Hand (1982, Chapter 7), examining the performance of the following discriminant analysers:

- linear discriminant (labelled LD)
- quadratic discriminant (labelled QD)
- kernel discriminant with 2-stage AMSE diagonal bandwidth matrices (labelled KDD2)
- kernel discriminant with 2-stage SAMSE full bandwidth matrices (labelled KDS2)
- kernel discriminant with 1-stage SCV full bandwidth matrices (labelled KDSC)

The code for the kernel discriminant analysers are based on the bandwidth matrix selection and density estimation functions in the `ks` library whose details are found in Appendix

C. The code for LDA and QDA are supplied within the base `R` software, namely `lda` and `qda`.

We simulate from the following normal mixture densities for 1000 trials (rather than the 400 trials as previously), using training sample sizes $n = 100$ and 1000, and test data sample size $m = 1000$. We use target densities D and E from previous chapters except that now we keep track of which mixture component an observation is drawn from. Density D contains fairly distinct components and any reasonable discriminant analyser is expected to perform well here. Density E has three components of various shapes and sizes and so is a more challenging case than density D. Density K is a pair of bimodal normal mixtures, with alternating modes. Density L is a large mode separating a bimodal density with narrower modes. For these two latter densities we expect the linear and quadratic discriminant analysers to perform poorly since it is difficult to distinguish the different components using only linear or quadratic cuts. Alternatively we can view densities K and L as being highly non-normal so the assumptions of normality for the parametric methods are invalid. Thus we hope that the kernel methods will demonstrate their efficacy here. The formulas for these target densities are in Table 5.1 and their contour plots are in Figure 5.5.

Before we investigate the long term properties of these discriminant analysers, we look at more detail at the construction of an individual density estimate for density K for $n = m = 100$ points in Figure 5.6 (the size of m is reduced for clarity of presentation). The contours of the different density estimates are denoted by the solid and dashed lines. The circles and triangles are the two groups of test data. The kernel discriminant analysers are all able to detect the alternating bimodality whereas the parametric analysers are unable to do so.

The average and standard deviation of misclassification rates are in Table 5.2. These rates are computed using the simple method. From this table, we see for density D and E, LD has inferior performance compared to QD and the kernel discriminant analysers. For density K, our initial expectations are confirmed: KDD2, KDS2, KDSC all outperform the linear and quadratic counterparts. For density L, the advantage of the kernel methods over the linear method is maintained whilst it is reduced compared to the quadratic method. The increased performance of the kernel discriminant analysers for the latter two densities is apparent for both sample sizes. Moreover, even with the increased burden of selecting an increased number of bandwidths which comprise the bandwidth matrix, the full matrix selectors overall produce smaller standard deviations.

The differences between the diagonal matrix KDD2 and the full matrix KDSC and KDS2 are more subtle than the differences between the kernel methods and the parametric methods. We can see that both full bandwidth matrix methods KDS2 and KDSC in the

Target density	Formula
D	$\pi_1 = \frac{1}{2}, f \sim N\left(\begin{bmatrix}1\\-1\end{bmatrix}, \begin{bmatrix}\frac{4}{9} & \frac{14}{45}\\ \frac{14}{45} & \frac{4}{9}\end{bmatrix}\right)$; $\pi_2 = \frac{1}{2}, f_2 \sim N\left(\begin{bmatrix}-1\\1\end{bmatrix}, \begin{bmatrix}\frac{4}{9} & 0\\0 & \frac{4}{9}\end{bmatrix}\right)$
E	$\pi_1 = \frac{3}{7}, f_1 \sim N\left(\begin{bmatrix}-1\\0\end{bmatrix}, \begin{bmatrix}\frac{9}{25} & \frac{63}{250}\\ \frac{63}{250} & \frac{49}{100}\end{bmatrix}\right)$; $\pi_2 = \frac{3}{7}, f_2 \sim N\left(\begin{bmatrix}1\\\frac{2}{\sqrt{3}}\end{bmatrix}, \begin{bmatrix}\frac{9}{25} & 0\\0 & \frac{49}{100}\end{bmatrix}\right)$; $\pi_3 = \frac{1}{7}, f_3 \sim N\left(\begin{bmatrix}1\\-\frac{2}{\sqrt{3}}\end{bmatrix}, \begin{bmatrix}\frac{9}{25} & 0\\0 & \frac{49}{100}\end{bmatrix}\right)$
K	$\pi_1 = \frac{1}{2}, f_1 \sim \frac{1}{2}N\left(\begin{bmatrix}-\frac{3}{2}\\-\frac{3}{2}\end{bmatrix}, \begin{bmatrix}\frac{4}{5} & -\frac{1}{2}\\-\frac{1}{2} & \frac{4}{5}\end{bmatrix}\right) + \frac{1}{2}N\left(\begin{bmatrix}\frac{1}{2}\\\frac{1}{2}\end{bmatrix}, \begin{bmatrix}\frac{4}{5} & -\frac{1}{2}\\-\frac{1}{2} & \frac{4}{5}\end{bmatrix}\right)$ $\pi_2 = \frac{1}{2}, f_2 \sim \frac{1}{2}N\left(\begin{bmatrix}\frac{3}{2}\\\frac{3}{2}\end{bmatrix}, \begin{bmatrix}\frac{4}{5} & -\frac{1}{2}\\-\frac{1}{2} & \frac{4}{5}\end{bmatrix}\right) + \frac{1}{2}N\left(\begin{bmatrix}-\frac{1}{2}\\-\frac{1}{2}\end{bmatrix}, \begin{bmatrix}\frac{4}{5} & -\frac{1}{2}\\-\frac{1}{2} & \frac{4}{5}\end{bmatrix}\right)$
L	$\pi_1 = \frac{1}{3}, f_1 \sim \frac{1}{2}N\left(\begin{bmatrix}-\frac{3}{2}\\0\end{bmatrix}, \begin{bmatrix}\frac{3}{10} & \frac{1}{4}\\\frac{1}{4} & \frac{3}{10}\end{bmatrix}\right) + \frac{1}{2}N\left(\begin{bmatrix}\frac{3}{2}\\0\end{bmatrix}, \begin{bmatrix}\frac{3}{10} & \frac{1}{4}\\\frac{1}{4} & \frac{3}{10}\end{bmatrix}\right)$ $\pi_2 = \frac{2}{3}, f_2 \sim N\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}\frac{4}{5} & \frac{2}{5}\\\frac{2}{5} & 1\end{bmatrix}\right)$

Table 5.1: Formulas for target densities D, E, K & L

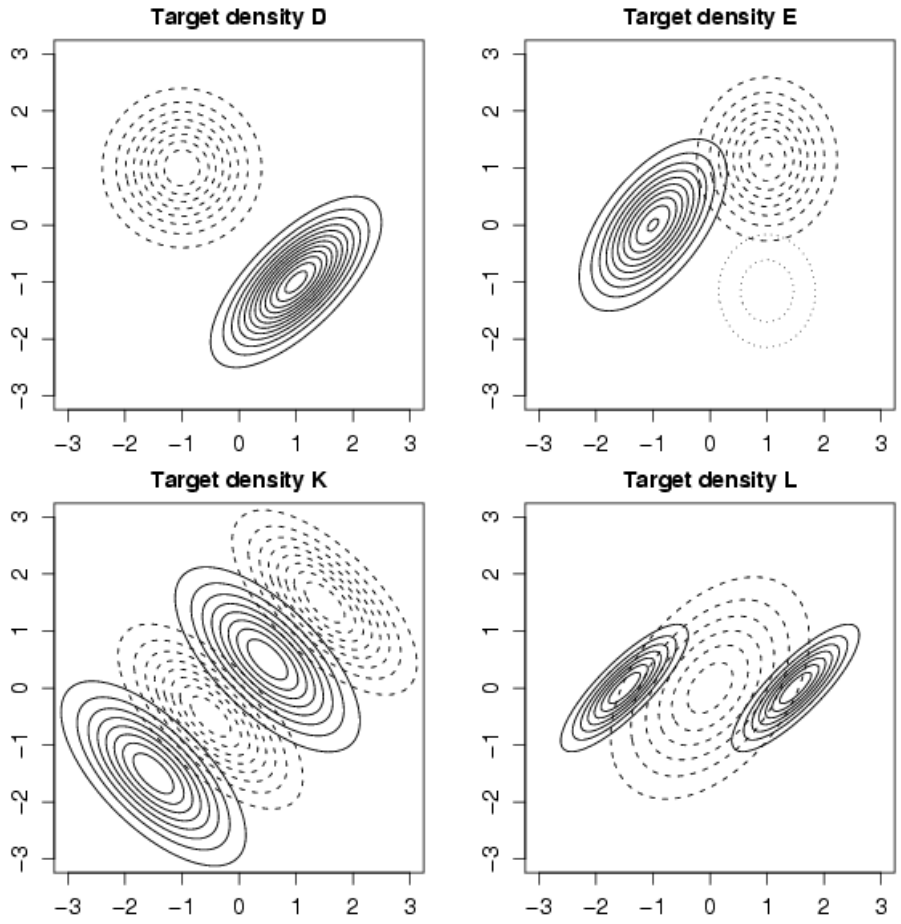

Figure 5.5: Contour plots for target densities D, E, K, L for discriminant analysis: solid contours – $\pi_1 f_1$, dashed lines – $\pi_2 f_2$ and dotted lines – $\pi_3 f_3$.

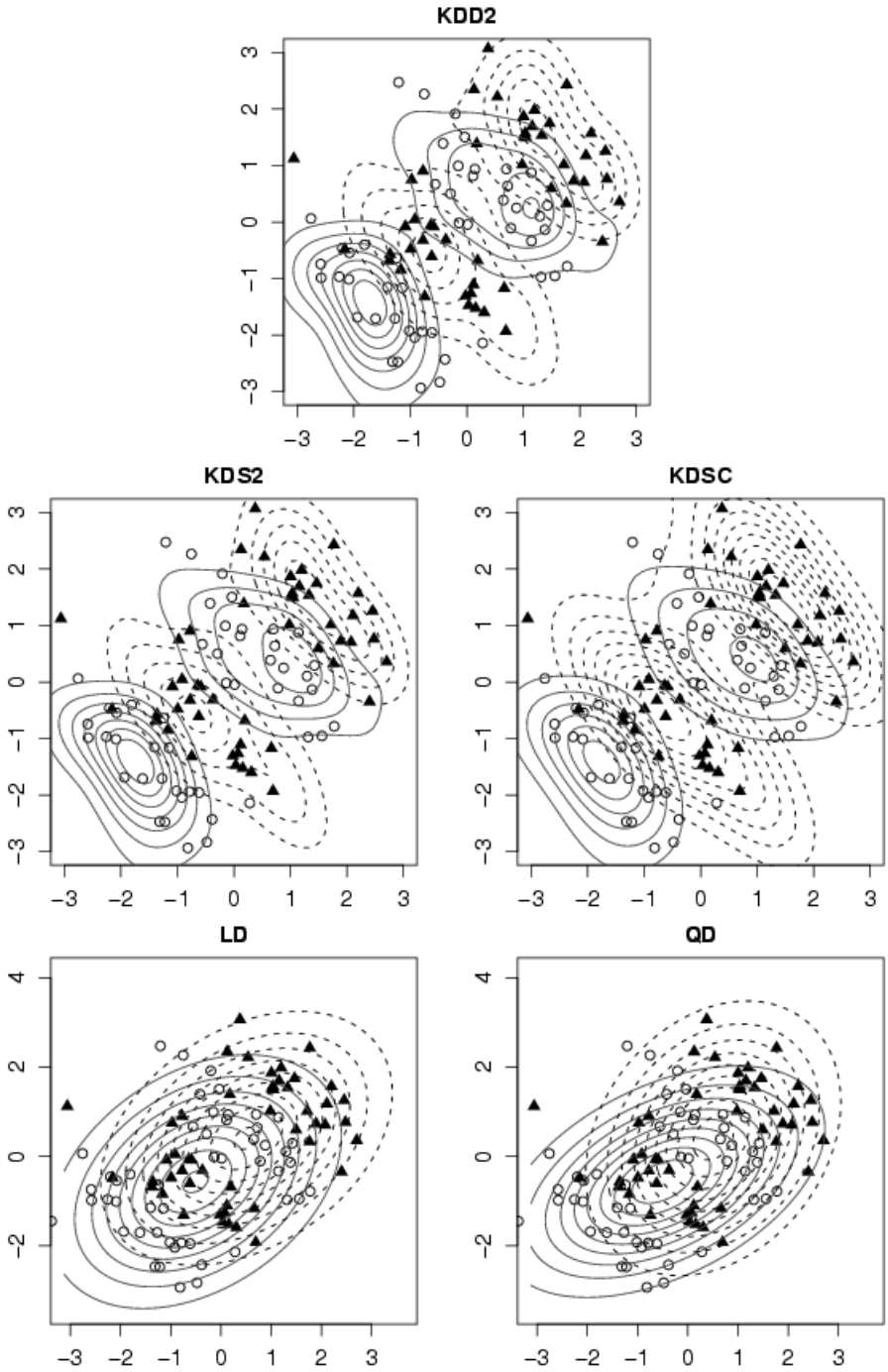

Figure 5.6: Kernel density estimates for discriminant analysers for density K: circle – solid line, triangle – dotted line. Circles and triangles are test data

Target density		Misclassification rate				
		KDD2	KDS2	KDSC	LD	QD
		$n = 100, m = 1000$				
D	mean	0.0051	0.0049	0.0041	0.0089	0.0036
	SD	0.0031	0.0029	0.0024	0.0036	0.0020
E	mean	0.0741	0.0738	0.0709	0.0701	0.0675
	SD	0.0109	0.0108	0.0099	0.0093	0.0091
K	mean	0.1094	0.1032	0.0994	0.4505	0.4431
	SD	0.0141	0.0127	0.0120	0.0232	0.0203
L	mean	0.1514	0.1495	0.1503	0.3408	0.1669
	SD	0.0160	0.0157	0.0171	0.0179	0.0205
		$n = 1000, m = 1000$				
D	mean	0.0032	0.0032	0.0031	0.0084	0.0029
	SD	0.0017	0.0017	0.0017	0.0029	0.0017
E	mean	0.0640	0.0640	0.0635	0.0678	0.0625
	SD	0.0080	0.0079	0.0078	0.0079	0.0078
K	mean	0.0895	0.0885	0.0878	0.4684	0.4666
	SD	0.0090	0.0088	0.0088	0.0066	0.0068
L	mean	0.1287	0.1272	0.1265	0.3340	0.1544
	SD	0.0108	0.0108	0.0107	0.0000	0.0116

Table 5.2: Misclassification rates for discriminant analysers

majority of cases considered here have lower mean misclassification rates than KDD2. Table 5.3 contains the value and standard errors for the pairwise differences in mean misclassification rate for KDD2, KDS2 and KDSC. Our guide to statistical significance is if the the absolute value of the difference in mean misclassification rates is more than twice the standard error. From the table, KDS2 has significantly lower misclassification rates than KDD2 for densities K and L; and that KDSC is significantly lower than both KDD2 for the same densities (except for density L, $n = 100$). As for the differences between KDSC and KDS2, it is not clear that they are overall significantly different.

5.3.2 Results for real data

A real data set that has been previously analysed with kernel discriminants is the MBA GMAT–GPA (Master of Business Administration Graduate Management Admissions Test – Grade Point Average) data from Simonoff (1996). The data consist of pairs of GMAT and GPA scores for 61 second year students at the Stern Business School at New York University in 1995. There are 13 women and 48 men with prior probabilities $\pi_{\text{female}} = 0.35, \pi_{\text{male}} = 0.65$. Simonoff investigates the performance of a kernel discriminant anal-

Target density		Difference in mean misclassification rate KDD2 – KDS2	KDD2 – KDSC	KDS2 – KDSC
		$n = 100, m = 1000$		
D	difference in means	0.00017	0.00091*	0.00074*
	SE (difference)	0.00013	0.00012	0.00012
E	difference in means	0.00030	0.00320*	0.00290*
	SE (difference)	0.00049	0.00047	0.00046
K	difference in means	0.00614*	0.00993*	0.00380*
	SE (difference)	0.00060	0.00059	0.00055
L	difference in means	0.00184*	0.00107	-0.00077
	SE (difference)	0.00071	0.00074	0.00073
		$n = 1000, m = 1000$		
D	difference in means	0.00000	0.00012	0.00013
	SE (difference)	0.00008	0.00008	0.00008
E	difference in means	0.00002	0.00052	0.00050
	SE (difference)	0.00036	0.00036	0.00035
K	difference in means	0.00099*	0.00167*	-0.00068
	SE (difference)	0.00040	0.00040	0.00039
L	difference in means	0.00149*	0.00219*	0.00070
	SE (difference)	0.00048	0.00048	0.00048

Table 5.3: Difference in mean misclassification rates for kernel discriminant analysers – the asterisk indicates approximate statistical significance at 5%

ysers with diagonal bandwidth matrices when attempting to discriminate by sex. We supplement this by adding our versions with full bandwidth matrices. Simonoff uses the subjectively chosen bandwidths $\mathbf{H}_{\text{female}} = \text{diag}(2025, 0.0144), \mathbf{H}_{\text{male}} = \text{diag}(625, 0.0225)$ and bandwidths from a diagonal LSCV criterion. Simonoff reports a cross validated the misclassification rate for the diagonal LSCV selectors to be 0.21. However he does not report a rate for his subjectively chosen bandwidth. So we simply calculate a cross validated MR estimate using this same bandwidth for each leave-one-out stage; and we obtain 0.23. For our selectors, we have 0.21 for KDD2, 0.18 for KDS2 and 0.16 for KDSC. For the parametric estimators, LD has rate 0.28 and QD 0.20. We can see that the kernel methods, with appropriately chosen bandwidth matrices, outperform the parametric methods; and that the kernel methods with full bandwidth matrices outperform those with diagonal bandwidth matrices. The partitions obtained for these discriminant analysers are in Figure 5.7, with females represented by triangles and males circles. The partitions classes are grey for females and white for males.

The other real data set with which we compare the different discriminant analysers is the reef data, taken from Bowman & Azzalini (1997). These data were collected during

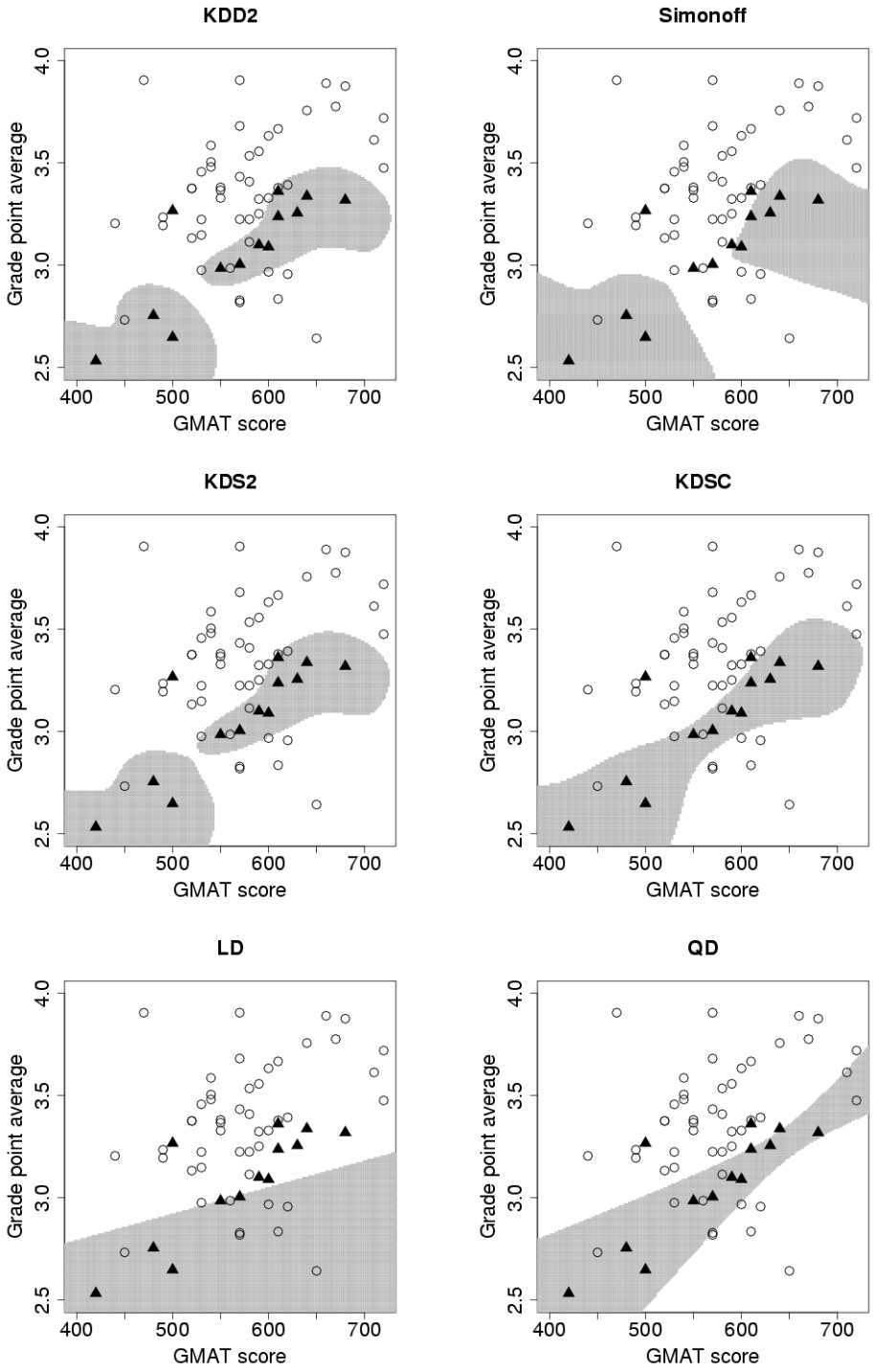

Figure 5.7: Partition of MBA GMAT–GPA data: females – triangles, grey; males – circles, white

a survey of the sea bed wildlife lying between the coast and the Great Barrier Reef in north-eastern Australia. We use a subset of this data set for our analysis: the 149 latitude and longitude measurements (in degrees) of the sampling point (with negative degrees signifying south of the Equator) along with the depth (in metres) of the sea bed. The pairs of longitude and latitude are classified into three categories of sea bed depth: depth ≤ 20 m, 20 m $\leq$ depth < 32 m and depth ≥ 32 m. There are 30, 82 and 37 observations respectively. We wish to classify all points to a depth class based solely on their longitude and latitude. This time we do not have any prior probabilities so we use the sample proportions.

The cross validation misclassification rates for the kernel discriminants are KDD2 – 0.309, KDS2 – 0.309 and KDSC – 0.322. For the parametric discriminants, they are LD – 0.443 and QD – 0.430. Like the MBA–GMAT data, the kernel discriminant analysers substantially outperform their parametric versions. Within the kernel discriminant analysers, all three exhibit similar performance. The partitions that result are in Figure 5.8. The three depth classes are denoted by the circles, triangles and pluses; and their partitions classes are white, light grey and dark grey.

5.4 Conclusion

The flexibility of kernel density estimators to reproduce a wide range of density shapes has been implemented advantageously in the non-parametric discriminant analysis setting. The usual linear and quadratic discriminant analysers are unable to cope with highly non-normal data whereas kernel discriminants encounter no such problem. As is usual for kernel methods, diagonal bandwidth matrices are currently the norm. Our novel contribution has been to apply full bandwidth selectors to the problem. This can possibly lead to improvements in the performance of unconstrained kernel discriminant analysers over their restricted diagonal counterparts.

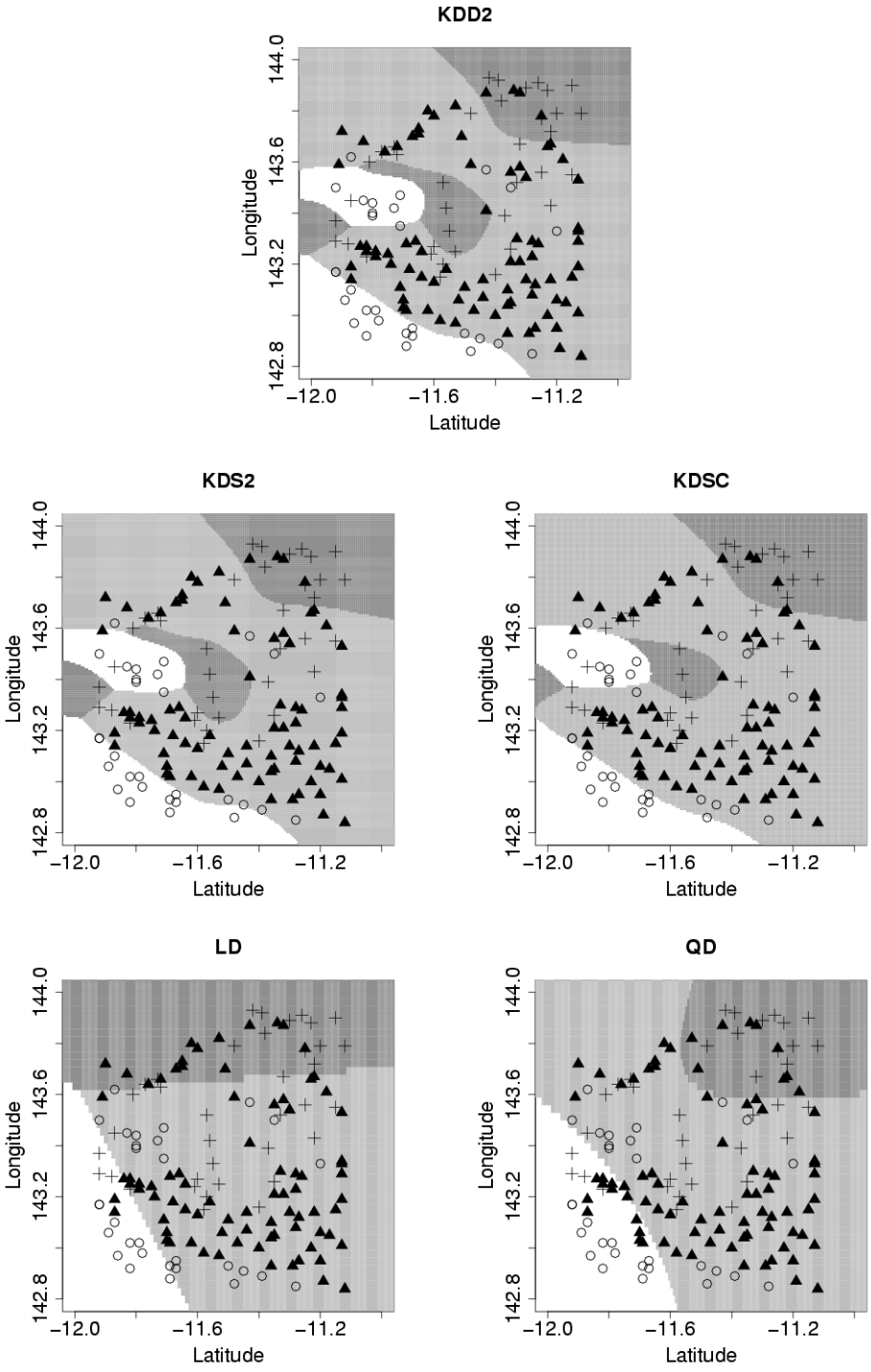

Figure 5.8: Partition of reef longitude–latitude data: shallow depth – circle, white; middle depth – triangle, light grey; deepest depth – plus, dark grey

Chapter 6

Conclusion

Our stated aim for this thesis was to develop solid theory for full bandwidth matrices for multivariate kernel density estimation and then to demonstrate their utility in practice. At this point, we summarise the progress we have made towards this aim.

6.1 Fixed bandwidth selectors

Fixed bandwidth selectors occupy the bulk of this thesis and it is for them that we make the most substantial theoretical and practical progress. The current method for fixed plug-in selectors is based on the AMSE pilot selection of Wand & Jones (1994). This pilot selector works well with diagonal bandwidth matrices but is less effective for full bandwidth matrices. Our innovation has been to provide an alternative SAMSE pilot selector for full bandwidth matrices. This pilot benefits from the positive definiteness of $\hat{\mathbf{\Psi}}_4$ and parsimony (when compared to the AMSE pilot). The current method for pilot estimation for smoothed cross validation is restricted to the scalar bandwidths. Our innovation has been to extend it to full bandwidth matrices. To provide the theoretical justifications for our innovations, we supplied asymptotic relative convergence rates. Although we did not provide any new LSCV or BCV selectors, we supplied their convergence rates, using the same mathematical framework thus providing a unified analysis of all selectors considered in this thesis.

For fixed univariate bandwidth selectors, the understanding of their behaviour with respect to MISE, is fairly complete. The plug-in selectors of Sheather & Jones (1991) can be considered to have the overall best performance. On the theoretical side, these selectors have small asymptotic variance and have the fast asymptotic relative rates of convergence to h_{MISE}. On the practical side, they have good performance for finite samples, considered over a wide range of simulated and real data sets.

For the fixed multivariate bandwidth selectors we considered, we saw that in Chapter 2 the (2-stage) plug-in selectors again show themselves to be efficacious and in Chapter 3

smoothed cross-validation selectors do so likewise. These selectors however still have two unresolved issues that prevent a similar claim being made for the best overall bandwidth matrix selector, as we now describe.

The first issue is how to measure the closeness between a bandwidth selector and the MISE optimal bandwidth. In the univariate case an expansion of the $\operatorname{MISE}(\hat{h})$ about h_{MISE} is

$$\operatorname{MISE}(\hat{h}) = \operatorname{MISE}(h_{\operatorname{MISE}}) + \tfrac{1}{2}(\hat{h} - h_{\operatorname{MISE}})^2 \left[\frac{\partial^2}{\partial h^2}\operatorname{MISE}(h_{\operatorname{MISE}})\right]^{-1} [1 + o(1)]$$

From this expansion we can see that finding the $\hat{h}$ such that $\operatorname{MISE}(\hat{h})$ is as close as possible to $\operatorname{MISE}(h_{\operatorname{MISE}})$ is asymptotically equivalent to finding the $\hat{h}$ that is as close as possible to h_{MISE} i.e. minimising $(\hat{h} - h_{\operatorname{MISE}})^2$. On the other hand, a multivariate expansion of the MISE is

$$\begin{aligned}\operatorname{MISE}(\hat{\mathbf{H}}) = \operatorname{MISE}(\mathbf{H}_{\operatorname{MISE}}) + \tfrac{1}{2}\operatorname{vech}^T(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{MISE}}) \left[D^2_{\mathbf{H}}\operatorname{MISE}(\mathbf{H}_{\operatorname{MISE}})\right]^{-1} \\ \times \operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{MISE}})[1 + o(1)].\end{aligned}$$

We can see that if we wish to find $\hat{\mathbf{H}}$ such that $\operatorname{MISE}(\hat{\mathbf{H}})$ is as close as possible to $\operatorname{MISE}(\mathbf{H}_{\operatorname{MISE}})$ then asymptotically we should be looking for $\hat{\mathbf{H}}$ such that the quadratic term is as small as possible. Of course this is impossible without knowing $D^2_{\mathbf{H}}\operatorname{MISE}(\mathbf{H}_{\operatorname{MISE}})$ which is difficult to estimate. In this thesis, we have simplified the situation by seeking instead to find the smallest unweighted sum of the differences between $\hat{\mathbf{H}}$ and $\mathbf{H}_{\operatorname{MISE}}$ i.e. $\operatorname{vech}^T(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{MISE}})\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{MISE}})$ which is taking a direct analogue from the one-dimensional case. We believe that taking into account the weighting of the Hessian, i.e. selecting $\hat{\mathbf{H}}$ based on minimising $\operatorname{vech}^T(\hat{\mathbf{H}}-\mathbf{H}_{\operatorname{MISE}})[D^2_{\mathbf{H}}\operatorname{MISE}(\mathbf{H}_{\operatorname{MISE}})]^{-1}\operatorname{vech}(\hat{\mathbf{H}}-\mathbf{H}_{\operatorname{MISE}})$, may improve the performance of these selectors.

The second issue concerns the parameterisation of pilot bandwidth matrices. We believe that improvements may be possible if we use a more general parameterisation, especially for the first stage of pilot bandwidth selection. For plug-in selectors, we have supplied an algorithm for selecting an appropriate scalar pilot bandwidth. We started with $\operatorname{MSE}\hat{\psi}_{\boldsymbol{r}}(\mathbf{G})$ in Section 2.2.1 and seek the minimiser of this. Our task is simplified by using $\mathbf{G} = g^2\mathbf{I}$. For smoothed cross validation selectors, we start with

$$\operatorname{tr}\operatorname{MSE}(\operatorname{vech}\hat{\mathbf{H}}; \mathbf{G}) = \mathbb{E}[\operatorname{vech}^T(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{AMISE}})\operatorname{vech}(\hat{\mathbf{H}} - \mathbf{H}_{\operatorname{AMISE}})]$$

where $\hat{\mathbf{H}} = \hat{\mathbf{H}}(\mathbf{G})$ and seek the $\mathbf{G}$ that minimises this. Again we simplify our task by restricting $\mathbf{G}$ to be $g^2\mathbf{I}$. If we were to use the full matrix form for pilot selectors then full bandwidth matrices would be entrenched throughout the entire bandwidth selection algorithm. Implementing these would be future avenues of investigation.

6.2 Variable bandwidth selectors

The ideas behind variable bandwidth matrices are conceptually simple. It appears that varying the bandwidth to vary to the amount of smoothing according to the local conditions would lead to improvements in performance. Unfortunately implementing these variable selectors is extremely difficult. The sample point selector has had more success with practical algorithms than the balloon version. Abramson's selector is the benchmark in variable kernel density estimation. Instead of generalising this for full bandwidth matrices, we have taken a side path into partitioned kernel density estimators where the bandwidth matrix function is a fixed (full) bandwidth matrix within each partition class. We select our partition using multivariate clustering so the performance depends heavily on the latter. These selectors have shown some promise, outperforming Abramson's selector in certain cases. We have only considered only a small range of possibilities for these partitioned selectors and so further research is required.

6.3 Discriminant analysis

Non-parametric discriminant analysis is widely recognised as superior to parametric discriminant analysis. Most attempts so far at kernel discriminant analysis have focused on diagonal bandwidth matrices. We apply the advantages of full bandwidth matrices for density estimation to discriminant analysis. We see that in more complicated discriminant problems, full bandwidth matrices can give extra flexibility to yield a more accurate discrimination.

Appendix A

Notation

Vectors and matrices

Let $\mathbf{A}$ be a $d \times d$ matrix with elements $[\mathbf{A}]_{ij}$ and $\boldsymbol{a}$ be a d-vector with elements $[\boldsymbol{a}]_i$.

vec $\mathbf{A}$ is vector obtained by vertically stacking elements of $\mathbf{A}$
vech $\mathbf{A}$ is vector obtained by vertically stacking elements of lower triangular half of $\mathbf{A}$
dg $\mathbf{A}$ is $\mathbf{A}$ with all its non-diagonal elements set to zero
$\mathbf{D}_d$ is duplication matrix of order d
$\mathbf{I}, \mathbf{I}_d$ is $d \times d$ identity matrix
$\mathbf{J}, \mathbf{J}_d$ is $d \times d$ matrix of ones
$|\boldsymbol{a}|$ is sum of elements of $\boldsymbol{a}$
$\|\boldsymbol{a}\|$ is Euclidean norm of $\boldsymbol{a}$
$|\mathbf{A}|$ is determinant of $\mathbf{A}$
$\boldsymbol{e}_i$ is i-th elementary vector
$\mathbf{E}_{ij}$ is (i, j)-th elementary matrix
$\otimes$ is Kronecker product operator
d' is $\frac{1}{2}d(d+1)$ is dimension of vech'ed $d \times d$ matrix

Functions, constants, variables

f is unknown target density function
K is unscaled kernel function
$K_{\mathbf{H}}$ is scaled kernel function, scaled with bandwidth $\mathbf{H}$
$\mu_j(K)$ is j-th central moment of K
$\operatorname{supp}(K, \boldsymbol{x})$ is support of $K(\cdot - \boldsymbol{x})$
$f * g$ is convolution of functions f and g
$R(f)$ is $\int_{\mathbb{R}^d} f(\boldsymbol{x})^2 \, d\boldsymbol{x}$
$Df(\boldsymbol{x})$ is derivative of f with respect to $\boldsymbol{x}$

$D^2 f(\boldsymbol{x})$ is Hessian of f with respect to $\boldsymbol{x}$
$D_{\mathbf{H}} f(\boldsymbol{x})$ is derivative of f with respect to $\operatorname{vech} \mathbf{H}$
$D^2_{\mathbf{H}} f(\boldsymbol{x})$ is Hessian of f with respect to $\operatorname{vech} \mathbf{H}$
$f^{(\boldsymbol{r})}(\boldsymbol{x})$ is $\boldsymbol{r}$-th partial derivative of f with respect to $\boldsymbol{x}$ where $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$
$\psi_{\boldsymbol{r}}$ is integrated density derivative functional
$\boldsymbol{\Psi}_4$ is matrix of fourth order $\psi_{\boldsymbol{r}}$ functionals
$\boldsymbol{\Theta}_6$ is matrix of sixth order $\psi_{\boldsymbol{r}}$ functionals
$\phi_{\boldsymbol{\Sigma}}(\boldsymbol{x} - \boldsymbol{\mu})$ is multivariate normal density with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$
$\boldsymbol{\varphi}_{\mathbf{A}}(\boldsymbol{x})$ is $\phi_{\mathbf{A}}(\boldsymbol{x}) \operatorname{vec}(\mathbf{A}^{-1}\boldsymbol{x}\boldsymbol{x}^T\mathbf{A}^{-1} - \mathbf{A}^{-1})$
$\mathcal{H}$ is the space of all symmetric positive definite matrices
$B(\boldsymbol{x}, \epsilon)$ is ball with centre $\boldsymbol{x}$ and radius ϵ
$\mathcal{P} = \{P_1, P_2, \ldots, P_\nu\}$ is partition with ν classes of sample space
f_{P_j} is f restricted to P_j
$\boldsymbol{\Psi}_{4,P_j}$ is $\boldsymbol{\Psi}_4$ restricted to P_j
π_j is probability mass of f in P_j, for variable kernel density estimation

Error measures

ISE is Integrated Squared Error
MSE is Mean Squared Error
AMSE is Asymptotic Mean Squared Error
RMSE is Relative Mean Squared Error
SAMSE is Sum of Asymptotic Relative Mean Squared Error
SAMSE_j is j-th order SAMSE
MISE is Mean Integrated Squared Error
MIAE is Mean Integrated Absolute Error
AMISE is Asymptotic Mean Integrated Squared Error
AMISE$'$ is a higher order expansion of AMISE
ABias is Asymptotic Bias
AVar is Asymptotic Variance
ABias$'$ is higher order Asymptotic Bias
AVar$'$ is higher order Asymptotic Variance
AMSE$'$ is higher order Asymptotic Mean Squared Error
PI is Plug-In
LSCV is Least Squares Cross Validation
BCV is Biased Cross Validation
SCV is Smoothed Cross Validation

Data

$\boldsymbol{X}$ is data vector of dimension d

$\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n$ is random sample of size n

$\boldsymbol{X}^*$ is pre-scaled/pre-sphered version of $\boldsymbol{X}$

$\mathbf{S}$ is sample variance

$\mathbf{S}_{\mathcal{D}}$ is $\operatorname{dg} \mathbf{S}$

$\mathbf{S}^*_{\mathcal{D}}$ is variance of pre-scaled data

$\{C_1, C_2, \ldots, C_\nu\}$ is set of ν clusters

$d(C_i, C_j)$ is dissimilarity between clusters C_i and C_j

$W(\nu)$ is within cluster sum of squares for ν clusters

Kernel estimates

$\hat{f}(\boldsymbol{x}; \mathbf{H})$ is fixed kernel density estimate

$\hat{f}_{-i}(\boldsymbol{x}; \mathbf{H})$ is leave-one-out estimate

$\hat{f}_{\mathrm{SP}}(\boldsymbol{x}; \boldsymbol{\Omega})$ is sample point estimate

$\hat{f}_{\mathrm{B}}(\boldsymbol{x}; \mathbf{H}(\boldsymbol{x}))$ is balloon estimate

$\hat{f}_P(\boldsymbol{x}; \mathbf{G})$ is pilot kernel density estimate with pilot bandwidth $\mathbf{G}$

$\hat{f}^*(\boldsymbol{x}; \mathbf{H})$ is bootstrap kernel density estimate

$\hat{f}^*(\boldsymbol{x}^*; \mathbf{H}^*)$ is kernel density estimate on pre-transformed scale

$\hat{f}_{\mathrm{PT}}(\boldsymbol{x}; \boldsymbol{\Omega})$ is partitioned kernel density estimate with bandwidth function $\boldsymbol{\Omega}$

$\hat{f}_{\mathrm{PT}}(\boldsymbol{x}; \boldsymbol{\Omega}, \mathcal{P})$ is partitioned kernel density estimate with bandwidth $\boldsymbol{\Omega}$, partition $\mathcal{P}$

$\hat{\psi}_{\boldsymbol{r}}$ is leave-in-diagonals estimate, using $K^{(\boldsymbol{r})}$

$\tilde{\psi}_{\boldsymbol{r}}$ is leave-out-diagonals estimate, using $K^{(\boldsymbol{r})}$

$\check{\psi}_{\boldsymbol{r}}$ is leave-out-diagonals estimate, using $K^{(\boldsymbol{r})} * K^{(\boldsymbol{r})}$

$\hat{\psi}^{\mathrm{NR}}_{\boldsymbol{r}}$ is normal reference estimate of $\psi_{\boldsymbol{r}}$

$\hat{\boldsymbol{\Psi}}_4$ is estimate of $\boldsymbol{\Psi}_4$ with $\hat{\psi}_{\boldsymbol{r}}$

$\tilde{\boldsymbol{\Psi}}_4$ is estimate of $\boldsymbol{\Psi}_4$ with $\tilde{\psi}_{\boldsymbol{r}}$

$\check{\boldsymbol{\Psi}}_4$ is estimate of $\boldsymbol{\Psi}_4$ with $\check{\psi}_{\boldsymbol{r}}$

Bandwidth selectors

$\mathbf{H}$ is bandwidth matrix

$\mathbf{H}^*$ is pre-scaled/pre-sphered bandwidth

$\mathbf{H}_{\mathrm{MISE}}$ is MISE-optimal bandwidth

$\mathbf{H}_{\mathrm{AMISE}}$ is AMISE-optimal bandwidth

$\hat{\mathbf{H}}_{\mathrm{PI}}$ is plug-in bandwidth selector

$\hat{\mathbf{H}}_{\text{PI,AMSE}}$ is plug-in bandwidth selector, with AMSE pilot
$\hat{\mathbf{H}}_{\text{PI,SAMSE}}$ is plug-in bandwidth selector, with SAMSE pilot
$\hat{\mathbf{H}}_{\text{MS}}$ is maximally smoothed bandwidth selector
$\hat{\mathbf{H}}_{\text{LSCV}}$ is LSCV bandwidth selector
$\hat{\mathbf{H}}_{\text{BCV}}$ is BCV bandwidth selector
$\hat{\mathbf{H}}_{\text{SCV}}$ is SCV bandwidth selector
$\boldsymbol{\Omega}$ is bandwidth selector function
$\mathbf{G}$ is pilot bandwidth selector
$g_{\boldsymbol{r},\text{AMSE}}$ is $\boldsymbol{r}$-th order AMSE pilot, for plug-in
$g_{j,\text{SAMSE}}$ is j-th order SAMSE pilot, for plug-in
g_0 is optimal pilot, for SCV

Bandwidth selectors labels

Fm, Fm^* is label for m-stage full AMSE selectors: pre-scaled, pre-sphered
Dm, Dm^* is label for m-stage diagonal AMSE selectors: pre-scaled, pre-sphered
Sm, Sm^* is label for m-stage full SAMSE selectors: pre-scaled pre-sphered
L is label for full LSCV selector
DL is label for diagonal LSCV selector
B1, B2 are labels for full BCV1 and BCV2 selectors
DB2 is label for diagonal BCV2 selectors
SC, SC* is label for 1-stage full SCV selectors: pre-scaled pre-sphered
AL is label for Abramson's LSCV selector
PL is label for pre-clustered LSCV selector
SL is label for Sain's LSCV selector
KDXX is label for kernel discriminant analyser, with XX selector

Discriminant analysis

$\boldsymbol{\mathcal{X}}_1, \boldsymbol{\mathcal{X}}_2, \ldots, \boldsymbol{\mathcal{X}}_\nu$ is ν training data samples
$\boldsymbol{\mathcal{X}}_j = \{\boldsymbol{X}_{j1}, \boldsymbol{X}_{j2}, \ldots, \boldsymbol{X}_{jn_j}\}$ is j-th training data sample of size n_j
$\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_m$ is test data sample of size m
f_j is density for j-th discriminant group
π_j is prior probability of f_j
$\hat{f}_j(\boldsymbol{x}; \mathbf{H}_j)$ is kernel density estimate for j-th training data sample
$\hat{\pi}_j$ is sample proportion for j-th training data sample
$\hat{f}_{j,-i}(\boldsymbol{x}; \mathbf{H}_{j,-i})$ is kernel density estimate for j-th training data sample, leaving out $\boldsymbol{X}_{ji}$
$\hat{\pi}_{j,-i}$ is sample proportion for j-th training data sample, leaving out $\boldsymbol{X}_{ji}$

KDR is kernel discriminant rule

KDR_{-ji} is kernel discriminant rule KDR, leaving out $\boldsymbol{X}_{ji}$

MR is misclassification rate

$\widehat{\text{MR}}$ is simple estimate of MR

$\widehat{\text{MR}}_{\text{CV}}$ is cross validated estimate of MR

Appendix B

Supplementary results

Tables B.1 and B.2 contain the plug-in bandwidth matrix that attains the median of the simulations trials. Table B.1 is for pre-sphered data. The top half is for sample size $n = 100$ and the lower half is for $n = 1000$. The first column is the density label, the next is $\mathbf{H}_{\text{MISE}}$, the next four are the bandwidths which achieve the median ISE($\hat{\mathbf{H}}$) for F1*, S1*, F2*, S2* respectively. Table B.2 is for pre-scaled data i.e. F1, S1, F2, S2 and D2. Tables B.3 and B.4 contain the results from ISE calculations. The second column is the optimal MISE (i.e. MISE($\mathbf{H}_{\text{MISE}}$)) which is then followed by the mean and standard deviation of the ISEs. Table B.5 is similar to Tables B.1 and B.2 but for the cross-validation selectors DL, DB2, L, B1, B2, SC and SC*. Table B.6 is the cross-validation counterpart to Tables B.3 and B.4, whereas Table B.7 is for the variable selectors AL, PL, and SL, along with S2, L and SC for comparison.

	$\mathbf{H}_{\text{MISE}}$	F1*	S1*	F2*	S2*
			Median $\hat{\mathbf{H}}$		
			$n = 100$		
A	$\begin{bmatrix} 0.0631 & 0 \\ 0 & 0.2522 \end{bmatrix}$	$\begin{bmatrix} 0.0449 & 0.0044 \\ 0.0044 & 0.1577 \end{bmatrix}$	$\begin{bmatrix} 0.0584 & -0.0061 \\ -0.0061 & 0.2205 \end{bmatrix}$	$\begin{bmatrix} 0.0495 & -0.0034 \\ -0.0034 & 0.1273 \end{bmatrix}$	$\begin{bmatrix} 0.0414 & -0.0184 \\ -0.0184 & 0.1418 \end{bmatrix}$
B	$\begin{bmatrix} 0.2012 & 0 \\ 0 & 0.1348 \end{bmatrix}$	$\begin{bmatrix} 0.2961 & 0.0126 \\ 0.0126 & 0.0995 \end{bmatrix}$	$\begin{bmatrix} 0.2937 & -0.0025 \\ -0.0025 & 0.0932 \end{bmatrix}$	$\begin{bmatrix} 0.2413 & -0.0093 \\ -0.0093 & 0.1169 \end{bmatrix}$	$\begin{bmatrix} 0.2876 & 0.0091 \\ 0.0091 & 0.1037 \end{bmatrix}$
C	$\begin{bmatrix} 0.0209 & 0 \\ 0 & 0.3351 \end{bmatrix}$	$\begin{bmatrix} 0.1738 & 0.0139 \\ 0.0139 & 0.2106 \end{bmatrix}$	$\begin{bmatrix} 0.1740 & -0.0087 \\ -0.0087 & 0.3108 \end{bmatrix}$	$\begin{bmatrix} 0.0914 & 0.0326 \\ 0.0326 & 0.2470 \end{bmatrix}$	$\begin{bmatrix} 0.1060 & 0.0051 \\ 0.0051 & 0.1768 \end{bmatrix}$
D	$\begin{bmatrix} 0.1363 & 0.0718 \\ 0.0718 & 0.1363 \end{bmatrix}$	$\begin{bmatrix} 0.2157 & -0.1351 \\ -0.1351 & 0.2157 \end{bmatrix}$	$\begin{bmatrix} 0.1930 & -0.0582 \\ -0.0582 & 0.2064 \end{bmatrix}$	$\begin{bmatrix} 0.1645 & -0.1095 \\ -0.1095 & 0.1681 \end{bmatrix}$	$\begin{bmatrix} 0.1376 & -0.0511 \\ -0.0511 & 0.1400 \end{bmatrix}$
E	$\begin{bmatrix} 0.1387 & 0.0726 \\ 0.0726 & 0.1840 \end{bmatrix}$	$\begin{bmatrix} 0.2036 & 0.0503 \\ 0.0503 & 0.1653 \end{bmatrix}$	$\begin{bmatrix} 0.2627 & 0.1299 \\ 0.1299 & 0.2389 \end{bmatrix}$	$\begin{bmatrix} 0.2055 & 0.0509 \\ 0.0509 & 0.1379 \end{bmatrix}$	$\begin{bmatrix} 0.1982 & 0.0712 \\ 0.0712 & 0.1328 \end{bmatrix}$
F	$\begin{bmatrix} 0.2522 & 0.2269 \\ 0.2269 & 0.2522 \end{bmatrix}$	$\begin{bmatrix} 0.2497 & 0.2484 \\ 0.2484 & 0.2886 \end{bmatrix}$	$\begin{bmatrix} 0.2158 & 0.1977 \\ 0.1977 & 0.2264 \end{bmatrix}$	$\begin{bmatrix} 0.2002 & 0.1478 \\ 0.1478 & 0.1578 \end{bmatrix}$	$\begin{bmatrix} 0.1267 & 0.1262 \\ 0.1262 & 0.1587 \end{bmatrix}$
			$n = 1000$		
A	$\begin{bmatrix} 0.0269 & 0 \\ 0 & 0.1077 \end{bmatrix}$	$\begin{bmatrix} 0.0263 & -0.0010 \\ -0.0010 & 0.1041 \end{bmatrix}$	$\begin{bmatrix} 0.0253 & 0.0032 \\ 0.0032 & 0.1050 \end{bmatrix}$	$\begin{bmatrix} 0.0262 & -0.0018 \\ -0.0018 & 0.0857 \end{bmatrix}$	$\begin{bmatrix} 0.0232 & -0.0016 \\ -0.0016 & 0.0997 \end{bmatrix}$
B	$\begin{bmatrix} 0.0727 & 0 \\ 0 & 0.0588 \end{bmatrix}$	$\begin{bmatrix} 0.0970 & -0.0001 \\ -0.0001 & 0.0497 \end{bmatrix}$	$\begin{bmatrix} 0.1020 & 0.0006 \\ 0.0006 & 0.0505 \end{bmatrix}$	$\begin{bmatrix} 0.0696 & 0.0018 \\ 0.0018 & 0.0417 \end{bmatrix}$	$\begin{bmatrix} 0.0949 & -0.0040 \\ -0.0040 & 0.0417 \end{bmatrix}$
C	$\begin{bmatrix} 0.0087 & 0 \\ 0 & 0.1386 \end{bmatrix}$	$\begin{bmatrix} 0.0492 & -0.0014 \\ -0.0014 & 0.1170 \end{bmatrix}$	$\begin{bmatrix} 0.0470 & -0.0023 \\ -0.0023 & 0.1303 \end{bmatrix}$	$\begin{bmatrix} 0.0209 & 0.0020 \\ 0.0020 & 0.1369 \end{bmatrix}$	$\begin{bmatrix} 0.0261 & 0.0027 \\ 0.0027 & 0.0817 \end{bmatrix}$
D	$\begin{bmatrix} 0.0558 & 0.0299 \\ 0.0299 & 0.0558 \end{bmatrix}$	$\begin{bmatrix} 0.0733 & -0.0422 \\ -0.0422 & 0.0704 \end{bmatrix}$	$\begin{bmatrix} 0.0681 & -0.0001 \\ -0.0001 & 0.0728 \end{bmatrix}$	$\begin{bmatrix} 0.0503 & -0.0303 \\ -0.0303 & 0.0525 \end{bmatrix}$	$\begin{bmatrix} 0.0481 & -0.0040 \\ -0.0040 & 0.0449 \end{bmatrix}$
E	$\begin{bmatrix} 0.0526 & 0.0266 \\ 0.0266 & 0.0723 \end{bmatrix}$	$\begin{bmatrix} 0.0771 & 0.0245 \\ 0.0245 & 0.0817 \end{bmatrix}$	$\begin{bmatrix} 0.0804 & 0.0345 \\ 0.0345 & 0.0895 \end{bmatrix}$	$\begin{bmatrix} 0.0592 & 0.0245 \\ 0.0245 & 0.0780 \end{bmatrix}$	$\begin{bmatrix} 0.0608 & 0.0236 \\ 0.0236 & 0.0721 \end{bmatrix}$
F	$\begin{bmatrix} 0.1077 & 0.0969 \\ 0.0969 & 0.1077 \end{bmatrix}$	$\begin{bmatrix} 0.0987 & 0.0881 \\ 0.0881 & 0.0974 \end{bmatrix}$	$\begin{bmatrix} 0.1007 & 0.0943 \\ 0.0943 & 0.1067 \end{bmatrix}$	$\begin{bmatrix} 0.0898 & 0.0815 \\ 0.0815 & 0.0935 \end{bmatrix}$	$\begin{bmatrix} 0.0977 & 0.0899 \\ 0.0899 & 0.1005 \end{bmatrix}$

Table B.1: Plug-in bandwidth matrices with pre-sphering for normal mixture densities.

		Median $\hat{\mathbf{H}}$				
	$\mathbf{H}_{\text{MISE}}$	F1	S1	F2	S2	D2
			$n = 100$			
A	$\begin{bmatrix} 0.0631 & 0 \\ 0 & 0.2522 \end{bmatrix}$	$\begin{bmatrix} 0.0487 & -0.0014 \\ -0.0014 & 0.2245 \end{bmatrix}$	$\begin{bmatrix} 0.0581 & -0.0010 \\ -0.0010 & 0.2207 \end{bmatrix}$	$\begin{bmatrix} 0.0487 & -0.0015 \\ -0.0015 & 0.1503 \end{bmatrix}$	$\begin{bmatrix} 0.0290 & 0.0106 \\ 0.0106 & 0.1829 \end{bmatrix}$	$\begin{bmatrix} 0.0549 & 0 \\ 0 & 0.1663 \end{bmatrix}$
B	$\begin{bmatrix} 0.2012 & 0 \\ 0 & 0.1348 \end{bmatrix}$	$\begin{bmatrix} 0.2812 & 0.0040 \\ 0.0040 & 0.0818 \end{bmatrix}$	$\begin{bmatrix} 0.3064 & 0.0027 \\ 0.0027 & 0.0803 \end{bmatrix}$	$\begin{bmatrix} 0.1959 & -0.0006 \\ -0.0006 & 0.0724 \end{bmatrix}$	$\begin{bmatrix} 0.2056 & 0.0166 \\ 0.0166 & 0.0649 \end{bmatrix}$	$\begin{bmatrix} 0.2424 & 0 \\ 0 & 0.0841 \end{bmatrix}$
C	$\begin{bmatrix} 0.0209 & 0 \\ 0 & 0.3351 \end{bmatrix}$	$\begin{bmatrix} 0.1865 & -0.0016 \\ -0.0016 & 0.1702 \end{bmatrix}$	$\begin{bmatrix} 0.1682 & -0.0018 \\ -0.0018 & 0.1974 \end{bmatrix}$	$\begin{bmatrix} 0.0905 & -0.0010 \\ -0.0010 & 0.2874 \end{bmatrix}$	$\begin{bmatrix} 0.1056 & -0.0026 \\ -0.0026 & 0.1720 \end{bmatrix}$	$\begin{bmatrix} 0.0883 & 0 \\ 0 & 0.2232 \end{bmatrix}$
D	$\begin{bmatrix} 0.1363 & 0.0718 \\ 0.0718 & 0.1363 \end{bmatrix}$	$\begin{bmatrix} 0.1886 & 0.0622 \\ 0.0622 & 0.1630 \end{bmatrix}$	$\begin{bmatrix} 0.1291 & 0.0288 \\ 0.0288 & 0.1518 \end{bmatrix}$	$\begin{bmatrix} 0.1011 & 0.0204 \\ 0.0204 & 0.1102 \end{bmatrix}$	$\begin{bmatrix} 0.1241 & 0.0162 \\ 0.0162 & 0.1147 \end{bmatrix}$	$\begin{bmatrix} 0.0854 & 0 \\ 0 & 0.0909 \end{bmatrix}$
E	$\begin{bmatrix} 0.1387 & 0.0726 \\ 0.0726 & 0.1840 \end{bmatrix}$	$\begin{bmatrix} 0.2878 & 0.1012 \\ 0.1012 & 0.1888 \end{bmatrix}$	$\begin{bmatrix} 0.1588 & 0.0024 \\ 0.0024 & 0.1208 \end{bmatrix}$	$\begin{bmatrix} 0.1976 & 0.0471 \\ 0.0471 & 0.1853 \end{bmatrix}$	$\begin{bmatrix} 0.1875 & 0.0424 \\ 0.0424 & 0.1515 \end{bmatrix}$	$\begin{bmatrix} 0.1220 & 0 \\ 0 & 0.1547 \end{bmatrix}$
F	$\begin{bmatrix} 0.2522 & 0.2269 \\ 0.2269 & 0.2522 \end{bmatrix}$	$\begin{bmatrix} 0.0477 & 0.0194 \\ 0.0194 & 0.0621 \end{bmatrix}$	$\begin{bmatrix} 0.0411 & 0.0067 \\ 0.0067 & 0.0446 \end{bmatrix}$	$\begin{bmatrix} 0.0411 & 0.0062 \\ 0.0062 & 0.0401 \end{bmatrix}$	$\begin{bmatrix} 0.0505 & 0.0024 \\ 0.0024 & 0.0378 \end{bmatrix}$	$\begin{bmatrix} 0.0367 & 0 \\ 0 & 0.0462 \end{bmatrix}$
			$n = 1000$			
A	$\begin{bmatrix} 0.0269 & 0 \\ 0 & 0.1077 \end{bmatrix}$	$\begin{bmatrix} 0.0249 & 0.0002 \\ 0.0002 & 0.0980 \end{bmatrix}$	$\begin{bmatrix} 0.0249 & -0.0010 \\ -0.0010 & 0.0979 \end{bmatrix}$	$\begin{bmatrix} 0.0267 & -0.0009 \\ -0.0009 & 0.0874 \end{bmatrix}$	$\begin{bmatrix} 0.0263 & -0.0008 \\ -0.0008 & 0.0884 \end{bmatrix}$	$\begin{bmatrix} 0.0217 & 0 \\ 0 & 0.1016 \end{bmatrix}$
B	$\begin{bmatrix} 0.0727 & 0 \\ 0 & 0.0588 \end{bmatrix}$	$\begin{bmatrix} 0.1028 & 0.0003 \\ 0.0003 & 0.0545 \end{bmatrix}$	$\begin{bmatrix} 0.1018 & -0.0010 \\ -0.0010 & 0.0504 \end{bmatrix}$	$\begin{bmatrix} 0.0694 & 0.0000 \\ 0.0000 & 0.0418 \end{bmatrix}$	$\begin{bmatrix} 0.0890 & 0.0017 \\ 0.0017 & 0.0425 \end{bmatrix}$	$\begin{bmatrix} 0.0829 & 0 \\ 0 & 0.0522 \end{bmatrix}$
C	$\begin{bmatrix} 0.0087 & 0 \\ 0 & 0.1386 \end{bmatrix}$	$\begin{bmatrix} 0.0470 & 0.0000 \\ 0.0000 & 0.1302 \end{bmatrix}$	$\begin{bmatrix} 0.0470 & -0.0023 \\ -0.0023 & 0.1303 \end{bmatrix}$	$\begin{bmatrix} 0.0203 & -0.0001 \\ -0.0001 & 0.1440 \end{bmatrix}$	$\begin{bmatrix} 0.0264 & 0.0003 \\ 0.0003 & 0.0775 \end{bmatrix}$	$\begin{bmatrix} 0.0213 & 0 \\ 0 & 0.1769 \end{bmatrix}$
D	$\begin{bmatrix} 0.0558 & 0.0299 \\ 0.0299 & 0.0558 \end{bmatrix}$	$\begin{bmatrix} 0.0915 & 0.0565 \\ 0.0565 & 0.0885 \end{bmatrix}$	$\begin{bmatrix} 0.0661 & 0.0231 \\ 0.0231 & 0.0636 \end{bmatrix}$	$\begin{bmatrix} 0.0575 & 0.0274 \\ 0.0274 & 0.0629 \end{bmatrix}$	$\begin{bmatrix} 0.0505 & 0.0167 \\ 0.0167 & 0.0528 \end{bmatrix}$	$\begin{bmatrix} 0.0461 & 0 \\ 0 & 0.0463 \end{bmatrix}$
E	$\begin{bmatrix} 0.0526 & 0.0266 \\ 0.0266 & 0.0723 \end{bmatrix}$	$\begin{bmatrix} 0.0802 & 0.0288 \\ 0.0288 & 0.0788 \end{bmatrix}$	$\begin{bmatrix} 0.0745 & 0.0186 \\ 0.0186 & 0.0804 \end{bmatrix}$	$\begin{bmatrix} 0.0539 & 0.0251 \\ 0.0251 & 0.0770 \end{bmatrix}$	$\begin{bmatrix} 0.0562 & 0.0127 \\ 0.0127 & 0.0545 \end{bmatrix}$	$\begin{bmatrix} 0.051 & 0 \\ 0 & 0.0646 \end{bmatrix}$
F	$\begin{bmatrix} 0.1077 & 0.0969 \\ 0.0969 & 0.1077 \end{bmatrix}$	$\begin{bmatrix} 0.0221 & 0.0040 \\ 0.0040 & 0.0228 \end{bmatrix}$	$\begin{bmatrix} 0.0219 & 0.0055 \\ 0.0055 & 0.0226 \end{bmatrix}$	$\begin{bmatrix} 0.0199 & 0.0024 \\ 0.0024 & 0.0187 \end{bmatrix}$	$\begin{bmatrix} 0.0224 & 0.0056 \\ 0.0056 & 0.0229 \end{bmatrix}$	$\begin{bmatrix} 0.0206 & 0 \\ 0 & 0.0232 \end{bmatrix}$

Table B.2: Median plug-in bandwidth matrices with pre-scaling for normal mixture densities.

	MISE($\mathbf{H}_{\text{MISE}}$)		ISE($\hat{\mathbf{H}}$)			
			F1*	S1*	F2*	S2*
			$n = 100$			
A	0.00863	mean	0.01009	0.01019	0.01065	0.01066
		SD	0.00403	0.00405	0.00425	0.00425
B	0.00717	mean	0.00806	0.00813	0.00828	0.00840
		SD	0.00250	0.00253	0.00263	0.00264
C	0.01404	mean	0.04184	0.04175	0.02620	0.03016
		SD	0.00399	0.00397	0.00482	0.00456
D	0.01034	mean	0.02101	0.01707	0.01818	0.01482
		SD	0.00332	0.00362	0.00350	0.00368
E	0.00864	mean	0.00975	0.00969	0.00935	0.00932
		SD	0.00255	0.00251	0.00263	0.00260
F	0.00990	mean	0.01154	0.01168	0.01215	0.01222
		SD	0.00460	0.00464	0.00484	0.00487
			$n = 1000$			
A	0.00212	mean	0.00220	0.00222	0.00224	0.00224
		SD	0.00066	0.00066	0.00066	0.00066
B	0.00181	mean	0.00193	0.00194	0.0019	0.00194
		SD	0.00050	0.00050	0.00049	0.00049
C	0.00341	mean	0.01048	0.01048	0.00478	0.00575
		SD	0.00115	0.00115	0.00104	0.00106
D	0.00253	mean	0.00581	0.00378	0.00456	0.00314
		SD	0.00082	0.00075	0.00075	0.00068
E	0.00216	mean	0.00239	0.00238	0.00222	0.00223
		SD	0.00060	0.00060	0.00055	0.00055
F	0.00244	mean	0.00253	0.00254	0.00256	0.00257
		SD	0.00075	0.00076	0.00076	0.00076

Table B.3: ISEs for plug-in bandwidth matrices with pre-sphering for normal mixture densities.

	MISE ($\mathbf{H}_{\text{MISE}}$)		ISE($\hat{\mathbf{H}}$)				
			F1	S1	F2	S2	D2
				$n = 100$			
A	0.00863	mean	0.01011	0.01015	0.01067	0.01063	0.00976
		SD	0.00405	0.00405	0.00429	0.00426	0.00419
B	0.00717	mean	0.00805	0.00809	0.00828	0.00837	0.00789
		SD	0.00252	0.00253	0.00265	0.00264	0.00259
C	0.01404	mean	0.04141	0.04143	0.02583	0.02998	0.02597
		SD	0.00396	0.00396	0.00478	0.00454	0.00429
D	0.01034	mean	0.01195	0.01204	0.01174	0.01174	0.01226
		SD	0.00354	0.00353	0.00348	0.00348	0.00336
E	0.00864	mean	0.00982	0.00984	0.00960	0.00957	0.00981
		SD	0.00258	0.00257	0.00268	0.00267	0.00255
F	0.00990	mean	0.02177	0.02138	0.02443	0.02291	0.02263
		SD	0.00628	0.00617	0.00693	0.00670	0.00668
				$n = 1000$			
A	0.00212	mean	0.00221	0.00221	0.00224	0.00224	0.00216
		SD	0.00066	0.00066	0.00066	0.00066	0.00066
B	0.00181	mean	0.00193	0.00194	0.00190	0.00194	0.00186
		SD	0.00050	0.00050	0.00049	0.00049	0.00048
C	0.00341	mean	0.01046	0.01047	0.00477	0.00575	0.00485
		SD	0.00115	0.00115	0.00104	0.00106	0.00098
D	0.00253	mean	0.00295	0.00280	0.00269	0.00267	0.00298
		SD	0.00073	0.00066	0.00065	0.00062	0.00063
E	0.00216	mean	0.00240	0.00239	0.00225	0.00226	0.00240
		SD	0.00060	0.00059	0.00055	0.00055	0.00055
F	0.00244	mean	0.00424	0.00427	0.00457	0.00436	0.00479
		SD	0.00098	0.00094	0.00101	0.00098	0.00099

Table B.4: ISEs for plug-in bandwidth matrices with pre-scaling for normal mixture densities.

	Median $\hat{\mathbf{H}}$															
	$\mathbf{H}_{\text{MISE}}$		L		B1		B2		SC		SC∗		DL		DB2	
							$n = 100$									
A	0.0631	0	0.1026	−0.0485	0.0290	−0.0080	0.0517	0.0069	0.0649	−0.0016	0.0613	−0.0024	0.0663	0	0.0676	0
	0	0.2522	−0.0485	0.6640	−0.0080	0.1889	0.0069	0.2149	−0.0016	0.3492	−0.0024	0.3454	0	0.3761	0	0.3026
B	0.2012	0	0.3477	0.1632	0.2839	0.0219	0.3090	0.0067	0.2013	−0.0000	0.3160	−0.0240	0.2450	0	0.3071	0
	0	0.1348	0.1632	0.1661	0.0219	0.0939	0.0067	0.0929	−0.0000	0.1348	−0.0240	0.1189	0	0.0326	0	0.0953
C	0.0209	0	0.0096	−0.0039	0.0180	−0.0026	0.4763	0.0174	0.1362	0.0718	0.1470	−0.0030	0.0206	0	0.5409	0
	0	0.3351	−0.0039	0.3018	−0.0026	0.0537	0.0174	0.2248	0.0718	0.1364	−0.003	0.3934	0	0.1923	0	0.2411
D	0.1363	0.0718	0.1678	0.1057	0.0753	0.0230	0.2774	0.1290	0.1915	0.0641	0.2240	−0.0071	0.2537	0	0.1499	0
	0.0718	0.1363	0.1057	0.1965	0.0230	0.0499	0.1290	0.2499	0.0641	0.1941	−0.0071	0.1856	0	0.0424	0	0.2122
E	0.1387	0.0726	0.3669	0.0064	0.0060	0.0080	0.2855	0.1331	0.2608	0.0844	0.3619	0.2056	0.0495	0	0.3243	0
	0.0726	0.1840	0.0064	0.2007	0.0080	0.2711	0.1331	0.2550	0.0844	0.2328	0.2056	0.3107	0	0.2649	0	0.3543
F	0.2522	0.2269	0.1821	0.1873	0.1897	0.1772	0.2205	0.1938	0.1219	0.0679	0.3386	0.3162	0.0459	0	0.2716	0
	0.2269	0.2522	0.1873	0.2687	0.1772	0.1967	0.1938	0.2085	0.0679	0.1164	0.3162	0.3428	0	0.0849	0	0.2897
							$n = 1000$									
A	0.0269	0	0.0346	−0.0027	0.0241	0.0013	0.0264	0.0011	0.0275	0.0003	0.0275	0.0000	0.0248	0	0.0273	0
	0	0.1077	−0.0027	0.1013	0.0013	0.0943	0.0011	0.0923	0.0003	0.1242	0.0000	0.1242	0	0.0796	0	0.1200
B	0.0727	0	0.0724	0.0015	0.1281	0.0011	0.1416	0.0006	0.1156	−0.0105	0.1114	0.0006	0.0560	0	0.1031	0
	0	0.0588	0.0015	0.0472	0.0011	0.0359	0.0006	0.0424	−0.0105	0.0550	0.0006	0.0589	0	0.0607	0	0.0520
C	0.0087	0	0.0090	−0.0088	0.0108	−0.0008	0.0240	0.0002	0.0328	0.0001	0.0322	−0.0001	0.0074	0	0.0111	0
	0	0.1386	−0.0088	0.1428	−0.0008	0.0418	0.0002	0.0222	0.0001	0.1234	−0.0001	0.1378	0	0.1559	0	0.1091
D	0.0558	0.0299	0.0508	0.0299	0.0303	0.0157	0.1551	0.0154	0.0660	0.0265	0.0681	0.0085	0.0408	0	0.0454	0
	0.0299	0.0558	0.0299	0.0554	0.0157	0.0342	0.0154	0.1572	0.0265	0.0658	0.0085	0.0659	0	0.0431	0	0.0545
E	0.0526	0.0266	0.0288	0.0260	0.0407	0.0296	0.0230	0.0150	0.0747	0.0338	0.0727	0.0306	0.0521	0	0.0558	0
	0.0266	0.0723	0.0260	0.1034	0.0296	0.1336	0.0150	0.0120	0.0338	0.0918	0.0306	0.0809	0	0.0423	0	0.0789
F	0.1077	0.0969	0.1338	0.1310	0.0957	0.0802	0.0940	0.0824	0.0797	0.0654	0.1236	0.1125	0.0248	0	0.0313	0
	0.0969	0.1077	0.1310	0.1476	0.0802	0.0969	0.0824	0.1076	0.0654	0.0794	0.1125	0.1234	0	0.0237	0	0.0311

Table B.5: Median cross-validation bandwidth matrices for normal mixture densities.

	MISE ($\mathbf{H}_{\text{MISE}}$)		ISE($\hat{\mathbf{H}}$)						
			L	B1	B2	SC	SC*	DL	DB2
					$n = 100$				
A	0.00863	mean	0.01746	0.01100	0.01003	0.00974	0.00979	0.01265	0.00907
		SD	0.01579	0.00451	0.00377	0.00399	0.00400	0.00796	0.00387
B	0.00717	mean	0.01340	0.00879	0.00799	0.00835	0.00840	0.00934	0.00782
		SD	0.00863	0.00363	0.00236	0.00236	0.00236	0.00428	0.00211
C	0.01404	mean	0.02433	0.03704	0.07303	0.03665	0.03692	0.01748	0.08023
		SD	0.01478	0.02279	0.00259	0.00421	0.00425	0.00756	0.00222
D	0.01034	mean	0.01676	0.01514	0.01406	0.01262	0.01749	0.01409	0.02039
		SD	0.00885	0.00748	0.00431	0.00393	0.00408	0.00533	0.00573
E	0.00864	mean	0.01438	0.01212	0.01067	0.01069	0.01066	0.01195	0.01384
		SD	0.00850	0.00923	0.00232	0.00286	0.00280	0.00535	0.00240
F	0.00990	mean	0.02105	0.01341	0.01154	0.01352	0.01123	0.02328	0.03806
		SD	0.01860	0.00760	0.00430	0.00532	0.00459	0.00889	0.00926
					$n = 1000$				
A	0.00212	mean	0.00283	0.00236	0.00219	0.00218	0.00218	0.00255	0.00220
		SD	0.00121	0.00082	0.00074	0.00074	0.00074	0.00092	0.00080
B	0.00181	mean	0.00226	0.00222	0.00213	0.00199	0.00199	0.00214	0.00200
		SD	0.00085	0.00055	0.00047	0.00053	0.00053	0.00065	0.00055
C	0.00341	mean	0.00412	0.00589	0.01890	0.00696	0.00697	0.00396	0.00357
		SD	0.00143	0.00325	0.01927	0.00112	0.00112	0.00116	0.00088
D	0.00253	mean	0.00303	0.00395	0.00477	0.00274	0.00330	0.00328	0.00312
		SD	0.00101	0.00247	0.00301	0.00067	0.00074	0.00079	0.00064
E	0.00216	mean	0.00251	0.00295	0.00288	0.00240	0.00238	0.00259	0.00247
		SD	0.00077	0.00110	0.00104	0.00063	0.00063	0.00060	0.00061
F	0.00244	mean	0.00324	0.00273	0.00252	0.00276	0.00250	0.00516	0.00489
		SD	0.00138	0.00096	0.00084	0.00089	0.00085	0.00128	0.00114

Table B.6: ISEs for cross-validation bandwidth matrices for normal mixture densities.

	MISE ($\mathbf{H}_{\text{MISE}}$)		ISE					
			S2	L	SC	AL	SL	PL
			$n = 100$					
A	0.00863	mean	0.01063	0.01746	0.00974	0.01021	0.01090	0.01802
		SD	0.00426	0.01579	0.00399	0.00693	0.00582	0.01326
B	0.00717	mean	0.00837	0.01340	0.00835	0.00921	0.01039	0.01638
		SD	0.00264	0.00863	0.00236	0.00383	0.00664	0.01012
D	0.01034	mean	0.01174	0.01676	0.01262	0.01303	0.01502	0.02100
		SD	0.00348	0.00885	0.00393	0.00473	0.00746	0.01192
E	0.00864	mean	0.00957	0.01438	0.01069	0.01048	0.01307	0.01731
		SD	0.00267	0.00850	0.00286	0.00494	0.00741	0.01219
G	0.07166	mean	0.08546	0.09355	0.10153	0.07960	0.12373	0.09148
		SD	0.01474	0.04040	0.01577	0.01981	0.06958	0.05525
H	-	mean	0.09378	0.09060	0.10934	0.07168	0.09373	0.08247
		SD	0.01638	0.02951	0.01752	0.01745	0.03405	0.03977
			$n = 1000$					
A	0.00212	mean	0.00224	0.00283	0.00218	0.00223	0.00226	0.00295
		SD	0.00066	0.00121	0.00074	0.00074	0.00072	0.00127
B	0.00181	mean	0.00194	0.00226	0.00199	0.00209	0.00223	0.00254
		SD	0.00049	0.00085	0.00053	0.00059	0.00059	0.00094
D	0.00253	mean	0.00267	0.00303	0.00274	0.00282	0.00310	0.00344
		SD	0.00062	0.00101	0.00067	0.00072	0.00079	0.00126
E	0.00216	mean	0.00226	0.00251	0.00240	0.00207	0.00265	0.00286
		SD	0.00055	0.00077	0.00063	0.00051	0.00090	0.00099
G	0.01837	mean	0.02179	0.02041	0.02383	0.01932	0.05079	0.01498
		SD	0.00344	0.00408	0.00359	0.00354	0.02755	0.00550
H	-	mean	0.02620	0.01980	0.02929	0.01650	0.03305	0.01937
		SD	0.00399	0.00343	0.00426	0.00302	0.01731	0.00369

Table B.7: ISEs for fixed and variable bandwidth matrices for mixture densities.

Appendix C

Software

All the algorithms for the fixed selectors are now available in an `R` library: `ks`. The current release is 1.1 and is available in two versions from:

- Unix – `http://www.maths.uwa.edu.au/~martin/software/ks_1.1.tar.gz`
- Windows – `http://www.maths.uwa.edu.au/~martin/software/ks_1.1.zip`

The variable selectors are not included in this release since there is still much development required till they can be made for general use (though they are available from the author). This appendix contains the help guide for this library, included as part of the release.

`Hbcv, Hbcv.diag`	*Biased cross-validation (BCV) bandwidth matrix selector for bivariate data*

Description

BCV bandwidth matrix for bivariate data.

Usage

```
Hbcv(x, whichbcv=1, Hstart)
Hbcv.diag(x, whichbcv=1, Hstart)
```

Arguments

`x`	matrix of data values
`whichbcv`	1 = BCV1, 2 = BCV2. See details below
`Hstart`	initial bandwidth matrix, used in numerical optimisation

Details

Use `Hbcv` for full bandwidth matrices and `Hbcv.diag` for diagonal bandwidth matrices.

There are two types of BCV criteria considered here. They are known as BCV1 and BCV2, from Sain, Baggerly & Scott (1994) and they only differ slightly. These BCV surfaces can have multiple minima and so it can be quite difficult to locate the most appropriate minimum.

If `Hstart` is not given then it defaults to `k*var(x)` where k = $\left[\frac{4}{n(d+2)}\right]^{2/(d+4)}$, n = sample size, d = dimension of data.

Value

BCV bandwidth matrix.

Note

It can be difficult to find an appropriate (local) minimum of the BCV criterion. Some times, there can be no local minimum at all so there may be no finite BCV selector.

References

Sain, S.R, Baggerly, K.A. & Scott, D.W. (1994) *Cross-validation of multivariate densities.* Journal of the American Statistical Association. **82**, 1131-1146.

Duong, T. & Hazelton, M.L. (2004) *Cross-validation bandwidth matrices for multivariate kernel density estimation.* Submitted for publication.

See Also

`Hlscv`, `Hscv`

Examples

```
data(faithful)
Hbcv(faithful)
Hbcv.diag(faithful)
```

`Hkda, Hkda.diag` *Bandwidth matrix selectors for kernel discriminant analysis for bivariate data*

Description

Bandwidth matrices for kernel discriminant analysis for bivariate data.

Usage

```
Hkda(x, x.group, Hstart, bw="plugin", nstage=2, pilot="samse",
     pre="sphere")
Hkda.diag(x, x.group, bw="plugin", nstage=2, pilot="samse",
          pre="sphere")
```

Arguments

`x`	matrix of training data values
`x.group`	vector of group labels for training data
`bw`	bandwidth: `"plugin"` = plug-in, `"lscv"` = LSCV, `"scv"` = SCV
`nstage`	number of stages in the plug-in bandwidth selector (1 or 2)
`pilot`	`"amse"`=AMSE-optimal pilot bandwidths, `"samse"`=single SAMSE-optimal pilot bandwidth
`pre`	`"scale"` = pre-scaling, `"sphere"` = pre-sphering
`Hstart`	(stacked) matrix of initial bandwidth matrices, used in numerical optimisation

Details

The values that valid for `bw` are `"plugin", "lscv"` and `"scv"` for `Hkda`. These in turn call `Hpi`, `Hlscv` and `Hscv`. For plugin selectors, all of `nstage`, `pilot` and `pre` need to be set. For SCV selectors, currently `nstage` is always programmed to be one but the other two need to be set. For LSCV selectors, none of them are required.

For `Hkda.diag`, only `"plugin"` or `"lscv"` are valid which in turn call `Hpi.diag` and `Hlscv.diag`. Again, `nstage`, `pilot` and `pre` are available for `Hpi.diag` but not required for `Hlscv.diag`.

Value

Stacked matrix of bandwidth matrices for each training data group.

References

Simonoff, J. S. (1996) *Smoothing Methods in Statistics*. Springer-Verlag. New York.

See Also

`kda.kde`, `Hpi`, `Hpi.diag`, `Hlscv`, `Hlscv.diag`, `Hscv`

Examples

```
library(MASS)
data(iris)
iris.mat <- rbind(iris[,,1], iris[,,2], iris[,,3])
ir <- iris.mat[,c(1,2)]
ir.gr <- iris.mat[,5]

Hkda(ir, ir.gr, bw="scv", pre="scale")
Hkda.diag(ir, ir.gr, bw="plugin", pre="scale")
```

Hlscv — *Least-squares cross-validation (LSCV) bandwidth matrix selector for bivariate data*

Description

LSCV bandwidth matrix for bivariate data.

Usage

```
Hlscv(x, Hstart)
Hlscv.diag(x, Hstart)
```

Arguments

`x`	matrix of data values
`Hstart`	initial bandwidth matrix, used in numerical optimisation

Details

Use `Hlscv` for full bandwidth matrices and `Hlscv.diag` for diagonal bandwidth matrices.

If `Hstart` is not given then it defaults to `k*var(x)` where k = $\left[\frac{4}{n(d+2)}\right]^{2/(d+4)}$, n = sample size, d = dimension of data.

Value

LSCV bandwidth matrix.

References

Sain, S.R, Baggerly, K.A & Scott, D.W. (1994) *Cross-validation of multivariate densities.* Journal of the American Statistical Association. **82**, 1131-1146.

Duong, T. & Hazelton, M.L. (2004) *Cross-validation bandwidth matrices for multivariate kernel density estimation.* Submitted for publication.

See Also

`Hbcv`, `Hscv`

Examples

```
data(faithful)
Hlscv(faithful)
Hlscv.diag(faithful)
```

`Hmise.mixt, Hamise.mixt`

MISE- and AMISE-optimal bandwidth matrix selectors for normal mixture densities

---

Description

For normal mixture densities, we have a closed form for the MISE and AMISE. So in these cases, we can numerically minimise these criteria to find MISE- and AMISE-optimal matrices.

Usage

```
Hmise.mixt(mus, Sigmas, props, samp, Hstart)
Hamise.mixt(mus, Sigmas, props, samp, Hstart)
```

Arguments

`mus`	(stacked) matrix of mean vectors
`Sigmas`	(stacked) matrix of variance matrices
`props`	vector of mixing proportions
`samp`	sample size
`Hstart`	initial bandwidth matrix, used in numerical optimisation

Details

For normal mixture densities, the MISE and AMISE have exact formulas. See Wand & Jones (1995).

If `Hstart` is not given then it defaults to `k*var(x)` where k = $\left[\frac{4}{n(d+2)}\right]^{2/(d+4)}$, n = sample size, d = dimension of data.

Value

Full MISE- or AMISE-optimal bandwidth matrix. Please note that diagonal forms of these matrices are not available.

References

Wand, M.P. & Jones, M.C. (1995) *Kernel Smoothing*. Chapman & Hall. London.

Examples

```
mus <- rbind(c(-3/2,0), c(3/2,0))
Sigmas <- rbind(diag(c(1/16, 1)), rbind(c(1/16, 1/18), c(1/18, 1/16)))
props <- c(2/3, 1/3)
samp <- 100
Hmise.mixt(mus, Sigmas, props, samp)
Hamise.mixt(mus, Sigmas, props, samp)
```

`Hpi, Hpi.diag` *Plug-in bandwidth matrix selector for bivariate data*

Description

Plug-in bandwidth matrix for bivariate data.

Usage

```
Hpi(x, nstage=2, pilot="samse", pre="sphere", Hstart)
Hpi.diag(x, nstage=2, pilot="amse", pre="scale")
```

Arguments

`x`	matrix of data values
`nstage`	number of stages in the plug-in bandwidth selector (1 or 2)
`pilot`	`"amse"`=AMSE-optimal pilot bandwidths, `"samse"`=single SAMSE-optimal pilot bandwidth
`pre`	`"scale"` = pre-scaling, `"sphere"` = pre-sphering
`Hstart`	initial bandwidth matrix, used in numerical optimisation

Details

Use `Hpi` for full bandwidth matrices and `Hpi.diag` for diagonal bandwidth matrices.

For AMSE pilot bandwidths, see Wand & Jones (1994). For SAMSE pilot bandwidths, see Duong & Hazelton (2003). The latter is a modification of the former, in order to remove any possible problems with non-positive definiteness. Both of these pilot bandwidths require numerical optimisation.

For details on the pre-transformations in `pre`, see `pre.sphere` and `pre.scale`.

If `Hstart` is not given then it defaults to `k*var(x)` where k = $\left[\frac{4}{n(d+2)}\right]^{2/(d+4)}$, n = sample size, d = dimension of data.

Value

Plug-in bandwidth matrix.

References

Wand, M.P. & Jones, M.C. (1994) *Multivariate plugin bandwidth selection.* Computational Statistics **9**, 97-116.

Duong, T. & Hazelton, M.L. (2003) *Plug-in bandwidth matrices for bivariate kernel density estimation.* Journal of Nonparametric Statistics **15**, 17-30.

Examples

```
data(faithful)
Hpi(faithful, nstage=1, pilot="amse", pre="scale")
Hpi(faithful, nstage=2, pilot="samse", pre="sphere")
Hpi.diag(faithful, nstage=2, pilot="amse", pre="scale")
```

`Hscv` *Smoothed cross-validation (SCV) bandwidth matrix selector for bivariate data*

Description

SCV bandwidth matrix for bivariate data.

Usage

```
Hscv(x, pre="sphere", Hstart)
```

Arguments

`x`	matrix of data values
`pre`	`"scale"` = pre-scaling, `"sphere"` = pre-sphering
`Hstart`	initial bandwidth matrix, used in numerical optimisation

Details

This SCV selector is a generalisation of the univariate SCV selector of Jones, Marron & Park (1991).

For details on the pre-transformations in `pre`, see `pre.sphere` and `pre.scale`.

If `Hstart` is not given then it defaults to `k*var(x)` where k = $\left[\frac{4}{n(d+2)}\right]^{2/(d+4)}$, n = sample size, d = dimension of data.

Value

Full SCV bandwidth matrix. Please note that a diagonal version of this selector is not available.

References

Jones, M.C., Marron, J. S. & Park, B.U. (1991) A simple root n bandwidth selector. *The Annals of Statistics* **19**, 1919–1932.

Duong, T. & Hazelton, M.L. (2004) *Cross-validation bandwidth matrices for multivariate kernel density estimation*. Submitted for publication.

See Also

`Hlscv`, `Hbcv`

Examples

```
data(faithful)
Hscv(faithful)
```

`kda, pda, compare` *Kernel and parametric discriminant analysis*

Description

Kernel and parametric discriminant analysis.

Usage

```
kda(x, x.group, Hs, y, prior.prob=NULL)
pda(x, x.group, y, prior.prob=NULL, type="quad")
compare(x.group, est.group)
```

Arguments

`x`	matrix of training data values
`x.group`	vector of group labels for training data
`est.group`	vector of estimated group labels
`y`	matrix of test data
`Hs`	(stacked) matrix of bandwidth matrices
`prior.prob`	vector of prior probabilities
`type`	`"line"` = linear discriminant, `"quad"` = quadratic discriminant

Details

If you have prior probabilities then set `prior.prob` to these. Otherwise set `prior.prob =NULL` (the default) and the sample proportions are used as the estimates for the prior probabilities.

The parametric discriminant analysers use the code from the `MASS` library namely `lda` and `qda` for linear and quadratic discriminants.

Value

The discriminant analysers are `kda` and `pda` and these return a vector of group labels assigned via discriminant analysis. If the test data `y` are given then these are classified. Otherwise the training data `x` are classified.

The function `compare` creates a comparison between the true group labels `x.group` and the estimated ones `est.group`. It returns a list with fields

`cross` cross-classification table with the rows indicating the true group and the columns the estimated group

`error` misclassification rate (MR) where

$$\text{MR} = \frac{\text{number of points wrongly classified}}{\text{total number of points}}$$

Note that this MR is only suitable when we have test data. If we don't have test data, then the cross validated estimate is more appropriate. See Silverman (1986).

References

Silverman, B. W. (1986) *Data Analysis for Statistics and Data Analysis*. Chapman & Hall. London.

Simonoff, J. S. (1996) *Smoothing Methods in Statistics*. Springer-Verlag. New York

Venables, W.N. & Ripley, B.D. (1997) *Modern Applied Statistics with S-PLUS*. Springer-Verlag. New York.

See Also

`kda.kde`, `pda.pde`

Examples

```
library(MASS)
data(iris)
iris.mat <- rbind(iris[,,1], iris[,,2], iris[,,3])
ir <- iris.mat[,c(1,2)]
```

```
ir.gr <- iris.mat[,5]

H <- Hkda(ir, ir.gr, bw="plugin", pre="scale")
kda.gr <- kda(ir, ir.gr, H, ir)
lda.gr <- pda(ir, ir.gr, ir, type="line")
qda.gr <- pda(ir, ir.gr, ir, type="quad")

compare(kda.gr, ir.gr)
compare(qda.gr, ir.gr)
compare(lda.gr, ir.gr)
```

`kda.kde, pda.pde` *Density estimates for kernel and parametric discriminant analysis*

Description

Density estimates for kernel and parametric discriminant analysis.

Usage

```
kda.kde(x, x.group, Hs, gridsize, supp=3.7, eval.points=NULL)
pda.pde(x, x.group, gridsize, type="quad", xlim, ylim)
```

Arguments

`x`	matrix of training data values
`x.group`	vector of group labels for training data
`Hs`	(stacked) matrix of bandwidth matrices
`gridsize`	vector of number of grid points
`supp`	effective support for standard normal is [`-supp, supp`]
`eval.points`	points that density estimate is evaluated at
`type`	`"line"` = linear discriminant, `"quad"` = quadratic discriminant
`xlim, ylim`	x-axis, y-axis limits

Details

The kernel density estimate is based on `kde`.

If `gridsize` is not set to a specific value, then it defaults to 100 grid points in each co-ordinate direction i.e. `c(100,100)`. Not required to be set if specifying `eval.points`.

If `eval.points` is not specified, then the density estimate is automatically computed over a grid whose resolution is controlled by `gridsize` (a grid is required for plotting).

The parametric discriminant analysers use the code from the `MASS` library namely `lda` and `qda` for linear and quadratic discriminants.

If `xlim` and `ylim` are not specified then they default to be 10 % bigger than the range of the data values.

Value

Density estimate for discriminant analysis is an object of class `dade` which is a list with 6 fields

`x`	data points - same as input
`eval.points`	points that density estimate is evaluated at
`estimate`	density estimate at `eval.points`
`H`	bandwidth matrices
`prior.prob`	sample proportions of each group
`type`	one of `"kernel", "linear", "quadratic"` indicating the type of discriminant analyser used.

References

Simonoff, J. S., (1996) *Smoothing Methods in Statistics*, Springer-Verlag. New York.

Venables, W.N. & Ripley, B.D. (1997) *Modern Applied Statistics with S-PLUS (3rd ed.)*, Springer-Verlag. New York.

See Also

`plot.dade`, `pda`, `kda`, `kde`

Examples

```
library(MASS)
data(iris)
iris.mat <- rbind(iris[,,1], iris[,,2], iris[,,3])
ir <- iris.mat[,c(1,2)]
ir.gr <- iris.mat[,5]

H <- Hkda(ir, ir.gr, bw="plugin", pre="scale")
kda.gr <- kda(ir, ir.gr, H, ir)
fhat <- kda.kde(ir, ir.gr, H, gridsize=c(250,250))

qda.gr <- pda(ir, ir.gr, ir, type="quad")
qda.fhat <- pda.pde(ir, ir.gr, gridsize=c(250,250))
```

`kde` *Kernel density estimate for bivariate data*

Description

Kernel density estimate for bivariate data.

Usage

```
kde(x, H, gridsize, supp=3.7, eval.points)
```

Arguments

`x`	matrix of data values
`H`	bandwidth matrix
`gridsize`	vector of number of grid points
`supp`	effective support for standard normal is [`-supp, supp`]
`eval.points`	points that density estimate is evaluated at

Details

The kernel density estimate is computed exactly i.e. binning is not used.

If `gridsize` is not set to a specific value, then it defaults to 50 grid points in each co-ordinate direction i.e. `c(50,50)`. Not required to be set if specifying `eval.points`.

If `eval.points` is not specified, then the density estimate is automatically computed over a grid whose resolution is controlled by `gridsize` (a grid is required for plotting).

Value

Kernel density estimate is an object of class `kde` which is a list with 4 fields

`x`	data points - same as input
`eval.points`	points that density estimate is evaluated at
`estimate`	density estimate at `eval.points`
`H`	bandwidth matrix

References

Wand, M.P. & Jones, M.C. (1995) *Kernel Smoothing*. Chapman & Hall. London.

See Also

`plot.kde`

Examples

```
data(faithful)
Hpi <- Hpi(faithful)
fhat <- kde(faithful, Hpi)
```

`ise, mise, amise` *ISE, MISE and AMISE of kernel density estimates for normal and t mixture densities*

Description

The global errors ISE (Integrated Squared Error), MISE (Mean Integrated Squared Error) and AMISE (Asymptotic Mean Integrated Squared Error) of kernel density estimates for normal and t mixture densities.

Usage

```
ise.mixt(x, H, mus, Sigmas, props, lower, upper, gridsize, stepsize)
iset.mixt(x, H, mus, Sigmas, dfs, props, lower, upper, gridsize,
          stepsize)
mise.mixt(H, mus, Sigmas, props, samp)
amise.mixt(H, mus, Sigmas, props, samp)
```

Arguments

`x`	matrix of data values
`H`	bandwidth matrix
`mus`	(stacked) matrix of mean vectors
`Sigmas`	(stacked) matrix of variance matrices
`dfs`	vector of degrees of freedom
`props`	vector of mixing proportions
`samp`	sample size
`lower, upper`	vectors of lower, upper bounds for numerical integration
`gridsize`	vector of number of points in each dimension
`stepsize`	vector of step sizes in each dimension

Details

For normal mixture densities, the ISE, MISE and AMISE all have exact formulas. See Wand & Jones (1995). For the t mixture densities, we resort to using numerical integration, using a simple Riemann sum. A grid is set up and the function values are computed and then multiplied by the area of the grid element to give an approximation of the volume under the curve. The resolution of the grid is given either by `gridsize` or `stepsize`.

Value

ISE, MISE or AMISE value.

Note

Remember that ISE is a random variable that depends on the data `x`; and that MISE and AMISE are non-random and don't depend on the data.

References

Wand, M.P. & Jones, M.C. (1995) *Kernel Smoothing.* Chapman & Hall. London.

Examples

```
samp <- 100
mus <- rbind(c(-3/2,0), c(3/2,0))
Sigmas <- rbind(diag(c(1/16, 1)), rbind(c(1/16, 1/18), c(1/18, 1/16)))
props <- c(2/3, 1/3)
x <- rmvnorm.mixt(samp, mus, Sigmas, props)
H <- Hpi(x)
ise.mixt(x, H, mus, Sigmas, props, stepsize=0.01)
mise.mixt(H, mus, Sigmas, props, samp)
amise.mixt(H, mus, Sigmas, props, samp)

dfs <- c(7,5)
x <- rmvt.mixt(samp, mus, Sigmas, dfs, props)
H <- Hpi(x)
iset.mixt(x, H, mus, Sigmas, dfs, props, lower=c(-5,-5), upper=c(5,5))
```

`rmvnorm.mixt, dmvnorm.mixt`

Multivariate normal mixture distribution

Description

Random generation and density values from multivariate normal mixture distribution.

Usage

```
rmvnorm.mixt(n=100, mus=c(0,0), Sigmas=diag(2), props=1)
dmvnorm.mixt(x, mus, Sigmas, props)
```

Arguments

`n`	number of random variates
`x`	matrix of quantiles
`mus`	(stacked) matrix of mean vectors
`Sigmas`	(stacked) matrix of variance matrices
`props`	vector of mixing proportions

Details

`rmvnorm.mixt` is based on the `rmvnorm` function from the `mvtnorm` library.

Value

Multivariate normal mixture random vectors and density values.

See Also

`rmvt.mixt`, `dmvt.mixt`

Examples

```
mus <- rbind(c(-3/2,0), c(3/2,0))
Sigmas <- rbind(diag(c(1/16, 1)), rbind(c(1/16, 1/18), c(1/18, 1/16)))
props <- c(2/3, 1/3)
x <- rmvnorm.mixt(1000, mus, Sigmas, props)
dens <- dmvnorm.mixt(x, mus, Sigmas, props)
```

`rmvt.mixt, dmvt.mixt`

Multivariate t mixture distribution

Description

Random generation and density values from multivariate t mixture distribution.

Usage

```
rmvt.mixt(n=100, mus=c(0,0), Sigmas=diag(2), dfs=3, props=1)
dmvt.mixt(x, mus, Sigmas, dfs, props)
```

Arguments

`n`	number of random variates
`x`	matrix of quantiles
`mus`	(stacked) matrix of location vectors
`Sigmas`	(stacked) matrix of dispersion matrices
`dfs`	vector of degrees of freedom
`props`	vector of mixing proportions

Details

`rmvt.mixt` is based on the `rmvt` function from the `mvtnorm` library.

The formula for a d-variate t density with location vector $\boldsymbol{\mu}$, dispersion matrix $\boldsymbol{\Sigma}$ and df degrees of freedom is

$$k \left[1 + \frac{1}{df} (\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) \right]^{-(d+df)/2}$$

where

$$k = \frac{\Gamma((df + d)/2)}{(df\pi)^{d/2} \Gamma(df/2) |\boldsymbol{\Sigma}^{1/2}|}.$$

Value

Multivariate t mixture random vectors and density values.

See Also

`rmvnorm.mixt`, `dmvnorm.mixt`

Examples

```
mus <- rbind(c(-3/2,0), c(3/2,0))
Sigmas <- rbind(diag(c(1/16, 1)), rbind(c(1/16, 1/18), c(1/18, 1/16)))
props <- c(2/3, 1/3)
dfs <- c(7,3)
x <- rmvt.mixt(1000, mus, Sigmas, dfs, props)
dens <- dmvt.mixt(x, mus, Sigmas, dfs, props)
```

`plot.dade` *Density estimates and partition plot for discriminant analysis for bivariate data*

Description

Density estimate plot and partition for bivariate data for kernel, linear and quadratic discriminant analysis

Usage

```
## S3 method for class 'dade':
plot(fhat, y, y.group, prior.prob=NULL, display="part", cont,
     ncont=NULL, ...)
```

Arguments

`fhat`	an object of class `dade` i.e. output from `kda.kde` or `pda.pde`
`display`	include plot of partition classes
`y`	matrix of test data points
`y.group`	vector of group labels for test data points
`prior.prob`	vector of prior probabilities
`cont`	vector of percentages (of maximum height) for contour level curves
`ncont`	number of contour level curves
`...`	other graphics parameters

Details

If `prior.prob` is set to a particular value then this is used. The default is `NULL` which means that the sample proportions are used.

If `display="part"` then a partition induced by the discriminant analysis is also plotted. If this is not desired then set `display=""`. Its colours are controlled by `col` (the default is 2 to $\nu + 1$, where ν is the number of groups).

Unlike `plot.kde`, the contour plots are automatically added to the plot. The line types are set by `lty` (the default is 1 to ν). Also, `cont` and `ncont` control the number of level curves (only one of these needs to be set).

The object `fhat` contains the training data and its group labels. If `y` and `y.group` are missing then the training data points are plotted. Otherwise, the test data `y` are

plotted. The plotting symbols are set by `pch` (the default is 1 to ν), one for each group.

Value

Plot of density estimates (and partition) for discriminant analysis is sent to graphics window.

References

Simonoff, J. S., (1996) *Smoothing Methods in Statistics*. Springer-Verlag. New York.

See Also

`kda.kde`, `pda.pde`, `kda`, `pda`

Examples

```
library(MASS)
data(iris)
iris.mat <- rbind(iris[,,1], iris[,,2], iris[,,3])
ir <- iris.mat[,c(1,2)]
ir.gr <- iris.mat[,5]
xlab <- "Sepal length (mm)"
ylab <- "Sepal width (mm)"
xlim <- c(4,8)
ylim <- c(2,4.5)

H <- Hkda(ir, ir.gr, bw="plugin", pre="scale")
fhat <- kda.kde(ir, ir.gr, H, gridsize=c(250,250))
lda.fhat <- pda.pde(ir, ir.gr, type="line")
qda.fhat <- pda.pde(ir, ir.gr, type="quad")

layout(rbind(c(1,2), c(3,4)))
plot(fhat, cont=0, xlab=xlab, ylab=ylab, xlim=xlim, ylim=ylim, pch=c(1,5,10))
plot(fhat, ncont=6, xlab=xlab, ylab=ylab, xlim=xlim, ylim=ylim,
     col=c("transparent", "grey", "#8f8f8f"), drawlabels=FALSE)
plot(lda.fhat, ncont=6, xlim=xlim, ylim=ylim,  xlab=xlab, ylab=ylab, disp="")
plot(qda.fhat, ncont=6, xlim=xlim, ylim=ylim,  xlab=xlab, ylab=ylab,
     lty=c(2,5,3))
layout(1)
```

`plot.kde` *Kernel density estimate plot for bivariate data*

Description

Kernel density estimate plot for bivariate data.

Usage

```
## S3 method for class 'kde':
plot(fhat, display="slice", ...)
```

Arguments

`fhat`	an object of class `kde` i.e. output from `kde` function
`display`	type of display
`...`	other graphics parameters - see details below

Details

There are three types of plotting displays available, controlled by the `display` parameter.

If `display="slice"` then a slice/contour plot is generated using `contour`. The default contours are at 25%, 50%, 75% or `cont=c(25,50,75)`. The user can also set the number of contour level curves by changing the value set to `ncont`. See examples below.

If `display="persp"` then a perspective/wire-frame plot is generated. The default z-axis limits `zlim` are determined by the range of the z values i.e. default from the usual `persp` command.

If `display="image"` then an image plot is generated. The colours are the default from the usual `image` command.

Value

Plot of kernel density estimate is sent to graphics window.

References

Bowman, A.W. & Azzalini, A. (1997) *Applied Smoothing Techniques for Data Analysis.* Clarendon Press. Oxford.

Simonoff, J. S., (1996) *Smoothing Methods in Statistics.* Springer-Verlag. New York.

See Also

`kde`

Examples

```
data(faithful)
Hpi <- Hpi(faithful)
fhat <- kde(faithful, Hpi)

layout(rbind(c(1,2), c(3,4)))
plot(fhat, display="slice", cont=seq(10,90, by=20), cex=0.3)
plot(fhat, display="slice", ncont=5, cex=0.3, drawlabels=FALSE)
plot(fhat, display="persp")
plot(fhat, display="image", col=rev(heat.colors(15)))
layout(1)
```

---

pre.sphere, pre.scale

Pre-sphering and pre-scaling

Description

Pre-sphered or pre-scaled version of data.

Usage

```
pre.sphere(x)
pre.scale(x)
```

Arguments

x	matrix of data values

Details

For pre-scaling, the data values are pre-multiplied by $\mathbf{S}^{-1/2}$ and for pre-scaling, by $(\mathbf{S}_D)^{-1/2}$ where $\mathbf{S}$ is the sample variance and $\mathbf{S}_D$ is diag (S_1^2, S_2^2) and S_1^2, S_2^2 are the marginal sample variances.

If $\mathbf{H}^*$ is the bandwidth matrix for the pre-transformed data and $\mathbf{H}$ is the bandwidth matrix for the original data, then $\mathbf{H} = \mathbf{S}^{1/2}\mathbf{H}^*\mathbf{S}^{1/2}$ or $\mathbf{H} = \mathbf{S}_D^{1/2}\mathbf{H}^*\mathbf{S}_D^{1/2}$ as appropriate.

Value

Pre-sphered or pre-scaled version of data. These pre-transformations are required for implementing the plug-in `Hpi` selectors and the smoothed cross validation `Hscv` selectors.

References

Wand, M.P. & Jones, M.C. (1994) *Multivariate plugin bandwidth selection.* Computational Statistics **9**, 97-116.

Duong, T. & Hazelton, M.L. (2003) *Plug-in bandwidth matrices for bivariate kernel density estimation.* Journal of Nonparametric Statistics **15**, 17-30.

Examples

```
x <- rmvnorm.mixt(1000, mus=c(0,0), Sigmas=rbind(c(1,0.2), c(0.2, 0.5)))
x.sp <- pre.sphere(x)
x.sc <- pre.scale(x)
var(x.sp)
var(x.sc)
```

Bibliography

Abdous, B. & Berlinet, A. (1998), 'Pointwise improvement of multivariate kernel density estimates', *Journal of Multivariate Analysis* **65**, 109–128.

Abramson, I. S. (1982), 'On bandwidth variation in kernel estimates—a square root law', *The Annals of Statistics* **10**, 1217–1223.

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

Bowman, A. W. & Azzalini, A. (1997), *Applied Smoothing Techniques for Data Analysis*, Oxford University Press, Oxford.

Breiman, L., Meisel, W. & Purcell, E. (1977), 'Variable kernel estimates of probability density estimates', *Technometrics* **19**, 135–144.

Cacoullos, T. (1966), 'Estimation of a multivariate density', *Annals of the Institute of Statistical Mathematics* **18**, 179–189.

Cao, R., Cuevas, A. & Manteiga, W. G. (1994), 'A comparative study of several smoothing methods in density estimation', *Computational Statistics and Data Analysis* **17**, 153–176.

Chiu, S.-T. (1991), 'Bandwidth selection for kernel density estimation', *The Annals of Statistics* **19**, 1883–1905.

Chiu, S.-T. (1996), 'A comparative review of bandwidth selection for kernel density estimation', *Statistica Sinica* **6**, 126–145.

Cwik, J. & Koronacki, J. (1997*a*), 'A combined adaptive-mixtures/plug-in estimator of multivariate probability densities', *Computational Statistics and Data Analysis* **26**, 199–218.

Cwik, J. & Koronacki, J. (1997*b*), 'Multivariate density estimation: A comparative study', *Neural Computing and Applications* **6**, 173–185.

Deheuvels, P. (1977), 'Estimation non paramétrique de la densité par histogrammes généralisés. II', *Publications de l'Institut de Statistique de l'Université de Paris* **22**, 1–23.

Devroye, L. & Györfi, L. (1985), *Nonparametric Density Estimation: the L_1 View*, John Wiley & Sons Inc., New York.

Duda, R. C. & Hart, P. E. (1973), *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York.

Epanechnikov, V. A. (1969), 'Non-parametric estimation of a multivariate probability density', *Theory of Probability and its Applications* **14**, 153–158.

Everitt, B. S. (1993), *Cluster analysis*, 3rd edn, Edward Arnold, London.

Faraway, J. J. & Jhun, M. (1990), 'Bootstrap choice of bandwidth for density estimation', *Journal of the American Statistical Association* **85**, 1119–1122.

Foster, P. (1995), 'A comparative study of some bias correction techniques for kernel-based density estimators', *Journal of Statistical Computation and Simulation* **51**, 137–152.

Gordon, A. D. (1999), *Classification*, 2nd edn, Chapman & Hall/CRC, London.

Grund, B., Hall, P. & Marron, J. S. (1994), 'Loss and risk in smoothing parameter selection', *Journal of Nonparametric Statistics* **4**, 107–132.

Hall, P. & Marron, J. S. (1987), 'Extent to which least-squares cross-validation minimises integrated square error in nonparametric density estimation', *Probability Theory and Related Fields* **74**, 567–581.

Hall, P. & Marron, J. S. (1991), 'Lower bounds for bandwidth selection in density estimation', *Probability Theory and Related Fields* **90**, 149–173.

Hall, P., Marron, J. S. & Park, B. U. (1992), 'Smoothed cross-validation', *Probability Theory and Related Fields* **92**, 1–20.

Hall, P., Sheather, S. J., Jones, M. C. & Marron, J. S. (1991), 'On optimal data-based bandwidth selection in kernel density estimation', *Biometrika* **78**, 263–269.

Hall, P. & Wand, M. P. (1988), 'On nonparametric discrimination using density differences', *Biometrika* **75**, 541–547.

Hand, D. J. (1982), *Kernel discriminant analysis*, Vol. 2 of *Electronic & Electrical Engineering Research Studies: Pattern Recognition & Image Processing Series*, Research Studies Press [John Wiley & Sons], Chichester.

Hazelton, M. L. (1996), 'Bandwidth selection for local density estimators', *Scandinavian Journal of Statistics. Theory and Applications* **23**, 221–232.

Hazelton, M. L. (1999), 'An optimal local bandwidth selector for kernel density estimation', *Journal of Statistical Planning and Inference* **77**, 37–50.

Hinkley, D. V. (1969), 'On the ratio of two correlated normal random variables', *Biometrika* **56**, 635–639.

Jones, M. C. (1990), 'Variable kernel density estimates and variable kernel density estimates', *The Australian Journal of Statistics* **32**, 361–371.

Jones, M. C. (1991), 'The roles of ISE and MISE in density estimation', *Statistics and Probability Letters* **12**, 51–56.

Jones, M. C. (1992), 'Potential for automatic bandwidth choice in variations on kernel density estimation', *Statistics & Probability Letters* **13**, 351–356.

Jones, M. C. & Kappenman, R. F. (1992), 'On a class of kernel density estimate bandwidth selectors', *Scandinavian Journal of Statistics. Theory and Applications* **19**, 337–349.

Jones, M. C., Marron, J. S. & Park, B. U. (1991), 'A simple root n bandwidth selector', *The Annals of Statistics* **19**, 1919–1932.

Jones, M. C., Marron, J. S. & Sheather, S. J. (1996), 'A brief survey of bandwidth selection for density estimation', *Journal of the American Statistical Association* **91**, 401–407.

Loader, C. R. (1999), 'Bandwidth selection: classical or plug-in?', *The Annals of Statistics* **27**, 415–438.

Loftsgaarden, D. O. & Quesenberry, C. P. (1965), 'A nonparametric estimate of a multivariate density function', *Annals of Mathematical Statistics* **36**, 1049–1051.

Magnus, J. R. & Neudecker, H. (1988), *Matrix Differential Calculus with Applications in Statistics and Econometrics*, John Wiley & Sons Ltd., Chichester.

Marchette, D. J., Priebe, C. E., Rogers, G. W. & Solka, J. L. (1996), 'Filtered kernel density estimation', *Computational Statistics* **11**, 95–112.

Marron, J. S. & Tsybakov, A. B. (1995), 'Visual error criteria for qualitative smoothing', *Journal of the American Statistical Association* **90**, 499–507.

Mathsoft (1999), *S-PLUS 2000 Guide to Statistics, Volume I*, Data Analysis Products Division, MathSoft, Seattle, WA.

Miller, K. S. (1987), *Some Eclectic Matrix Theory*, Robert E. Krieger Publishing Co. Inc., Melbourne, FL.

Milligan, G. W. & Cooper, M. C. (1985), 'An examination of procedures for determining the number of clusters in a data det', *Pyschometrika* **50**, 159–179.

Park, B. U. & Marron, J. S. (1990), 'Comparison of data-driven bandwidth selectors', *Journal of the American Statistical Society* **85**, 66–72.

Park, B. U. & Turlach, B. A. (1992), 'Practical performance of several data driven bandwidth selectors (with discussion)', *Computational Statistics* **7**, 251–270. Correction in Vol. 9, p. 79.

Parzen, E. (1962), 'On estimation of a probability density function and mode', *The Annals of Mathematical Statistics* **33**, 1065–1076.

R Development Core Team (2003), *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria.

Rosenblatt, M. (1956), 'Remarks on some nonparametric estimates of a density function', *The Annals of Mathematical Statistics* **27**, 832–837.

Rudemo, M. (1982), 'Empirical choice of histograms and kernel density estimators', *Scandinavian Journal of Statistics. Theory and Applications* **9**, 65–78.

Sain, S. R. (2002), 'Multivariate locally adaptive density estimation', *Computational Statistics & Data Analysis* **39**, 165–186.

Sain, S. R., Baggerly, K. A. & Scott, D. W. (1994), 'Cross-validation of multivariate densities', *Journal of the American Statistical Association* **89**, 807–817.

Sain, S. R. & Scott, D. W. (1996), 'On locally adaptive density estimation', *Journal of the American Statistical Association* **91**, 1525–1534.

Schimek, M. G., ed. (2000), *Smoothing and Regression*, John Wiley & Sons Inc., New York.

Schucany, W. R. (1989), 'Locally optimal window widths for kernel density estimation with large samples', *Statistics & Probability Letters* **7**, 401–405.

Scott, D. W. (1992), *Multivariate Density Estimation: Theory, Practice, and Visualization*, John Wiley & Sons Inc., New York.

Scott, D. W. & Terrell, G. R. (1987), 'Biased and unbiased cross-validation in density estimation', *Journal of the American Statistical Association* **82**, 1131–1146.

Sheather, S. J. (1992), 'The performance of six popular bandwidth selection methods on some real data sets (with discussion)', *Computational Statistics* **7**, 225–250, 271–81.

Sheather, S. J. & Jones, M. C. (1991), 'A reliable data-based bandwidth selection method for kernel density estimation', *Journal of the Royal Statistical Society. Series B. Methodological* **53**, 683–690.

Silverman, B. W. (1986), *Density Estimation for Statistics and Data Analysis*, Chapman & Hall, London.

Simonoff, J. S. (1996), *Smoothing Methods in Statistics*, Springer-Verlag, New York.

Stone, C. J. (1984), 'An asymptotically optimal window selection rule for kernel density estimates', *The Annals of Statistics* **12**, 1285–1297.

Taylor, C. C. (1989), 'Bootstrap choice of the smoothing parameter in kernel density estimation', *Biometrika* **76**, 705–712.

Terrell, G. R. (1990), 'The maximal smoothing principle in density estimation', *Journal of the American Statistical Association* **85**, 470–477.

Terrell, G. R. & Scott, D. W. (1992), 'Variable kernel density estimation', *The Annals of Statistics* **20**, 1236–1265.

Turlach, B. (1993), 'Bandwidth selection in kernel density estimation: a review', *Discussion paper 9317*. Institut de Statistique, Voie du Roman Pays, B-1348, Louvain-la-Neuve.

UNICEF (2003), *The State of the World's Children 2003*, Oxford University Press for UNICEF, New York.

Victor, N. (1976), Nonparametric allocation rules, *in* F. T. Dombal & F. Grémy, eds, 'Decision Making and Medical care: Can Information Science Help?', North-Holland, Amsterdam, pp. 515–529.

Wagner, T. J. (1975), 'Nonparametric estimates of probability densities', *IEEE Transactions on Information Theory* **IT-21**, 438–440.

Wand, M. P. (1992), 'Error analysis for general multivariate kernel estimators', *Journal of Nonparametric Statistics* **2**, 1–15.

Wand, M. P. & Jones, M. C. (1993), 'Comparison of smoothing parameterizations in bivariate kernel density estimation', *Journal of the American Statistical Association* **88**, 520–528.

Wand, M. P. & Jones, M. C. (1994), 'Multivariate plug-in bandwidth selection', *Computational Statistics* **9**, 97–116.

Wand, M. P. & Jones, M. C. (1995), *Kernel Smoothing*, Chapman and Hall Ltd., London.