BUMP HUNTING IN REGRESSION REVISITED

by

Jaroslaw Harezlak

B.Sc., Simon Fraser University, Burnaby, B.C., 1995

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

Department of Statistics

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

August, 1998

Department of Statistics

The University of British Columbia
Vancouver, Canada

Date August 12, 1998

Abstract

Suppose bivariate data $\{(t_i, Y_i), i = 1, \ldots, n\}$ are observed at times $a \leq t_1 \leq t_2 \leq \ldots \leq t_n \leq b$. Given a nonparametric regression model $Y_i = m(t_i) + \varepsilon_i$ with ε_i i.i.d. mean 0, variance σ^2, for $i = 1, 2, \ldots, n$, we want to estimate the number of modes of the underlying regression function $m(\cdot)$ or its derivative. We use the penalized least squares technique to get an estimate of $m(\cdot)$, i.e. the function minimizing

$$\sum_{i=1}^{n}(Y_i - m(t_i))^2 + \lambda \int_a^b Lm(t)\, dt$$

where $\int_a^b Lm(t)\, dt$ is a penalty function and λ is a smoothing parameter. The estimate of the derivative, $\widehat{m'}$, is simply the derivative of the estimate, $(\hat{m})'$.

A new test of multimodality is introduced and its performance is studied. Our idea is motivated by the test proposed by Silverman (1981) concerning the number of modes in the density function. He used a "critical bandwidth" as a test statistic in a kernel smoothing context. He noted that if the data are strongly bimodal, we would need a large value of a bandwidth to obtain a unimodal density estimate. In our case we define the "critical smoothing parameter" λ_{crit} as the smallest λ giving an estimate with the specified number of modes. We use λ_{crit} as a test statistic in our new test $\mathcal{C}$ri$\mathcal{SP}$.

We use bootstrap techniques to assess the performance of our test. We study the effects of the penalty L on the quality of our test via simulation using different regression functions and we compare it with Bowman et.al.'s monotonicity test (1998). $\mathcal{C}$ri$\mathcal{SP}$ is also applied to the children's growth data in studying the number of bumps in the derivatives of the growth functions. In a sample of 43 boys and 50 girls, our test procedure gives an automatic classification rule in about 80% of the growth curves analyzed.

Contents

List of Tables

List of Figures

Acknowledgements

First and foremost, I would like to thank my supervisor, Nancy Heckman, for her support and guidance throughout the development of this thesis. Nancy was always willing to give her advice and encouragement which were greatly appreciated.

Secondly, I would like to thank Paul Gustafson for his comments and suggestions on improving this manuscript. My gratitude extends to all other faculty members and particularly to Jean Meloche and Michael Schulzer for expanding my understanding of statistics.

In addition, I must acknowledge the work of Jim Ramsay who provided us with the `Splus` programs `Pspline` and `Lspline`, and the Berkeley growth data.

Thanks also to all fellow graduate students who were always ready to share their knowledge and expertise, and especially to Matías Salibián-Barrera for his amity, wit and patience.

Finally, I would like to thank the Natural Science and Engineering Research Council of Canada for its financial support during my graduate studies.

Chapter 1

Introduction

Statisticians often want to find an underlying "true" regression function when given a set of noisy data. We assume, very often, that this function is of a parametric form. For instance, in the case of polynomial regression, we search for the "truth" in the class of polynomials. However once we adopt a specific parametric form of a function, we cannot detect features of the data beyond those found in the specified model. In the event we adopt the exponential parametric model, the regression function is monotone, i.e. implicitly we restrict the regression relationship to be of increasing or decreasing form. However, on a number of occasions we want to allow more flexibility in modeling. We can achieve that by not assuming any specific model, i.e. by taking advantage of nonparametric regression. This approach permits a detection of fine features in the function, e.g. bumps and valleys.

Let us consider the growth data collected on children in Berkeley, California. This data set consists of height measurements taken at least once per year from an age of one till age eighteen. Our main goal is to discover interesting features in the speed and acceleration of growth. A few parametric models for the speed of growth have been proposed in the literature. Gasser et.al. (1984) give a good review of the existing models. However, all of the models assume the knowledge of the shape of the regression function. For example, we know that most children have pubertal spurts of growth. Therefore we include this spurt in the model. However, the studies in 1930's and 1940's indicated that some children experience prepubertal growth acceleration. Such growth spurts cannot occur in the "pubertal growth spurt only" model. A new parametric model would be needed.

Thus we approach the problem from the nonparametric point of view. We allow ourselves the flexibility of modeling the growth curve but without prespecification of the number of spurts. Unfortunately, a problem arises in the nonparametric approach. We are forced to choose a so called "smoothing parameter" controlling the balance between goodness of fit to the data and smoothness of the regression function. Largo et.al. (1978) pioneered the use of smoothing splines functions and Gasser et.al. (1984) introduced kernel estimation in the analysis of growth data. Largo et.al. selected the "optimal" smoothing parameter using a cross validation procedure. Gasser et.al. described a complicated procedure for choosing the "optimal" smoothing parameter. Ramsay et.al. (1995) in their analysis chose the smoothing parameter: "in part by a data-driven process called generalized cross validation, and in part by experiments with simulated data". We can look at Figure 1.1 to see the importance of smoothing parameter selection. At low values of the smoothing parameter, we produce very wiggly estimates, and at large values we lose all the important features of the curve. Construction of these function estimates is discussed in Chapter 3. In our approach, we avoid this problem entirely. We actually use a specially defined smoothing parameter to test for the existence of a certain number of bumps in the regression function.

We use a testing method inspired by the work of Silverman (1981, 1983). His interest was in testing for the number of modes of an underlying density function. He

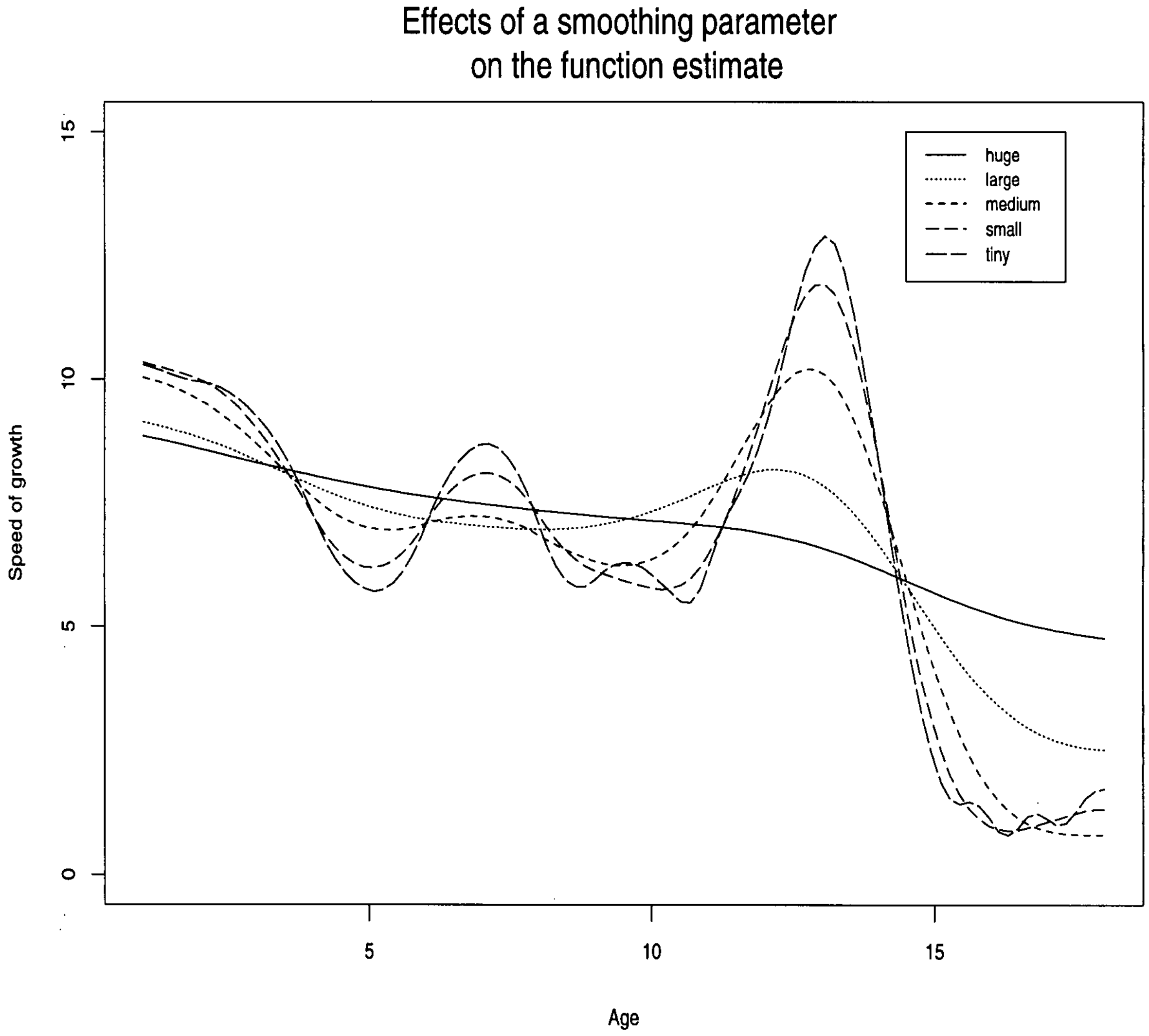


Figure 1.1: Estimates of the speed of growth of male # 1.

used the kernel density estimation method. This local approach produces a smooth histogram, with the smoothing parameter (also called a *bandwidth*), h, playing the role of bin-width. Now, if the data are strongly bimodal, we would need a large value of h to obtain a unimodal estimate. Silverman used this idea to test the null hypothesis that the density f has k modes versus the alternative that f has more than k modes.

Bowman et.al. (1997) extended Silverman's idea to regression functions. They were interested in testing for the monotonicity of the underlying regression function. Here, the nonparametric regression function estimate used is found by a local linear kernel method. This procedure fits a local least squares line to the data. A bandwidth h controls how many adjacent points to the point of interest are used in the local fit. Again, to have a monotone regression function estimate when the "truth" is non-monotone, the bandwidth would have to be very large. Both, Silverman's and Bowman's procedures are described in more detail in Chapter 4.

We extend Bowman's monotonicity test to $\mathcal{C}$ri$\mathcal{SP}$ (Critical Smoothing Parameter), a new test of

$$H_0 : \#\text{ of bumps in } m(\cdot) \ \leq k$$

vs.

$$H_1 : \#\text{ of bumps in } m(\cdot) \ > k$$

where $m(\cdot)$ is the "true" regression function. We use a spline smoothing procedure in the estimation step. Our testing method is based on the critical value of a smoothing parameter which is defined as the smallest smoothing parameter yielding a regression curve with k bumps. The p-value of the test is calculated via bootstrapping.

The general scenario that we follow in this thesis has three parts:

1. Defining a *bump* in $m(\cdot)$.

2. Estimating the regression function $m(\cdot)$.

3. Defining a test statistic and conducting a hypothesis test.

In Chapter 2, we discuss possible definitions of a *bump*, and we select the most practical working definition. Methods of estimation are presented in Chapter 3. We mainly use spline smoothing methods, but for comparison purposes kernel smoothing is also utilized. Chapter 4 gives an overview of existing testing procedures. There we also define our test $\mathcal{C}\mathrm{ri}\mathcal{SP}$, and we provide the way for calculating the p-value associated with it. Extensive simulation studies and the analysis of the Berkeley growth data are presented in Chapter 5.

Chapter 2

Bumps: what are they?

In this chapter we outline the framework of our problem. The next section presents the definitions and facts about modes and bumps in densities, and section 2.2 extends the definitions to regression functions.

2.1 Bumps in Density Functions

Let us focus our attention for the moment on the distribution $F(\cdot)$ of a random variable X. Let $f(\cdot)$ be its density and $f^{(k)}(\cdot)$ the k-th derivative of the density $f(\cdot)$ when it exists. M is a mode of the distribution of X if $f(\cdot)$ has a local maximum at M. In practical applications and in the rest of the thesis, we assume $f(\cdot)$ to be continuous and at least twice continuously differentiable. Thus, we can define a *mode* as follows:

Definition 2.1 *M is a mode if and only if $f'(M) = 0$ and $\exists\, \varepsilon > 0$ such that $f'(t) > 0$ for $t \in (M - \varepsilon, M)$ and $f'(t) < 0$ for $t \in (M, M + \varepsilon)$.*

Usually, each mode has two associated *antimodes* $t_L < M$ and $t_U > M$ (or local minima), defined by

$$t_L = \max_{t<M} t \quad \text{such that} \quad f'(t_L) = 0,\ f''(t_L) > 0 \tag{2.1}$$

$$t_U = \min_{t>M} t \quad \text{such that} \quad f'(t_U) = 0,\ f''(t_U) > 0. \tag{2.2}$$

We want to study properties of bumps, but we have not clarified yet what we mean when we say that a function has a "bump". The first difficulty arises in the definition of a *bump*. There have been a few definitions proposed in the literature, and we discuss here the approaches taken to define a *bump*. Since $t_L < M < t_U$, we could give the following definition of a *bump*.

Definition 2.2 *$f(\cdot)$ has one bump in $[t_L, t_U]$ if and only if there exists a mode $M \in (t_L, t_U)$ and t_L, t_U satisfy properties* (2.1) *and* (2.2) *respectively.*

However, definition 2.2 of a bump does not cover all the possibilities, since not every mode has two antimodes (e.g. the Gaussian density has a mode, but no antimodes). Another way to define a *bump* in $f(\cdot)$ is:

Definition 2.3 *$f(\cdot)$ has a bump in $[t_L, t_U]$ if and only if*

$$f''(t_L) = 0 = f''(t_U) \tag{2.3}$$

$$\forall t \in (t_L, t_U) \;\; f''(t) < 0 \tag{2.4}$$

and $[t_L, t_U]$ is the shortest interval with properties (2.3) *and* (2.4).

Definition 2.3 extends the class of functions with bumps. According to this definition, it is clear that every mode M with $f''(M) \neq 0$ has a surrounding bump, but the converse is not true. We can encounter a bump in a density function where the second derivative changes sign, but the first derivative does not. In this case, we say that there is a *shoulder* in $[t_L, t_U]$. Therefore, we have to adjust the definition of a *bump* to avoid the situations when we have mere changes of concavity on the slopes of a "true" bump.

Combining the features of definitions 2.2 and 2.3, we give the final definition of a *bump.*

Definition 2.4 *$f(\cdot)$ has one bump in $[t_L, t_U]$ if and only if*

- *There exists a mode $M \in (t_L, t_U)$,*
- $\forall t \in (t_L, M)$ $f'(t) > 0$, *and*
- $\forall t \in (M, t_U)$ $f'(t) < 0$

The second problem is the difficulty of estimating the number of bumps in a density function. Let us consider a random variable X with a probability distribution $F(\cdot)$. When we take a sample from it, we get little information about the tails of the distribution. In a simple case of estimating the mean μ of X in a nonparametric way, Bahadur and Savage (1956) showed that there is no effective way to get a truly nonparametric point estimate of μ or a confidence interval for μ.

Similar problems are encountered when estimating nonlinear functionals of a density function $f(\cdot)$ or its derivatives. Two integer-valued functionals that are commonly looked at are mixture complexity $K(F)$ (the number of mixture terms needed to represent a density) and $M(F)$, number of modes in $f(\cdot)$. In the applied setting, it is of interest to estimate such functionals. However, there is a logical difficulty with this, since all the functionals depend on the existence of a density with some amount of regularity. *And we cannot verify empirically that a density is well-behaved.* Donoho (1988) showed that we can still make inferences about the values of $K(F)$ or $M(F)$

with a non-parametric validity. We have a restriction, however, that all the statements are of a one-sided nature. In the $M(F)$ example, we can make a statement: "I have 90% confidence that the number of modes of the underlying distribution is *at least* 2". The statement above does not depend on the assumptions about the underlying distribution. It is totally empirical and it has at least the indicated coverage. Moreover, Donoho showed that using his procedure we get one-sided bounds that are consistent and converge at rapid rates.

The main idea behind the non-existence of upper limits of some functionals is that close to any distribution of interest, there are indistinguishable distributions for which the functional takes on arbitrarily large values. Given a fixed sample size, it is not possible to put an upper bound on the functional based only on the empirical evidence. We would have to make some a priori assumptions that are not testable by data on hand.

There is an ambiguity arising in the definition of the functional $M(F)$ counting the number of modes. When a distribution is flat on some interval, we can argue that it has infinitely many or just one mode there. Donoho gave a definition which yields only one mode in such a situation. He used Silverman's idea (1981) of a convolution $\Phi_h * F$ of a normal distribution function Φ_h with variance h and an arbitrary distribution function F. $G = \Phi_h * F$ is a smooth distribution function with derivatives of all orders. Therefore, Donoho defined the number of modes of G, $M(G)$, as the number of downcrossings in G''. Using Silverman's result (Theorem 4.1) about monotonicity of $M(G)$ with respect to h, $M(F)$ was defined as

$$M(F) = \lim_{h \to 0} M(\Phi_h * F) \tag{2.5}$$

In the light of Definition 2.5, we can make statements of the form: the correct number of modes (bumps) may be larger than a certain number, but I have a high confidence that it is not smaller. In general, the data can invalidate simple models, e.g. a distribution with one mode. However, data cannot usually rule out complex models. Therefore, inference about most measures of complexity pertain to lower bounds, but not upper ones.

2.2 Bumps in Regression Functions

Let us turn our attention to the regression problem. We define $m(\cdot)$ to be the true (smooth) regression function and we denote its k-th derivative by $m^{(k)}(\cdot)$. We assume $m(\cdot)$ is defined on an open interval (a, b) where either a can be $-\infty$ or b can be $+\infty$. We define a *mode* in $m^{(k)}(\cdot)$ for $k = 0, 1, 2, \ldots$ by

Definition 2.5 *$m^{(k)}(\cdot)$ has a mode M if and only if M is a local maximum of $m^{(k)}(\cdot)$.*

Following the definition 2.4 for a *bump* in a density function, we define a *bump* in the regression function $m(\cdot)$ or in its k-th derivative $m^{(k)}(\cdot)$ by

Definition 2.6 *$m^{(k)}(\cdot)$ has one bump in $[t_L, t_U]$ if and only if*

- *there exists a mode $M \in (t_L, t_U)$,*
- *$\forall t \in (t_L, M)$ $m^{(k+1)}(t) > 0$, and*
- *$\forall t \in (M, t_U)$ $m^{(k+1)}(t) < 0$.*

In section 4.3, we discuss ways to estimate the number of bumps in regression functions and their derivatives.

2.3 Testing in Density versus Regression

In the last two sections, we defined modes and bumps as well as presented some results pertaining to them. So far, we have looked at the similarities between density and regression functions. However, the questions asked in the two mentioned contexts are usually different. In the density framework, the main questions arising are "Is the density function unimodal?" or "Should we make use of the bimodal mixture?". In the regression setting, we are more interested in the monotonicity, or possibly in the existence of a few local maxima in the function. Therefore, we can test for 0 bumps in the regression context, but not in the density function environment. Besides, very often we are interested in the number of bumps in the derivatives of the regression function. For example, in the growth data analyzed in section 5.3 we inquire about possible spurts of growth, requiring the estimation or testing for the number of bumps in the speed of growth. We can be concerned as well with the change in the speed of growth, i.e. with acceleration. We will show in Chapter 4 how our testing procedure can be applied in tests for regression functions or their derivatives.

Chapter 3

Methods of Estimation

In this chapter, we give a brief summary of the nonparametric techniques used in univariate regression estimation. Some of the most popular methods are based on kernel functions, spline functions and wavelets. The first two of these methods have become commonly used in recent years. In section 3.1, we give a review of prevalent kernel smoothing methods with special emphasis on the local linear approach. Spline smoothing is reviewed in section 3.2. We concentrate mostly on smoothing splines.

3.1 Kernel Smoothing

Before giving an overview of kernel-based nonparametric regression, we will introduce some relevant terminology and notation. We focus on the *fixed design* context with equally spaced non-random numbers $(t_1, ..., t_n) \in [a, b]$. The response variable is assumed to satisfy

$$Y_i = m(t_i) + \varepsilon_i, \quad i = 1, ..., n \tag{3.1}$$

where $\varepsilon_1, ..., \varepsilon_n$ are independent random variables with

$$E(\varepsilon_i) = 0 \text{ and } \mathrm{Var}(\varepsilon_i) = \sigma^2.$$

We call $m(\cdot)$ the *mean (or true) regression function.*

The fundamental idea behind kernel smoothing lies in the following premise: data points far away from t carry little information about the value of $m(t)$. Thus, the intuitive estimator for the *mean regression function* is the running local average. An improved version of it is the locally weighted average. Let $K(\cdot)$ be a function which will be used to assign weights. Usually $K(\cdot)$ is a probability density symmetric about 0 and is called a *kernel function.* Let h be a *bandwidth* (sometimes called a *smoothing parameter*), which is a nonnegative number controlling the size of a window around the point of interest t. With notation $K_h(\cdot) = K(\cdot/h)/h$, the Nadaraya-Watson kernel regression estimator is given by

$$\hat{m}_h(t) = \frac{\sum_{i=1}^n K_h(t_i - t)Y_i}{\sum_{i=1}^n K_h(t_i - t)}. \tag{3.2}$$

Most commonly used kernel functions include the normal kernel

$$K(t) = (\sqrt{2\pi})^{-1} \exp(-t^2/2),$$

and the "symmetric Beta family"

$$K(t) = \frac{1}{\mathrm{Beta}(1/2, \gamma+1)}(1-t^2)_+^{\gamma}, \quad \gamma = 0, 1, ..., \tag{3.3}$$

where the subscript + denotes the positive part taken before the exponentiation. The choices γ = 0,1,2, and 3 in (3.3) lead respectively to the *uniform*, the *Epanechnikov*, the *biweight* and the *triweight* kernel function. The selection of the kernel function is not very important in general. However, we choose to work with the normal kernel, since it enables us to use convolution theorems true only in the normal case (see Silverman (1981) and Babaud et.al. (1986)). On the other hand, the choice of the bandwidth h is crucial for the estimation. Small values of h give us an *undersmoothed* estimate, and large values of h give us an *oversmoothed* estimate.

Gasser and Müller (1979) proposed another way to introduce weights. Their estimator (also called the convolution weighted estimator) is given by:

$$\hat{m}_h(t) = \sum_{i=1}^{n} Y_i \int_{s_{i-1}}^{s_i} K_h(t-u)du, \tag{3.4}$$

where $s_0 = -\infty$, $s_n = \infty$, $t_i \leq s_i \leq t_{i+1}$ for $j = 1, ..., n-1$ and $K(\cdot)$ is a density function.

Both kernel estimators mentioned so far use a local constant approximation. We can see that by applying a local least squares regression when approximating $m(\cdot)$ by a constant θ. We obtain an estimate

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^{n}(Y_i - \theta)^2 w_i = \sum_{i=1}^{n} w_i Y_i / \sum_{i=1}^{n} w_i,$$

where $w_i = K_h(t_i - t)$ in the case of the Nadaraya-Watson estimator

and $w_i = \int_{s_{i-1}}^{s_i} K_h(t-u)du$ in the case of Gasser-Müller estimator.

Local constant approximation gives us a large bias in estimation of $m(t)$ when t is close to either a or b. A local linear or a higher order local polynomial fit does not have this problem, and so is recommended. Higher order polynomials are preferable when we want to accurately fit finer features of the data, e.g. peaks and valleys. The main drawbacks of higher order local polynomial fitting are its computational complexity and higher degree of sample variability.

Let p be the degree of the local polynomial being fit. We estimate $m(t)$ by $\hat{m}_h(t;p)$ obtained by fitting the polynomial

$$\beta_0 + \beta_1(\cdot - t) + ... + \beta_p(\cdot - t)^p$$

to the (t_i, Y_i)'s using weighted least squares with kernel function $K(\cdot)$. The estimator $\hat{m}_h(t;p)$ is $\hat{\beta}_0$ where $\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, ..., \hat{\beta}_p)'$ is the minimizer of

$$\sum_{i=1}^{n}\{Y_i - \beta_0 - ... - \beta_p(t_i - t)^p\}^2 K_h(t_i - t).$$

The general solution to the minimization problem is given by

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{T}_t' \boldsymbol{W}_t \boldsymbol{T}_t)^{-1} \boldsymbol{T}_t' \boldsymbol{W}_t \boldsymbol{Y}$$

where $\boldsymbol{Y} = (Y_1, ..., Y_n)'$ is the response vector,

$$\boldsymbol{T}_t = \begin{bmatrix} 1 & t_1 - t & \cdots & (t_1 - t)^p \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_n - t & \cdots & (t_n - t)^p \end{bmatrix}$$

is an $n \times (p+1)$ design matrix and

$$\boldsymbol{W}_t = \text{diag}\{K_h(t_1 - t), \ldots, K_h(t_n - t)\}$$

is an $n \times n$ diagonal matrix of weights.

We can derive explicit formulae for the Nadaraya-Watson ($p = 0$) and the local linear ($p = 1$) estimators. In the latter case, we obtain

$$\hat{m}_h(t; 1) = n^{-1} \sum_{i=1}^{n} \frac{\{\hat{s}_2(t; h) - \hat{s}_1(t; h)(t_i - t)\} K_h(t_i - t) Y_i}{\hat{s}_2(t; h) \hat{s}_0(t; h) - \hat{s}_1(t; h)^2} \tag{3.5}$$

where

$$\hat{s}_r(t; h) = n^{-1} \sum_{i=1}^{n} (t_i - t)^r K_h(t_i - t).$$

An important problem in estimation is the choice of the degree p of the polynomial to be locally fitted. For sufficiently smooth regression functions, $\hat{m}_h(t; p)$ has better asymptotic performance for larger values of p. In practice, however, we would need a very large sample size to see the substantial performance improvement. Since odd degree fits have good bias and boundary properties, it is recommended that $p = 1$ or $p = 3$ be used (see Wand and Jones (1995)). In our work, we use $p = 1$, i.e. a local linear kernel regression estimator.

Our interest extends beyond the estimation of the regression functions themselves. In some problems, the estimation of the derivatives of the regression function is more important than estimation of the function itself. For example, in analyzing human growth curves, the first two derivatives of height as a function of age have significant biological meaning. Estimation of the derivatives is also needed for plug-in bandwidth selection procedures (see Wand and Jones (1995)). We can easily use local polynomial fitting to estimate the rth derivative of the regression function. We use the coefficient of the rth derivative of the local polynomial being fitted at t to estimate $m^{(r)}(t)$. For instance, $\hat{m}'_h(t;p) = \hat{\beta}_1$, and in general, the local pth degree estimate of $m^{(r)}(t)$ is $\widehat{m_h^{(r)}}(t;p) = r!\hat{\beta}_r$. It should be noted that $\widehat{m_h^{(r)}}(t;p) \neq (\widehat{m_h}(t;p))^{(r)}$ in general. The preferred selection of the degree p of the local polynomial fit is such that $(p-r)$ is odd. In the case of $(p-r)$ even, we obtain a more complicated expression for the bias of $\widehat{m_h^{(r)}}(t;p)$. More seriously, the bias for t close to the boundary of (a,b) is of higher order than the bias for the t in the interior of (a,b).

As mentioned before, the value of the bandwidth h plays an important role in kernel regression estimation. In many situations a satisfactory choice can be made subjectively by eye. This involves looking at several regression estimates over a range of bandwidths and choosing the "best" in some sense. However, it is also desirable to have an automatic procedure for the bandwidth selection. Currently available methods can be divided into two categories: plug-in and cross-validation. The former technique has many attractive theoretical and practical properties (see Ruppert, Sheather, Wand (1995)), and can be easily implemented in `S-PLUS`.

M.P. Wand developed a function `locpoly` using a plug-in procedure in the bandwidth choice. We use `locpoly` in our simulation study, discussed in detail in Chapter 5.

3.2 Spline Estimation

We turn our attention to another method of nonparametric estimation: splines. Intuitively, we can think about splines as piecewise polynomial functions. The name "spline" comes from the similarity of a spline fit to the curve obtained by a draftsman using a mechanical spline - a thin flexible rod with weights, used to fix the rod at points through which one draws a smooth interpolating curve.

For simplicity, we assume our true regression function $m(\cdot)$ is defined on a closed interval $[0, 1]$. Of course, any closed interval can be chosen.

We first define *a general polynomial spline of degree D*:

Definition 3.1 *A general polynomial spline g is a function on* $[0, 1]$ *with the following properties: Given an integer* $D \geq 1$, *and J points* $0 < t_1 < t_2 < \ldots < t_J < 1$, *called "knots"*

$$\begin{aligned} & g \in \pi^D, \quad t \in [t_i, t_{i+1}], \ \textit{for} \ i = 0, \ldots, J, \\ & \qquad\qquad \textit{where} \ t_0 \equiv 0, \ \textit{and} \ t_{J+1} \equiv 1 \\ \textit{and} \quad & g \in C^{D-1}, \quad t \in [0, 1], \end{aligned}$$

where π^D *is the class of polynomials of degree at most D and* C^D *is the class of functions with D continuous derivatives.*

Thus, g is a D^{th} degree piecewise polynomial with the pieces joined at the knots in such a way that g has $D - 1$ continuous derivatives. This general idea of polynomial splines was of interest to numerical analysts for over 30 years. Many interpolation results have been proved about the convergence of general splines, as well as their derivatives and integrals, as the number of knots becomes dense in $[0, 1]$. Because of these properties as well as good features in smoothing noisy data, splines are of interest to statisticians. The two principal classes of splines used in statistics are *regression splines* and *smoothing splines*. We will present the former briefly in the next section, and the latter in more detail in section 3.2.2.

3.2.1 Regression Splines

We could define regression splines in general, but to be more concrete, we present a cubic regression spline. Let $t_1, \ldots, t_J$ be a knot sequence such that $0 < t_1 < \ldots < t_J < 1$. A cubic spline function g satisfies the two following properties:

- g is a cubic polynomial on each of the intervals $[0, t_1], [t_1, t_2], \ldots, [t_{J-1}, t_J], [t_J, 1]$.
- It is twice continuously differentiable.

The collection of cubic spline functions is a $(J+4)$-dimensional linear space. The two most popular bases used for this space are:

- *power basis*: $(t - t_j)^3_+,\ (j = 1, \ldots, J), 1, t, t^2, t^3$;
- *B-spline basis*: Let $B_i^{(D)}$ denote the i^{th} B-spline basis function of degree D. Therefore, the basis functions for a cubic spline are $\{B_{-3}^{(3)}, \ldots, B_J^{(3)}\}$. We can define $B_i^{(3)}$'s using a recursive formula. Let

  $$\ldots = t_{-1} = t_0 = 0 < t_1 < \ldots < t_J < 1 = t_{J+1} = \ldots$$

  We start with the degree 0 local polynomial, i.e. local constant,

  $$B_i^{(0)}(t) = I\{t \in [t_i, t_{i+1})\} \text{ for } i = 0, \ldots, J-1$$

  and

  $$B_J^{(0)}(x) = I\{t \in [t_J, 1]\}.$$

  Now, given $B_i^{(D-1)}, i = -(D-1), \ldots, J$, define

  $$B_i^{(D)}(t) = \frac{t - t_i}{t_{i+D} - t_i} B_i^{(D-1)}(t) + \frac{t_{i+D+1} - t}{t_{i+D+1} - t_{i+1}} B_{i+1}^{D-1}(t),$$

where we take 0/0=0 and undefined/0=0.

We can observe that $B_i^{(D)}(t)$ is equal to 0 for $t \notin [t_i, t_{i+D+1}]$. Thus, design matrices based on B-splines are sparser than those using the power basis, and so we achieve a better numerical stability with the B-spline basis. This fact plays an important role in the estimation procedure described below.

Using the B-spline basis we can express a cubic spline function as

$$g(t) = \sum_{j=-3}^{J} \theta_j B_j^{(3)}(t).$$

For given knots $t_1, \ldots, t_J$, the spline regression method finds the best spline approximation by the following least squares regression:

$$\min_{(\theta_{-3},\ldots,\theta_J)} \sum_{i=1}^{n} \{Y_i - \sum_{j=-3}^{J} \theta_j B_j^{(3)}(t_i)\}^2.$$

Using this approach, we obtain an estimate of $m(\cdot)$ (unique if $J \geq D$) of the form

$$\hat{m}(t) = \sum_{j=-3}^{J} \hat{\theta}_j B_j(t).$$

The procedure just described is very sensitive to the choice of knots as well as to their location. Knots are usually placed in the spots where curvature of $m(\cdot)$ changes rapidly. One can find in the literature a few methods to deal with knot selection. Most of them are based on the *knot deletion* idea (see Fan and Gijbels (1996)).

3.2.2 Smoothing Splines

Our goal is to estimate the regression function $m(\cdot)$ using the least squares fit, but with the restriction that $\hat{m}(\cdot)$ is not too rough or wiggly. Smoothing splines provide one of the ways to find such an estimate. We want to find a minimizer $\hat{m}_\lambda(\cdot)$ over an appropriate space of functions of

$$\sum_{i=1}^{n} (Y_i - m(t_i))^2 + \lambda \int_0^1 [Lm(t)]^2 dt, \tag{3.6}$$

where $\lambda \geq 0$ is called a smoothing parameter, and L is a linear differential operator of order $k \geq 1$ defined as

$$(Lm)(t) = m^k(t) + \sum_{j=0}^{k-1} w_j m^j(t) \tag{3.7}$$

where $w_j \in \mathbb{R}$ for $j = 0, \ldots, k-1$.

The minimization is carried out over all $m(\cdot)$ in the Sobolev space

$$\mathcal{H}^k[0,1] = \{m : [0,1] \to \mathbb{R} : m^{(j)}, j = 0, \ldots, k-1 \text{ are absolutely continuous and } \int_0^1 (m^{(k)}(t))^2 dt < \infty\}.$$

Going back to the expression (3.6), we can see that the first part of it penalizes for the lack of fit, and the second part penalizes for m when Lm is far from 0. We can see clearly from (3.6) that setting $\lambda = 0$ would give us an interpolation to the data. For large λ, one is, essentially, minimizing the second term, and as $\lambda \to +\infty$, our estimate converges to m with $Lm(t) = 0$ almost everywhere $t \in [0,1]$. We say that m_{def} is a *default model* if and only if $Lm_{def}(t) = 0$ almost everywhere t.

Before looking at the general solution to the minimization problem, let us look at a few examples of the linear differential operator L. The most popular one is $Lm = m''$ with the default model $m_{def}(t) = \alpha_0 + \alpha_1 t$, $(\alpha_0, \alpha_1 \in \mathbb{R})$. The function $\hat{m}_\lambda(\cdot)$ minimizing (3.6) is known as a natural cubic spline function, a general cubic spline with knots $t_1, \ldots, t_n$, with $\hat{m}_\lambda$ linear on $[0, t_1] \cup [t_n, 1]$. Using $m''(t)$ in (3.6) corresponds to penalizing visual roughness. It is very difficult for us to see by eye the behaviour of third or higher derivatives. Penalizing by the second derivative is adequate to enable us to exclude rough and wiggly functions. However, the drawback of this approach is in assuming the default model is a line. In many situations, we can gain considerably in estimation if we use other linear differential operators. One of the classes of L, which includes $Lm = m''$, is $Lm = m^{(k)}$ for $k = 1, 2, 3, \ldots$ with the default model $m_{def}(t) = \alpha_0+\alpha_1 t+\ldots+\alpha_{k-1}t^{k-1}$ $(\alpha_0, \alpha_1, \ldots, \alpha_{k-1} \in \mathbb{R})$. The functions minimizing (3.6) with $Lm = m^{(k)}$ are known as natural splines (general splines of degree $2k$-1, with knots $t_1, \ldots, t_n$, but with zero k^{th} derivative on $[0, t_1] \cup [t_n, 1]$). In another example, used in the growth data analysis (described in chapter 5), the default model is commonly assumed to be of the form $m_{def}(t) = \alpha_0 + \alpha_1 \exp(-\gamma t)$ with the corresponding $Lm = m'' + \gamma m$.

In our work, we use $Lm = m''$ and $Lm = m'' + \gamma m$ as penalty terms in (3.6). For both L's, the default models are monotone. Therefore, when the smoothing parameter $\lambda \to +\infty$ we obtain estimated functions with 0 bumps.

The general solution to the minimization problem (3.6) involves the theory of reproducing kernel Hilbert spaces. The details and proofs of the main results can be found in Wahba (1990) or in Heckman (1997). We present here the most important points. The minimizer of (3.6) is of the form

$$\hat{\mu}(t) = \sum_{i=1}^{k} \hat{\alpha}_i u_i(t) + \sum_{j=1}^{n} \hat{\beta}_j \nu_j(t). \tag{3.8}$$

where the u_i's $(i = 1, \ldots, k)$ form a basis for the default model space, and $\nu_j(t)$ can be calculated by the integration of the Green's function $G(\cdot, \cdot)$ associated with L. The Green's function satisfies

$$\begin{aligned} f(t) &= \int_{u=0}^{1} G(t,u)(Lf)(u)du \\ & \quad \forall f \in \mathcal{H}^k \text{ with } f^{(i)}(0) = 0, \quad i = 0, \ldots, k-1. \end{aligned}$$

G is easily computed, given $u_1, \ldots, u_k$. Then

$$\begin{aligned} \nu_j(t) &= K(t_j, t), \\ K(s,t) &= \int_0^1 G(s,u)G(t,u)du = K(t,s). \end{aligned}$$

The only remaining task is to choose a smoothing parameter λ in such a way that we obtain a "good" fit. As mentioned in the last section, our estimate will go from interpolating the data (when $\lambda \to 0$) to taking the form of the default model (when $\lambda \to +\infty$). Therefore, we would like to find an optimum smoothing parameter, that allows us flexibility in the modeling, but does not let the model dominate the estimation procedure.

A few methods for estimating the optimum smoothing parameter have been proposed in the literature on smoothing, but there is no general agreement on one method being superior to the others. Two possibilities are *cross validation* (CV) and *generalized cross validation* (GCV). Both ideas are based on minimizing the prediction error $MSPE$. Let $\hat{m}_\lambda(\cdot)$ denote the regression function fitted to the data $\{Y_i\}_1^n$. We would like to choose λ in such a way as to minimize

$$MSPE_\lambda = \sum_{i=1}^{n}(Y_i' - \hat{m}_\lambda(t_i))^2, \tag{3.9}$$

where the Y_i''s are new observations at each t_i respectively. The difficulty with this approach is that the Y_i''s are unknown, so we have to replace them with the observed values $\{Y_i\}_1^n$. However, minimizing (3.9) with direct substitution of Y_i for Y_i' would result in the choice of $\lambda = 0$, since we fit the regression model to the "old" data, thus overfitting the data on hand. To adjust for that unpleasant feature, we divide our data set into a training sample ($n-1$ observations) and a testing sample (1 observation) in every possible way, i.e. we estimate the regression function with case i removed, and we compare the fit with the original data point. Therefore, we define the cross-validation function as

Definition 3.2

$$CV_\lambda = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{m}_\lambda^{(-i)}(t_i))^2, \tag{3.10}$$

where $\hat{m}_\lambda^{(-i)}(t_i)$ is the estimate of the regression function with case (t_i, Y_i) removed.

We select λ according to the CV criterion by

$$\lambda_{CV} = \arg\min_{\lambda \in \mathbb{R}} CV_\lambda. \tag{3.11}$$

In practice, this procedure looks very time-consuming, since it seems that we would have to fit n separate curves. However we can use a shortcut making use of only one fit for each value of λ. It can be shown that the solution to the minimization problem (3.6) can be written as a linear combination of the observed values Y_i

$$\hat{\mathbf{m}}_\lambda(\mathbf{t}) = S(\lambda)\mathbf{Y} \tag{3.12}$$

where $\hat{\mathbf{m}}_\lambda(\mathbf{t}) \equiv (\hat{m}_\lambda(t_1), \ldots, \hat{m}_\lambda(t_n))'$ and $\mathbf{Y} = (Y_1, \ldots, Y_n)'$. We call $S(\lambda)$ a hat matrix, because it maps the vector of observed values Y_i into fitted values $\hat{m}_\lambda(t_i)$. The following result

$$Y_i - \hat{m}_\lambda^{(-i)}(t_i) = \frac{Y_i - \hat{m}_\lambda(t_i)}{1 - S_{ii}(\lambda)} \tag{3.13}$$

and the theorem using it can be found in Green and Silverman (1994).

Theorem 3.1 *The cross-validation function CV_λ satisfies*

$$CV_\lambda = n^{-1} \sum_{i=1}^{n} \left(\frac{Y_i - \hat{m}_\lambda(t_i)}{1 - S_{ii}(\lambda)} \right)^2 \tag{3.14}$$

where $\hat{m}_\lambda(\cdot)$ is calculated from the full data set using a smoothing parameter λ.

Using Theorem 3.1, we can write an efficient computer program to calculate CV_λ on a grid of λ values. We do not have a guarantee that the function CV_λ is convex in λ, so it can have many local minima. Therefore a grid search is the safest method of correctly calculating the smoothing parameter λ_{CV}.

Another method used frequently in smoothing parameter choice is *generalized cross validation*, abbreviated by GCV. It is a modified version of cross validation described above. In GCV the factor $1-S_{ii}(\lambda)$ in the denominator of (3.14) is replaced by the average value of $1 - S_{ii}(\lambda)$'s, i.e. by $1 - n^{-1}\mathrm{tr}S(\lambda)$. The generalized cross validation function is

$$GCV_\lambda = n^{-1}\frac{\sum_{i=1}^{n}(Y_i - \hat{m}_\lambda(t_i))^2}{(1-n^{-1}\mathrm{tr}\,S(\lambda))^2}. \tag{3.15}$$

We choose the smoothing parameter λ_{GCV} by minimizing (3.15) over values of λ. One of the reasons GCV was introduced was computational. We can find the trace of the hat matrix without finding its diagonal elements. Nowadays, however there is little difference between computing time for CV and GCV.

The slight advantage of using GCV can be seen when we think, analogously to standard regression, of the diagonal elements of the matrix $S(\lambda)$, the $S_{ii}(\lambda)$'s, as the *leverage values.* We know that at the points with high leverage the predicted values are very sensitive to the observations made at these points. Using (3.13) we can rewrite GCV function in the form

$$GCV_\lambda = n^{-1}\sum_{i=1}^{n}\left\{\left(\frac{1-S_{ii}(\lambda)}{(1-n^{-1}\mathrm{tr}\,S(\lambda))^2}\right)^2 (Y_i - \hat{m}_\lambda^{(-i)}(t_i))^2\right\}, \tag{3.16}$$

We observe from (3.16) that GCV when compared to the CV criterion (Definition 3.2), downweighs the deleted residuals at points with large leverage values.

Another way of looking at GCV is possible when we invoke classical least squares regression, and its idea of the number of model parameters. For a parametric model, the least squares vector of fitted values is $\hat{\boldsymbol{Y}} = H\boldsymbol{Y}$ with H the so-called hat matrix. The number of parameters is the sum of the diagonal elements of the hat matrix, i.e. for a model with k parameters, we have $k = \mathrm{tr}(H)$. We can introduce a similar concept in nonparametric regression. The degrees of freedom of the model can be defined by analogy to standard regression as the $\mathrm{tr}(S(\lambda))$ (see Hastie and Tibshirani (1990) or Green and Silverman (1994)). Therefore, we can define the *equivalent* *degrees of freedom for noise* as

$$EDF = \mathrm{tr}\,(I - S(\lambda)).$$

The *EDF* goes from 0 when $\lambda = 0$ (interpolation to the data), to $(n - k)$ when $\lambda \to +\infty$ (k is the dimension of the default model). The following expression for GCV_λ can be easily derived

$$GCV_\lambda = n \times \frac{\sum_{i=1}^{n}(Y_i - \hat{m}_\lambda(t_i))^2}{\mathrm{EDF}^2}.$$

In our simulation studies we use the *GCV* function to choose the λ value. We selected this method because it has more favorable properties than *CV* (see Wahba (1990)). However, the preference for one or the other criterion is not crucial in our work, since we deal with hypothesis testing and not the true regression function estimation. We use two `S-PLUS` modules `Lspline` and `Pspline` written by J.O. Ramsay. `Lspline` enables us to use the penalties L that are linear combinations of the derivatives of $m(\cdot)$, e.g. in our study we use $Lm = m'' + \gamma m'$. `Pspline` allows us to fit a smoothing spline associated with the penalty $Lm = m^{(k)}$ for $k \in \mathbb{N}$.

Chapter 4

Testing for Bumps

As mentioned in section 2.3, we have to look separately at the density estimation and the regression estimation problems. The differences between them carry through to hypotheses testing. In the next section I describe the methods used in bump-hunting in the density context. Review of bump testing in regression follows in section 4.2, and our method is presented in section 4.3.

4.1 Density Function

Consider a data set $\{X_1, ..., X_n\}$ coming from a population with a density function $f(\cdot)$. Firstly, we want to estimate the density $f(\cdot)$ using one of the available methods. Secondly, we wish to make inferences about the features of the true density. We are interested in the existence of modes and associated bumps. There have been different approaches proposed to the bump-hunting problem, starting from the early parametric approach in the 1960's through nonparametric methods developed mainly in the 1980's and 1990's.

Cox (1966) has suggested constructing a histogram from the data choosing a bin width broad enough to smooth over spurious bumps, but not so broad as to obscure the effects under study. Let f_i denote the number of observations falling in cell i, i.e. $f_i = \sum_{j=1}^{n} \mathrm{I}(X_j \in \mathcal{C}_i)$. Therefore $f_i \sim$ Binomial $(n, p = \int_{\mathcal{C}_i} f(x)\,dx)$ and

$$\begin{aligned} f_i \,|\, f_{i-1} + f_i + f_{i+1} &\sim \text{Binomial } (n = f_{i-1} + f_i + f_{i+1}, p) \\ \text{with } p &= \mathrm{P}(X \in \mathcal{C}_i | X \in \mathcal{C}_{i-1} \cup \mathcal{C}_i \cup \mathcal{C}_{i+1}) \\ &= \frac{\mathrm{E}(f_i)}{\mathrm{E}(f_{i-1}) + \mathrm{E}(f_i) + \mathrm{E}(f_{i+1})}. \end{aligned}$$

We can define a test statistic

$$T_i = \frac{(f_{i+1} - 2f_i + f_{i-1})}{\sqrt{2(f_{i-1} + f_i + f_{i+1})}}. \tag{4.1}$$

Using the normal approximation to the binomial distribution, it can be shown that T_i has an approximately normal distribution. Values of T_i significantly different from zero will indicate either convexity ($T_i < 0$) or concavity ($T_i > 0$) of the true density function around cell i. When we examine the sequence of T_i's, we can get some evidence on the number of bumps in f.

There were two approaches proposed to test the hypothesis

$$H_0 : \# \text{ bumps} = 1 \text{ versus } H_a : \# \text{ bumps} = 2$$

using normal models. Wolfe (1970) used the likelihood ratio to test the normal null hypothesis against the alternative of a two-component normal mixture. The proposed test, however, is very sensitive to the normality assumption, and it may decide with a high probability that a long-tailed unimodal distribution has two modes. Engelman and Hartigan (1969) proposed dividing the sample into the two subsets which maximize the likelihood ratio that the two subsets are sampled from normal distributions with different means, against the null hypothesis that the means are equal.

In the 1980's nonparametric methods in bump-hunting started gaining popularity due to greater availability of computing resources.

Good and Gaskins (1980) compared density estimation and bump hunting. Their notion was that bump-hunting differed from density estimation, since the former involved significance testing, and the latter estimation. They used the maximum penalized-likelihood (MPL) method for estimating the probability density and for the bump-hunting. Let $\hat{f}$ be the estimated density using MPL. They then used *iterative surgery* to construct another estimate f^* in which the bump is absent. The difference between the penalized log likelihood of $\hat{f}$ and f^* gave log-odds ratio in favour of the bump.

Silverman (1981) proposes a test statistic for hypotheses concerning the number of modes in the density. His statistic is calculated by application of a kernel density estimator defined at a point x as

$$\hat{f}(x;h) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{4.2}$$

where $K(\cdot)$ is known as a kernel function. Using this definition, we can say the window width h controls the amount of data used to calculate the estimate of $f(x)$. Now, if the data are strongly bimodal, we would need a large value of h to obtain a unimodal estimate. Using this idea, we can test the null hypothesis that the density f has at most k modes versus the alternative that f has more than k modes.

Formally, we define the *k-critical window width* h_{crit} by

$$h_{crit} = \inf\{h : \hat{f}(\cdot, h) \text{ has at most } k \text{ modes}\} \tag{4.3}$$

Large values of h_{crit} will lead to rejection of the null hypothesis that $f(\cdot)$ has at most k modes. The existence of h_{crit} is guaranteed, if the kernel used in estimation is normal. Silverman proved

Theorem 4.1 *Given any fixed* $\{X_1, ..., X_n\}$*, define* $\hat{f}(x;h)$ *as above, using the normal kernel* $K(x) = (\sqrt{2\pi})^{-1}\exp(-x^2/2)$*. For each integer* $m \geq 0$*, the number of local* *maxima in $\partial^m \hat{f}(x;h)/\partial x^m$ as a function of x is a right continuous decreasing function of h.*

The following corollary can be used to simplify computation of h_{crit}.

Corollary 4.1 *Using the definition of h_{crit} as above and $\hat{f}(\cdot,h)$ calculated using the same data set, $\hat{f}(\cdot,h)$ has more than k modes if and only if $h < h_{crit}$.*

Thus we can use a simple binary search procedure to find h_{crit}. For any value h we can tell if $h < h_{crit}$ by counting the number of modes in $\hat{f}(\cdot;h)$. However some caution should be exercised, since the cited result does not apply to other kernels or to other smoothing techniques. Babaud, Witkin, Baudei, and Duda (1986) examined different kernels used in kernel density estimation. They proved that in the class of symmetric, infinitely differentiable kernels with tails vanishing faster than an inverse of any polynomial, the normal kernel is unique in guaranteeing the conclusion of Theorem 4.1.

Hartigan and Hartigan (1985) have proposed a technique called the *dip* test. They defined a statistic D_H by

$$D_H = \min_{F \in \mathcal{U}} \sup_x |F_n(x) - F(x)| \tag{4.4}$$

where $F(\cdot)$ is a cumulative distribution function, $F_n(\cdot)$ is the empirical cumulative distribution function of the data and $\mathcal{U}$ is the class of all distribution functions with unimodal densities. They define a unimodal distribution function $F(\cdot)$ with mode at $x = m$ as $F(\cdot)$ being convex in $(-\infty, m]$ and concave in $[m, \infty)$. Additionally, for comparison purposes, the *depth* test and the *likelihood ratio* test are specified. The *depth* test identifies three intervals of equal length such that the middle interval has low empirical probability relative to both the outside intervals. The *likelihood ratio* test statistic is defined as

$$\sup_{f \in \mathcal{U}_{1C}} \sum_{i=1}^{n} \log f(X_i) / \sup_{f \in \mathcal{U}_{2C}} \sum_{i=1}^{n} \log f(X_i) \tag{4.5}$$

where $\mathcal{U}_{1C}$ is the class of unimodal densities and $\mathcal{U}_{2C}$ is the class of bimodal densities, constrained by $\max f(\cdot) < C$. The *dip* test performed slightly better than the *depth* test, and much better than the *likelihood ratio* test.

Hartigan (1987) generalized his *dip* test to higher dimensions. He proposed a SPAN statistic, which used an analogue of the empirical distribution function on the minimum spanning tree. However, the SPAN statistic is computationally and conceptually complex.

Hartigan and Mohanty (1992) offered yet another test called RUNT, which is based on single linkage clusters of $\{X_1, ..., X_n\}$ (the maximal connected subgraphs formed by a clustering method based on a distance matrix model). The test is defined as follows. We consider all the single linkage clusters. They form a hierarchical tree with each nonsingleton cluster dividing into a few subclusters. Each cluster C has a smallest subcluster (or "runt"). Let $n(C)$ be the number of points in this subcluster. We define the RUNT statistic as the $max_C n(C)$. Hartigan (1981) justified the RUNT test statistic by the asymptotics of the single linkage clusters. If there are at least two modes in the population density, then asymptotically, just one of the single linkage clusters will split into two clusters of points corresponding to the different modes, with the smaller of the clusters being the runt. A large number of points in the smaller cluster would indicate bimodality. On the other hand, if there is a single mode, we expect each cluster to divide into two clusters, the smaller of which contains very few points. Therefore, a large value of the RUNT statistic should indicate multimodality.

Müller and Sawitzki (1991) proposed a method for analyzing the modality of a possibly multivariate distribution based on the excess mass functional. This functional measures excessive empirical mass defined by a comparison of the empirical distribution to multiples of a uniform distribution. In one dimension the resulting test is equivalent to Hartigan's *dip* test.

Minnotte (1993, 1997) introduced a whole new class of tests. He used kernel density estimation with varying bandwidths h. Due to the fact that data-driven choice of the bandwidth h is difficult, he proposed a new graphical method, a mode tree, to assist in bandwidth selection. A mode tree is a graphical tool relating the locations of modes in density estimates with the bandwidths of those estimates. For each value of the bandwidth, a density estimate and location of the mode are calculated. The mode locations are then plotted against the logarithm of the bandwidth. The usual scale is not appropriate here, since large changes in a bandwidth when the bandwidth is large have less of an effect on the density estimate than smaller changes in h when h is small. Additional functionality can be added to a simple mode tree. Firstly, we can draw lines to show the connections between the new modes and the old ones from which they split. Minotte suggested a second enhancement which helps us visualize the presence and the importance of the bumps by drawing the widths of mode traces. It may be more appropriate to consider the mode tree idea as an exploratory device useful in indicating modes being worth further study. It is therefore unclear how we can apply this approach to testing, since we make multiple comparisons at different bandwidths.

Chaudhuri and Marron (1997) offered us a method based on scale space ideas developed first in computer vision literature. Here "scale" means "level of resolution" or in a kernel density context a bandwidth - h. Their proposal, the SiZer, assesses the significant zero crossings of derivatives of estimates of the density. The density estimation is carried out simultaneously at different values of the bandwidth h. Marron noted that only "blurred signal" is available from a finite sample in the presence of noise. Therefore, the focus of his analysis was on the convolution $f * K_h$ representing the "blurring" of the signal. At each level of h, the confidence limit for the derivative of a density $f_h'(x)$ was obtained, where $f_h'(\cdot)$ denoted the derivative of $f * K_h$. The behaviour of the estimator $\hat{f}_h'(\cdot)$ at each x and h is presented via the SiZer colour map. Chaudhuri and Marron's method can be applied in the regression framework by looking at the first derivative of the regression function. The advantage of their technique is not only in detecting the number of significant bumps, but also in showing their locations. Chaudhuri and Marron mentioned, however, that their approach had less power than methods devised specifically for mode testing, since mode tests are not impeded by trying to be simultaneous at each x and h. As with the mode tree idea, SiZer should be used mostly as an exploratory tool, with follow up analysis of important features of interest.

4.2 Regression Function

It is very surprising that tests for bumps are so much more prevalent in the density estimation context than in the regression estimation. Procedures for bump-hunting in the regression context are few, and only some of them can be readily implemented to the problem on hand. An early reference is Schlee (1982) who proposed a test based on the greatest discrepancy of an estimate of the derivative of the regression function from zero. He used a modified version of kernel estimators, and based his tests on the asymptotic distribution of the maximal deviation. Schlee's tests can be used to test for constancy, monotonicity, or convexity of the underlying regression function. The drawback of the methodology is in the lack of discussion of practical implementation.

Heckman (1992) suggested that one might recognize an occurrence of a bump in the regression function if an estimate of the regression function rises and then falls over a range of independent variable. Letting $\hat{Y}_i,\ i = 1, \dots, N$ denote these estimates, she defined a "bump" of size k at the j-th estimate as the event

$$\left\{\hat{Y}_{j-k} < \hat{Y}_{j-k+1} < \cdots < \hat{Y}_j > \hat{Y}_{j+1} > \cdots > \hat{Y}_{j+k}\right\} \tag{4.6}$$

She studied the asymptotic distribution of S_N^k, the number of bumps of size k, and K_N, the maximum bump size observed. S_N^k was proven to be a consistent estimator of the number of local maxima of a regression function provided $k \to \infty$ and $k/N \to 0$. However in practical applications, S_N^k was very sensitive to the choice of k and N in determining the correct number of bumps.

Bowman et.al. (1998) constructed a test for the monotonicity of the regression function that is analogous to "Silverman's test" (1981) of multimodality of a probability density function. Further discussion of "Bowman's test" is given in section 4.3.

4.3 New Test: $\mathcal{C}$ri$\mathcal{SP}$

As mentioned in section 4.2, results on bump-hunting in regression are few, when compared to the results in density estimation. These few results mainly concern testing for monotonicity. We would like to explore here bump-hunting in regression using spline methods of estimation. In the first place in section 4.3.1, we present the discussion on estimating the number of bumps in the regression function. Then in section 4.3.2, we propose a testing method inspired by Silverman's idea of defining the h_{crit} statistic. In section 4.3.3 we give a description of a testing procedure for the derivatives of the regression function.

4.3.1 Counting Bumps

Following on the idea (4.6) of Heckman (1992) , we can decide that a bump in $m^{(k)}(\cdot)$ (k=0, 1, 2, ...) has occurred if the estimates of $m^{(k)}(\cdot)$ rise and then fall over a range of the independent variable. If we let $\hat{Y}_i^{(k)} = \hat{m}^{(k)}(t_i)$, $i = 1, ..., n$, be our estimates, then we say $\hat{m}^{(k)}(\cdot)$ has one bump in (t_{j-l}, t_{j+l}), if

$$\left\{\hat{Y}_{j-l}^{(k)} < \hat{Y}_{j-l+1}^{(k)} < \cdots < \hat{Y}_j^{(k)} > \hat{Y}_{j+1}^{(k)} > \cdots > \hat{Y}_{j+l}^{(k)}\right\} \tag{4.7}$$

where k=0, 1, 2, 3,.... We call it an l-bump to show how many points we take into account when estimating the number of bumps.

It is very appealing to use (4.7) in the testing procedure for the number of bumps. However, the restriction of the estimates to be strictly rising and then falling over a range of an independent variable does not seem to work well in practice. We do not have any guarantee that in the vicinity of a mode the estimates of the regression function will rise on one side of a mode and fall on the other side at the grid points. In addition, the monotonicity property proven by Silverman for modes (Theorem 4.1) does not hold for regression estimates. Empirical studies show that often the estimate of the number of bumps increases and then decreases as the the smoothing parameter increases (see section 5.1). Another problem arises in the definition of λ_{crit}, which is needed to calculate the estimate of the regression function $\hat{m}_{\lambda_{crit}}$ for our bootstrap procedure (step 1). For some data sets, $\hat{m}_\lambda$ may have fewer than k l-bumps for all values of λ. Thus λ_{crit} does not exist.

Furthermore, time of computation required to estimate the number of bumps according to (4.7) is prohibitively large. Therefore in our testing procedure, we utilize

$$\hat{Y}_j^{(k)} > \max\left\{\hat{Y}_{j-l}^{(k)}, \cdots, \hat{Y}_{j-1}^{(k)}, \hat{Y}_{j+1}^{(k)}, \cdots, \hat{Y}_{j+l}^{(k)}\right\} \tag{4.8}$$

as a basis for saying that $\hat{m}^{(k)}(\cdot)$ has one bump in (t_{j-l}, t_{j+l}). We call this an l-max.

In our simulation studies, we found that l-max did not have the same problems as l-bump.

For l=1, l-bump and l-max are of course equivalent. However, for all other values of l, if a bump occurs according to l-max then it will occur according to l-bump, but not necessarily vice versa. If the true underlying regression function is not constant over an interval, the differences in the estimated number of bumps according to (4.7) and (4.8) happen mostly for small values of the smoothing parameter λ. In section 5.1.1, we compare the performance of the $\mathcal{C}$ri$\mathcal{SP}$ using definitions (4.7) and (4.8) on one of the regression functions, and we discuss further the advantages and disadvantages of (4.8).

4.3.2 $\mathcal{C}$ri$\mathcal{SP}$ in Regression

Formally, the following tests are performed for a fixed non-negative integer k:

$$H_0 : \# \text{ of bumps in } m(\cdot) \ \leq k$$

vs.

$$H_1 : \# \text{ of bumps in } m(\cdot) \ > k$$

Using the test statistic: $\lambda_{crit} = \inf\{\lambda : \hat{m}_\lambda \text{ has } k \text{ bumps}\}$, we reject H_0 if λ_{crit} is too big. The testing procedure would be reasonable if the number of bumps were monotone in λ, i.e. the greater the λ the fewer bumps of $\hat{m}_\lambda(\cdot)$. As λ approaches infinity the limiting value of the number of bumps is the number of bumps in the default model $m(\cdot)$.

The p-value of the test is then

$$p = P_{H_0}\{\lambda_{crit} > \lambda_{crit}^{obs}\} \tag{4.9}$$

where λ_{crit}^{obs}, viewed as non-random in the probability, is the observed value of λ_{crit}. In the definition of p, we actually calculate the probability as a supremum under the null hypothesis, if H_0 is composite. Unfortunately, the distribution of λ_{crit} is unknown, and the estimation of p is extremely computing-intensive. Therefore, we use the following shortcut. Define $p^\star$ by

$$p^\star = P_{H_0}\{\hat{m}_{\lambda_{crit}^{obs}} \text{has more than } k \text{ bumps}\} \tag{4.10}$$

If the number of bumps in $\hat{m}_\lambda$ is monotone in λ then $p = p^\star$. The set in (4.10) is fairly simple, since it depends on $\hat{m}_\lambda$ for a fixed value of $\lambda = \lambda_{crit}^{obs}$.

The monotonicity of the number of modes as a function of the smoothing parameter or the bandwidth was established in a few cases. In the case of kernel estimators, both Gasser-Müller and Nadaraya-Watson estimators have the monotonicity property, if the kernel is normal. However, this feature does not hold for the local linear regression estimator or for the spline regression estimator. Fortunately, simulation studies indicate that, when defining a bump as in (4.8), data sets where the number of bumps in $\hat{m}_\lambda$ is non-monotone are very unusual, so we use $p^\star$ as our estimator of the p-value.

Theoretical derivation of the exact expression for p or $p^\star$ would be hard or even impossible, therefore we will use bootstrapping to estimate $p^\star$.

In order to keep the description of the procedure described below comprehensible, let us fix a smoothing method we use. We abbreviate the generic smoothing method utilized by `SM`.

The **bootstrap procedure** involves the following steps:

1. Calculate λ_{crit}^{obs} and $\hat{m}_{\lambda_{crit}^{obs}}$ using `SM`.

2. Find $\hat{m}_{unr}$, an unrestricted estimate of $m(\cdot)$ with smoothing parameter for `SM` chosen by generalized cross-validation.

3. Find residuals $\varepsilon_i^* = Y_i - \hat{m}_{unr}(t_i)$, $i = 1, ..., n$.

4. Create a bootstrap sample $Y_i^* = \hat{m}_{\lambda_{crit}^{obs}}(t_i) + \varepsilon_i^*$, $i = 1, ..., n$ where the ε_i^*'s are drawn at random with replacement from the ε_j^*'s, $j = 1, 2, \ldots, n$.

5. Find the new estimate $\hat{m}^*_{\lambda_{crit}^{obs}}$ for the bootstrap sample $\{Y_i^*\}_1^n$ using SM.

6. Find the number of bumps in $\hat{m}^*_{\lambda_{crit}^{obs}}$.

7. Repeat 4-6 a large number of times, and calculate the estimate of the p-value by counting the number of estimates $\hat{m}^*_{\lambda_{crit}^{obs}}$ having more than k bumps, i.e.

$$\hat{p} = \frac{\#\text{ of estimates } \hat{m}^*_{\lambda_{crit}^{obs}} \text{ with more than } k \text{ bumps}}{\#\text{ of bootstrap samples}}.$$

The choice of SM depends on the default model we assume. For instance, if the default model is a line, we use penalized least squares with $Lm = m''$, and SM is then cubic Pspline. The smoothing method to calculate the residuals in step 3 need not be the same as the smoothing method in steps 1 and 5. However, steps 1 and 5 should use the same smoothing method. Therefore, we can experiment by using either spline smoothing or the local linear kernel technique in steps 1 and 5. When we use the kernel smoothing the definition of the critical smoothing parameter is analogous to λ_{crit}. We define $h_{crit} = \inf\{h : \hat{m}_h \text{ has } k \text{ bumps}\}$ and we use the same bootstrap steps 4-7 with $\hat{m}_{\lambda_{crit}^{obs}}$ replaced by $\hat{m}_{h_{crit}^{obs}}$.

In step 3 we calculate the residuals using the estimates $\hat{Y}_i^{unr} = \hat{m}_{unr}(t_i)$. Another possibility would be to use the restricted estimates $\hat{Y}_i^{crit} = \hat{m}_{\lambda_{crit}^{obs}}(t_i)$ in the residual computation. The two methods of generating the residuals may be equivalent when the null hypothesis is true, although this is not clear. The first method has an advantage of being appropriate under both null and alternative hypothesis while the second method is proper only when the null hypothesis is true. The results of our simulation study (see Chapter 5) indicate that using the unrestricted estimates to obtain the residuals gives the right size of the test.

Determining the "correct" residuals for hypothesis testing using a bootstrap technique has not yet been done satisfactorily. There exist complex ways to adjust the bootstrap procedure in the simpler testing problems. However, no consensus has been reached on resolving the residual problem in more complicated testing procedures.

Bowman et.al. (1998) used a similar bootstrap procedure in their test of the null hypothesis that $m(\cdot)$ is monotone. They utilized the same residual generating method as we did in $\mathcal{C}$ri$\mathcal{SP}$. The problem of the residual choice was mentioned in their work, but no satisfactory way to settle the issue was provided. Bowman et.al. used local linear kernel regression estimator in all steps of their bootstrap procedure. When the bandwidth $h \to +\infty$ this estimator tends to a least squares line fit, which is of course monotone. In a new test, $\mathcal{C}$ri$\mathcal{SP}$, a default model plays the role of the line. The default model may not be monotone. Therefore, we can test for any number of bumps being greater than or equal to the number of bumps in the default model.

4.3.3 $\mathcal{C}$ri$\mathcal{SP}$ for Derivatives

So far we have concentrated on the bump-hunting problem in the regression function $m(\cdot)$. However we can extend our procedure to the derivatives of $m(\cdot)$. For example, we can test for any fixed integer-valued k

$$H_0 : \# \text{ of bumps in } m'(\cdot) \ \leq k$$

vs.

$$H_1 : \# \text{ of bumps in } m'(\cdot) \ > k.$$

We use the same test as for the regression function with $m'(\cdot)$ replacing $m(\cdot)$. Again, we have to use the bootstrap procedure to estimate the p-value of the test. Now, instead of calculating the number of bumps in $m(\cdot)$, we compute the number of bumps in $m'(\cdot)$.

Let us denote by SM, a generic smoothing method used in the procedure described below. The following steps are used in the **bootstrap procedure** testing for bumps in the derivatives:

1. Calculate λ_{crit}^{obs} for $\hat{m}'$ and $\hat{m}_{\lambda_{crit}^{obs}}$ using SM.

2. Find $\hat{m}_{unr}$, an unrestricted estimate of $m(\cdot)$ with smoothing parameter for SM chosen by either generalized cross-validation or the plug-in method.

3. Find residuals $\varepsilon_i^* = Y_i - \hat{m}_{unr}(t_i)$, $i = 1, ..., n$.

4. Create a bootstrap sample $Y_i^* = \hat{m}_{\lambda_{crit}^{obs}}(t_i) + \varepsilon_i^*$, $i = 1, ..., n$ where the ε_i^*'s are drawn at random with replacement from the ε_j^*'s, $j = 1, 2, \ldots, n$

5. Find the new estimate $\hat{m}'^*_{\lambda_{crit}^{obs}}$ for the bootstrap sample $\{Y_i^*\}_1^n$ using SM.

6. Find the number of bumps in $\hat{m}'^*_{\lambda_{crit}^{obs}}$.

7. Repeat 4-6 a large number of times, and calculate the estimate of the p-value by counting the number of estimates $\hat{m}'^*_{\lambda_{crit}^{obs}}$ having more than k bumps, i.e.

$$\hat{p} = \frac{\#\text{ of estimates } \hat{m}'^*_{\lambda_{crit}^{obs}} \text{ with more than } k \text{ bumps}}{\#\text{ of bootstrap samples}}.$$

Again, we use the same smoothing technique throughout the simulations, but we can experiment with a different smoothing method in steps 1, 2 and 6. In our simulation study, we employ Lspline with the penalty $Lm = m'' + \gamma m'$ with the default model $m_{def}(t) = \beta_0 + \beta_1 \exp\{-\gamma t\}$. In the case of the Berkeley growth data, local linear kernel smoothing using the Splus function locpoly, is used as SM in all steps. For comparison purposes, the data are re-analyzed, using `locpoly` in step 2, and `Lspline` in steps 1 and 5.

In Chapter 5 we present the results of the simulations for the regression functions and their derivatives as well as the analysis of the speed of growth in the Berkeley growth data.

Chapter 5

Results

In order to examine the proposed testing procedure for the number of bumps, we conducted a simulation study. We used several functions to assess the performance of a new test $\mathcal{C}\text{ri}\mathcal{SP}$. In section 5.1, we present the results of the simulations for two classes of regression functions. In section 5.2, we show the performance of the test in the context of derivatives of regression functions, and in section 5.3, the $\mathcal{C}\text{ri}\mathcal{SP}$ is applied to the Berkeley growth data.

5.1 Regression Function Simulations

We generated the data

$$Y_i = m(t_i) + \sigma\varepsilon_i \tag{5.1}$$

with $t_i = i/101$ and $\varepsilon_i \sim N(0, 1)$ for $i = 1, 2, \ldots, 101$. Two levels of noise were used, $\sigma = 0.05$ and $\sigma = 0.10$ for one of the functions, and one level of noise $\sigma = 0.05$ for the other. Functions $m(\cdot)$ used in the simulation study are described below. For each model, we produced 500 data sets, and to each applied the bootstrap procedure (see pages 36 and 39) in sections 5.1.1 - 5.1.3, with bootstrap sample size equal to 500.

We have chosen two classes of monotone functions to work with, one being the linear function, and the other the exponential function. To each of them, we added one or more bumps. In order to preserve the continuity of the regression functions, we have chosen a function producing a bump decreasing at the exponential rate to zero. We define a function generating a bump by

$$B(t|c) = \exp\left\{-\frac{(t-c)^2}{2\,(0.1)^2}\right\}. \tag{5.2}$$

As we can see $B(t|c)$ is the kernel of the normal density function. The parameter c controls the place the bump is inserted in the function. The choice of (0.1) in the denominator of the exponent enables us to separate bumps easily, in case we want to add more than one bump to the studied function.

Definition of the functions

We considered two classes of functions

$$m_1(t|a) \quad = 1 + t + a * B(t|0.5) \tag{5.3}$$

and

$$m_2(t|a,b) \quad = 1 + \exp(-4t) + a * B(t|0.25) + b * B(t|0.75). \tag{5.4}$$

Introduction of the variable a in $m_1(\cdot|a)$ allows us to introduce a bump in the otherwise a monotone function. Plots of the functions $m_1(\cdot|a)$ used are shown in Figure 5.1. The value of $a = 0$ gives us a straight line and $a = 0.15$ produces a curve which is on a border-line of having a bump. Cases $a = 0.25$ and $a = 0.45$ produce curves with small and large bump respectively. Bowman et.al. (1998) used functions similar to (5.3) in their simulation studies of a test of monotonicity of a regression function. In the first two cases of the parameter $a = 0, 0.15$, the function $m_1(\cdot)$ is monotone, and in the other two cases, $a = 0.25, 0.45$, non-monotone.

Using a combination of variables a and b in the function $m_2(t|a,b)$, we can produce a function with 0, 1, or 2 bumps. When $a = 0$ or $a = 0.32$ the resulting function

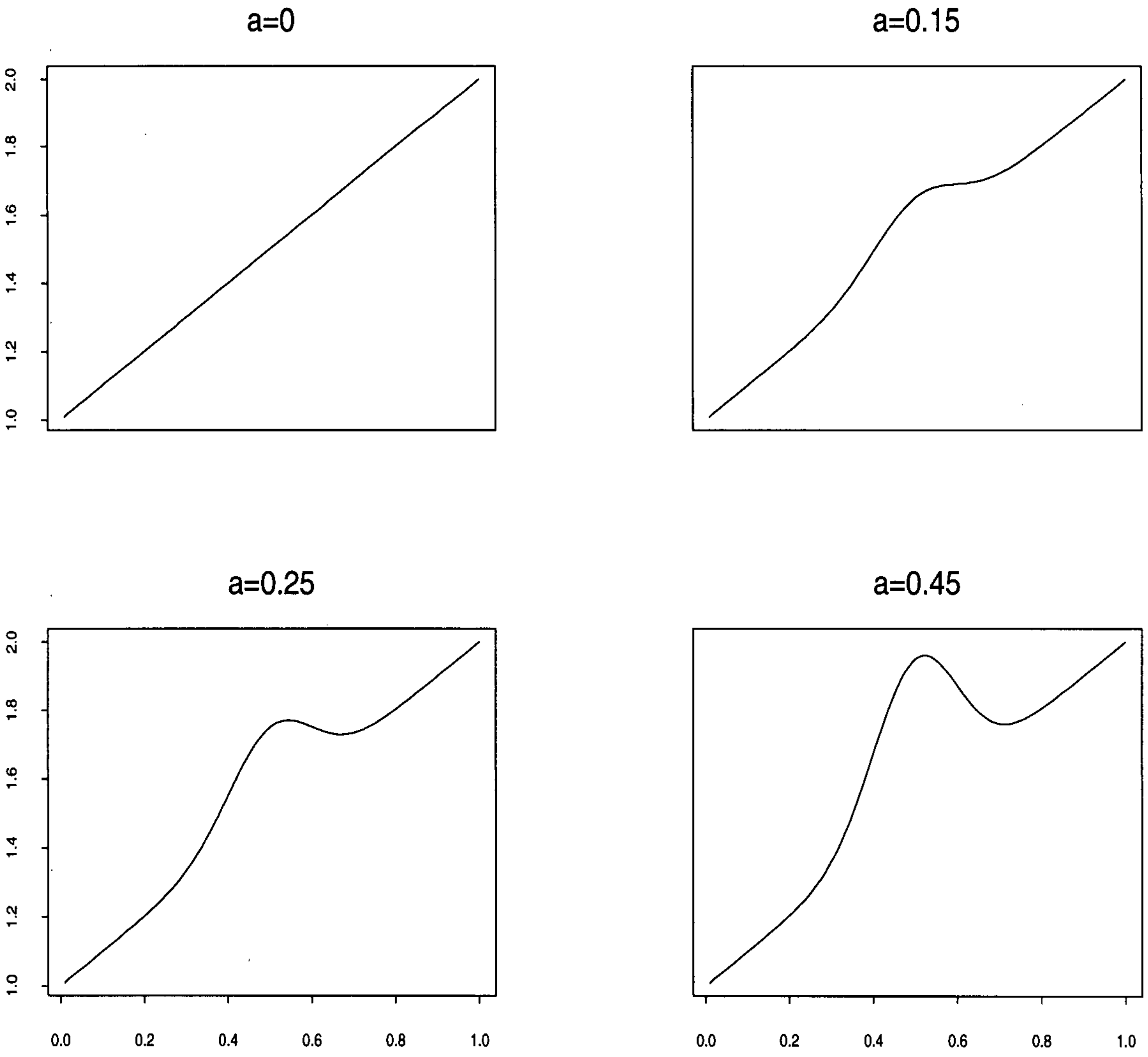


Figure 5.1: Function $m_1(t) = 1 + t + a * B(t|0.5)$.

has no bump at t=0.25. The values $a = 0.48$ and $a = 0.64$ yield a medium and large bump respectively. When $b = 0$, there is no bump in the regression function at t=0.75. However values of $b = 0.10, 0.15, 0.20$ give rise to a small, medium and large bump respectively. An interaction between two bumps introduced in the equation by $B(t|0.25)$ and $B(t|0.75)$ is negligible, since the exponential functions are the kernels of a normal density, and the distance between their maxima is equal to $0.5 = 5\times$ the standard deviation, if thought of as a normal density function. We can see the functions with all the combinations of a and b in Figure 5.2. In our simulation study, we used 9 (indicated by the asterisks "**") out of 16 combinations to assess the performance of $\mathcal{C}$ri$\mathcal{SP}$.

Counting bumps in estimates

We would like to look at a few aspects of our test in the simulation study. First, we want to investigate the behaviour of $\mathcal{C}$ri$\mathcal{SP}$ when we utilize two different methods for detection of a bump. Let $\hat{Y}_i = \hat{m}(t_i)$, i.e. we take the estimates at the data points. As pointed out in chapter 3, we can employ either an l-bump

$$\left\{\hat{Y}_{j-l} < \hat{Y}_{j-l+1} < \cdots < \hat{Y}_j > \hat{Y}_{j+1} > \cdots > \hat{Y}_{j+l}\right\} \tag{5.5}$$

OR an l-max

$$\hat{Y}_j > \max\left\{\hat{Y}_{j-l}, \cdots \hat{Y}_{j-1}, \hat{Y}_{j+1}, \cdots, \hat{Y}_{j+l}\right\} \tag{5.6}$$

with $l = 1, 2, 3, \ldots$, to define a bump at $\hat{Y}_j$ in our testing procedure.

We use the function $m_2(\cdot \mid a = 0.48, b = 0.15)$ to investigate in detail the behaviour of the $\mathcal{C}$ri$\mathcal{SP}$. We employ the Lspline smoothing technique in all the steps of the bootstrap procedure, using the penalty $L = m'' + \gamma m'$. The values of $l = 2, 3, 4$ and 5 are used in l-max as well as $l = 2$ in l-bump for comparison purposes. We present detailed discussion and results in section 5.1.1. Based on that section, for all the

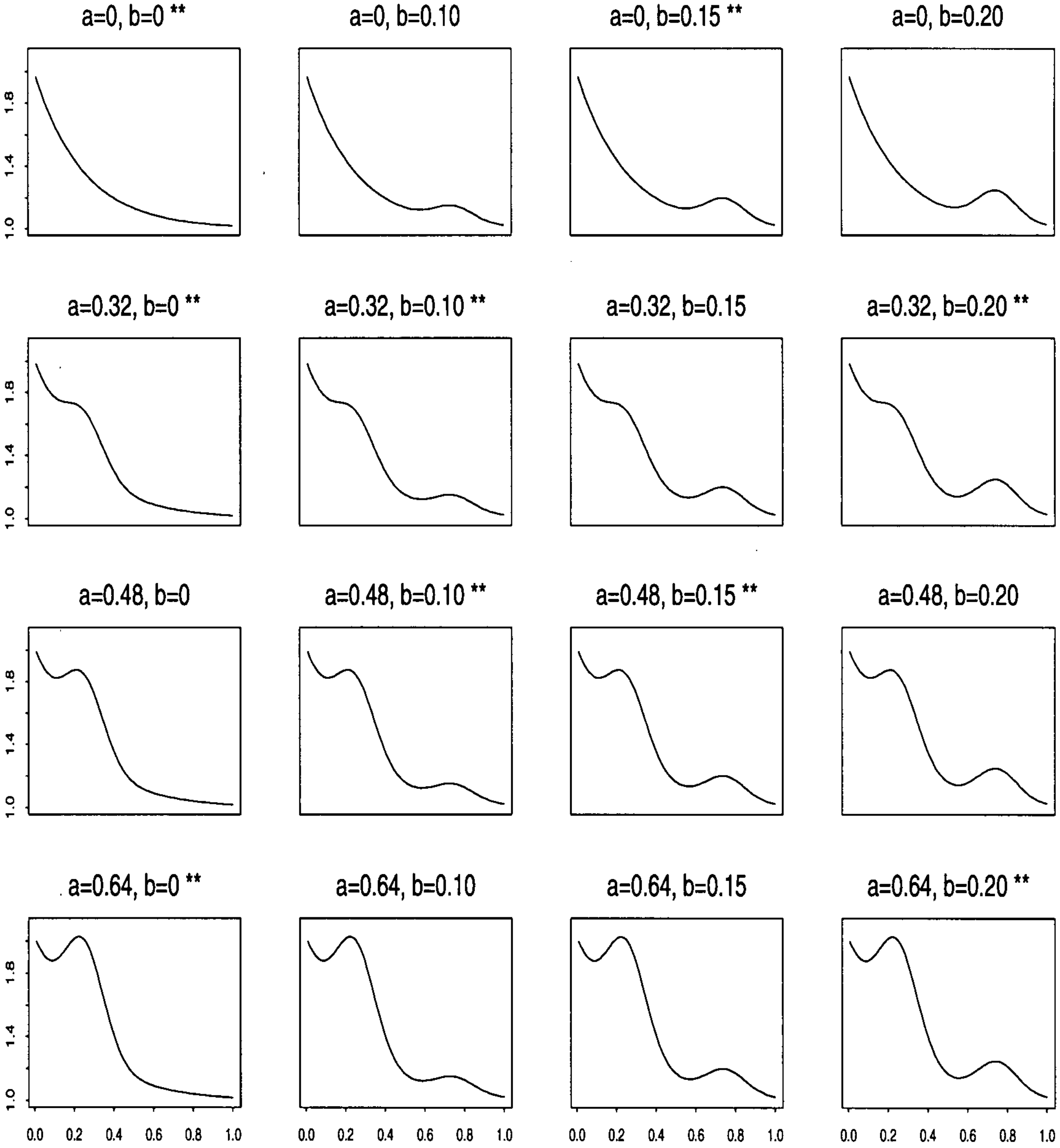


Figure 5.2: Function $m_2(t|a,b) = 1 + \exp(-4t) + a * B(t|0.25) + b * B(t|0.75)$.

other functions we utilized 3-max as a basis for counting bumps in the $\mathcal{C}\mathrm{ri}\mathcal{SP}$ testing procedure.

5.1.1 Function $m_2(\cdot|a = 0.48, b = 0.15)$

We test the hypothesis

$$\begin{aligned} H_0 &: m_2(\cdot) \text{ has } \leq k \text{ bumps} \\ &\text{vs.} \\ H_a &: m_2(\cdot) \text{ has } > k \text{ bumps} \end{aligned} \tag{5.7}$$

for $k = 0, 1, 2, 3$, where

$$\begin{aligned} m_2(t \,|\, a = 0.48, b = 0.15) &= 1 + \exp\{-4t\} + 0.48 * B(t|c = 0.25) \\ &\quad + 0.15 * B(t|c = 0.75). \end{aligned} \tag{5.8}$$

We use l-bump with $l = 2$, and l-max with $l = 2, 3, 4$ and 5 to count the number of bumps in the estimates $\hat{m}_2(\cdot \,|\, 0.48, 0.15)$. Results of the test (5.7) are displayed in Table 5.1. The proportions of the simulations with p-values falling below 5%, 10% and 25% are displayed. Thus, the table entries give the proportion of times we reject the null hypothesis at levels 5%, 10% and 25%. The standard error of the estimates of the rejection probabilities is not more than 2.25% at all significance levels.

As we can see, there is little difference among definitions, but 5-max seems slightly better due to high power (for k=0, 1) and low rejection when k=2, 3.

Consequently, it would be reasonable to use 5-max in our testing procedure. However, for 2 out of 500 data sets used in the simulations, the estimates $\hat{m}_\lambda$, $\lambda \geq 0$, had at most 2 bumps. Thus $\lambda_{crit}^{(2)} = 0$. Therefore, $\lambda_{crit}^{(3)}$ did not exist and so it was impossible to carry out tests. In 26 out of 500 data sets, $\lambda_{crit}^{(3)} = 0$, since the estimates

Table 5.1: Two Bumps: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_2(t_i|a = 0.48, b = 0.15) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0,1)$, $\sigma = 0.05$, using different definitions of a bump.

$\boldsymbol{a = 0.48, b = 0.15}$		$\boldsymbol{H_0 : \# \text{ of bumps } \leq k}$			
Level	Width	$k = 0$	$k = 1$	$\boldsymbol{k = 2}$	$k = 3$
$\boldsymbol{\alpha = 0.05}$	2-bump	1.000	0.358	0.032	0.016
	2-max	1.000	0.362	0.036	0.016
	3-max	1.000	0.368	0.034	0.020
	4-max	1.000	0.366	0.034	0.022
	5-max	0.998	0.382	0.024	0.008
$\boldsymbol{\alpha = 0.10}$	2-bump	1.000	0.590	0.080	0.058
	2-max	1.000	0.592	0.088	0.056
	3-max	1.000	0.586	0.074	0.060
	4-max	1.000	0.586	0.070	0.042
	5-max	1.000	0.596	0.056	0.026
$\boldsymbol{\alpha = 0.25}$	2-bump	1.000	0.836	0.212	0.174
	2-max	1.000	0.838	0.222	0.184
	3-max	1.000	0.832	0.206	0.172
	4-max	1.000	0.840	0.202	0.150
	5-max	1.000	0.838	0.188	0.092

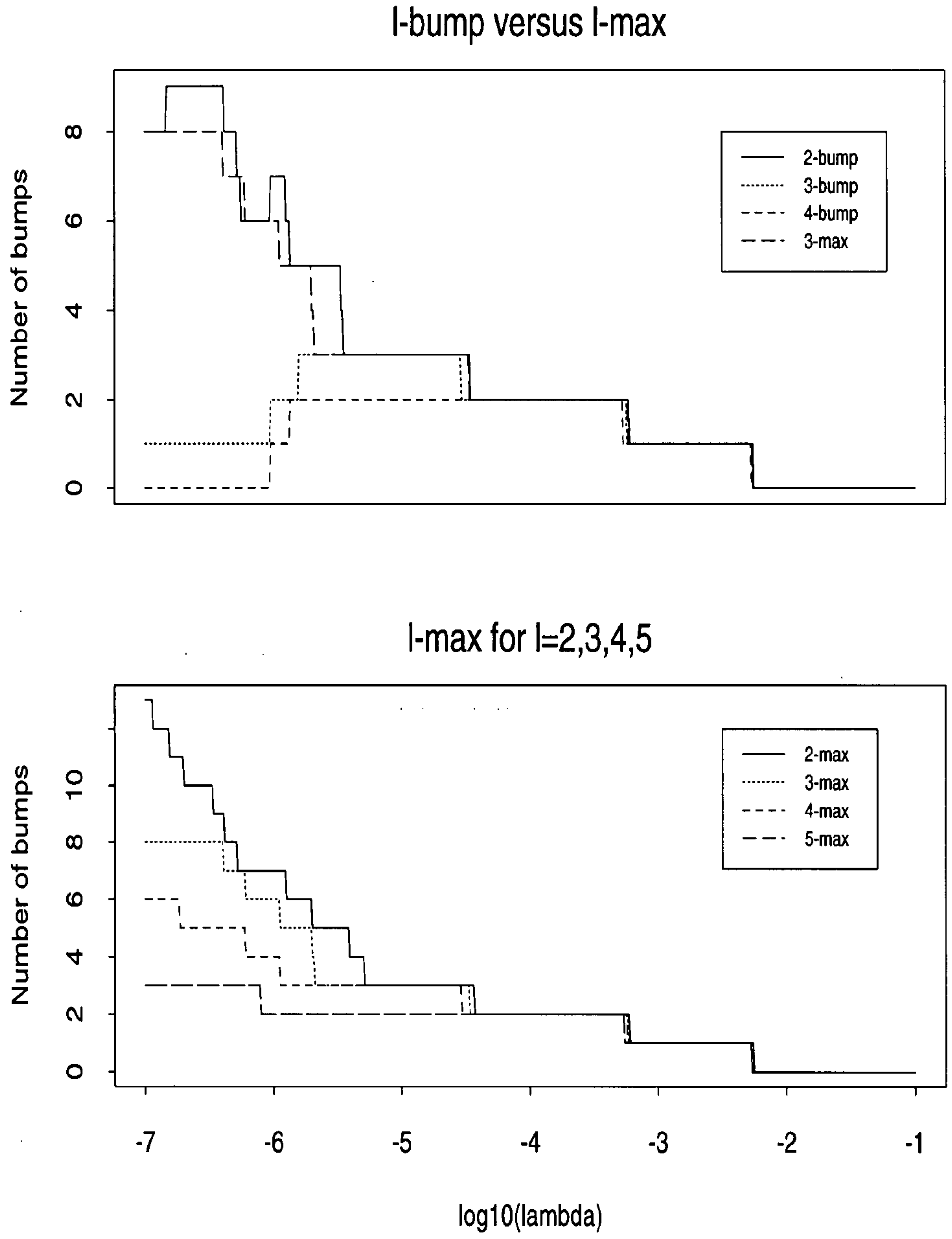


Figure 5.3: Number of bumps as a function of a smoothing parameter λ.

$\hat{m}_\lambda$ had at most 3 bumps for $\lambda \geq 0$. When we used $\mathcal{C}$ri$\mathcal{SP}$ with 4-max, 1 out of 500 datasets had $\lambda_{crit}^{(3)} = 0$. In all other situations, i.e. 2-bump, 2-max and 3-max, we were able to obtain $\lambda_{crit}^{(k)} > 0$ for $k \leq 3$. Due to the similar performance of the $\mathcal{C}$ri$\mathcal{SP}$ test in all the cases, and difficulties encountered with 5-max and 4-max, we had to decide between values of l=2 or 3 in l-max or we could use l-bump in the further studies.

In the simulations, we found some problems with testing while utilizing l-bump when l=3 or 4. Our testing procedure is based on the existence of $\lambda_{crit}^{(k)}$ and on the monotonicity property of the number of bumps as a function of λ. For some data sets, using l-bump with $l \geq 3$ seriously violates the monotonicity property as can be seen in Figure 5.3. When we use 2-bump, the monotonicity is violated mostly at the smaller values of λ.

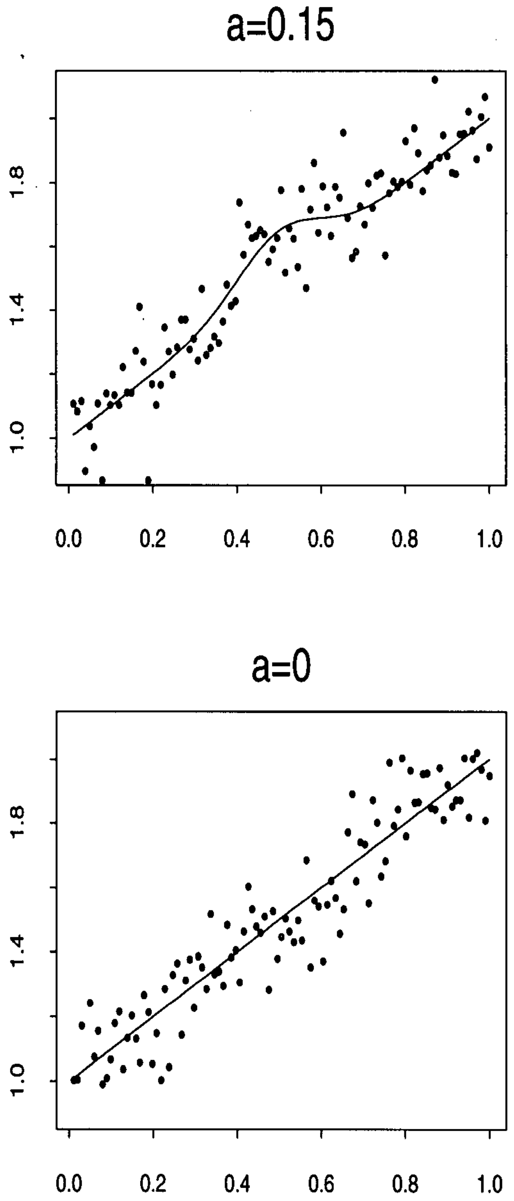
The range of λ on the plot extends from $10^{(-7)}$ to $10^{(-1)}$. Values of $\lambda < 10^{(-7)}$ produce a very wiggly estimate, since they correspond to using more than 60 (=tr $S(\lambda)$) parameters, and we do not want to model a data set with 101 observations by a regression function with so many parameters.

Therefore, for our testing purposes we can choose either 2-max or 3-max. Since we are more interested in detecting "true" bumps, and not the spurious ones, we decided to use 3-max in all other simulation studies.

5.1.2 Zero-One Bump Function

As introduced in section 5.1, we used regression functions $m_1(\cdot|a)$ in the first simulation study with $a = 0, 0.15, 0.25$ and 0.45. Sample data sets are given in Figures 5.4 and 5.5.

In the testing procedure, we use the Pspline method with a penalty $L = m''$ giving a default model $m_{def}(t) = \alpha + \beta t$. For comparison purposes, we substitute


a=0.15

a=0

Figure 5.4: Function $m_1(t|a) = 1 + t + a * B(t|0.5)$ with 0 bumps and simulated data with the level of noise $\sigma = 0.10$.

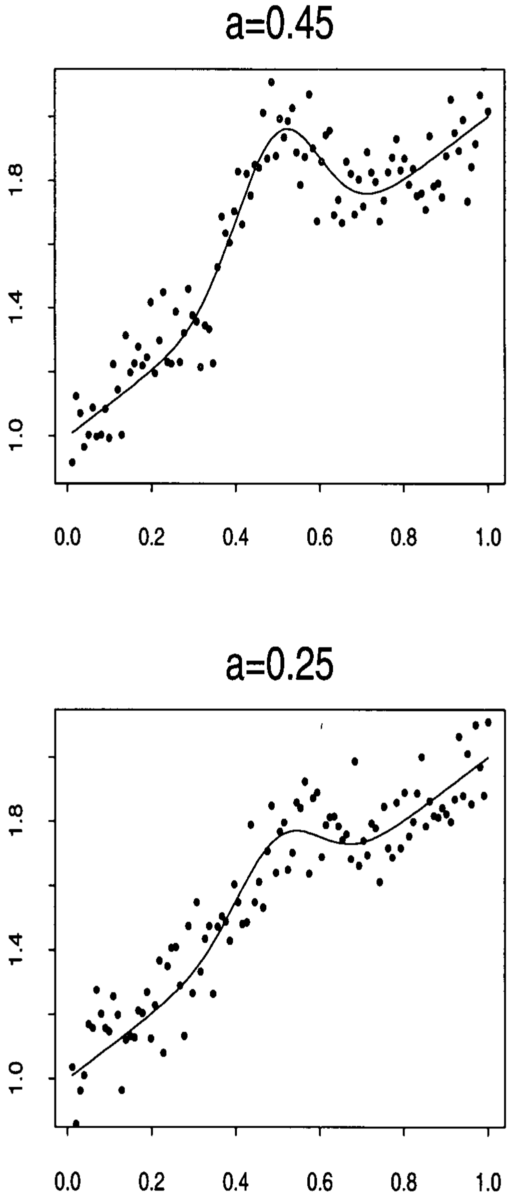


Figure 5.5: Function $m_1(t|a) = 1 + t + a * B(t|0.5)$ with 1 bump and simulated data with the level of noise $\sigma = 0.10$.

Table 5.2: Zero Bumps: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_1(t_i|a = 0) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0,1)$, $\sigma = 0.05, 0.10$.

$\boldsymbol{a = 0}$		$\boldsymbol{H_0 : \# \text{ of bumps } \leq k}$		
Noise	**Level**	Method	$\boldsymbol{k = 0}$	$k = 1$
$\boldsymbol{\sigma = 0.05}$	$\alpha = 0.05$	Pspline	0.002	0.002
		Kernel	0.002	0.006
	$\alpha = 0.10$	Pspline	0.004	0.010
		Kernel	0.006	0.010
	$\alpha = 0.25$	Pspline	0.030	0.024
		Kernel	0.044	0.036
$\boldsymbol{\sigma = 0.10}$	$\alpha = 0.05$	Pspline	0.006	0.008
		Kernel	0.006	0.018
	$\alpha = 0.10$	Pspline	0.010	0.020
		Kernel	0.014	0.038
	$\alpha = 0.25$	Pspline	0.080	0.088
		Kernel	0.112	0.122

Local Linear Kernel Smoothing for Pspline in steps 1 and 5 of the bootstrap procedure (see page 36).

The results are exhibited in Tables 5.2 - 5.5. For $a = 0$ and $a = 0.15$, we carried out the tests with $k = 0$ and $k = 1$, and for $a = 0.25$ and $a = 0.45$ we used $k = 0, 1$ and 2.

Overall the performance of $\mathcal{C}$ri$\mathcal{SP}$ using either Pspline or Kernel Smoothing is very good at all combinations of a and σ. In most cases, the size of the test is well below the significance level, and the power is at acceptable levels .

Table 5.3: Zero Bumps: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_1(t_i|a = 0.15) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0, 1)$, $\sigma = 0.05, 0.10$.

$\boldsymbol{a = 0.15}$		$\boldsymbol{H_0}$ **: # of bumps** $\boldsymbol{\le k}$		
Noise	**Level**	Method	$\boldsymbol{k = 0}$	$k = 1$
$\boldsymbol{\sigma = 0.05}$	$\alpha = 0.05$	Pspline	0.032	0.010
		Kernel	0.060	0.014
	$\alpha = 0.10$	Pspline	0.082	0.026
		Kernel	0.106	0.038
	$\alpha = 0.25$	Pspline	0.240	0.114
		Kernel	0.298	0.162
$\boldsymbol{\sigma = 0.10}$	$\alpha = 0.05$	Pspline	0.024	0.026
		Kernel	0.030	0.032
	$\alpha = 0.10$	Pspline	0.066	0.050
		Kernel	0.076	0.060
	$\alpha = 0.25$	Pspline	0.208	0.156
		Kernel	0.250	0.204

In the first case, $a = 0$, the rejection levels are far below the specified significance levels of 0.05, 0.10, and 0.25. As might be expected, when the noise level is higher, the null hypothesis is rejected more often. However this difference is only significant at level $\alpha = 0.25$. Both methods (Pspline and Kernel Smoothing) give comparable rejection levels, with the first method being slightly superior.

For $a = 0.15$ our test performs well given the fact that $m(\cdot|a = 0.15)$ is on a border line of having a bump. The proportions of rejection of H_0 are larger than when $a = 0$, but in most cases are still below the specified significance levels. The

Table 5.4: One Bump: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_1(t_i|a = 0.45) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0, 1)$, $\sigma = 0.05, 0.10$.

$\boldsymbol{a = 0.45}$		$\boldsymbol{H_0 : \# \text{ of bumps } \leq k}$			
Noise	**Level**	Method	$k = 0$	$\boldsymbol{k = 1}$	$k = 2$
$\boldsymbol{\sigma = 0.05}$	$\alpha = 0.05$	Pspline	0.996	0.010	0.008
		Kernel	1.000	0.006	0.018
	$\alpha = 0.10$	Pspline	1.000	0.026	0.020
		Kernel	1.000	0.022	0.036
	$\alpha = 0.25$	Pspline	1.000	0.124	0.096
		Kernel	1.000	0.154	0.132
$\boldsymbol{\sigma = 0.10}$	$\alpha = 0.05$	Pspline	0.498	0.002	0.012
		Kernel	0.916	0.006	0.010
	$\alpha = 0.10$	Pspline	0.686	0.016	0.042
		Kernel	0.974	0.024	0.046
	$\alpha = 0.25$	Pspline	0.924	0.138	0.150
		Kernel	0.998	0.136	0.154

Kernel Smoothing method rejects more often than the Pspline method. However the difference is significant only at $\alpha = 0.25$. It seems surprising that we reject the true null hypothesis more often with a smaller noise.

Thus far, we dealt with functions with zero bumps, and the rejection of the null hypothesis was expected in all the tests performed. When $a = 0.25$ or $a = 0.45$ the problem becomes more challenging, since we would like to reject the null hypothesis when $k = 0$, but not in the other instances $k = 1$ and $k = 2$.

In the easier situation of $a = 0.45$, both Pspline and Kernel Smoothing give very

Table 5.5: One Bump: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_1(t_i|a = 0.25) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0,1)$, $\sigma = 0.05, 0.10$.

$a = 0.25$		**H_0 : # of bumps $\leq k$**			
Noise	**Level**	Method	$k = 0$	**$k = 1$**	$k = 2$
$\sigma = 0.05$	$\alpha = 0.05$	Pspline	0.782	0.018	0.010
		Kernel	0.822	0.040	0.016
	$\alpha = 0.10$	Pspline	0.902	0.040	0.034
		Kernel	0.866	0.066	0.038
	$\alpha = 0.25$	Pspline	0.962	0.138	0.126
		Kernel	0.936	0.170	0.158
$\sigma = 0.10$	$\alpha = 0.05$	Pspline	0.302	0.028	0.016
		Kernel	0.298	0.054	0.030
	$\alpha = 0.10$	Pspline	0.462	0.050	0.056
		Kernel	0.446	0.110	0.072
	$\alpha = 0.25$	Pspline	0.702	0.182	0.186
		Kernel	0.676	0.248	0.196

good results. At a smaller level of noise $\sigma = 0.05$, the estimated power of the test for H_0 : $k = 0$ is 100% at all levels, except at $\alpha = 0.05$ where the Pspline method rejects 99.6% of the time. When the noise level is increased to $\sigma = 0.10$, the power decreases slightly for the Kernel method and quite a bit for the Pspline method. When we test with $k = 1$ or 2, at both noise levels the estimated sizes are far below the specified significance levels, and both methods (Pspline and Kernel) give very similar results. Thus, based on size and power, Kernel Smoothing seems superior to the Pspline method here. As expected, we falsely reject $H_0 : k \leq 2$ more often at the higher noise level. However the false rejection of $H_0 : k \leq 1$ happens at virtually the same frequency at both levels of noise.

We encountered by far the most difficult task in testing with $a = 0.25$. When the noise is at a lower level, we reject $H_0 : k = 0$ a large proportion of time. However, power decreases dramatically when $\sigma = 0.10$. The rejection levels for the true null hypothesis fall well below the specified significance levels, with performance better at the lower noise level. The Pspline method seems to be superior to the Kernel method, but the differences are not significant. In Figure 5.6 we can observe that the power of the test rises slowly as a function of significance level. Therefore, it would be more reasonable to choose a threshold value for α to be more than 10%. For a threshold of $\alpha = 0.15$ (indicated by the vertical line), the power is high (around 0.50) and the size is fairly small (around 0.10).

5.1.3 Multi-Bump Function

In the case of the first regression function $m_1(\cdot \mid a)$, we had either 0 bumps or 1 bump. However, we want to evaluate the $\mathcal{C}$ri$\mathcal{SP}$ in a more challenging situation. We study regression functions with 0, 1 or 2 bumps in our second attempt. We use the function $m_2(\cdot|a, b)$ introduced in section 5.1

$$m_2(t|a, b) = 1 + \exp(-4t) + a * B(t|0.25) + b * B(t|0.75).$$

We analyze the behaviour of the $\mathcal{C}$ri$\mathcal{SP}$ when the true function contains 0, 1 or 2 bumps. In order to accommodate all the situations, we have chosen 9 combinations of parameters a and b presented below:

- $m_2(\cdot \mid a, b)$ with 0 bumps (see Figure 5.7):
  - $m(\cdot|a = 0, b = 0)$

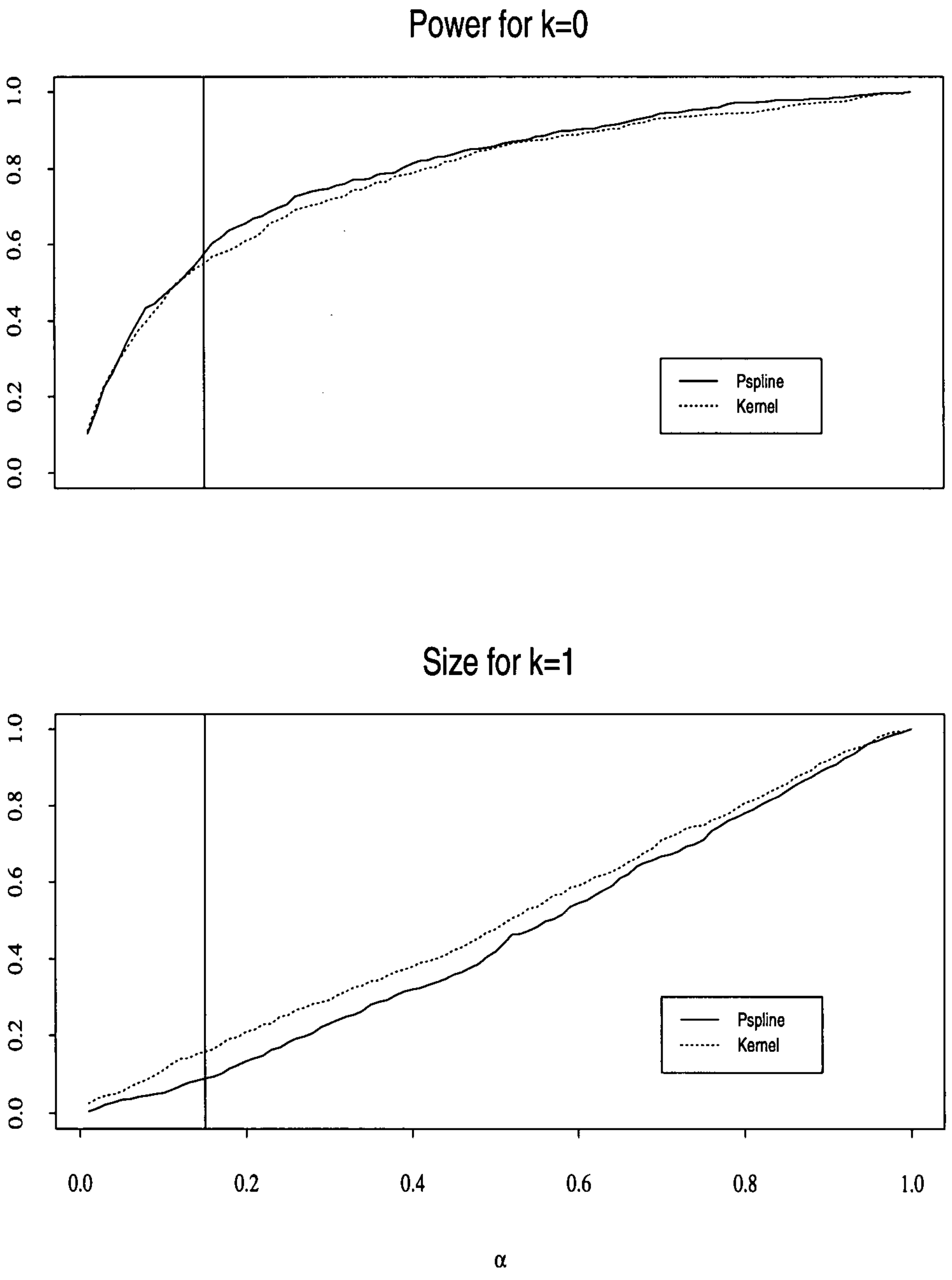
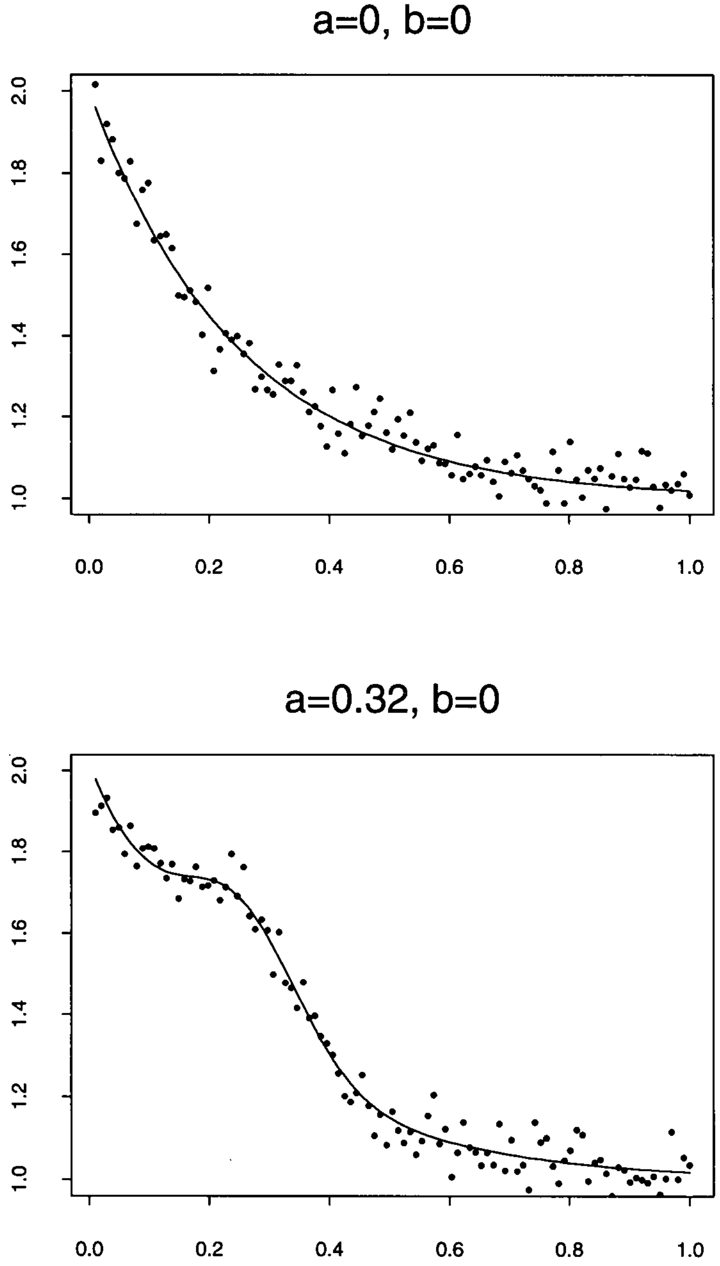


Figure 5.6: Power of the $\mathcal{C}$ri$\mathcal{SP}$ test with $k = 0$ and size of the test with $k = 1$ for the function $m_1 = 1 + t + 0.25 * B(t|0.5)$ with the noise level $\sigma = 0.10$.



a=0, b=0

a=0.32, b=0

Figure 5.7: Function $m_2(t|a,b) = 1 + \exp(-4t) + a * B(t|0.25) + b * B(t|0.75)$ with 0 bumps and simulated data with the level of noise $\sigma = 0.05$.

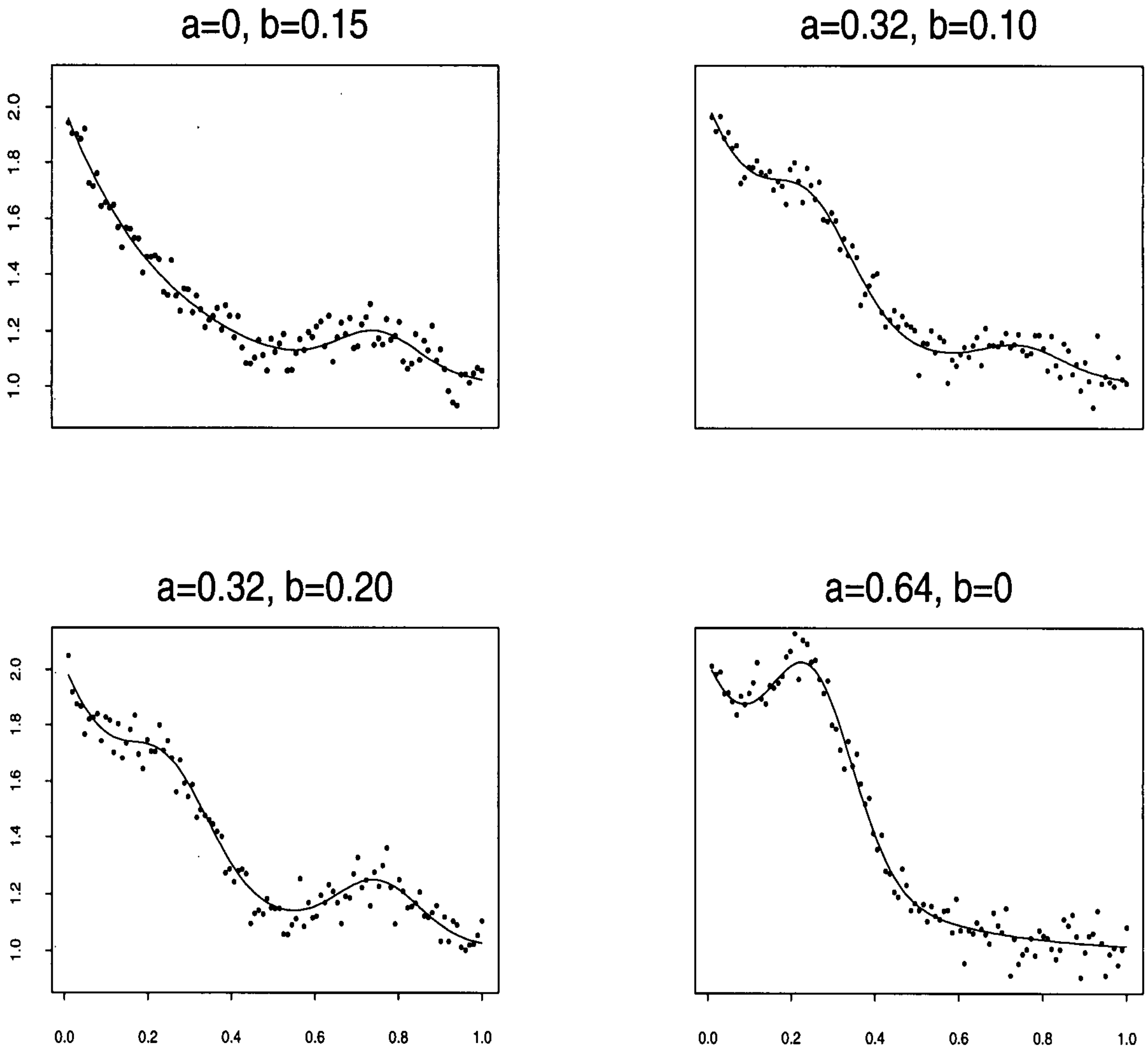
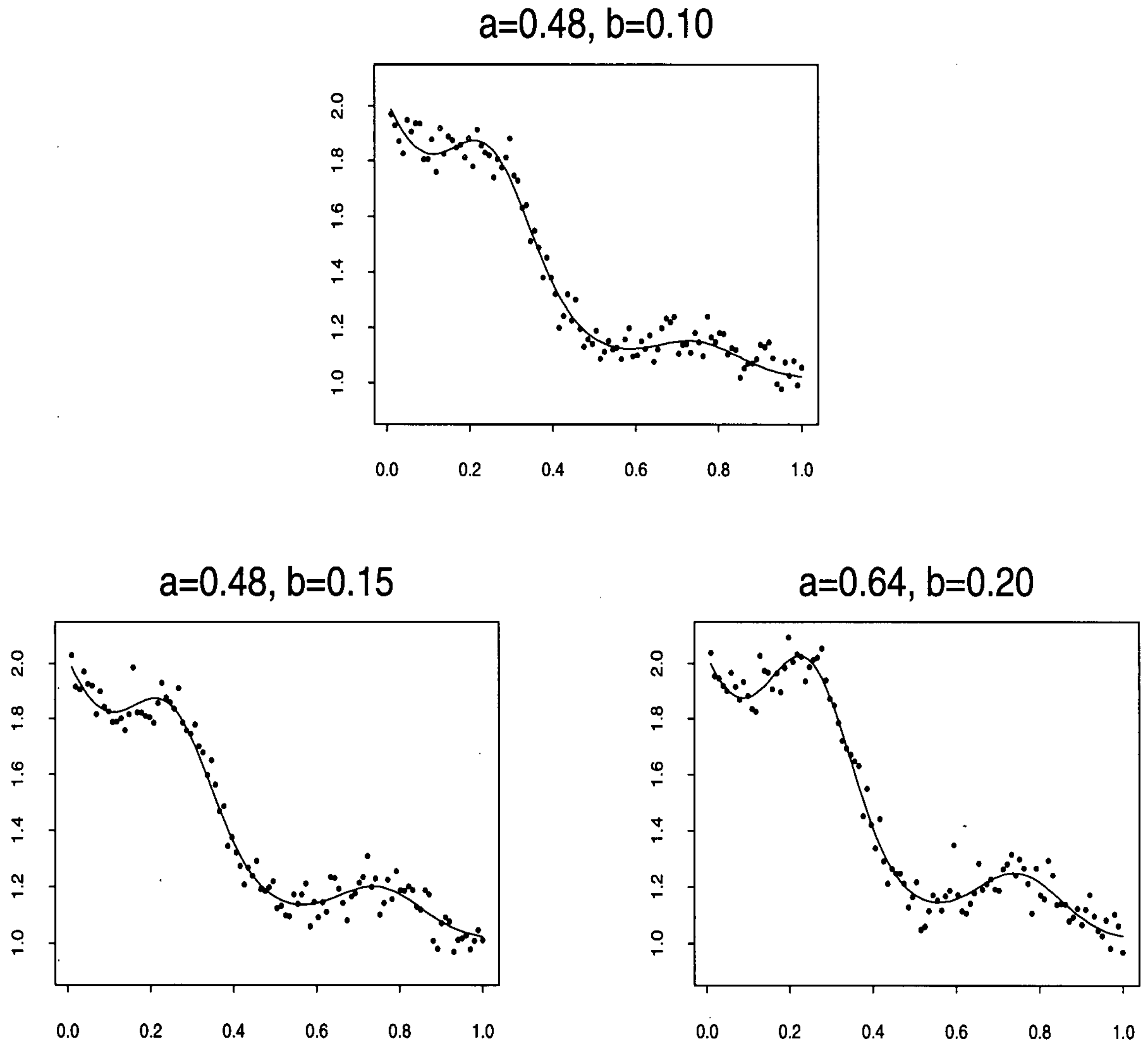


Figure 5.8: Function $m_2(t|a,b) = 1 + \exp(-4t) + a * B(t|0.25) + b * B(t|0.75)$ with 1 bump and the simulated data with the level of noise $\sigma = 0.05$.


a=0.48, b=0.10

a=0.48, b=0.15

a=0.64, b=0.20

Figure 5.9: Function $m_2(t|a,b) = 1 + \exp(-4t) + a * B(t|0.25) + b * B(t|0.75)$ with 2 bumps and simulated data with the level of noise $\sigma = 0.05$.

  - $m(\cdot|a = 0.32, b = 0)$

- $m_2(\cdot\,|\,a, b)$ with 1 bump (see Figure 5.8):
  - $m(\cdot|a = 0, b = 0.15)$
  - $m(\cdot|a = 0.64, b = 0)$
  - $m(\cdot|a = 0.32, b = 0.20)$
  - $m(\cdot|a = 0.32, b = 0.10)$
- $m_2(\cdot\,|\,a, b)$ with 2 bumps (see Figure 5.9):
  - $m(\cdot|a = 0.64, b = 0.20)$
  - $m(\cdot|a = 0.48, b = 0.15)$
  - $m(\cdot|a = 0.48, b = 0.10)$

Within each group, we present the functions in order of increasing difficulty for detecting a bump, e.g. for m_2 with one bump, when $a = 0$ and $b = 0.15$ the bump is easily detected, while the combination of $a = 0.32$ and $b = 0.10$ presents a hard testing problem. In all the simulations in this section the level of noise is fixed at $\sigma = 0.05$.

Results are displayed in Tables 5.6 - 5.8. The proportions of the simulations with p-values falling below 5%, 10 and 25% are exhibited. We use 3 smoothing methods, Lspline, Pspline and Local Linear Kernel Smoothing, in steps 1 and 5 of the bootstrap procedure (see page 36) to assess the performance of the $\mathcal{C}$ri$\mathcal{SP}$ testing procedure. Lspline is always used in step 3. In order to use Lspline, we need to estimate the parameter γ of the default model. We find $\hat{\gamma}$ for each dataset by a nonlinear least squares fit of the parametric model $m(t) = \alpha_0 + \alpha_1 \exp\{-\gamma t\}$ (see Heckman and

Ramsay (1996), for a discussion of estimation of unknown parameters in the default model).

ZERO BUMPS (Table 5.6)

In the first case of the combination of parameters $a = 0$ and $b = 0$, the sizes of the tests H_0 : # bumps $\leq k$, $k = 0, 1, 2$ fall well below the respective nominal levels of 5%, 10% and 25%. It is worth observing the unexpected fact that the size increases as k goes from 0 to 2. The performances of both the Lspline and Pspline methods are very similar, and slightly better than the performance of Kernel Smoothing.

The other case with $a = 0.32$ and $b = 0$ gives similar, but slightly worse results than the first case. The sizes of the test fall just below the nominal significance levels while testing with $k = 0$ and 1, except for $\alpha = 0.25$ and $k = 1$. However they are larger than α when $k = 2$, but in most cases only marginally. Again the performance of both spline methods is similar, and slightly superior to the Kernel method.

ONE BUMP (Table 5.7)

For $a = 0$ and $b = 0.15$, the estimated power of the test H_0 with k=0 is 100% in most cases, with the lowest value of 98.6% for the Lspline method at level 0.05. The size of the test is substantially lower than the prespecified significance level in all cases. The results for all the methods are very similar.

When $a = 0.32$ and $b = 0.20$, we achieve 100% power in all cases. The size of the test H_0 : # bumps ≤ 1 is very close to, but below the nominal significance level in all instances. For $k = 2$, the proportions of rejection of the true null hypotheses are lower than the respective levels. All the methods exhibit similar performance.

While keeping a at the same level of 0.32, we changed b to 0.10, thus trying to detect a bump that has half the amplitude of the one discussed in the previous paragraph. The size of the test is very similar to the one above on most occasions, expect for the Kernel method at level 0.25. However, the power has dropped, but is acceptable, ranging from around 65% to 90%. Therefore, even with a bump of very small amplitude the $\mathcal{C}$ri$\mathcal{SP}$ performs very well. Once more, there is practically no difference between the spline methods, with the Kernel method performing slightly worse in terms of both size and power.

The last situation we deal with is slightly different since we put $a = 0.64$ and $b = 0$, thus introducing a large bump on a steep part of an exponential function. In these circumstances, $\mathcal{C}$ri$\mathcal{SP}$'s behaviour depends on the smoothing method used. Pspline and Lspline have sizes close to and often below the respective significance levels. However, for the Kernel method the size is marginally larger than the level α in most cases. We have a reverse situation when it comes to the power of the test. The Kernel method outperforms the spline methods, especially at significance levels less than 0.15 (see Figure 5.10).

TWO BUMPS (Table 5.8)

We start off with the easiest combination of parameters $a = 0.64$ and $b = 0.20$. Both bumps are moderately big, i.e. they should be easily detected, based on our experience in the cases discussed so far. The power of the test for both $k = 0$ and 1 is over 95% in all cases but one (with the Kernel method for $k = 0$ and $\alpha = 0.05$). The size of the test is well below the prespecified significance level. All the methods perform very well with the only exception noted above for the Kernel method.

In the second function, we have two medium bumps introduced by setting $a = 0.48$ and $b = 0.15$. In the case with k=0, the power of the $\mathcal{C}$ri$\mathcal{SP}$ test stays close to 100% for all the methods at all considered levels. However, for the test with k=1, the power drops considerably for both spline methods and by some amount for the Kernel method. The reverse happens for the size: spline methods have sizes below the nominal significance levels, but the Kernel method has sizes above for the smaller values of α while testing with k=2 (see Figure 5.11).

In the last considered case, we shrink the amplitude of the bump introduced by b to 0.10, and we leave a at 0.48. We observe that the power of the test for k=0 drops substantially, if compared with the one in the previous case. However, the results obtained for $k = 1$ indicate little or no change when compared to the last case. The conclusions for the size are almost identical as in the case with b=0.15 with the Kernel method performing worse in the same circumstances (see Figure 5.12).

Table 5.6: Zero Bumps: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_2(t_i|a, b) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0, 1)$, $\sigma = 0.05$.

$\boldsymbol{a = 0, b = 0}$		$\boldsymbol{H_0 :}$ **# of bumps** $\boldsymbol{\le k}$		
Level $\boldsymbol{\alpha}$	Method	$\boldsymbol{k = 0}$	$k = 1$	$k = 2$
$\boldsymbol{\alpha = 0.05}$	Lspline	0.008	0.020	0.034
	Pspline	0.006	0.022	0.032
	Kernel	0.024	0.028	0.038
$\boldsymbol{\alpha = 0.10}$	Lspline	0.036	0.050	0.076
	Pspline	0.038	0.048	0.074
	Kernel	0.066	0.062	0.070
$\boldsymbol{\alpha = 0.25}$	Lspline	0.168	0.184	0.222
	Pspline	0.164	0.184	0.226
	Kernel	0.224	0.222	0.228

$\boldsymbol{a = 0.32, b = 0}$		$\boldsymbol{H_0 :}$ **# of bumps** $\boldsymbol{\le k}$		
Level $\boldsymbol{\alpha}$	Method	$\boldsymbol{k = 0}$	$k = 1$	$k = 2$
$\boldsymbol{\alpha = 0.05}$	Lspline	0.030	0.034	0.064
	Pspline	0.030	0.034	0.062
	Kernel	0.028	0.042	0.084
$\boldsymbol{\alpha = 0.10}$	Lspline	0.062	0.098	0.134
	Pspline	0.064	0.092	0.136
	Kernel	0.078	0.092	0.178
$\boldsymbol{\alpha = 0.25}$	Lspline	0.176	0.294	0.318
	Pspline	0.174	0.292	0.322
	Kernel	0.254	0.310	0.380

Table 5.7: One Bump: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_2(t_i|a, b) + \sigma\varepsilon_i,\ i = 1, \ldots, 101,\ \varepsilon_i \sim N(0, 1),\ \sigma = 0.05$.

$\boldsymbol{a = 0, b = 0.15}$		$\boldsymbol{H_0}$ **: # of bumps** $\boldsymbol{\leq k}$		
Level	Method	$k = 0$	$\boldsymbol{k = 1}$	$k = 2$
$\boldsymbol{\alpha = 0.05}$	Lspline	0.986	0.014	0.022
	Pspline	0.988	0.012	0.020
	Kernel	1.000	0.012	0.020
$\boldsymbol{\alpha = 0.10}$	Lspline	0.998	0.036	0.052
	Pspline	0.998	0.036	0.048
	Kernel	1.000	0.022	0.048
$\boldsymbol{\alpha = 0.25}$	Lspline	1.000	0.134	0.168
	Pspline	1.000	0.136	0.166
	Kernel	1.000	0.126	0.166

$\boldsymbol{a = 0.32, b = 0.20}$		$\boldsymbol{H_0}$ **: # of bumps** $\boldsymbol{\leq k}$		
Level	Method	$k = 0$	$\boldsymbol{k = 1}$	$k = 2$
$\boldsymbol{\alpha = 0.05}$	Lspline	1.000	0.048	0.032
	Pspline	1.000	0.050	0.034
	Kernel	1.000	0.038	0.042
$\boldsymbol{\alpha = 0.10}$	Lspline	1.000	0.088	0.056
	Pspline	1.000	0.092	0.058
	Kernel	1.000	0.076	0.068
$\boldsymbol{\alpha = 0.25}$	Lspline	1.000	0.244	0.202
	Pspline	1.000	0.250	0.202
	Kernel	1.000	0.236	0.212

Table 5.7: *(continued)*

$a = 0.32, b = 0.10$		H_0 : # of bumps $\leq k$		
Level	Method	$k = 0$	$k = 1$	$k = 2$
$\alpha = 0.05$	Lspline	0.678	0.042	0.048
	Pspline	0.652	0.042	0.046
	Kernel	0.698	0.056	0.048
$\alpha = 0.10$	Lspline	0.802	0.088	0.078
	Pspline	0.788	0.088	0.080
	Kernel	0.782	0.106	0.110
$\alpha = 0.25$	Lspline	0.930	0.264	0.206
	Pspline	0.922	0.258	0.212
	Kernel	0.878	0.274	0.310

$a = 0.64, b = 0$		H_0 : # of bumps $\leq k$		
Level	Method	$k = 0$	$k = 1$	$k = 2$
$\alpha = 0.05$	Lspline	0.546	0.078	0.032
	Pspline	0.554	0.078	0.034
	Kernel	0.838	0.066	0.044
$\alpha = 0.10$	Lspline	0.812	0.088	0.078
	Pspline	0.816	0.086	0.080
	Kernel	0.870	0.120	0.112
$\alpha = 0.25$	Lspline	0.968	0.152	0.254
	Pspline	0.972	0.154	0.258
	Kernel	0.992	0.292	0.302

Table 5.8: Two Bumps: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_2(t_i|a,b) + \sigma\varepsilon_i,\ i = 1,\ldots,101,\ \varepsilon_i \sim N(0,1),\ \sigma = 0.05$.

$\boldsymbol{a = 0.64, b = 0.20}$		$\boldsymbol{H_0 :}$ **# of bumps** $\boldsymbol{\leq k}$			
Level	Method	$k = 0$	$k = 1$	$\boldsymbol{k = 2}$	$k = 3$
$\boldsymbol{\alpha = 0.05}$	Lspline	1.000	0.958	0.020	0.024
	Pspline	1.000	0.956	0.018	0.022
	Kernel	0.922	0.996	0.016	0.016
$\boldsymbol{\alpha = 0.10}$	Lspline	1.000	0.978	0.044	0.050
	Pspline	1.000	0.976	0.042	0.050
	Kernel	0.972	1.000	0.032	0.050
$\boldsymbol{\alpha = 0.25}$	Lspline	1.000	0.998	0.158	0.172
	Pspline	1.000	0.998	0.168	0.164
	Kernel	1.000	1.000	0.146	0.172

$\boldsymbol{a = 0.48, b = 0.15}$		$\boldsymbol{H_0 :}$ **# of bumps** $\boldsymbol{\leq k}$			
Level	Method	$k = 0$	$k = 1$	$\boldsymbol{k = 2}$	$k = 3$
$\boldsymbol{\alpha = 0.05}$	Lspline	1.000	0.368	0.034	0.020
	Pspline	0.998	0.328	0.036	0.020
	Kernel	0.976	0.708	0.078	0.022
$\boldsymbol{\alpha = 0.10}$	Lspline	1.000	0.586	0.074	0.060
	Pspline	1.000	0.546	0.076	0.052
	Kernel	1.000	0.814	0.120	0.064
$\boldsymbol{\alpha = 0.25}$	Lspline	1.000	0.832	0.206	0.172
	Pspline	1.000	0.810	0.208	0.160
	Kernel	1.000	0.918	0.242	0.226

Table 5.8: *(continued)*

$\boldsymbol{a = 0.48, b = 0.10}$		$\boldsymbol{H_0 :}$ **# of bumps** $\boldsymbol{\leq k}$			
Level	Method	$k = 0$	$k = 1$	$\boldsymbol{k = 2}$	$k = 3$
$\boldsymbol{\alpha = 0.05}$	Lspline	0.650	0.420	0.036	0.036
	Pspline	0.616	0.360	0.036	0.034
	Kernel	0.640	0.738	0.100	0.044
$\boldsymbol{\alpha = 0.10}$	Lspline	0.816	0.614	0.076	0.070
	Pspline	0.800	0.544	0.078	0.066
	Kernel	0.748	0.828	0.162	0.092
$\boldsymbol{\alpha = 0.25}$	Lspline	0.934	0.864	0.224	0.234
	Pspline	0.932	0.844	0.226	0.230
	Kernel	0.914	0.910	0.316	0.244

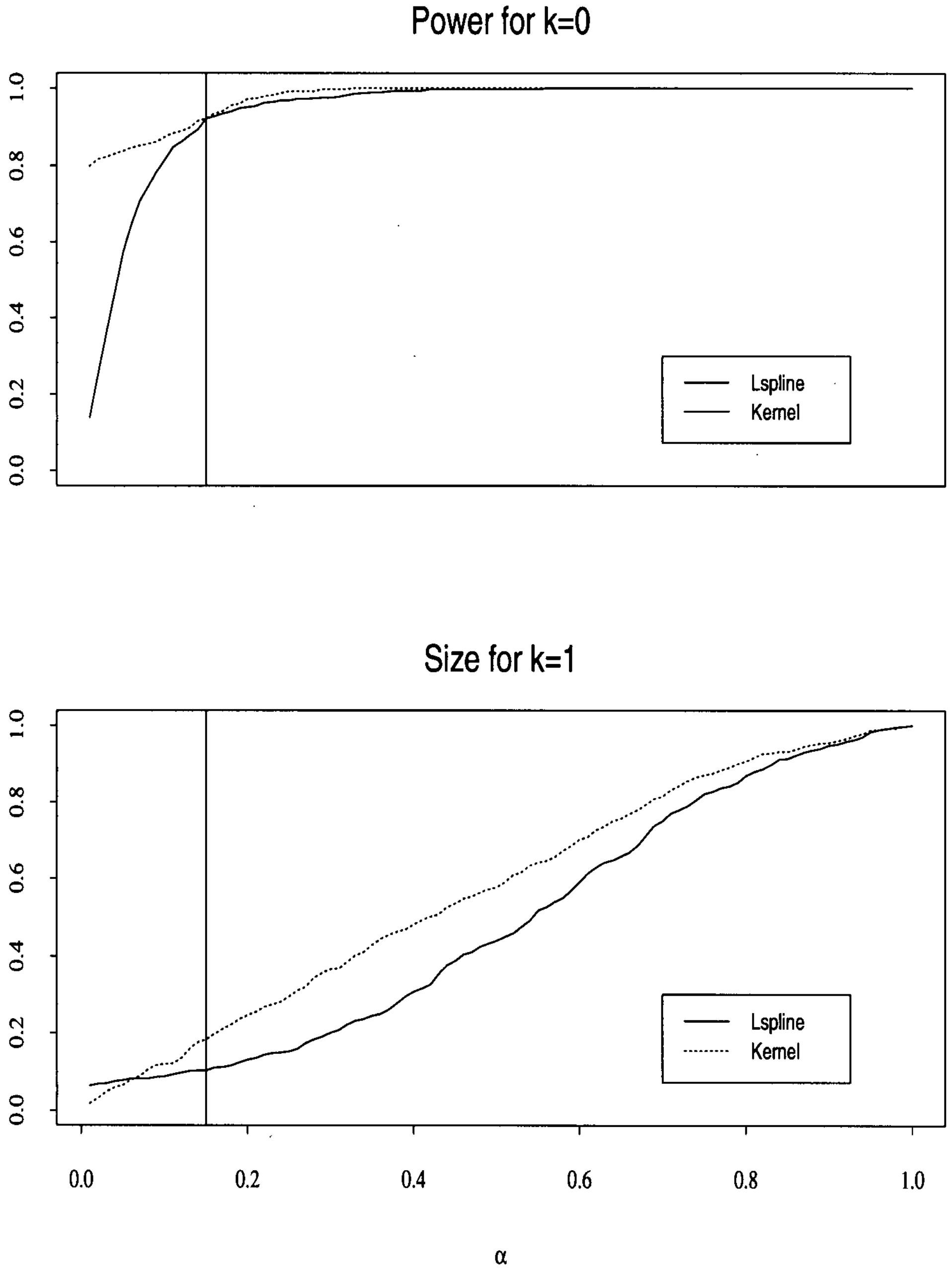


Figure 5.10: Power and size estimates for the function $m_2(t|a = 0.64, b = 0)$. The vertical line is drawn at $\alpha = 0.15$.

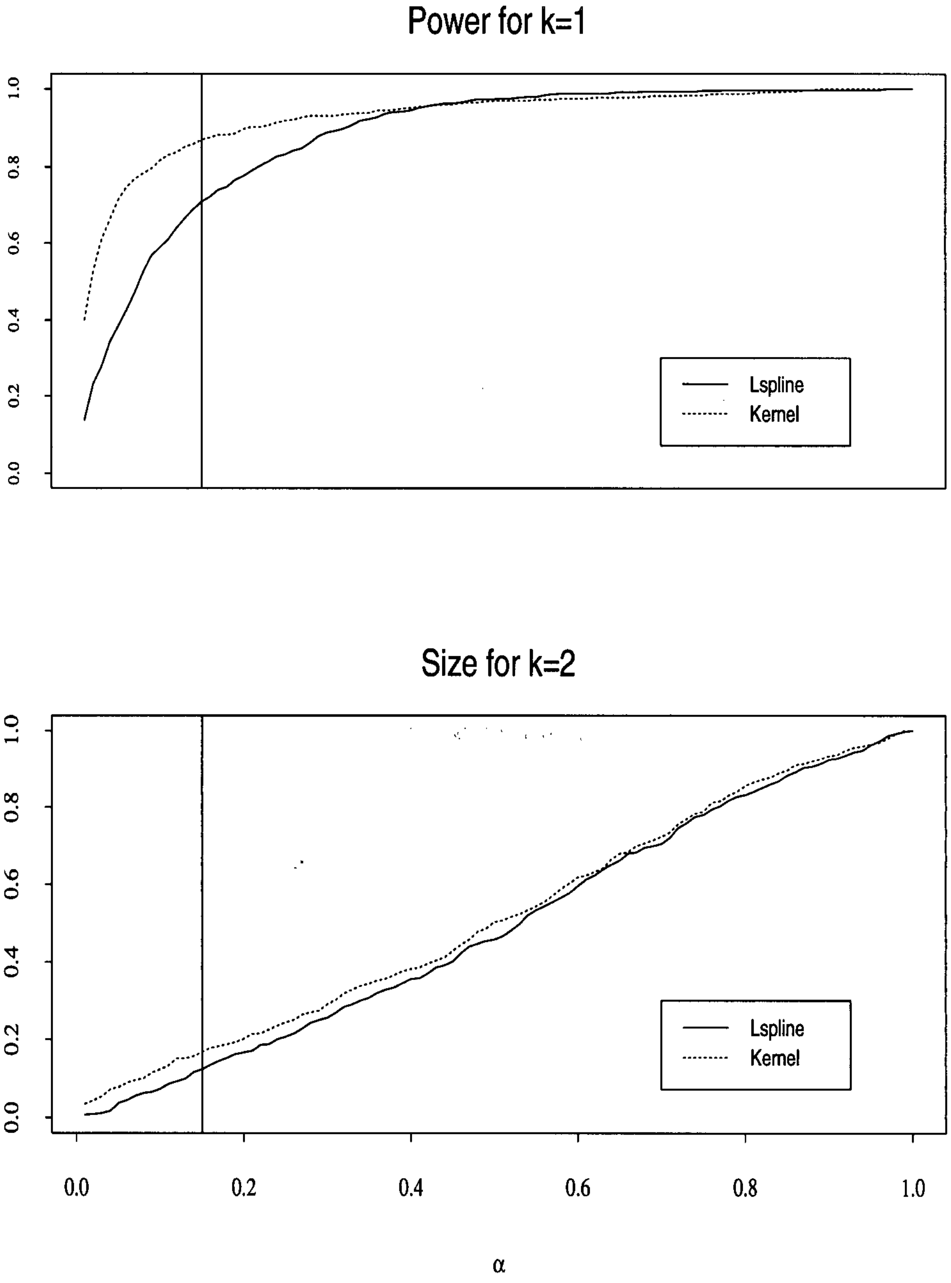
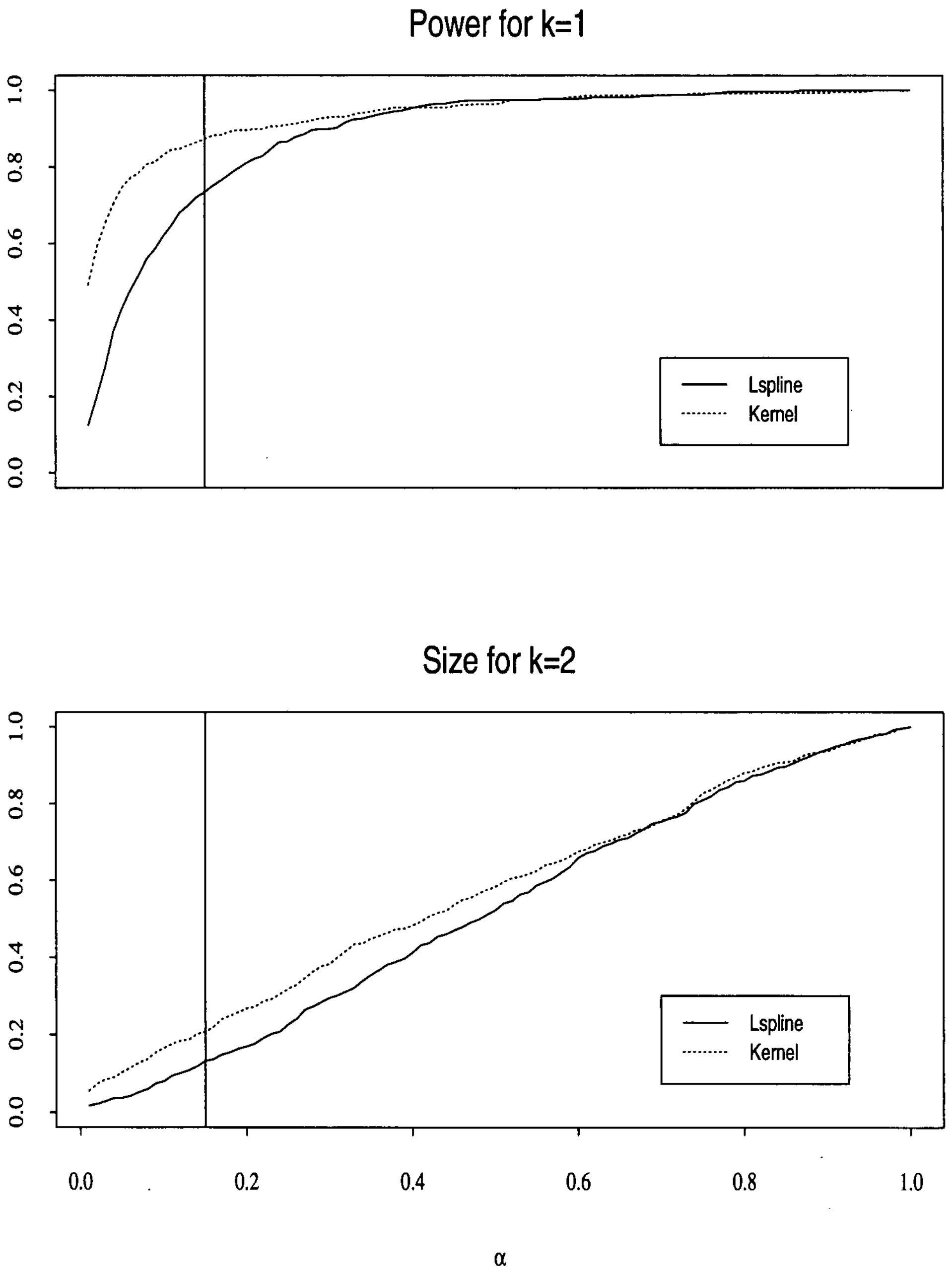


Figure 5.11: Power and size estimates for the function $m_2(t|a = 0.48, b = 0.15)$. The vertical line is drawn at $\alpha = 0.15$.


Power for k=1

Size for k=2

Figure 5.12: Power and size estimates for the function $m_2(t|a = 0.48, b = 0.10)$. The vertical line is drawn at $\alpha = 0.15$.

5.2 Derivative Simulations

As mentioned in Chapter 4 the $\mathcal{C}\mathrm{ri}\mathcal{SP}$ can be used in testing for bumps in the derivative of the regression function. As the example in section 5.3 will show, in certain situations we are interested in the rate of change of the regression function, and not in the function itself. In the simulation study we carry out, we have chosen to work with the antiderivatives of the function $(m_2(\cdot) - 1)$. Therefore we define the derivative as

$$\begin{aligned} m_3^{'}(t) &= m_2(t) - 1 \\ &= \exp\{-4t\} + a * B(t|0.25) + b * B(t|0.75) \qquad (5.9) \end{aligned}$$

and the regression function itself

$$\begin{aligned} m_3(t) &= \int_0^t m_3^{'}(s) ds \\ &= 0.25 - 0.25 \exp\{-4t\} + \\ &\quad \sqrt{(2\pi)}\,(0.1)\,\{a * \Phi((t-0.25)/0.1) + b * \Phi((t-0.75)/0.1\} \qquad (5.10) \end{aligned}$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal random variable.

We generated the data

$$Y_i = m_3(t_i) + \sigma\varepsilon_i \qquad (5.11)$$

with $t_i = i/101$ and $\varepsilon_i \sim N(0,1)$ for $i = 1, 2, \ldots, 101$. Two levels of noise were used: $\sigma = 0.0025$ and $\sigma = 0.0050$. However, if the results at the higher noise level indicated a good performance of the test, we limited the simulations at the lower noise level to one method (Lspline) and one function in each bump category. For each model, we produced 500 data sets and for each of them we applied a bootstrap procedure with a sample size equal to 500.

We have chosen 6 out of the 16 original combinations of parameters a and b defining functions $m_2(\cdot)$. We have chosen them in such a way as to have 2 curves with 0, 1, and 2 bumps. The functions $m_3(\cdot)$ are presented in Figures 5.13 - 5.15 together with their derivatives.

Throughout the study, the Lspline method of estimation was used with the penalty $L = m'' + \gamma m'$, yielding the default model $m_{def}(t) = \alpha_0 + \alpha_1 \exp\{-\gamma t\}$. The estimation of γ is necessary to utilize the Lspline smoothing method. We find $\hat{\gamma}$ for each dataset via a nonlinear least squares fit of the parametric model $m(t) = \alpha_0 + \alpha_1 \exp\{-\gamma t\}$. For comparison purposes we also use Local Quadratic Kernel Smoothing in steps 1 and 5 of the bootstrap procedure (see page 39).

ZERO BUMPS in the DERIVATIVE (Table 5.9)

In the first case with $a = 0$ and $b = 0$, the results show that the sizes of the test for $H_0 : \#\text{bumps} \leq k$ for k=0, 1, 2 are in most cases below the nominal significance levels. Both methods, Lspline and Kernel, perform well, and there is not much difference when the level of noise is increased.

In the second case with $a = 0.32$ and $b = 0$, i.e. with a "shoulder" on the slope of the derivative, for $k = 0$ and 1 the results are very similar to the previous case. The Kernel method holds an edge at the level $\alpha = 0.25$. When k=2, the Lspline method has consistently larger size than the prespecified level, and the Kernel method has size around the nominal levels.

ONE BUMP in the DERIVATIVE (Table 5.10)

We start evaluating $\mathcal{C}\text{ri}\mathcal{SP}$ in the one-bump case by setting $a = 0$ and $b = 0.20$. The estimated power of the test is equal to 100% in all the cases while testing with Lspline, and is almost as high while utilizing the Kernel method. The size of the test is well below the nominal level in most cases, and slightly below in a few situations. Size-wise the Kernel method seems to outperform the Lspline method. However we

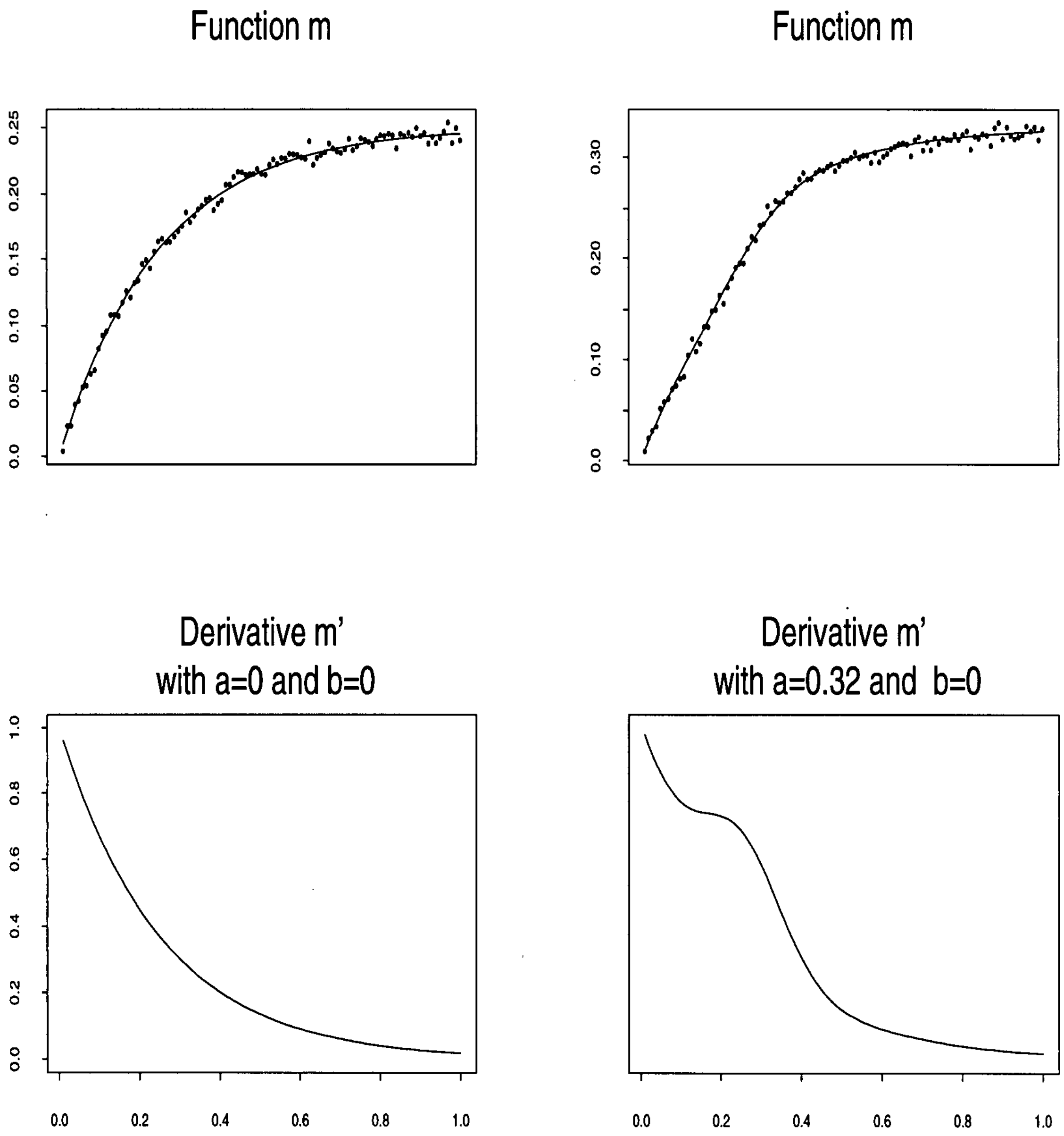


Figure 5.13: Functions $m_3(t|a,b)$ (Top) and its derivative m_3' (Bottom) with 0 bumps. Also included, data sets (Top) with noise level $\sigma = 0.005$.

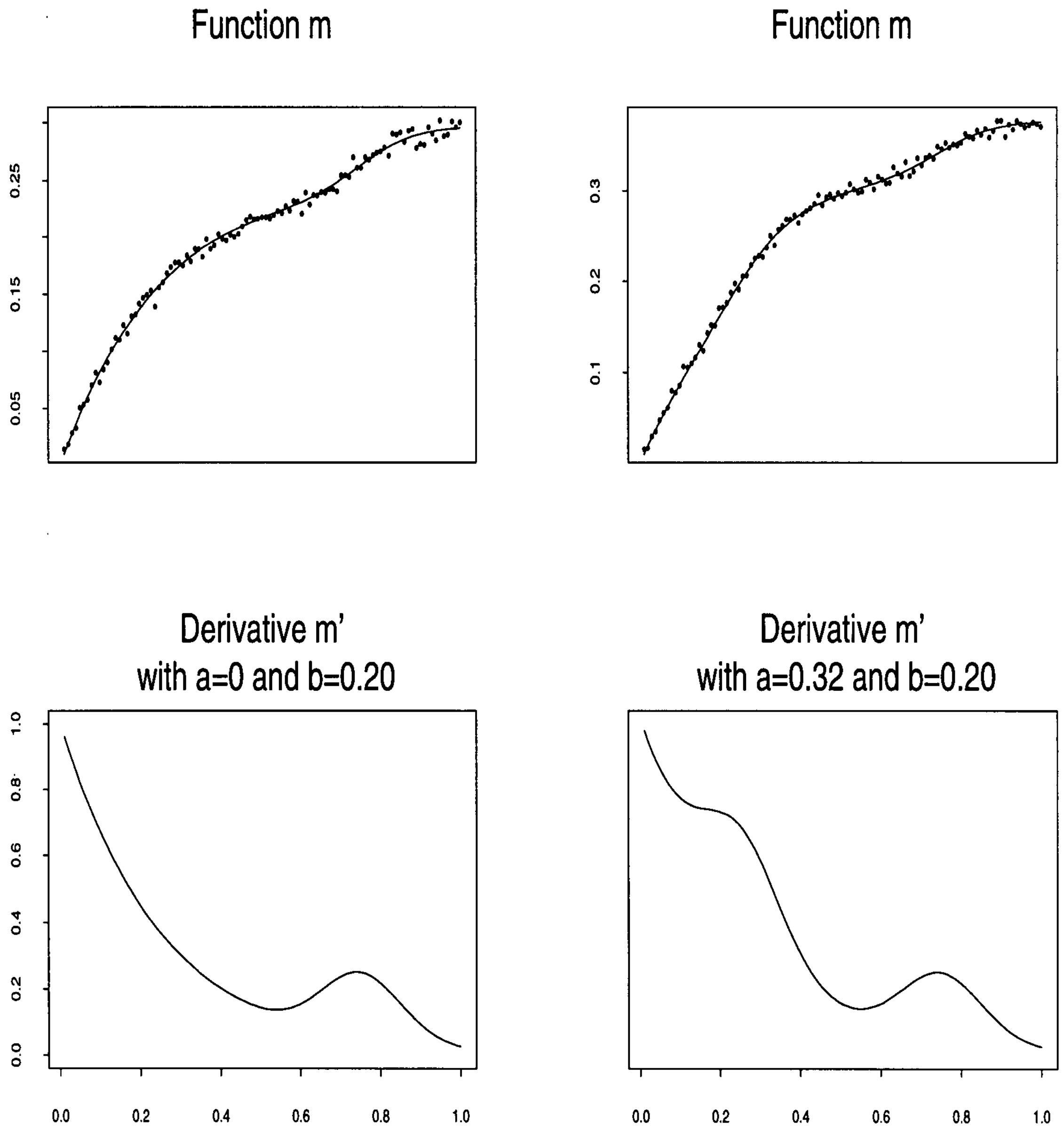


Figure 5.14: Function $m_3(t|a,b)$ (Top) and its derivative m_3' (Bottom) with 1 bump. Also included, data sets (Top) with noise level $\sigma = 0.005$.

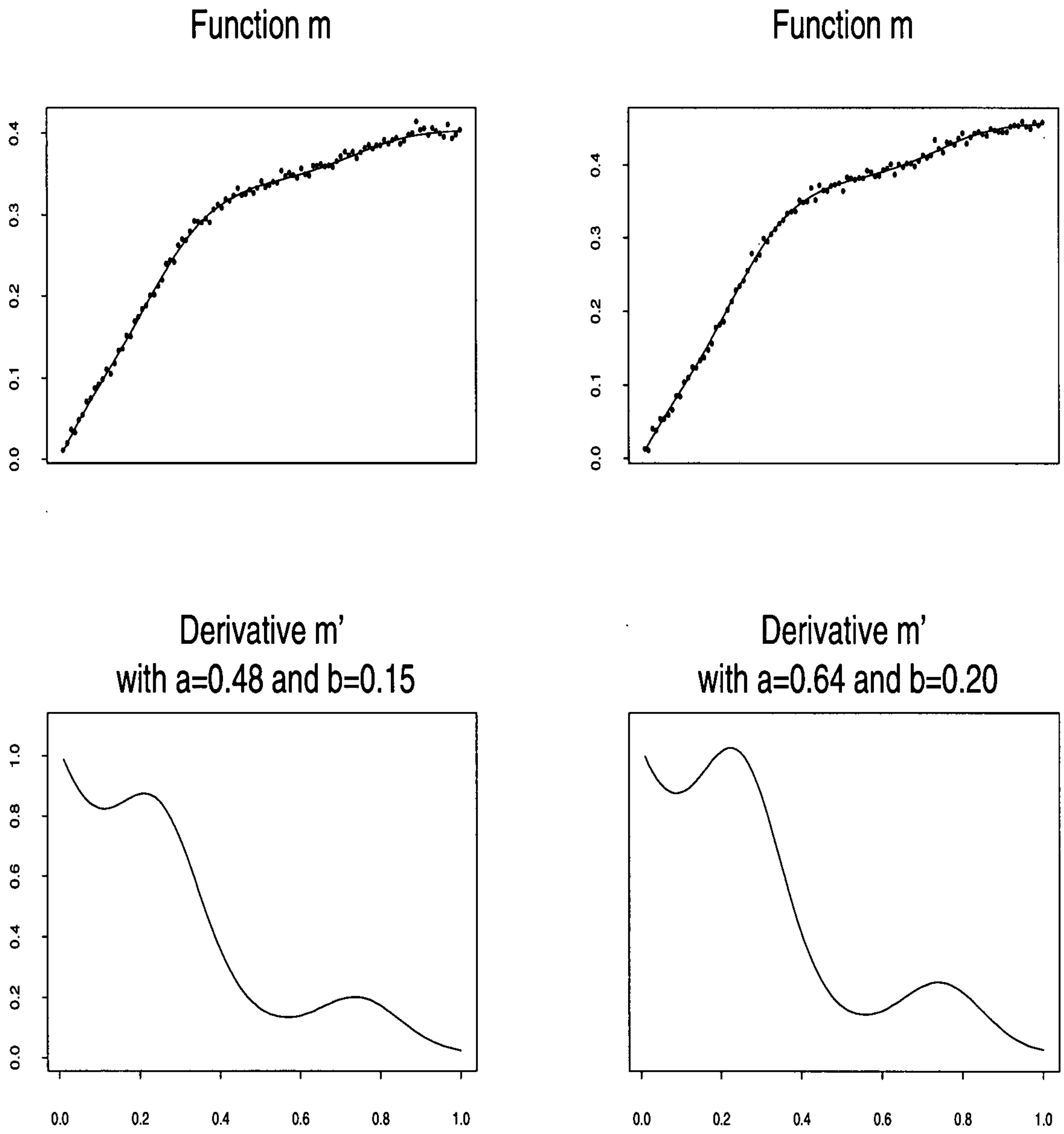


Figure 5.15: Function $m_3(t|a,b)$ (Top) and its derivative m_3' (Bottom) with 2 bumps. Also included, data sets (Top) with noise level $\sigma = 0.005$.

have to remember that Lspline has larger estimated power that is, Lspline to reject more often. To our surprise, the estimated size at the lower noise level is consistently higher than at the higher noise level. However power is about the same at both noise levels.

We try another setup of parameters a and b by introducing an "almost bump" on the steep part of $m_3(\cdot|a,b)$ by putting $a = 0.32$. We leave $b = 0.20$. Again, the power of the test is very high, reaching 100% when we use the Lspline method, and over 90% when the Kernel method is used. The sizes of the test are well below the nominal levels. In most cases, the Kernel method holds an advantage, when the size is considered, especially at the highest prespecified level $\alpha = 0.25$.

TWO BUMPS in the DERIVATIVE (Table 5.11)

We have come to by far the most difficult case of all considered. We would like to evaluate the $\mathcal{C}$ri$\mathcal{SP}$ test when we generate the data using a regression function with its derivative having two bumps. We start off with setting $a = 0.64$ and $b = 0.48$, i.e. introducing the two largest bumps considered in our study. We obtain results that are very satisfactory. The estimated power of the test is over 95% in all but one case, and the size of the test is well below the prespecified level. Once more, the Kernel method has an edge over Lspline when we compare the sizes of the test at $\alpha = 0.25$. However, the differences are only marginally significant.

We set $a = 0.48$ and $b = 0.15$, thus introducing only two moderate size bumps in $m_3^{'}(\cdot)$. At the lower noise level $\sigma = 0.0025$ the size of our test is about right in all the cases taken into account. However, when we look at the estimated power the Lspline method produces superior outcomes with the lowest estimated power over 85% at the level $\alpha = 0.05$, and k=1. The Kernel method achieves significantly lower power, especially for the lower significance levels or when we test for not more than one bump (see Figure 5.16).

When we increase the noise level to $\sigma = 0.005$, the power of $\mathcal{C}\mathrm{ri}\mathcal{SP}$ decreases slightly for the Lspline method when k=0. For k=1, the Kernel method has higher estimated power at the lower significance levels, but the Lspline method starts to be advantageous at the level $\alpha = 0.15$ (see Figure 5.16). When $k = 2$ the size of the test is higher than the nominal level, but in some cases only marginally. In the case of $k = 3$, the estimated size is at the right level in all but one case.

The simulation studies indicate that we should choose a rejection level of at least 10%. Otherwise, the power of our test will tend to be low. On the other hand, we do not want to commit a Type I error too often. Therefore the significance level should not be larger than 20%. In the next section, we settle for $\alpha = 0.15$, however any choice between 0.10 and 0.20 would be reasonable.

Table 5.9: Zero Bumps: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_3(t_i|a,b) + \sigma\varepsilon_i$, $i = 1,\ldots,101$, $\varepsilon_i \sim N(0,1)$, $\sigma = 0.0025, 0.0050$.

$\boldsymbol{a = 0, b = 0}$			$\boldsymbol{H_0 :\ \#}$ **of bumps** $\boldsymbol{\leq k}$		
Noise	**Level**	Method	$\boldsymbol{k = 0}$	$k = 1$	$k = 2$
$\boldsymbol{\sigma = 0.0025}$	$\alpha = 0.05$	Lspline	0.018	0.010	0.030
	$\alpha = 0.10$	Lspline	0.054	0.046	0.082
	$\alpha = 0.25$	Lspline	0.160	0.228	0.254
$\boldsymbol{\sigma = 0.005}$	$\alpha = 0.05$	Lspline	0.034	0.014	0.046
		Kernel	0.030	0.020	0.026
	$\alpha = 0.10$	Lspline	0.072	0.050	0.104
		Kernel	0.062	0.054	0.064
	$\alpha = 0.25$	Lspline	0.188	0.208	0.290
		Kernel	0.192	0.202	0.230

$\boldsymbol{a = 0.32, b = 0}$			$\boldsymbol{H_0 :\ \#}$ **of bumps** $\boldsymbol{\leq k}$		
Noise	**Level**	Method	$\boldsymbol{k = 0}$	$k = 1$	$k = 2$
$\boldsymbol{\sigma = 0.005}$	$\alpha = 0.05$	Lspline	0.054	0.018	0.088
		Kernel	0.044	0.020	0.040
	$\alpha = 0.10$	Lspline	0.084	0.074	0.180
		Kernel	0.062	0.052	0.090
	$\alpha = 0.25$	Lspline	0.204	0.258	0.400
		Kernel	0.144	0.152	0.284

Table 5.10: One Bump: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_3(t_i|a,b) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0,1)$, $\sigma = 0.0025, 0.0050$.

$\boldsymbol{a = 0, b = 0.20}$			$\boldsymbol{H_0 :}$ **# of bumps** $\boldsymbol{\leq k}$		
Noise	**Level**	Method	$k = 0$	$\boldsymbol{k = 1}$	$k = 2$
$\boldsymbol{\sigma = 0.0025}$	$\alpha = 0.05$	Lspline	1.000	0.028	0.028
	$\alpha = 0.10$	Lspline	1.000	0.064	0.064
	$\alpha = 0.25$	Lspline	1.000	0.238	0.248
$\boldsymbol{\sigma = 0.005}$	$\alpha = 0.05$	Lspline	1.000	0.008	0.014
		Kernel	0.944	0.000	0.004
	$\alpha = 0.10$	Lspline	1.000	0.032	0.038
		Kernel	0.998	0.002	0.022
	$\alpha = 0.25$	Lspline	1.000	0.140	0.204
		Kernel	1.000	0.064	0.140

$\boldsymbol{a = 0.32, b = 0.20}$			$\boldsymbol{H_0 :}$ **# of bumps** $\boldsymbol{\leq k}$		
Noise	**Level**	Method	$k = 0$	$\boldsymbol{k = 1}$	$k = 2$
$\boldsymbol{\sigma = 0.005}$	$\alpha = 0.05$	Lspline	1.000	0.020	0.042
		Kernel	0.910	0.038	0.010
	$\alpha = 0.10$	Lspline	1.000	0.054	0.076
		Kernel	0.978	0.048	0.032
	$\alpha = 0.25$	Lspline	1.000	0.220	0.244
		Kernel	1.000	0.076	0.148

Table 5.11: Two Bumps: Proportion of times H_0 is rejected at level α based on 500 simulations of $Y_i = m_3(t_i|a, b) + \sigma\varepsilon_i$, $i = 1, \ldots, 101$, $\varepsilon_i \sim N(0, 1)$, $\sigma = 0.0025, 0.0050$.

$a = 0.64, b = 0.20$			**H_0 : # of bumps $\leq k$**			
Noise	**Level**	Method	$k = 0$	$k = 1$	**$k = 2$**	$k = 3$
$\sigma = 0.0025$	$\alpha = 0.05$	Lspline	1.000	1.000	0.024	0.032
	$\alpha = 0.10$	Lspline	1.000	1.000	0.058	0.082
	$\alpha = 0.25$	Lspline	1.000	1.000	0.198	0.244
$\sigma = 0.005$	$\alpha = 0.05$	Lspline	0.986	0.898	0.026	0.026
		Kernel	0.956	0.960	0.044	0.026
	$\alpha = 0.10$	Lspline	0.996	0.982	0.064	0.076
		Kernel	0.964	0.968	0.070	0.060
	$\alpha = 0.25$	Lspline	0.998	0.998	0.186	0.242
		Kernel	0.980	0.976	0.138	0.176

$a = 0.48, b = 0.15$			**H_0 : # of bumps $\leq k$**			
Noise	**Level**	Method	$k = 0$	$k = 1$	**$k = 2$**	$k = 3$
$\sigma = 0.0025$	$\alpha = 0.05$	Lspline	1.000	0.874	0.040	0.042
		Kernel	0.460	0.240	0.046	0.036
	$\alpha = 0.10$	Lspline	1.000	0.938	0.090	0.122
		Kernel	0.690	0.270	0.084	0.094
	$\alpha = 0.25$	Lspline	1.000	0.972	0.238	0.288
		Kernel	0.970	0.410	0.258	0.238
$\sigma = 0.005$	$\alpha = 0.05$	Lspline	0.832	0.090	0.104	0.044
		Kernel	0.328	0.328	0.062	0.056
	$\alpha = 0.10$	Lspline	0.904	0.290	0.164	0.116
		Kernel	0.404	0.360	0.108	0.100
	$\alpha = 0.25$	Lspline	0.968	0.662	0.306	0.292
		Kernel	0.680	0.484	0.266	0.256

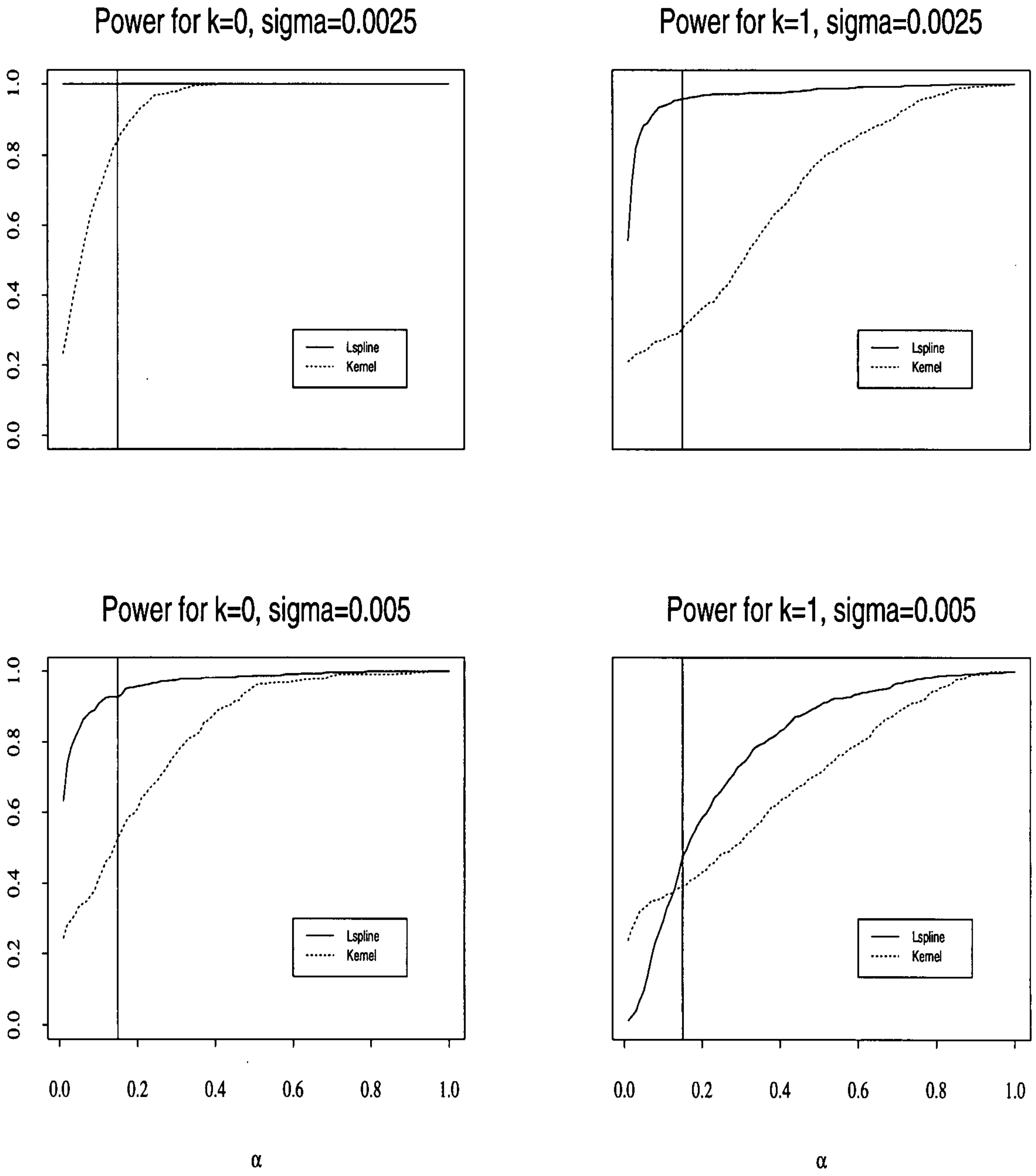


Figure 5.16: Power of the $\mathcal{C}$ri$\mathcal{SP}$ test when k=0 and k=1 for the function $m_3^{'}(t|a = 0.48, b = 0.15)$ at the two noise levels. The vertical line is drawn at $\alpha = 0.15$.

5.3 Analysis of the Growth Data

We consider the growth data coming from a longitudinal study conducted in Berkeley, California. These data are usually referred to in the literature as the Berkeley growth data. The complete description of the study can be found in Tuddenham and Snyder (1954). Here, we analyze the data coming from the records of 43 white males and 50 white females born in 1928 or 1929 in Berkeley, California. The data set includes, in most cases, measurements of recumbent length at three months and then at three-month intervals to 21 months. Standing heights were measured annually from age two until eight and thereafter semiannually until age eighteen.

Most of the analyses of human growth use the data coming from the Zurich longitudinal growth study (see Gasser et.al. (1984a, 1984b, 1985, 1990, 1991a, 1991b)). In the Zurich study, the children's heights were measured at a lesser frequency in infancy. In work making comparisons between the Berkeley and Zurich data sets, the common ages were chosen to make the comparison fair (see Ramsay et.al. (1995)). Therefore, in our analysis we have used only the measurements at ages 1, 1.5, 2 and annually thereafter until 9.5 in females and 10 in males, followed by semiannual measurements until age 18, for a total of 27 ages for the males and 28 for the females. The growth curves were analyzed for each child.

We have used the bootstrap procedure for derivatives outlined on page 39. We utilized local quadratic kernel smoothing in step 2 of the bootstrap procedure to find the unrestricted estimate $\hat{m}_{unr}(\cdot)$ of the regression function $m(\cdot)$. Local quadratic smoothing produces a good estimate of m', the rate of growth. The calculations were performed using the `locpoly` function in `Splus`. `locpoly` selects a bandwidth based on a plug-in procedure. We chose this method because the *GCV* method for smoothing parameter choice performs badly when the noise in the data is very small, as is the case here. The GCV choice of λ is usually too small, yielding residuals which are very close to zero.

Also, we employed `Lspline` smoothing in steps 1 and 5 of the bootstrap procedure. The penalty used was $Lm = m'' + \gamma m'$ with the default model $m_{def}(t) = \alpha_0 + \alpha_1 \exp\{-\gamma t\}$. We estimated γ by a non-linear least squares procedure. For each child, we generated 500 bootstrap samples from the original residuals.

Our data set has height measurements on either a semi-annual or annual basis. If we wanted to implement our testing procedure at the observed times t_i, it would be easier to detect a bump in an area with more frequent measurements. However, we do not want the frequency of observations to differ from the testing procedure developed for equally spaced independent variable values. Therefore, for testing purposes, we evaluate the estimate of the speed of growth at 101 equally spaced values of time between the ages 1 and 18.

We decide on the number of bumps in a curve using p-values from tests for the existence of 0, 1, 2, or 3 bumps in the speed of the growth. Following the discussion in section 5.2, we decided to choose the significance level of our test to be $\alpha = 0.15$, i.e. we reject H_0 if a p-value ≤ 0.15. For each child, we perform 4 tests: $H_0 : \#$ bumps $\leq k$ for $k = 0$, 1, 2, and 3. We base our decision on either a "strict" or "soft" rule, described below.

By the "strict" decision rule, we reach a conclusion that the speed of growth has exactly b bumps if, for the hypothesis tests $H_0 : \#$ bumps $\leq k$ the p-values are less than or equal to 0.15, for all $k < b$ and the p-values are greater than 0.15 for all $b \leq k \leq 3$. When we reject H_0 for all $k \leq 3$ we conclude that the number of bumps b is greater than three. In some cases, no classification can be made by the strict decision rule. For instance, we may reject the hypothesis $H_0 : \#$ bumps $\leq k$ for k=0 and 2, but we do not reject it for k=1 and 3 at the level $\alpha = 0.15$.

Therefore, we introduce the "soft" decision rule, which always gives us a classification. For b=0, 1, 2, 3, we say that the curve has b bumps if, for the hypothesis tests H_0 : # bumps $\leq k$, for all $k < b$ the p-values are less than or equal to 0.15, and the p-value is greater than 0.15 for $k = b$. As with the strict decision rule, we conclude that a curve has more than 3 bumps if, for all $k \leq 3$, the p-values are less than or equal to 0.15.

If we cannot reach a conclusion according to the strict decision rule, we report not only b, the soft rule decision, but also a range of values for the number of bumps. Thus, we say that a curve has $b_L, b_{L+1}, \dots, b_U$ bumps where $b_L = b$ according to the soft decision rule, and $b_U = 1 +$ (the biggest k for which the p-value ≤ 0.15). For example, when the p-value ≤ 0.15 for k=0 and 2, and the p-value > 0.15 for k=1 and 3, we have a range of bumps $b_L = 1, \dots, b_U = 1 + 2$, since k=2 is the largest k value for which the p-value ≤ 0.15.

We present here the results obtained using `Lspline` smoothing with 3-max as the definition of bump. Results are summarized in Tables 5.12 and 5.13 for males and females respectively. When the strict rule gives us a classification we have a single number in the "Conclusion" column. For other cases, we present a classification based on the range rule with the number of bumps as classified by the soft rule indicated in bold face.

The full results of all the tests performed are presented in Appendices A and B in Tables A.1 and B.1 for males and females respectively.

MALE DATA

Using the "strict" decision rule described in the last section, we classify the males into groups with 0, 1, 2, 3 and more than 3 bumps (or spurts of growth). We can make a decision in 34 out of 43 cases. Out of 34 males classified according to the strict decision rule, 10 males have one spurt of growth, 17 males - two spurts, 6 males

- three spurts, and 1 male more than three spurts.

In the remaining 9 cases, we do not reject the null hypothesis for some smaller values of k, but we do reject for larger k's. For instance, for 5 boys, we do not reject the H_0 with k=1 and 3, but we do reject H_0 for k=0 and 2. As mentioned before, in such a situation we give a range of values for b, the number of bumps.

The "soft" decision rule gives the following classification: 17 males with one growth spurt, 19 males with two growth spurts, 6 males with three, and 1 male with more than three spurts of growth.

Figures 5.17 - 5.19 contain plots of curves classified by the strict definition of having 1, 2, and 3 bumps, respectively. None of the 43 curves was classified as having 0 bumps. We plot the estimates for b=1, 2, and 3 bumps, and we indicate the one we decided on by a solid curve. The top plot in each figure shows an individual where our decision was very clear: p-values were much smaller than $\alpha = 0.15$ for k less than our choice of b, and much larger than 0.15 for the other values of k. The p-values for the bottom plot are less separated.

Figure 5.20 contains plots of two curves that could only be classified by the soft or range decision rule. The top plot shows curves for male # 18, with a range of the number of bumps: 1, 2, 3. The bottom plot plot for male # 23 has a range: 1, 2, 3, >3 bumps.

We cannot make a complete comparison of our test results with the work described in Ramsay et.al. (1995). The procedure outlined in their work deals with estimating so called "landmarks", or ages at which an important feature of growth occurs. The features consist of events associated with the mid-spurt of growth (MS), and the pubertal spurt of growth (PS). In total, eight landmarks are described, but only two are related to bumps in the rate of growth: $T3$ - age of maximal velocity during MS, and $T8$ - age of maximal velocity during PS.

Ramsay et.al. then find in how many cases $T3$ and $T8$ occur. From the information provided, we can use Ramsay et.al.'s estimation procedure to classify curves into two groups: in one group, curves have one bump (only $T8$ present), and in the other group, curves have two or more bumps (both $T3$ and $T8$ present with the possibility of $T3$ being multiple).

For the male data, they discovered the occurrence of $T3$ in 24 out of 43 cases, and $T8$ in all 43 cases. In our study, we found 1 bump in 10 cases out of 34 classified by the "strict" decision rule, and 17 out of 43 using the "soft" decision rule. The remaining curves were, of course, classified as having 2 or more bumps. Therefore, 71% of the cases classified by the "strict" rule give us 2 or more bumps in a curve, and in 60% of the cases "soft" rule indicates 2 or more bumps in the rate of growth curves. Ramsay et.al. had 56% of male growth curves with 2 or more bumps. We can conclude that our "soft" decision rule gives comparable results to the ones described in Ramsay et.al.

Table 5.12: p-value estimates based on 500 bootstrap samples for the male data using 3-max as a definition of a bump in the testing procedure.

	H_0 : # of bumps $\leq k$				
Males	$k = 0$	$k = 1$	$k = 2$	$k = 3$	**Conclusion**
1	0.000	0.002	0.938	0.994	2
2	0.000	0.908	0.680	0.316	1
3	0.000	0.130	0.380	0.718	2
4	0.000	0.068	0.274	0.996	2
5	0.000	0.310	0.094	0.884	**1**,2,3
6	0.000	0.000	0.126	0.560	3
7	0.000	0.010	0.114	0.558	3
8	0.000	0.590	0.442	0.998	1
9	0.000	0.000	0.864	0.454	2
10	0.000	0.862	0.478	0.476	1
11	0.000	0.000	0.220	0.588	2
12	0.000	0.030	0.460	0.310	2
13	0.000	0.012	0.798	0.802	2
14	0.000	0.012	0.162	0.126	**2**,3,>3
15	0.000	0.000	0.984	0.992	2
16	0.000	0.088	0.148	0.134	>3
17	0.000	0.830	0.292	0.384	1
18	0.000	0.380	0.010	0.224	**1**,2,3
19	0.000	0.154	0.706	0.718	1
20	0.000	0.184	0.204	0.036	**1**,2,3,>3
21	0.000	0.000	0.532	0.178	2
22	0.000	0.000	0.472	0.744	2

Table 5.12: *(continued)*

Males	$k = 0$	$k = 1$	$k = 2$	$k = 3$	**Conclusion**
23	0.000	0.750	0.296	0.086	**1**,2,3,>3
24	0.000	0.000	0.150	0.484	3
25	0.000	0.000	0.208	0.256	2
26	0.000	0.092	0.162	0.066	**2**,3,>3
27	0.000	0.748	0.798	0.416	1
28	0.000	0.390	0.078	0.776	**1**,2,3
29	0.000	0.000	0.172	0.566	2
30	0.000	0.010	0.038	0.364	3
31	0.000	0.050	0.686	0.998	2
32	0.000	0.152	0.994	0.978	1
33	0.000	0.534	0.302	0.484	1
34	0.000	0.222	0.028	0.286	**1**,2,3
35	0.000	0.006	0.856	0.938	2
36	0.000	0.296	0.030	0.188	**1**,2,3
37	0.000	0.002	0.034	0.516	3
38	0.000	0.000	0.018	0.400	3
39	0.000	0.100	0.538	0.994	2
40	0.000	0.106	0.658	0.440	2
41	0.000	0.824	0.408	0.260	1
42	0.000	0.212	0.592	0.488	1
43	0.000	0.028	0.388	0.576	2

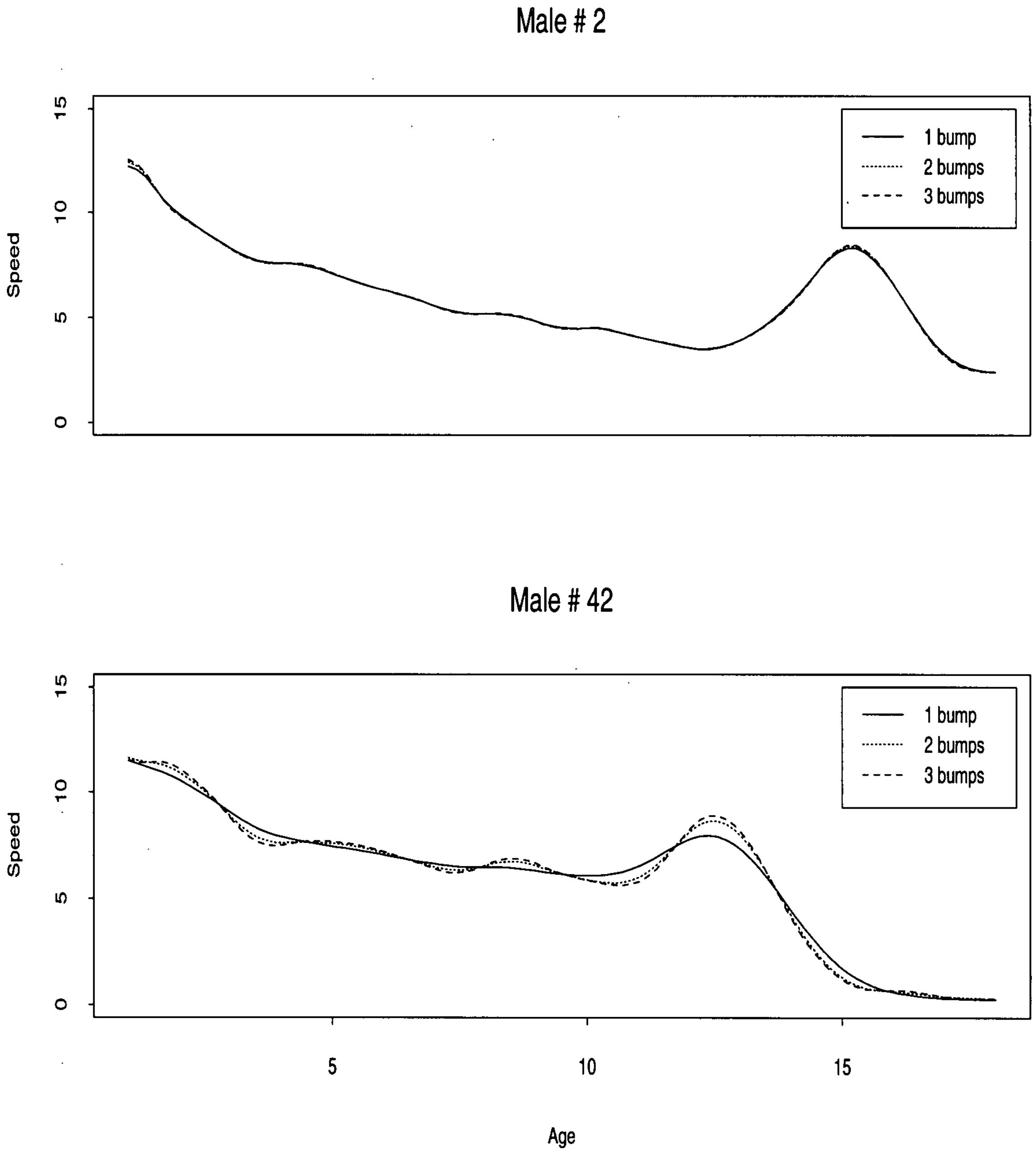


Figure 5.17: Estimates of the speed of growth of males with one bump.

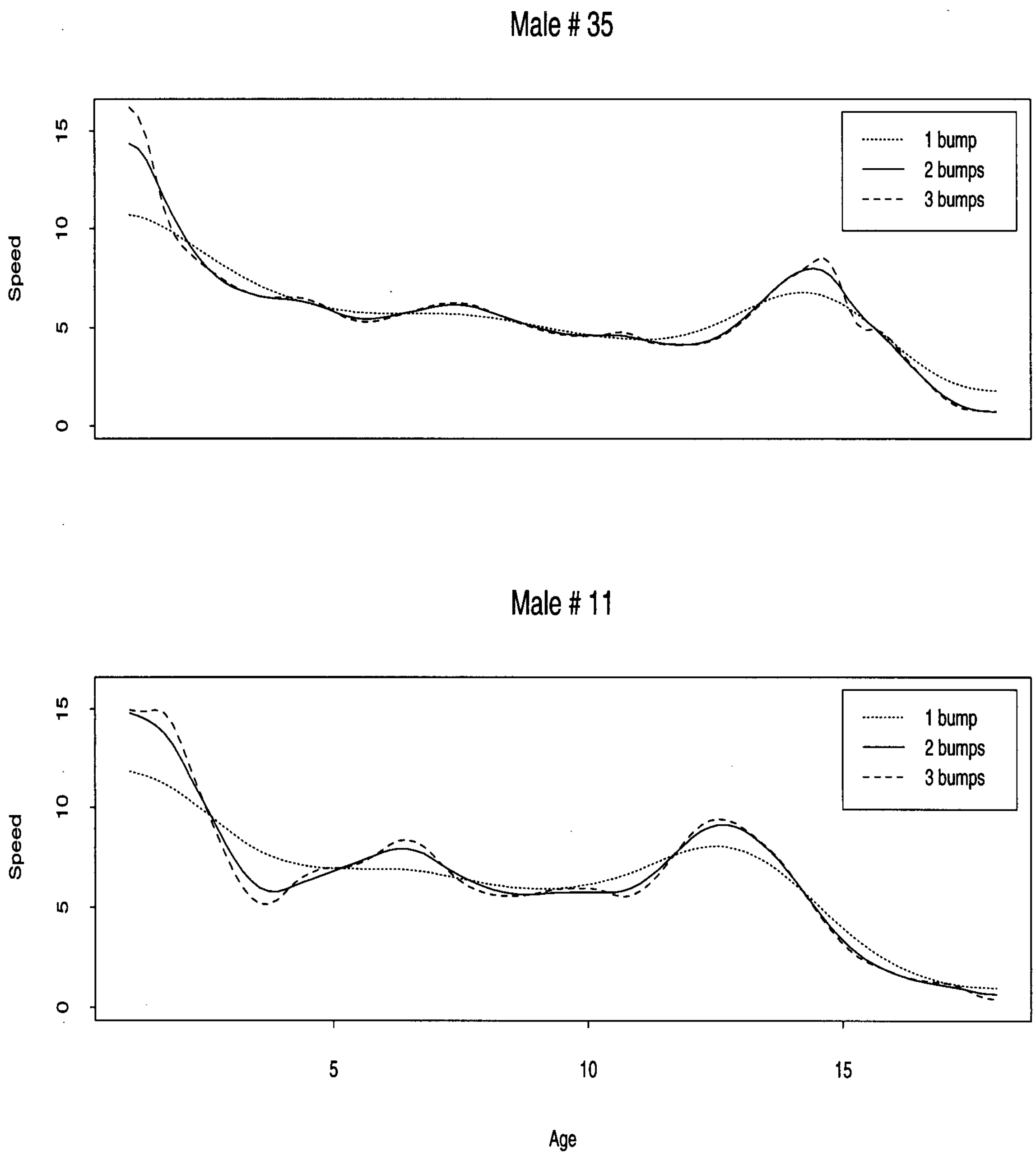


Figure 5.18: Estimates of the speed of growth of males with two bumps.

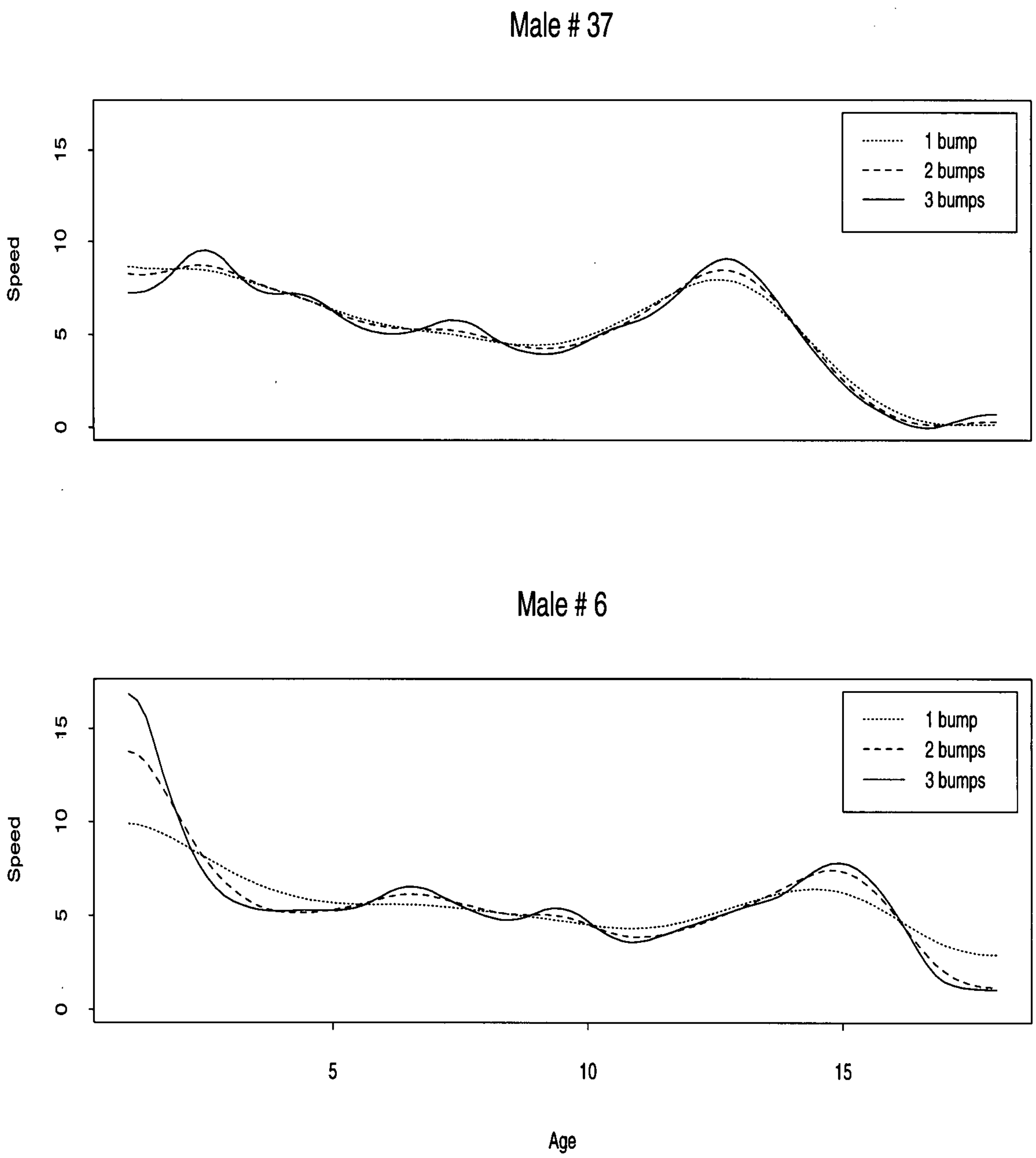
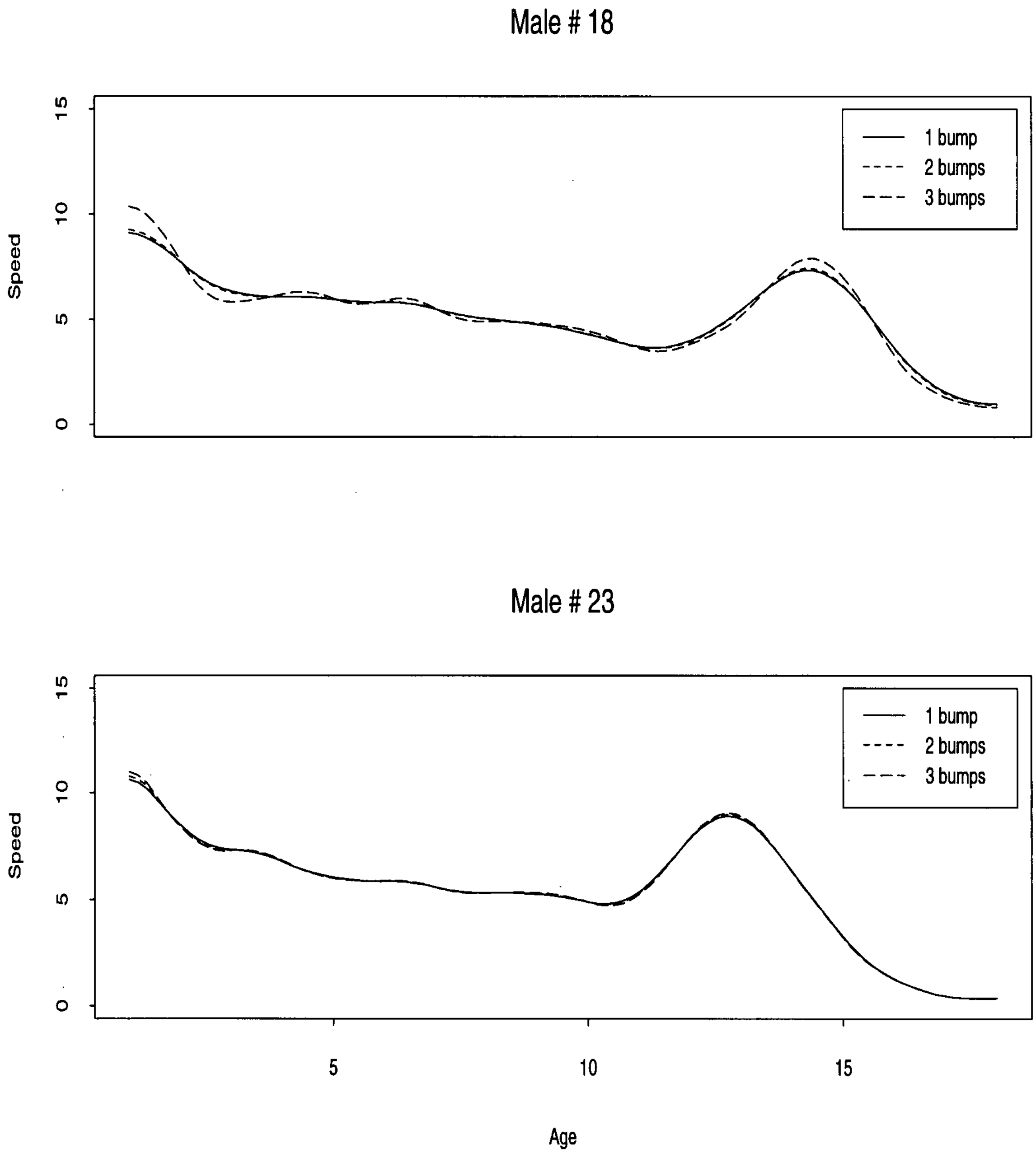


Figure 5.19: Estimates of the speed of growth of males with three bumps.



Male # 18

Male # 23

Figure 5.20: Estimates of the speed of growth of males when a decision cannot be reached by a "strict" criterion.

FEMALE DATA

When we apply the "strict" decision rule to the female growth curves, we can classify 39 out of 50 females. In 17 of these cases, we decide that there is one spurt of growth, in 15 cases - two spurts, in 5 cases - three spurts, and in 2 cases - more than three spurts of growth. The remaining 11 females cannot be classified into one of the mentioned groups.

Classification according to the "soft" rule gives: 1 case with zero spurts, 24 cases with one spurt of growth, 18 cases with two spurts, 5 cases with three, and 2 cases with more than three spurts of growth in the speed curve.

We plot two curves for each number of the bumps detected b=1, 2, 3 according to the "strict" rule (see Figures 5.21 - 5.23). Figure 5.24 contains the cases when the "strict" decision rule did not provide a b value.

In Appendix B the plots of the estimated growth functions for all 50 females are presented.

Ramsay et.al. found $T3$ present in 9 growth curves, and $T8$ in 49 growth curves, and this would classify 49 of the curves as having 1 bump and 9 of the curves as having 2 or more bumps. (Recall that there were multiple occurrences of $T3$ in some estimates of the female speed of growth.) Our results indicate more females having more than one episode of maximal velocity, 22 cases out of 39 using the "strict" criterion, and 25 out of 50 using the "soft" criterion.

Comparison of the female and male growth curves suggests that multi-bump curves are more prevalent for boys than for girls (71% of the boys versus 56% of the girls when the "strict" criterion is used, and 60% of the boys versus 50% of the girls when the "soft" criterion is used). Also, boys and girls are equally easy to classify by the "strict" definition, with only about 20% of each group being unclassified.

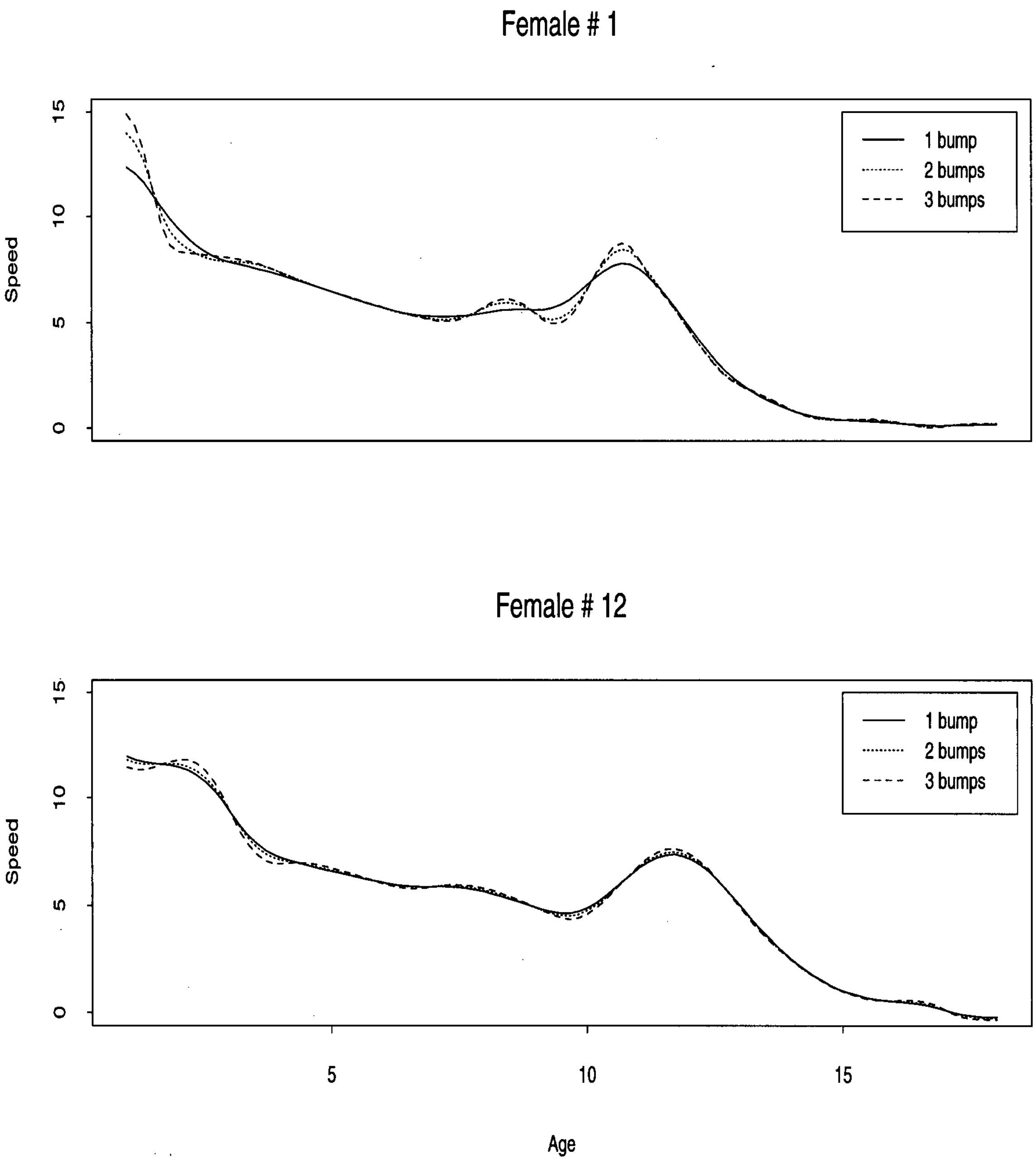


Figure 5.21: Estimates of the speed of growth of females with one bump.

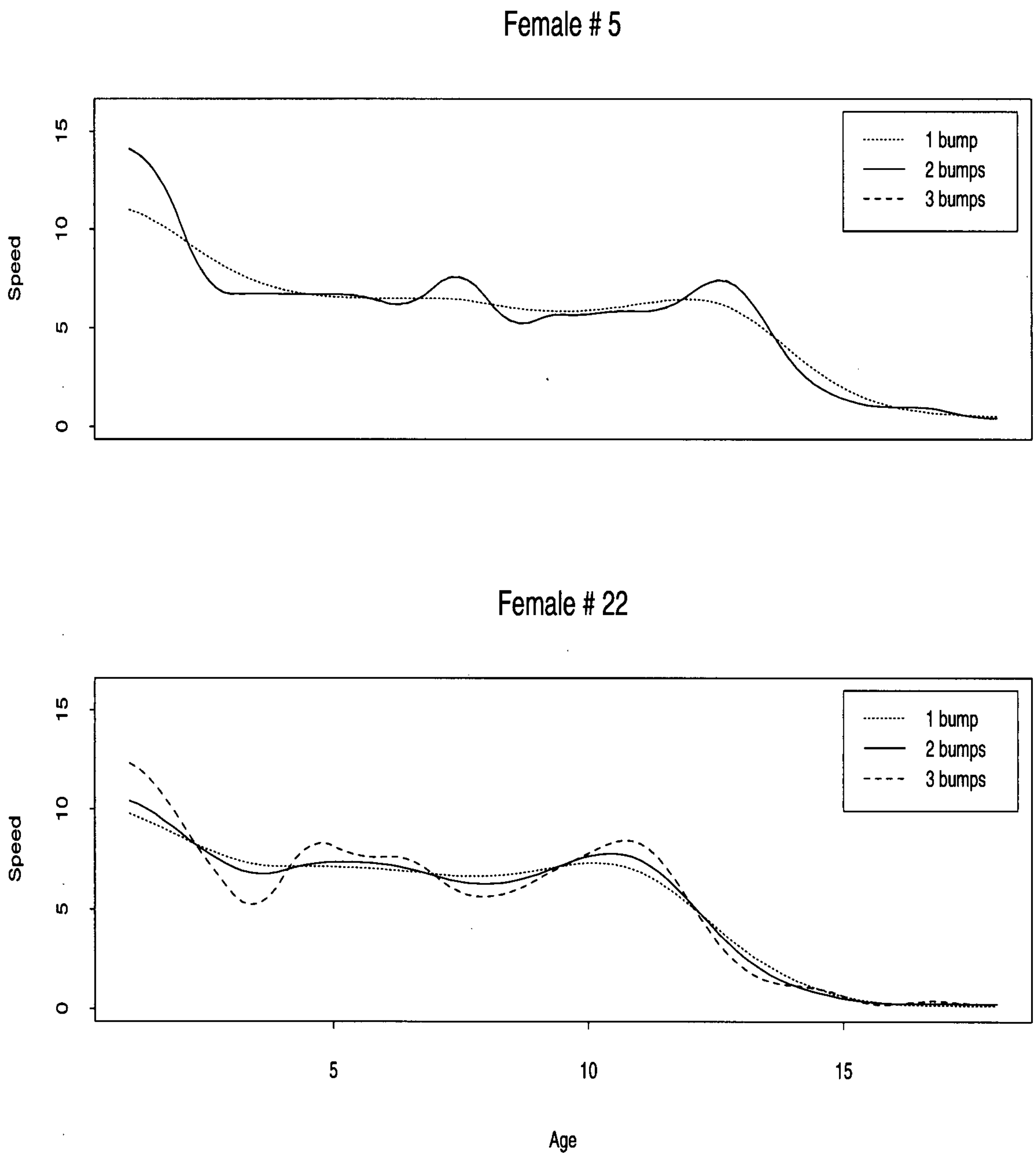
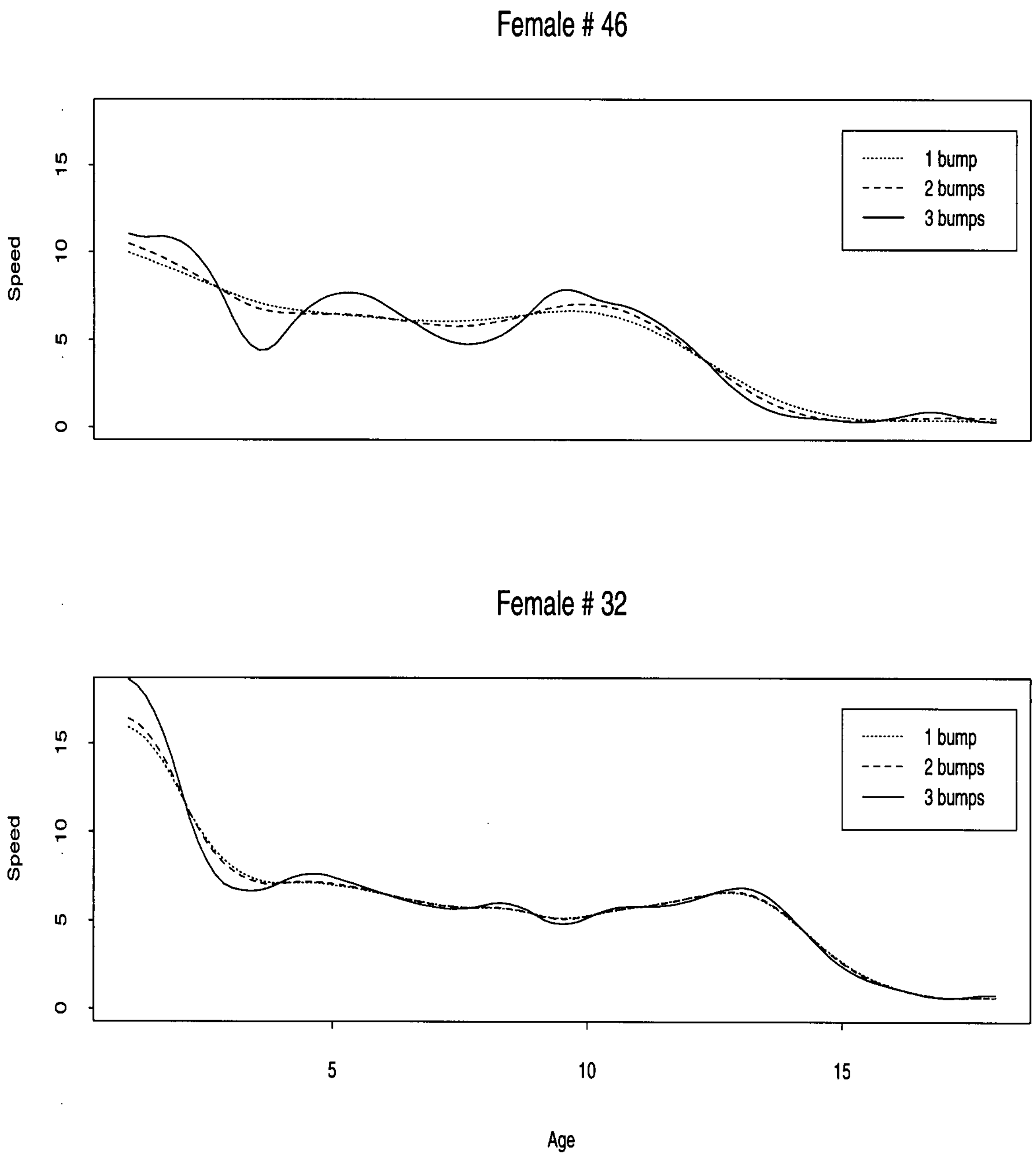


Figure 5.22: Estimates of the speed of growth of females with two bumps.



Female # 46

Female # 32

Figure 5.23: Estimates of the speed of growth of females with three bumps.

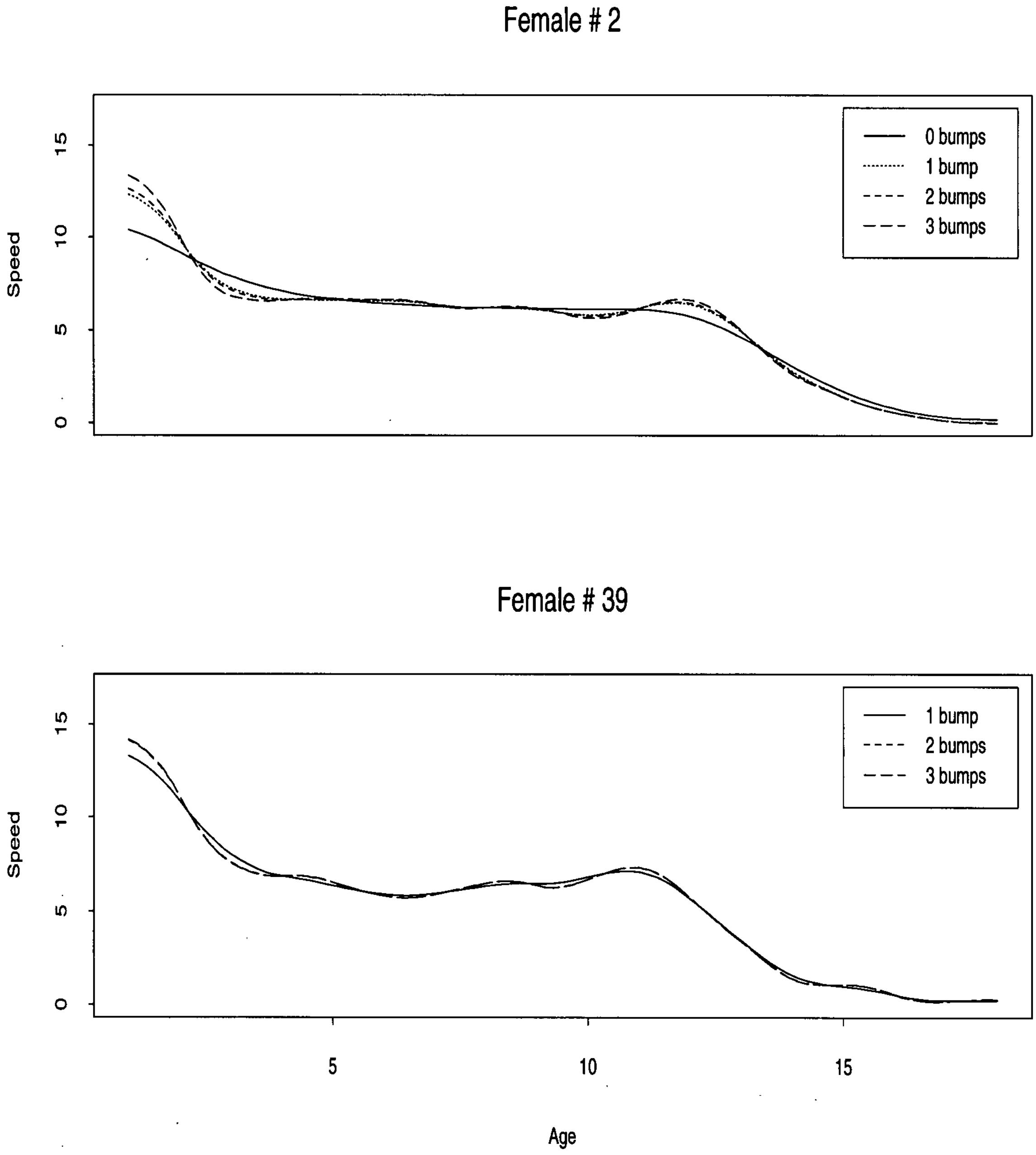


Figure 5.24: Estimates of the speed of growth of females when the decision could not be reached according to the "strict" criterion.

Table 5.13: p-value estimates based on 500 bootstrap samples for the female data using 3-max as a definition of a bump in the testing procedure.

	$\boldsymbol{H_0}$ **: # of bumps** $\boldsymbol{\leq k}$				
Females	$k=0$	$k=1$	$k=2$	$k=3$	**Conclusion**
1	0.000	0.872	0.998	0.994	1
2	0.586	0.630	0.116	0.030	**0**,1,2,3,>3
3	0.014	0.496	0.664	0.902	1
4	0.000	0.138	0.676	0.672	2
5	0.000	0.022	0.998	0.956	2
6	0.000	0.152	0.002	0.682	**1**,2,3
7	0.000	0.126	0.008	0.058	>3
8	0.000	0.388	0.722	0.884	1
9	0.000	0.426	0.420	0.606	1
10	0.000	0.320	0.508	1.000	1
11	0.000	0.214	0.168	0.102	**1**,2,3,>3
12	0.000	0.510	0.250	0.476	1
13	0.000	0.150	0.546	0.448	2
14	0.000	0.166	0.184	0.606	1
15	0.014	0.012	1.000	0.976	2
16	0.000	0.376	0.348	0.530	1
17	0.000	0.290	0.986	0.848	1
18	0.000	0.006	0.982	0.998	2
19	0.000	0.306	0.110	0.998	**1**,2,3
20	0.000	0.716	0.428	0.380	1
21	0.000	0.020	0.888	0.656	2
22	0.010	0.126	0.194	0.546	2

Table 5.13: *(continued)*

Females	$k = 0$	$k = 1$	$k = 2$	$k = 3$	**Conclusion**
23	0.000	0.006	0.390	1.000	2
24	0.000	0.008	0.772	0.324	2
25	0.000	0.014	0.984	0.948	2
26	0.000	0.144	0.000	0.470	3
27	0.000	0.918	0.816	1.000	1
28	0.000	0.080	0.094	0.162	3
29	0.000	0.042	0.170	0.018	**2**,3,>3
30	0.000	0.138	0.266	0.332	2
31	0.002	0.000	0.060	0.826	3
32	0.000	0.126	0.020	0.182	3
33	0.000	0.686	0.932	0.926	1
34	0.000	0.330	0.102	0.026	**1**,2,3,>3
35	0.000	0.042	0.172	0.954	2
36	0.000	0.088	0.148	0.026	>3
37	0.000	0.078	0.994	0.942	2
38	0.020	0.734	0.658	0.396	1
39	0.000	0.230	0.294	0.034	**1**,2,3,>3
40	0.000	0.008	0.210	0.112	2,3,>3
41	0.000	0.452	0.974	0.944	1
42	0.000	0.004	0.508	0.104	**2**,3,>3
43	0.000	0.218	0.080	0.902	**1**,2,3
44	0.000	0.148	0.984	0.970	2
45	0.000	0.340	0.092	0.002	**1**,2,3,>3
46	0.000	0.028	0.016	0.724	3

Table 5.13: *(continued)*

Females	$k = 0$	$k = 1$	$k = 2$	$k = 3$	**Conclusion**
47	0.000	0.634	0.548	0.898	1
48	0.000	0.086	0.778	0.958	2
49	0.010	0.728	0.242	0.796	1
50	0.000	0.804	0.652	0.938	1

Chapter 6

Conclusions

In this thesis, the new, smoothing parameter based test $\mathcal{C}$ri$\mathcal{SP}$ for the number of bumps in regression functions and their derivatives has been proposed. Bootstrap methods have been used to assess its performance. Simulation studies indicated that the $\mathcal{C}$ri$\mathcal{SP}$ test was successful in all considered setups. The behaviour of the test was better when the bump was of a bigger size and not situated on a steep slope of a regression function. As mentioned in Chapter 5, the significance level that should be chosen for $\mathcal{C}$ri$\mathcal{SP}$ is between 10% and 20%. At low significance levels ($\alpha < 0.05$) the power of the proposed test drops substantially, especially for the more challenging regression functions and their derivatives.

$\mathcal{C}$ri$\mathcal{SP}$ was applied to the Berkeley growth data, to detect the number of spurts of children's growth. Most of the results obtained from the completely data-driven test agree with what one might decide "by eye", by looking at an appropriately smoothed derivative estimate. When we compared our test with the results of Ramsay et.al. (1995), we arrived at very similar conclusions for the male data. For the female data, we detected spurts of growth on more occasions than Ramsay et.al. did.

It should be noted that the procedure introduced in this thesis is complicated, and many issues regarding it are still unresolved. Throughout the simulation studies, we obtained a few surprising results. For instance, in the simulations involving the first derivative of the regression function, for some models we got better results with a higher level of noise added to the true regression function than with a lower level of noise. Also, we were able to detect a small bump on a flat portion of the curve more often than for a large bump on a steep slope.

In the introduced procedure, the residuals were calculated using the GCV criterion for the smoothing parameter choice in spline smoothing. This criterion chooses the "optimal" smoothing parameter for the estimation of the function $m(\cdot)$. However, when we test for the number of bumps in the derivative of $m(\cdot)$, we could probably improve the performance of $\mathcal{C}$ri$\mathcal{SP}$ if we used a procedure tailored to choosing the "best" smoothing parameter for $m'(\cdot)$, rather than for $m(\cdot)$.

Another issue of interest is the definition of a bump. We defined it in terms of the values of the original function $m(\cdot)$. However, we could define it as a zero down-crossing of the derivative of m.

When we define a bump as (5.5), we encounter the unpleasant property of the number of bumps being a non-monotone function of a smoothing parameter. It causes some philosophical problems in the testing procedure. This property is not necessary for $\mathcal{C}$ri$\mathcal{SP}$ to work, but without the monotonicity property we cannot provide the intuitive reasons behind our test.

A modification of $\mathcal{C}$ri$\mathcal{SP}$ to deal with correlated or heteroscedastic errors is worth pursuing. This adjustment would most likely involve a new estimate of $\hat{m}_{unr}$ and a different method of bootstrapping the residuals. A version of $\mathcal{C}$ri$\mathcal{SP}$ for higher dimensional data would also be useful.

Bibliography

[1] Babaud, J., Witkin, A.P., Baudin, M., and Duda, R.O. (1986), "Uniqueness of the Gaussian Kernel for Scale-Space Filtering", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* **PAMI-8** 26-33.

[2] Bahadur, R.R. and Savage, L.J. (1956), "The Nonexistence of Certain Statistical Procedures in Nonparametric Problems", *Annals of Mathematical Statistics,* **27**, 1115-1122.

[3] Bowman, A.W., Jones, M.C. and Gijbels, I. (1998), "Testing Monotonicity of Regression", *Journal of Computational and Graphical Statistics*, to appear.

[4] Chaudhuri, P. and Marron, J.S. (1997), "SiZer for Exploration of Structures in Curves", unpublished manuscript.

[5] Cox, D.R. (1966), "Notes on the Analysis of Mixed Frequency Distributions", *British Journal of Mathematical and Statistical Psychology,* **19**, 39-47.

[6] Donoho, D.L. (1988), "One-sided Inference about Functionals of a Density", *The Annals of Statistics,* **16**, 1390-1420.

[7] Efron, B. and Tibshirani, R. (1993), *An Introduction to the Bootstrap*, Chapman and Hall.

[8] Fan, J., and Gijbels, I. (1995), "Data-driven Bandwidth Selection in Local Polynomial Fitting: Variable Bandwidth and Spatial Adaptation", *Journal of the Royal Statistical Society, Series B,* **57**, 371-394.

[9] Fan, J., and Gijbels, I. (1996), *Local Polynomial Modelling and Its Applications,* Chapman and Hall.

[10] Fan, Jianqing (1992), "Design-adaptive Nonparametric Regression", *Journal of the American Statistical Association,* **87**, 998-1004.

[11] Fisher, N.I., Mammen, E., and Marron, J.S. (1994), "Testing for Multimodality", *Computational Statistics & Data Analysis,* **18**, 499-512.

[12] Gasser, T., Kneip, A. and Köhler, W. (1991), "A Flexible and Fast Method for Automatic Smoothing", *Journal of the American Statistical Association,* **86**, 643-652.

[13] Gasser, T. and Müller, H.-G. (1979), "Kernel Estimation of Regression Functions", in *Smoothing Techniques for Curve Estimation* (eds. T. Gasser and M. Rosenblatt), Springer-Verlag.

[14] Gasser, T., Müller, H.-G., Köhler, W., Molinari L. and Prader A. (1984a), "Nonparametric Regression Analysis of Growth Curves", *The Annals of Statistics,* **12**, 210-229.

[15] Gasser, T., Köhler, W., Müller, H.-G., Kneip, A., Largo, R., Molinari L. and Prader A. (1984b), "Velocity and Acceleration of Height Growth Using Kernel Estimation", *The Annals of Human Biology,* **11**, 397-411.

[16] Gasser, T., Müller, H.-G., Köhler, W., Prader A., Largo, R. and Molinari L. (1985), "An Analysis of the Mid-growth and Adolescent Spurts of Height Based on Acceleration", *The Annals of Human Biology,* **12**, 129-148.

[17] Gasser, T., Kneip, A., Ziegler, P., Largo, R. and Prader A. (1990), "A Method for Determining the Dynamics and Intensity of Average Growth", *The Annals of Human Biology,* **17**, 459-474.

[18] Gasser, T., Kneip, A., Binding, A., Prader A. and Molinari L. (1991a), "The Dynamics of Linear Growth in Distance, Velocity and Acceleration", *The Annals of Human Biology,* **18**, 187-205.

[19] Gasser, T., Kneip, A., Ziegler, P., Largo, R., Molinari L. and Prader A. (1991b), "The Dynamics of Growth of Width in Distance, Velocity and Acceleration", *The Annals of Human Biology,* **18**, 449-461.

[20] Good, I.J. and Gaskins, R.A. (1980), "Density Estimation and Bump-Hunting by the Penalized Likelihood Method Exemplified by Scattering and Meteorite Data", *Journal of the American Statistical Association,* **75**, 42-56.

[21] Green P.J. and Silverman, B.W. (1994), *Nonparametric Regression and Generalized Linear Models - A Roughness Penalty Approach*, Chapman and Hall.

[22] Hall, P., Sheather, S.J., Jones, M.C. and Marron, J.S. (1991), "On Optimal Data-based Bandwidth Selection in Kernel Density Estimation", *Biometrika,* **78**, 263-271.

[23] Härdle, W., and Marron, J.S. (1985), "Optimal Bandwidth Selection in Nonparametric Regression Function Estimation", *The Annals of Statistics* **13**, 1465-1481.

[24] Härdle, W. (1989), *Applied Nonparametric Regression*, SIAM Cambridge University Press.

[25] Härdle, W. (1991), *Smoothing Techniques with Implementation in S*, Springer-Verlag.

[26] Hartigan, J.A. and Hartigan, P.M. (1985), "The Dip Test of Unimodality", *The Annals of Statistics*, **13**, 70-84.

[27] Hartigan, J.A. and Mohanty, S. (1992), "The RUNT Test for Multimodality", *Journal of Classification*, **9**, 63-70.

[28] Hastie, T.J. and Loader, C. (1993), "Local regression: Automatic Carpentry", *Statistical Science*, **8**, 120-143.

[29] Hastie, T.J. and Tibshirani, R.J. (1990), *Generalized Additive Models*, Chapman and Hall.

[30] Heckman, N.E. (1992), "Bump hunting in regression analysis", *Statistics and Probability Letters*, **14**, 141-152.

[31] Heckman, N.E. (1997), "The Theory and Application of Penalized Least Squares Methods or Reproducing Kernel Hilbert Spaces Made Easy", unpublished manuscript.

[32] Heckman, N.E. and Ramsay, J.O. (1996), "Penalized Regression with Model-Based Penalties", unpublished manuscript.

[33] Izenman, A.J., and Sommer, C.J. (1988), "Philatelic Mixtures and Multimodal Densities", *Journal of the American Statistical Association,* **83**, 941-953.

[34] Largo, R.H., Gasser, Th., Prader, A., Stuetzle, W. and Huber, P.J. (1978), "Analysis of the Adolescent Growth Spurt using Smoothing Spline Functions", *Annals of Human Biology*, **5**, 421-434.

[35] Mammen, E. (1991), "Estimating a Smooth Monotone Regression Function", *The Annals of Statistics*, **19**, 724-740.

[36] Mammen, E. (1991), "Nonparametric Regression Under Qualitative Smoothness Assumptions", *The Annals of Statistics*, **10**, 1217-1223.

[37] Mammen, E., and Marron, J.S. (1995), "Mass Recentered Kernel Smoothers", *Biometrika*, **84**, 765 - 778.

[38] Mammen, E., Marron, J.S., and Fisher, N.I. (1992), "Some Asymptotics for Multimodality Tests based on Kernel Density Estimates", *Probability Theory and Related Fields*, **91**, 115-132.

[39] Marron, J.S. and Chu, C. K. (1991), "Choosing a Kernel Regression Estimator", *Statistical Science*, **6**, 404-436, (with discussion).

[40] Minnotte, M.C. (1997), "Nonparametric Testing of the Existence of Modes", *The Annals of Statistics*, **25**, 1646-1660.

[41] Minnotte, M.C. and Scott, D.W. (1993), "The Mode Tree: A Tool for Visualization of Nonparametric Density Features", *Journal of Computational and Graphical Statistics*, **2**, 51-68.

[42] Mukerjee, H. (1988), "Monotone Nonparametric Regression", *The Annals of Statistics*, **16**, 741-750.

[43] Müller, D.W. (1992), "The Excess Mass Approach in Statistics", Preprint Nr.3, December 1992, Beiträge zur Statistik, Universität Heidelberg.

[44] Müller, D.W. and Sawitzki, G. (1991), "Excess Mass Estimates and Tests for Multimodality", *Journal of the American Statistical Association,* **86**, 738-746.

[45] Ramsay, J.O. (1996), "Estimating a Smooth Monotone Function", *Journal of the Royal Statistical Society, Series B*, **60**, 365-375.

[46] Ramsay, J.O., Altman, N. and Bock, R.D. (1994), "Variation in Height acceleration in the Fels Growth Data", *Canad. J. Statist.*, **22**, 89-102.

[47] Ramsay, J.O., Bock, R.D. and Gasser, T. (1994), "Comparison of Height Acceleration Curves in the Fels, Zurich, and Berkeley Growth Data", *Annals of Human Biology*, **22**, 413-426.

[48] Rice, J. (1984), "Bandwidth Choice for Nonparametric Regression", *The Annals of Statistics*, **12**, 1215-1230.

[49] Ruppert, D., Sheather, S.J., and Wand, M.P. (1993), "An Effective Bandwidth Selector for Local Least Squares Regression", *Journal of the American Statistical Association*, **90**, 522-534.

[50] Sain, S.R. (1994), "Adaptive Kernel Density Estimation", Ph.D. Thesis, Dept. of Statistics, Rice University.

[51] Schlee, W. (1982), "Nonparametric Tests of the Monotony and Convexity of Regression", in *Nonparametric Statistical Inference, vol. II* (eds: B.V. Gnedenko, M.L. Puri & I. Vincze), Amsterdam: North-Holland, 823-836.

[52] Silverman, B.W. (1981), "Using Kernel Density Estimates to investigate Multimodality", *Journal of Royal Statistical Society, Series B*, **43**, 97-99.

[53] Silverman, B.W. (1983), "Some Properties of a Test for Multimodality Based on Kernel Density Estimates", in London Mathematical Society Lecture Note

Series # 79 *Probability, Statistics and Analysis* (eds: J.F.C. Kingman & G.E.H. Reuter), 248-259.

[54] Silverman, B.W. (1986), *Density Estimation for Statistics and Data Analysis*, Chapman and Hall.

[55] Tuddenham, R.D., and Snyder, M.M. (1954), "Physical Growth of California Boys and Girls form Birth to Eighteen Years", *University of California Publications in Child Development*, **1**, 183-364.

[56] Wahba, G. (1990), *Spline Models for Observational Data*, SIAM.

[57] Wand, M.P. and Jones, M.C. (1995), *Kernel Smoothing*, Chapman and Hall.

[58] Wong, M.A. (1985), "A Bootstrap Testing Procedure for Investigating the Number of Subpopulations", *J. Statist. Comput. Simul.*, **22**, 99-112

[59] Woodroofe, M. and Sun, J. (1997), "Testing Uniformity versus a Monotone Density", submitted to *The Annals of Statistics*

Appendix A

Male results

In Table A.1 the results of the bootstrap procedure are summarized. We show the p-values of all the tests performed with different definitions of a bump (2-, 3-, 4-, and 5-max), and two smoothing methods used in steps 1 and 5 of the bootstrap procedure on page 39: Lspline and Kernel smoothing. In step 2 of the bootstrap procedure, we always used the Kernel smoothing.

Also, we exhibit here the estimated growth curves for all 43 males. Lspline smoothing is used to obtain these estimates. When the "strict" decision rule gives us the classification, we plot only the estimate with the value of a smoothing parameter being equal to the respective $\lambda_{crit}^{(b)}$. In 9 cases with no decision made according to the "strict" rule, we plot the estimates with the values of b arrived at using the range rule and we indicate the "soft" classification by a solid line.

Table A.1: Summary of 500 bootstrap samples for the male growth data.

			H_0 : # of bumps $\leq k$			
Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
1	2-max	Lspline	0.000	0.002	0.962	0.992
		Kernel	0.022	0.120	0.948	0.778
	3-max	Lspline	0.000	0.002	0.938	0.994
		Kernel	0.014	0.096	0.934	0.720
	4-max	Lspline	0.000	0.004	0.936	0.996
		Kernel	0.014	0.076	0.868	0.628
	5-max	Lspline	0.000	0.000	0.908	0.944
		Kernel	0.012	0.056	0.992	0.946
2	2-max	Lspline	0.000	0.930	0.632	0.372
		Kernel	0.006	0.874	0.610	0.266
	3-max	Lspline	0.000	0.908	0.680	0.316
		Kernel	0.004	0.882	0.638	0.274
	4-max	Lspline	0.000	0.914	0.738	0.382
		Kernel	0.004	0.862	0.628	0.264
	5-max	Lspline	0.000	0.934	0.632	0.556
		Kernel	0.000	0.876	0.650	0.274
3	2-max	Lspline	0.000	0.082	0.436	0.726
		Kernel	0.000	0.226	0.388	0.948
	3-max	Lspline	0.000	0.130	0.380	0.718
		Kernel	0.000	0.222	0.328	0.944
	4-max	Lspline	0.000	0.108	0.372	0.624
		Kernel	0.000	0.242	0.344	0.960
	5-max	Lspline	0.000	0.148	0.432	0.614
		Kernel	0.000	0.218	0.398	0.836

Table A.1: *(continued)*

Male	Width	Method	$k=0$	$k=1$	$k=2$	$k=3$
4	2-max	Lspline	0.000	0.068	0.300	0.998
		Kernel	0.008	0.052	0.802	0.418
	3-max	Lspline	0.000	0.068	0.274	0.996
		Kernel	0.004	0.058	0.802	0.404
	4-max	Lspline	0.000	0.072	0.242	0.992
		Kernel	0.000	0.056	0.836	0.484
	5-max	Lspline	0.000	0.104	0.226	0.958
		Kernel	0.000	0.088	0.832	0.482
5	2-max	Lspline	0.000	0.352	0.106	0.884
		Kernel	0.020	0.330	0.096	0.862
	3-max	Lspline	0.000	0.310	0.094	0.884
		Kernel	0.014	0.322	0.100	0.866
	4-max	Lspline	0.000	0.296	0.082	0.874
		Kernel	0.008	0.328	0.094	0.856
	5-max	Lspline	0.000	0.264	0.106	0.848
		Kernel	0.008	0.330	0.096	0.912
6	2-max	Lspline	0.000	0.000	0.138	0.632
		Kernel	0.000	0.000	0.182	0.550
	3-max	Lspline	0.000	0.000	0.126	0.560
		Kernel	0.000	0.000	0.158	0.590
	4-max	Lspline	0.000	0.000	0.130	0.924
		Kernel	0.000	0.000	0.162	0.588
	5-max	Lspline	0.000	0.000	0.130	0.864
		Kernel	0.000	0.000	0.176	0.906

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
7	2-max	Lspline	0.000	0.016	0.136	0.500
		Kernel	0.000	0.118	0.056	0.842
	3-max	Lspline	0.000	0.010	0.114	0.558
		Kernel	0.000	0.098	0.038	0.832
	4-max	Lspline	0.000	0.014	0.114	0.444
		Kernel	0.000	0.080	0.036	0.826
	5-max	Lspline	0.000	0.014	0.100	0.524
		Kernel	0.000	0.046	0.024	0.918
8	2-max	Lspline	0.000	0.630	0.492	0.996
		Kernel	0.030	0.510	0.348	0.994
	3-max	Lspline	0.000	0.590	0.442	0.998
		Kernel	0.024	0.506	0.328	0.998
	4-max	Lspline	0.000	0.554	0.396	0.998
		Kernel	0.010	0.518	0.318	1.000
	5-max	Lspline	0.000	0.528	0.456	1.000
		Kernel	0.012	0.510	0.370	0.940
9	2-max	Lspline	0.000	0.000	0.882	0.466
		Kernel	0.000	0.016	0.676	0.838
	3-max	Lspline	0.000	0.000	0.864	0.454
		Kernel	0.000	0.008	0.830	0.842
	4-max	Lspline	0.000	0.000	0.916	0.474
		Kernel	0.000	0.004	0.950	0.818
	5-max	Lspline	0.000	0.000	0.918	0.948
		Kernel	0.000	0.004	0.970	0.714

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
10	2-max	Lspline	0.000	0.884	0.522	0.520
		Kernel	0.030	0.910	0.698	0.416
	3-max	Lspline	0.000	0.862	0.478	0.476
		Kernel	0.020	0.914	0.710	0.454
	4-max	Lspline	0.000	0.854	0.536	0.420
		Kernel	0.014	0.894	0.626	0.472
	5-max	Lspline	0.000	0.854	0.550	0.468
		Kernel	0.010	0.886	0.624	0.614
11	2-max	Lspline	0.000	0.000	0.258	0.636
		Kernel	0.000	0.000	0.290	0.094
	3-max	Lspline	0.000	0.000	0.220	0.588
		Kernel	0.000	0.000	0.290	0.262
	4-max	Lspline	0.000	0.000	0.166	0.696
		Kernel	0.000	0.000	0.302	0.408
	5-max	Lspline	0.000	0.000	0.178	0.630
		Kernel	0.000	0.000	0.278	0.824
12	2-max	Lspline	0.000	0.042	0.428	0.152
		Kernel	0.004	0.018	0.350	0.064
	3-max	Lspline	0.000	0.030	0.460	0.310
		Kernel	0.000	0.014	0.434	0.134
	4-max	Lspline	0.000	0.026	0.518	0.270
		Kernel	0.000	0.014	0.440	0.562
	5-max	Lspline	0.000	0.012	0.558	0.272
		Kernel	0.000	0.012	0.388	0.670

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
13	2-max	Lspline	0.000	0.024	0.738	0.818
		Kernel	0.000	0.030	0.884	0.504
	3-max	Lspline	0.000	0.012	0.798	0.802
		Kernel	0.000	0.028	0.764	0.438
	4-max	Lspline	0.000	0.016	0.738	0.706
		Kernel	0.000	0.036	0.782	0.342
	5-max	Lspline	0.000	0.024	0.762	0.628
		Kernel	0.000	0.038	0.782	0.406
14	2-max	Lspline	0.000	0.034	0.184	0.092
		Kernel	0.000	0.054	0.002	0.062
	3-max	Lspline	0.000	0.012	0.162	0.126
		Kernel	0.000	0.050	0.006	0.050
	4-max	Lspline	0.000	0.014	0.140	0.080
		Kernel	0.000	0.046	0.044	0.036
	5-max	Lspline	0.000	0.018	0.106	0.034
		Kernel	0.000	0.048	0.086	0.036
15	2-max	Lspline	0.000	0.000	0.976	0.998
		Kernel	0.004	0.038	0.974	0.890
	3-max	Lspline	0.000	0.000	0.984	0.992
		Kernel	0.004	0.022	0.980	0.886
	4-max	Lspline	0.000	0.000	0.990	0.984
		Kernel	0.000	0.014	0.978	0.864
	5-max	Lspline	0.000	0.000	0.984	0.918
		Kernel	0.000	0.008	0.920	0.688

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
16	2-max	Lspline	0.000	0.104	0.150	0.118
		Kernel	0.034	0.150	0.014	0.072
	3-max	Lspline	0.000	0.088	0.148	0.134
		Kernel	0.032	0.130	0.014	0.070
	4-max	Lspline	0.000	0.066	0.130	0.150
		Kernel	0.028	0.096	0.034	0.066
	5-max	Lspline	0.000	0.054	0.084	0.086
		Kernel	0.020	0.066	0.060	0.040
17	2-max	Lspline	0.000	0.774	0.282	0.318
		Kernel	0.000	0.708	0.236	0.800
	3-max	Lspline	0.000	0.830	0.292	0.384
		Kernel	0.000	0.726	0.238	0.654
	4-max	Lspline	0.000	0.860	0.332	0.316
		Kernel	0.000	0.740	0.250	0.464
	5-max	Lspline	0.000	0.806	0.348	0.540
		Kernel	0.000	0.706	0.198	0.938
18	2-max	Lspline	0.000	0.382	0.010	0.242
		Kernel	0.002	0.480	0.054	0.140
	3-max	Lspline	0.000	0.380	0.010	0.224
		Kernel	0.000	0.492	0.064	0.158
	4-max	Lspline	0.000	0.440	0.036	0.238
		Kernel	0.000	0.446	0.096	0.136
	5-max	Lspline	0.000	0.478	0.058	0.270
		Kernel	0.000	0.466	0.048	0.146

Table A.1: *(continued)*

Male	Width	Method	$k=0$	$k=1$	$k=2$	$k=3$
19	2-max	Lspline	0.000	0.204	0.654	0.792
		Kernel	0.000	0.020	0.950	0.578
	3-max	Lspline	0.000	0.154	0.706	0.718
		Kernel	0.000	0.018	0.944	0.616
	4-max	Lspline	0.000	0.110	0.786	0.602
		Kernel	0.000	0.018	0.930	0.550
	5-max	Lspline	0.000	0.076	0.812	0.402
		Kernel	0.000	0.018	0.938	0.456
20	2-max	Lspline	0.000	0.218	0.214	0.032
		Kernel	0.000	0.160	0.012	0.148
	3-max	Lspline	0.000	0.184	0.204	0.036
		Kernel	0.000	0.142	0.014	0.172
	4-max	Lspline	0.000	0.184	0.192	0.026
		Kernel	0.000	0.138	0.018	0.148
	5-max	Lspline	0.000	0.206	0.126	0.030
		Kernel	0.000	0.138	0.014	0.186
21	2-max	Lspline	0.000	0.002	0.574	0.282
		Kernel	0.002	0.022	0.462	0.420
	3-max	Lspline	0.000	0.000	0.532	0.178
		Kernel	0.002	0.018	0.478	0.360
	4-max	Lspline	0.000	0.000	0.462	0.164
		Kernel	0.002	0.018	0.504	0.330
	5-max	Lspline	0.000	0.000	0.444	0.118
		Kernel	0.000	0.016	0.434	0.364

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
22	2-max	Lspline	0.000	0.000	0.548	0.756
		Kernel	0.022	0.002	0.140	0.808
	3-max	Lspline	0.000	0.000	0.472	0.744
		Kernel	0.006	0.000	0.114	0.852
	4-max	Lspline	0.000	0.000	0.446	0.766
		Kernel	0.004	0.000	0.100	0.754
	5-max	Lspline	0.000	0.000	0.390	0.766
		Kernel	0.004	0.000	0.072	0.644
23	2-max	Lspline	0.000	0.794	0.256	0.072
		Kernel	0.000	0.560	0.278	0.080
	3-max	Lspline	0.000	0.750	0.296	0.086
		Kernel	0.000	0.548	0.248	0.082
	4-max	Lspline	0.000	0.790	0.334	0.122
		Kernel	0.000	0.536	0.208	0.066
	5-max	Lspline	0.000	0.754	0.308	0.048
		Kernel	0.000	0.546	0.294	0.104
24	2-max	Lspline	0.000	0.000	0.164	0.382
		Kernel	0.000	0.008	0.222	0.712
	3-max	Lspline	0.000	0.000	0.150	0.484
		Kernel	0.000	0.008	0.214	0.648
	4-max	Lspline	0.000	0.000	0.118	0.360
		Kernel	0.000	0.010	0.206	0.548
	5-max	Lspline	0.000	0.000	0.104	0.486
		Kernel	0.000	0.008	0.204	NA

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
25	2-max	Lspline	0.000	0.000	0.232	0.302
		Kernel	0.052	0.030	0.200	0.404
	3-max	Lspline	0.000	0.000	0.208	0.256
		Kernel	0.012	0.016	0.160	0.360
	4-max	Lspline	0.000	0.000	0.292	0.256
		Kernel	0.000	0.012	0.188	0.360
	5-max	Lspline	0.000	0.000	0.336	0.294
		Kernel	0.000	0.002	0.300	0.334
26	2-max	Lspline	0.000	0.098	0.118	0.062
		Kernel	0.010	0.130	0.230	0.028
	3-max	Lspline	0.000	0.092	0.162	0.066
		Kernel	0.002	0.148	0.246	0.022
	4-max	Lspline	0.000	0.072	0.152	0.056
		Kernel	0.002	0.118	0.198	0.022
	5-max	Lspline	0.000	0.062	0.274	0.042
		Kernel	0.000	0.136	0.208	0.036
27	2-max	Lspline	0.000	0.788	0.758	0.454
		Kernel	0.016	0.918	0.548	0.320
	3-max	Lspline	0.000	0.748	0.798	0.416
		Kernel	0.010	0.922	0.530	0.336
	4-max	Lspline	0.000	0.728	0.798	0.426
		Kernel	0.006	0.920	0.530	0.388
	5-max	Lspline	0.000	0.726	0.702	0.322
		Kernel	0.004	0.918	0.524	0.462

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
28	2-max	Lspline	0.000	0.398	0.112	0.746
		Kernel	0.000	0.522	0.066	0.638
	3-max	Lspline	0.000	0.390	0.078	0.776
		Kernel	0.000	0.516	0.058	0.656
	4-max	Lspline	0.000	0.402	0.076	0.714
		Kernel	0.000	0.540	0.080	0.688
	5-max	Lspline	0.000	0.456	0.068	0.766
		Kernel	0.000	0.548	0.092	0.700
29	2-max	Lspline	0.000	0.000	0.210	0.660
		Kernel	0.000	0.018	0.134	0.106
	3-max	Lspline	0.000	0.000	0.172	0.566
		Kernel	0.000	0.016	0.124	0.230
	4-max	Lspline	0.000	0.000	0.160	0.610
		Kernel	0.000	0.016	0.152	0.478
	5-max	Lspline	0.000	0.000	0.174	NA
		Kernel	0.000	0.016	0.140	0.816
30	2-max	Lspline	0.000	0.006	0.036	0.318
		Kernel	0.008	0.104	0.104	0.446
	3-max	Lspline	0.000	0.010	0.038	0.364
		Kernel	0.002	0.090	0.098	0.470
	4-max	Lspline	0.000	0.012	0.062	0.446
		Kernel	0.002	0.078	0.108	0.444
	5-max	Lspline	0.000	0.026	0.050	0.476
		Kernel	0.000	0.064	0.122	0.426

Table A.1: *(continued)*

Male	Width	Method	$k=0$	$k=1$	$k=2$	$k=3$
31	2-max	Lspline	0.000	0.032	0.738	0.998
		Kernel	0.000	0.064	0.948	0.748
	3-max	Lspline	0.000	0.050	0.686	0.998
		Kernel	0.000	0.064	0.882	0.818
	4-max	Lspline	0.000	0.048	0.674	0.996
		Kernel	0.000	0.064	0.778	0.866
	5-max	Lspline	0.000	0.048	0.640	0.918
		Kernel	0.000	0.072	0.834	0.712
32	2-max	Lspline	0.000	0.114	0.996	0.982
		Kernel	0.000	0.208	0.964	0.960
	3-max	Lspline	0.000	0.152	0.994	0.978
		Kernel	0.000	0.216	0.966	0.974
	4-max	Lspline	0.000	0.136	0.994	0.984
		Kernel	0.000	0.192	0.962	0.972
	5-max	Lspline	0.000	0.176	0.986	0.976
		Kernel	0.000	0.186	0.968	0.900
33	2-max	Lspline	0.000	0.566	0.306	0.546
		Kernel	0.000	0.514	0.080	0.924
	3-max	Lspline	0.000	0.534	0.302	0.484
		Kernel	0.000	0.556	0.082	0.904
	4-max	Lspline	0.000	0.536	0.268	0.342
		Kernel	0.000	0.508	0.086	0.832
	5-max	Lspline	0.000	0.522	0.266	0.402
		Kernel	0.000	0.542	0.096	0.806

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
34	2-max	Lspline	0.000	0.160	0.024	0.264
		Kernel	0.000	0.370	0.040	0.070
	3-max	Lspline	0.000	0.222	0.028	0.286
		Kernel	0.000	0.396	0.034	0.060
	4-max	Lspline	0.000	0.358	0.016	0.234
		Kernel	0.000	0.360	0.040	0.092
	5-max	Lspline	0.000	0.458	0.016	0.352
		Kernel	0.000	0.386	0.048	0.108
35	2-max	Lspline	0.000	0.010	0.886	0.936
		Kernel	0.000	0.010	0.972	0.856
	3-max	Lspline	0.000	0.006	0.856	0.938
		Kernel	0.000	0.010	0.964	0.876
	4-max	Lspline	0.000	0.004	0.804	0.940
		Kernel	0.000	0.008	0.962	0.838
	5-max	Lspline	0.000	0.000	0.788	0.874
		Kernel	0.000	0.010	0.962	0.906
36	2-max	Lspline	0.000	0.268	0.024	0.176
		Kernel	0.068	0.464	0.002	0.082
	3-max	Lspline	0.000	0.296	0.030	0.188
		Kernel	0.064	0.460	0.000	0.090
	4-max	Lspline	0.000	0.328	0.022	0.158
		Kernel	0.060	0.464	0.000	0.054
	5-max	Lspline	0.000	0.378	0.016	0.156
		Kernel	0.056	0.464	0.002	0.066

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
37	2-max	Lspline	0.000	0.000	0.038	0.598
		Kernel	0.000	0.038	0.116	0.316
	3-max	Lspline	0.000	0.002	0.034	0.516
		Kernel	0.000	0.038	0.112	0.312
	4-max	Lspline	0.000	0.002	0.044	0.584
		Kernel	0.000	0.038	0.116	0.354
	5-max	Lspline	0.000	0.002	0.040	0.458
		Kernel	0.000	0.040	0.104	0.234
38	2-max	Lspline	0.000	0.000	0.014	0.390
		Kernel	0.000	0.000	0.012	0.360
	3-max	Lspline	0.000	0.000	0.018	0.400
		Kernel	0.000	0.000	0.016	0.294
	4-max	Lspline	0.000	0.000	0.014	0.478
		Kernel	0.000	0.000	0.012	0.326
	5-max	Lspline	0.000	0.000	0.016	0.400
		Kernel	0.000	0.000	0.018	0.356
39	2-max	Lspline	0.000	0.118	0.492	0.994
		Kernel	0.000	0.138	0.504	0.936
	3-max	Lspline	0.000	0.100	0.538	0.994
		Kernel	0.000	0.170	0.432	0.942
	4-max	Lspline	0.000	0.122	0.472	0.986
		Kernel	0.000	0.146	0.410	0.942
	5-max	Lspline	0.000	0.122	0.588	0.884
		Kernel	0.000	0.148	0.476	0.604

Table A.1: *(continued)*

Male	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
40	2-max	Lspline	0.000	0.106	0.694	0.416
		Kernel	0.000	0.076	0.540	0.202
	3-max	Lspline	0.000	0.106	0.658	0.440
		Kernel	0.000	0.092	0.668	0.220
	4-max	Lspline	0.000	0.108	0.644	0.560
		Kernel	0.000	0.080	0.610	0.534
	5-max	Lspline	0.000	0.066	0.558	0.530
		Kernel	0.000	0.090	0.592	0.786
41	2-max	Lspline	0.000	0.694	0.374	0.278
		Kernel	0.002	0.460	0.288	0.506
	3-max	Lspline	0.000	0.824	0.408	0.260
		Kernel	0.002	0.652	0.310	0.432
	4-max	Lspline	0.000	0.784	0.660	0.236
		Kernel	0.000	0.790	0.504	0.442
	5-max	Lspline	0.000	0.820	0.614	0.600
		Kernel	0.000	0.736	0.888	0.682
42	2-max	Lspline	0.000	0.184	0.656	0.572
		Kernel	0.000	0.284	0.114	0.106
	3-max	Lspline	0.000	0.212	0.592	0.488
		Kernel	0.000	0.266	0.262	0.090
	4-max	Lspline	0.000	0.224	0.564	0.504
		Kernel	0.000	0.248	0.372	0.182
	5-max	Lspline	0.000	0.188	0.480	0.640
		Kernel	0.000	0.240	0.336	0.858

Table A.1: *(continued)*

Male	Width	Method	$k=0$	$k=1$	$k=2$	$k=3$
43	2-max	Lspline	0.000	0.012	0.434	0.564
		Kernel	0.000	0.064	0.324	0.422
	3-max	Lspline	0.000	0.028	0.388	0.576
		Kernel	0.000	0.068	0.348	0.410
	4-max	Lspline	0.000	0.020	0.426	0.570
		Kernel	0.000	0.052	0.384	0.384
	5-max	Lspline	0.000	0.032	0.520	0.820
		Kernel	0.000	0.068	0.314	NA

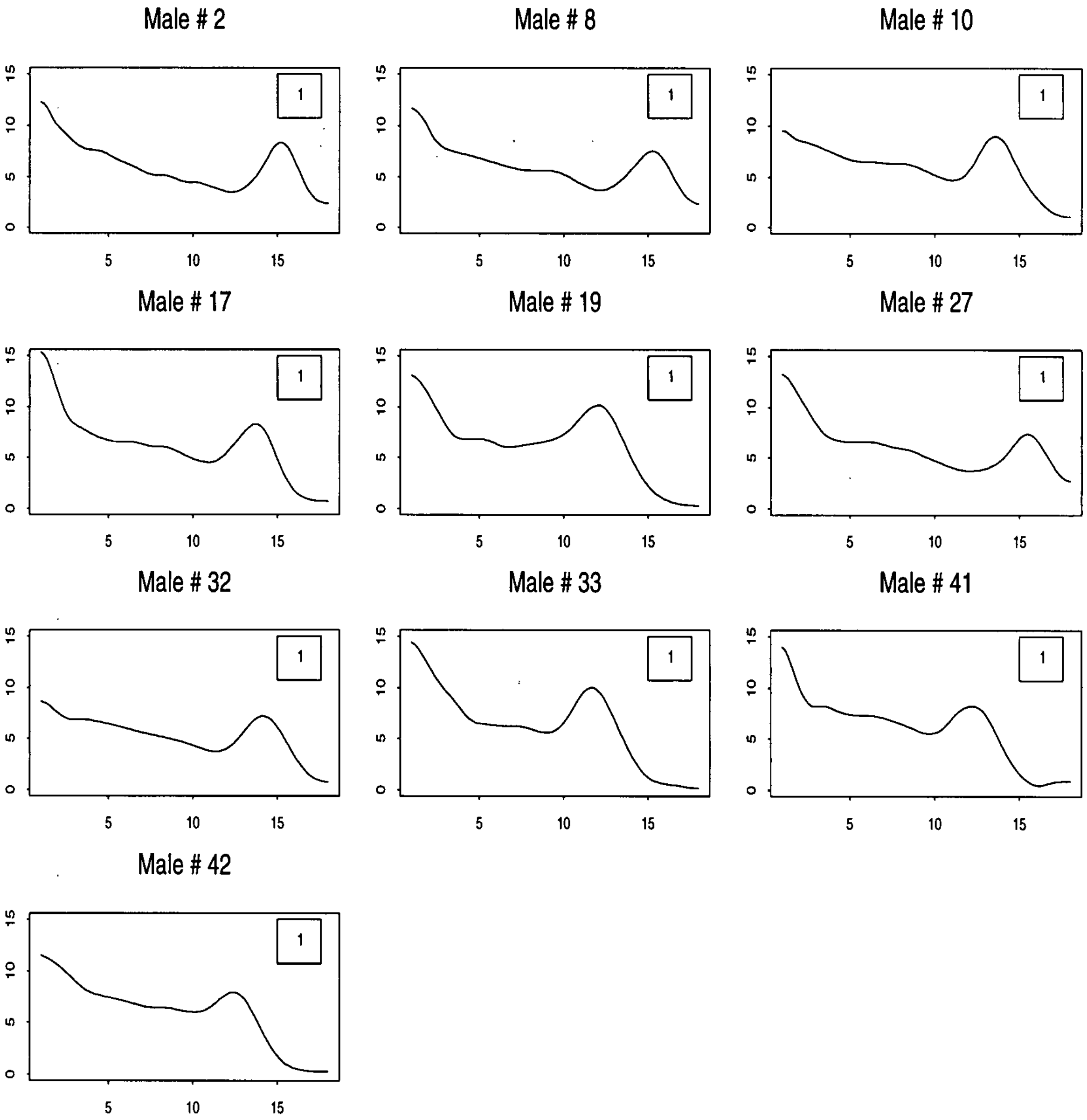


Figure A.1: Plot of male speed curves with 1 bump.

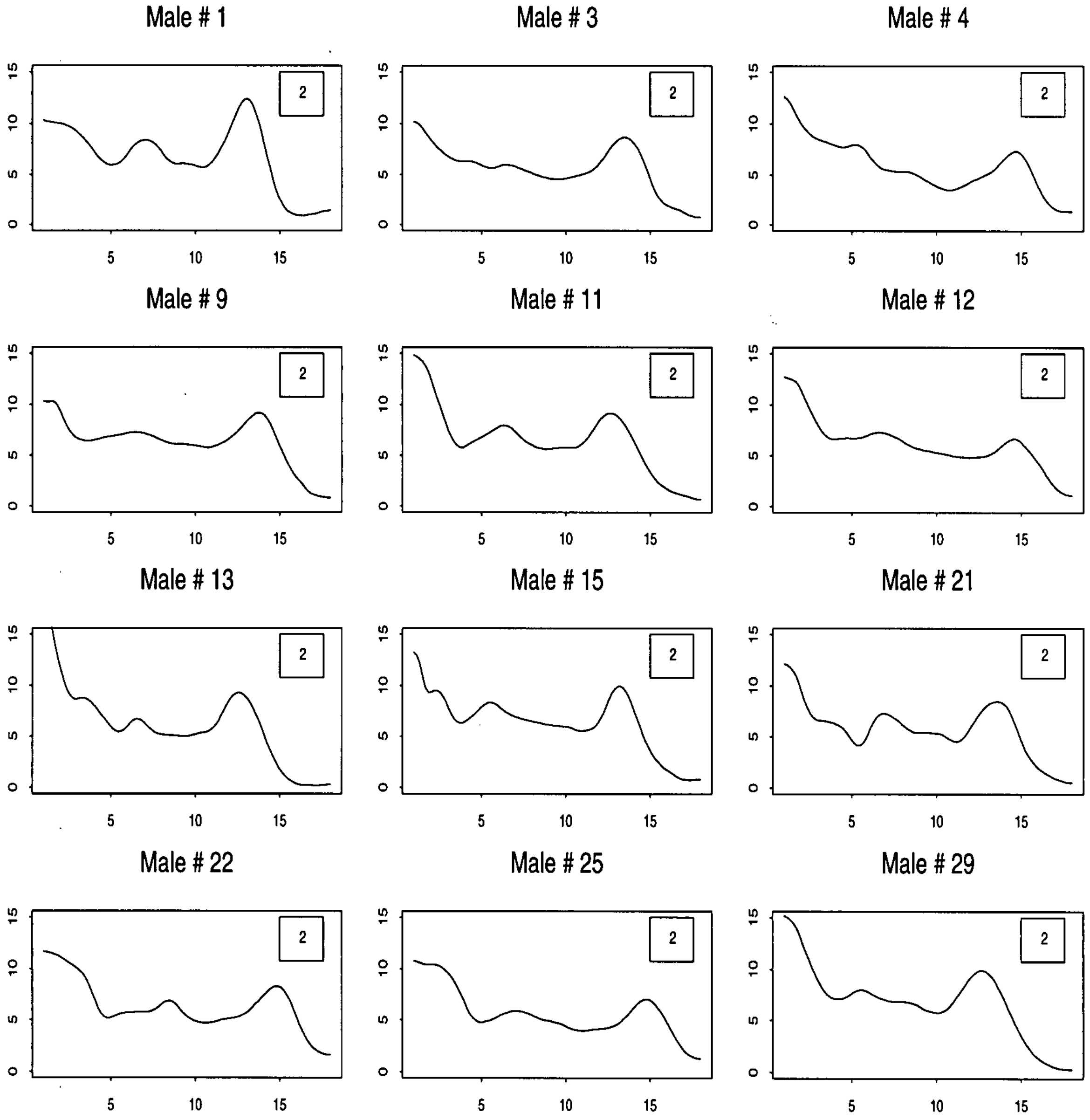


Figure A.2: Plot of male speed curves with 2 bumps.

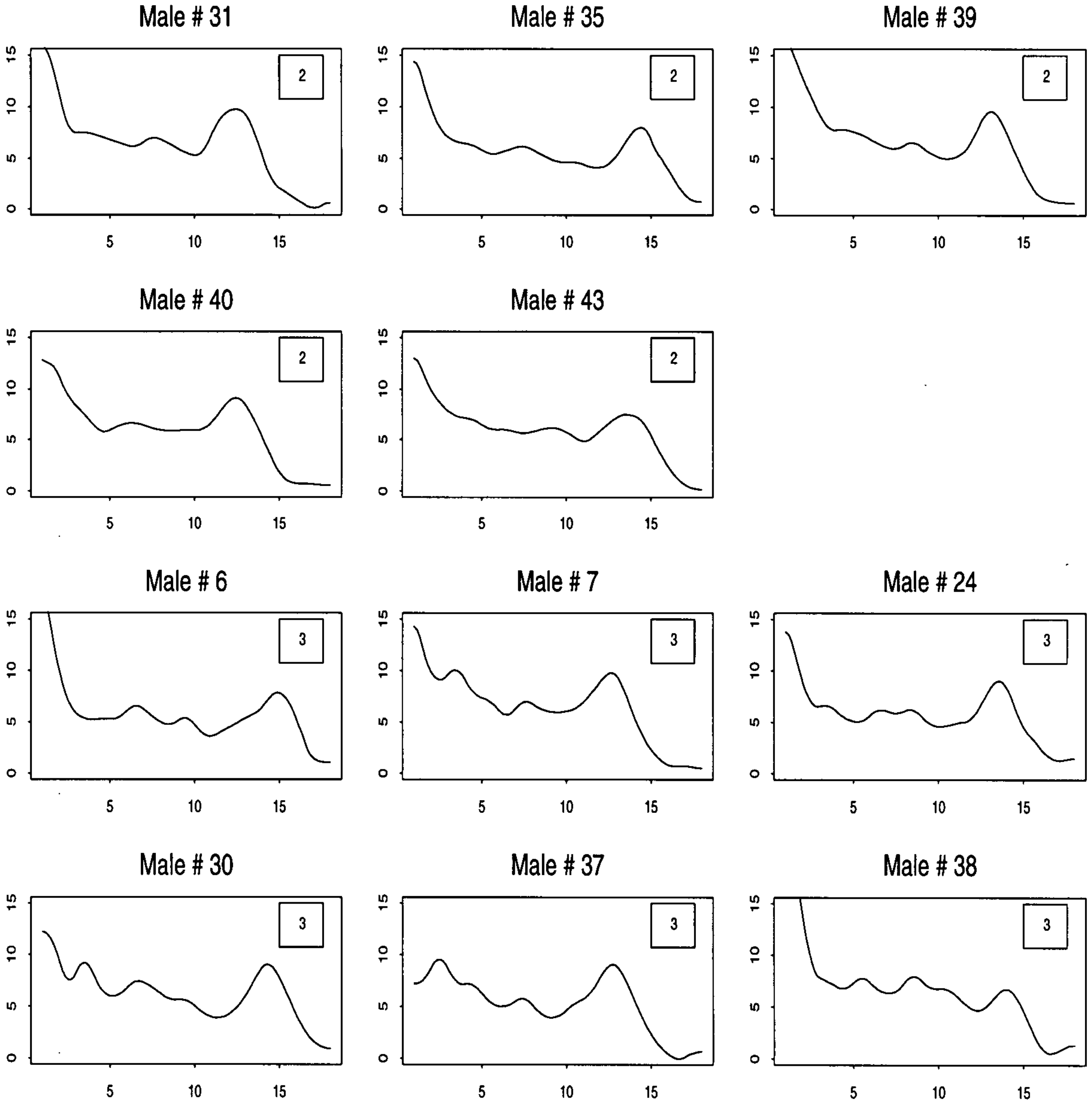


Figure A.3: Plot of male speed curves with 2 bumps (top) and 3 bumps (bottom).

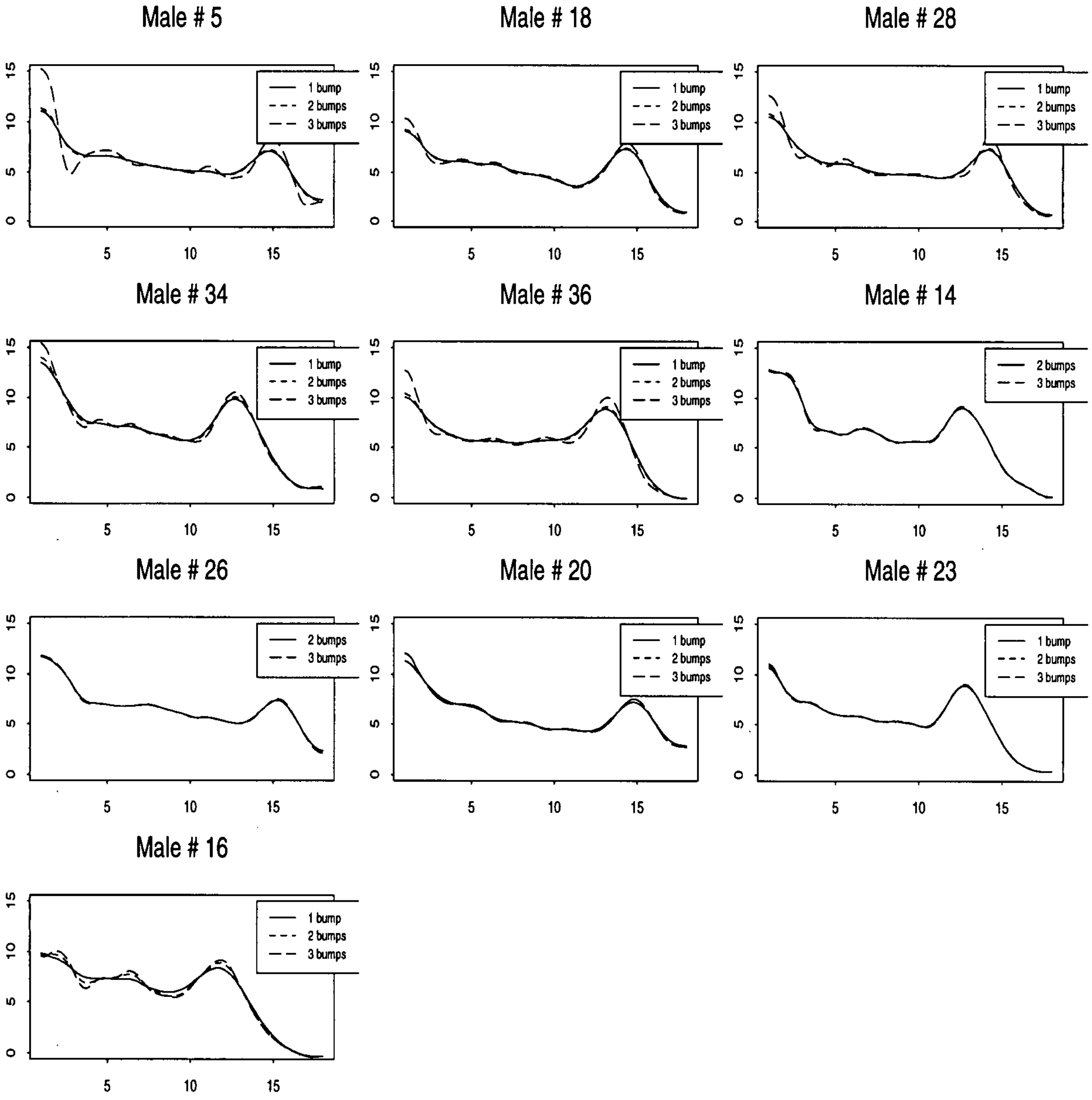


Figure A.4: Plots of the cases with no decision reached by the "strict" criterion.

Appendix B

Female results

In Table B.1 the results of the bootstrap procedure are summarized. We show the p-values of all the tests performed with different definitions of a bump (2-, 3-, 4-, and 5-max), and two smoothing methods used in steps 1 and 5 of the bootstrap procedure on page 39: Lspline and Kernel smoothing. In step 2 of the bootstrap procedure, we always used the Kernel smoothing.

Also, we exhibit here the estimated growth curves for all 50 females. Lspline smoothing is used to obtain these estimates. When the "strict" decision rule gives us the classification, we plot only the estimate with the value of a smoothing parameter being equal to the respective $\lambda_{crit}^{(b)}$. In 11 cases when the decision was not reached by using the "strict" rule we plot the estimates with the values of b attained when using the range rule. "Soft" rule classification is indicated by a solid line.

Table B.1: Summary of 500 bootstrap samples for the female growth data.

			H_0 : # of bumps $\le k$			
Female	Width	Method	$k=0$	$k=1$	$k=2$	$k=3$
1	2-max	Lspline	0.000	0.908	1.000	0.996
		Kernel	0.000	0.748	0.998	0.998
	3-max	Lspline	0.000	0.872	0.998	0.994
		Kernel	0.000	0.754	0.994	0.998
	4-max	Lspline	0.000	0.870	0.996	0.986
		Kernel	0.000	0.762	0.988	0.998
	5-max	Lspline	0.000	0.818	0.998	0.950
		Kernel	0.000	0.712	0.962	0.774
2	2-max	Lspline	0.578	0.520	0.156	0.034
		Kernel	0.074	0.172	0.170	0.026
	3-max	Lspline	0.586	0.630	0.116	0.030
		Kernel	0.066	0.186	0.174	0.020
	4-max	Lspline	0.582	0.638	0.096	0.052
		Kernel	0.058	0.196	0.148	0.018
	5-max	Lspline	0.600	0.698	0.090	0.118
		Kernel	0.068	0.198	0.160	0.018
3	2-max	Lspline	0.022	0.544	0.718	0.924
		Kernel	0.116	0.352	0.884	0.804
	3-max	Lspline	0.014	0.496	0.664	0.902
		Kernel	0.072	0.348	0.886	0.838
	4-max	Lspline	0.010	0.462	0.614	0.826
		Kernel	0.038	0.316	0.924	0.856
	5-max	Lspline	0.002	0.408	0.654	0.770
		Kernel	0.024	0.296	0.860	0.764

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
4	2-max	Lspline	0.000	0.070	0.728	0.688
		Kernel	0.000	0.006	0.222	0.430
	3-max	Lspline	0.000	0.138	0.676	0.672
		Kernel	0.000	0.004	0.462	0.398
	4-max	Lspline	0.000	0.118	0.700	0.610
		Kernel	0.000	0.004	0.752	0.476
	5-max	Lspline	0.000	0.084	0.722	0.966
		Kernel	0.000	0.004	0.882	NA
5	2-max	Lspline	0.000	0.044	0.364	0.944
		Kernel	0.000	0.012	0.874	0.630
	3-max	Lspline	0.000	0.022	0.998	0.956
		Kernel	0.000	0.008	0.886	0.550
	4-max	Lspline	0.000	0.018	0.996	0.910
		Kernel	0.000	0.008	0.894	0.508
	5-max	Lspline	0.000	0.012	0.982	0.804
		Kernel	0.000	0.012	0.914	0.656
6	2-max	Lspline	0.000	0.196	0.008	0.756
		Kernel	0.002	0.062	0.012	0.304
	3-max	Lspline	0.000	0.152	0.002	0.682
		Kernel	0.000	0.054	0.012	0.272
	4-max	Lspline	0.000	0.100	0.002	0.616
		Kernel	0.000	0.052	0.000	0.242
	5-max	Lspline	0.000	0.034	0.000	0.554
		Kernel	0.000	0.044	0.000	0.170

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
7	2-max	Lspline	0.000	0.174	0.020	0.038
		Kernel	0.000	0.082	0.000	0.008
	3-max	Lspline	0.000	0.126	0.008	0.058
		Kernel	0.000	0.084	0.000	0.012
	4-max	Lspline	0.000	0.128	0.010	0.032
		Kernel	0.000	0.102	0.000	0.008
	5-max	Lspline	0.000	0.132	0.008	0.034
		Kernel	0.000	0.104	0.000	0.012
8	2-max	Lspline	0.000	0.448	0.672	0.920
		Kernel	0.000	0.390	0.754	0.774
	3-max	Lspline	0.000	0.388	0.722	0.884
		Kernel	0.000	0.336	0.768	0.714
	4-max	Lspline	0.000	0.378	0.608	0.904
		Kernel	0.000	0.300	0.820	0.710
	5-max	Lspline	0.000	0.362	0.658	0.784
		Kernel	0.000	0.286	0.920	0.732
9	2-max	Lspline	0.000	0.474	0.518	0.616
		Kernel	0.000	0.868	0.776	0.732
	3-max	Lspline	0.000	0.426	0.420	0.606
		Kernel	0.000	0.864	0.888	0.806
	4-max	Lspline	0.000	0.550	0.376	0.652
		Kernel	0.000	0.810	0.974	0.836
	5-max	Lspline	0.000	0.744	0.968	0.838
		Kernel	0.000	0.790	0.860	0.784

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
10	2-max	Lspline	0.000	0.382	0.522	0.998
		Kernel	0.000	0.562	0.624	0.946
	3-max	Lspline	0.000	0.320	0.508	1.000
		Kernel	0.000	0.582	0.546	0.928
	4-max	Lspline	0.000	0.310	0.476	1.000
		Kernel	0.000	0.646	0.468	0.884
	5-max	Lspline	0.000	0.272	0.456	0.904
		Kernel	0.000	0.566	0.472	0.808
11	2-max	Lspline	0.000	0.246	0.206	0.140
		Kernel	0.014	0.466	0.216	0.216
	3-max	Lspline	0.000	0.214	0.168	0.102
		Kernel	0.010	0.448	0.194	0.192
	4-max	Lspline	0.000	0.220	0.208	0.068
		Kernel	0.006	0.422	0.162	0.150
	5-max	Lspline	0.000	0.258	0.154	0.062
		Kernel	0.006	0.450	0.156	0.090
12	2-max	Lspline	0.000	0.468	0.276	0.316
		Kernel	0.000	0.006	0.148	0.322
	3-max	Lspline	0.000	0.510	0.250	0.476
		Kernel	0.000	0.010	0.140	0.284
	4-max	Lspline	0.000	0.550	0.200	0.374
		Kernel	0.000	0.024	0.092	0.294
	5-max	Lspline	0.000	0.492	0.124	0.526
		Kernel	0.000	0.060	0.070	0.348

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
13	2-max	Lspline	0.000	0.180	0.624	0.450
		Kernel	0.000	0.356	0.438	0.684
	3-max	Lspline	0.000	0.150	0.546	0.448
		Kernel	0.000	0.318	0.406	0.672
	4-max	Lspline	0.000	0.200	0.520	0.416
		Kernel	0.000	0.310	0.448	0.616
	5-max	Lspline	0.000	0.258	0.496	0.404
		Kernel	0.000	0.352	0.388	0.620
14	2-max	Lspline	0.000	0.154	0.250	0.700
		Kernel	0.000	0.174	0.062	0.174
	3-max	Lspline	0.000	0.166	0.184	0.606
		Kernel	0.000	0.168	0.058	0.234
	4-max	Lspline	0.000	0.186	0.234	0.690
		Kernel	0.000	0.182	0.064	0.192
	5-max	Lspline	0.000	0.128	0.186	0.550
		Kernel	0.000	0.190	0.072	0.250
15	2-max	Lspline	0.010	0.018	1.000	0.984
		Kernel	0.374	0.006	1.000	1.000
	3-max	Lspline	0.014	0.012	1.000	0.976
		Kernel	0.356	0.004	1.000	0.988
	4-max	Lspline	0.014	0.008	1.000	0.962
		Kernel	0.346	0.004	0.998	0.964
	5-max	Lspline	0.014	0.008	0.990	0.930
		Kernel	0.332	0.004	0.996	NA

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
16	2-max	Lspline	0.000	0.446	0.276	0.492
		Kernel	0.000	0.350	0.164	0.600
	3-max	Lspline	0.000	0.376	0.348	0.530
		Kernel	0.000	0.346	0.180	0.534
	4-max	Lspline	0.000	0.410	0.416	0.516
		Kernel	0.000	0.348	0.156	0.502
	5-max	Lspline	0.000	0.412	0.302	0.602
		Kernel	0.000	0.274	0.122	0.674
17	2-max	Lspline	0.030	0.348	0.988	0.882
		Kernel	0.000	0.148	0.964	0.908
	3-max	Lspline	0.000	0.290	0.986	0.848
		Kernel	0.000	0.188	0.962	0.930
	4-max	Lspline	0.000	0.248	0.998	0.810
		Kernel	0.000	0.160	0.968	0.948
	5-max	Lspline	0.000	0.194	0.994	0.872
		Kernel	0.000	0.132	0.984	0.888
18	2-max	Lspline	0.000	0.006	0.932	0.998
		Kernel	0.000	0.028	0.968	0.870
	3-max	Lspline	0.000	0.006	0.982	0.998
		Kernel	0.000	0.032	0.972	0.828
	4-max	Lspline	0.000	0.002	1.000	0.996
		Kernel	0.000	0.030	0.928	0.896
	5-max	Lspline	0.000	0.002	0.998	NA
		Kernel	0.000	0.040	0.962	0.656

Table B.1: *(continued)*

Female	Width	Method	$k=0$	$k=1$	$k=2$	$k=3$
19	2-max	Lspline	0.000	0.346	0.116	0.998
		Kernel	0.000	0.210	0.064	0.990
	3-max	Lspline	0.000	0.306	0.110	0.998
		Kernel	0.000	0.196	0.070	0.988
	4-max	Lspline	0.000	0.282	0.084	0.988
		Kernel	0.000	0.206	0.090	0.974
	5-max	Lspline	0.000	0.226	0.096	0.958
		Kernel	0.000	0.226	0.068	0.912
20	2-max	Lspline	0.000	0.678	0.522	0.404
		Kernel	0.002	0.758	0.462	0.164
	3-max	Lspline	0.000	0.716	0.428	0.380
		Kernel	0.002	0.760	0.476	0.136
	4-max	Lspline	0.000	0.698	0.312	0.354
		Kernel	0.000	0.764	0.456	0.176
	5-max	Lspline	0.000	0.658	0.236	0.366
		Kernel	0.000	0.736	0.508	0.278
21	2-max	Lspline	0.000	0.038	0.912	0.650
		Kernel	0.000	0.014	0.838	0.576
	3-max	Lspline	0.000	0.020	0.888	0.656
		Kernel	0.000	0.020	0.818	0.570
	4-max	Lspline	0.000	0.008	0.914	0.712
		Kernel	0.000	0.008	0.776	0.598
	5-max	Lspline	0.000	0.012	0.838	0.646
		Kernel	0.000	0.014	0.938	0.724

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
22	2-max	Lspline	0.014	0.166	0.156	0.548
		Kernel	0.124	0.004	0.412	0.198
	3-max	Lspline	0.010	0.126	0.194	0.546
		Kernel	0.114	0.008	0.388	0.206
	4-max	Lspline	0.010	0.092	0.212	0.556
		Kernel	0.094	0.004	0.386	0.156
	5-max	Lspline	0.010	0.046	0.256	0.340
		Kernel	0.082	0.002	0.404	0.162
23	2-max	Lspline	0.000	0.012	0.444	1.000
		Kernel	0.048	0.044	0.182	0.888
	3-max	Lspline	0.000	0.006	0.390	1.000
		Kernel	0.010	0.032	0.188	0.880
	4-max	Lspline	0.000	0.000	0.364	1.000
		Kernel	0.006	0.022	0.166	0.846
	5-max	Lspline	0.000	0.000	0.312	1.000
		Kernel	0.006	0.016	0.150	0.870
24	2-max	Lspline	0.000	0.012	0.830	0.372
		Kernel	0.004	0.000	0.594	0.330
	3-max	Lspline	0.000	0.008	0.772	0.324
		Kernel	0.000	0.000	0.618	0.316
	4-max	Lspline	0.000	0.008	0.776	0.290
		Kernel	0.000	0.000	0.628	0.342
	5-max	Lspline	0.000	0.004	0.752	0.168
		Kernel	0.000	0.000	0.642	0.380

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
25	2-max	Lspline	0.000	0.060	0.986	0.928
		Kernel	0.000	0.006	0.830	0.590
	3-max	Lspline	0.000	0.014	0.984	0.948
		Kernel	0.000	0.006	0.912	0.780
	4-max	Lspline	0.000	0.016	0.988	0.982
		Kernel	0.000	0.006	0.904	0.788
	5-max	Lspline	0.000	0.008	0.988	NA
		Kernel	0.000	0.004	0.854	0.416
26	2-max	Lspline	0.000	0.198	0.010	0.288
		Kernel	0.000	0.030	0.000	0.770
	3-max	Lspline	0.000	0.144	0.000	0.470
		Kernel	0.000	0.034	0.000	0.686
	4-max	Lspline	0.000	0.126	0.000	0.400
		Kernel	0.000	0.036	0.000	0.752
	5-max	Lspline	0.000	0.084	0.000	0.614
		Kernel	0.000	0.026	0.000	0.786
27	2-max	Lspline	0.000	0.952	0.892	1.000
		Kernel	0.000	0.804	0.594	0.874
	3-max	Lspline	0.000	0.918	0.816	1.000
		Kernel	0.000	0.772	0.670	0.902
	4-max	Lspline	0.000	0.892	0.846	0.994
		Kernel	0.000	0.728	0.668	0.734
	5-max	Lspline	0.000	0.832	0.730	0.948
		Kernel	0.000	0.654	0.860	0.602

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
28	2-max	Lspline	0.000	0.114	0.080	0.172
		Kernel	0.000	0.082	0.188	0.010
	3-max	Lspline	0.000	0.080	0.094	0.162
		Kernel	0.000	0.072	0.204	0.028
	4-max	Lspline	0.000	0.078	0.058	0.166
		Kernel	0.000	0.064	0.200	0.146
	5-max	Lspline	0.000	0.056	0.060	0.764
		Kernel	0.000	0.056	0.176	0.780
29	2-max	Lspline	0.000	0.036	0.098	0.014
		Kernel	0.000	0.038	0.060	0.006
	3-max	Lspline	0.000	0.042	0.170	0.018
		Kernel	0.000	0.036	0.074	0.004
	4-max	Lspline	0.000	0.056	0.166	0.012
		Kernel	0.000	0.038	0.108	0.004
	5-max	Lspline	0.000	0.062	0.302	0.024
		Kernel	0.000	0.048	0.084	0.002
30	2-max	Lspline	0.000	0.142	0.378	0.390
		Kernel	0.000	0.094	0.206	0.114
	3-max	Lspline	0.000	0.138	0.266	0.332
		Kernel	0.000	0.084	0.222	0.156
	4-max	Lspline	0.000	0.178	0.360	0.314
		Kernel	0.000	0.082	0.176	0.100
	5-max	Lspline	0.000	0.118	0.356	0.236
		Kernel	0.000	0.054	0.184	0.250

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
31	2-max	Lspline	0.008	0.000	0.092	0.862
		Kernel	0.000	0.000	0.014	0.474
	3-max	Lspline	0.002	0.000	0.060	0.826
		Kernel	0.000	0.000	0.022	0.788
	4-max	Lspline	0.000	0.000	0.034	0.824
		Kernel	0.000	0.000	0.022	0.746
	5-max	Lspline	0.000	0.070	0.018	0.752
		Kernel	0.000	0.000	0.020	0.738
32	2-max	Lspline	0.000	0.180	0.026	0.200
		Kernel	0.000	0.204	0.000	0.196
	3-max	Lspline	0.000	0.126	0.020	0.182
		Kernel	0.000	0.230	0.004	0.178
	4-max	Lspline	0.000	0.104	0.026	0.180
		Kernel	0.000	0.196	0.004	0.174
	5-max	Lspline	0.000	0.106	0.036	0.164
		Kernel	0.000	0.190	0.004	0.232
33	2-max	Lspline	0.000	0.738	0.902	0.904
		Kernel	0.008	0.440	0.886	0.564
	3-max	Lspline	0.000	0.686	0.932	0.926
		Kernel	0.004	0.418	0.896	0.554
	4-max	Lspline	0.000	0.646	0.914	0.784
		Kernel	0.000	0.416	0.892	0.598
	5-max	Lspline	0.000	0.576	0.872	0.986
		Kernel	0.000	0.424	0.668	0.474

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
34	2-max	Lspline	0.000	0.302	0.110	0.020
		Kernel	0.000	0.214	0.066	0.344
	3-max	Lspline	0.000	0.330	0.102	0.026
		Kernel	0.000	0.224	0.110	0.340
	4-max	Lspline	0.000	0.478	0.142	0.036
		Kernel	0.000	0.206	0.164	0.256
	5-max	Lspline	0.000	0.414	0.190	0.352
		Kernel	0.000	0.206	0.386	0.140
35	2-max	Lspline	0.000	0.078	0.214	0.942
		Kernel	0.000	0.088	0.172	0.776
	3-max	Lspline	0.000	0.042	0.172	0.954
		Kernel	0.000	0.086	0.130	0.834
	4-max	Lspline	0.000	0.034	0.116	0.980
		Kernel	0.000	0.090	0.152	0.832
	5-max	Lspline	0.000	0.030	0.106	0.960
		Kernel	0.000	0.078	0.154	0.484
36	2-max	Lspline	0.000	0.162	0.140	0.020
		Kernel	0.000	0.044	0.000	0.000
	3-max	Lspline	0.000	0.088	0.148	0.026
		Kernel	0.000	0.042	0.000	0.000
	4-max	Lspline	0.000	0.052	0.134	0.016
		Kernel	0.000	0.014	0.000	0.000
	5-max	Lspline	0.000	0.016	0.130	0.002
		Kernel	0.000	0.008	0.000	0.000

Table B.1: *(continued)*

Female	Width	Method	$k=0$	$k=1$	$k=2$	$k=3$
37	2-max	Lspline	0.000	0.086	0.996	0.928
		Kernel	0.000	0.050	0.922	0.760
	3-max	Lspline	0.000	0.078	0.994	0.942
		Kernel	0.000	0.050	0.924	0.826
	4-max	Lspline	0.000	0.054	0.994	0.934
		Kernel	0.000	0.046	0.896	0.806
	5-max	Lspline	0.000	0.022	0.982	NA
		Kernel	0.000	0.052	0.762	0.442
38	2-max	Lspline	0.026	0.768	0.554	0.436
		Kernel	0.100	0.894	0.456	0.104
	3-max	Lspline	0.020	0.734	0.658	0.396
		Kernel	0.076	0.888	0.472	0.100
	4-max	Lspline	0.012	0.704	0.586	0.340
		Kernel	0.048	0.894	0.448	0.098
	5-max	Lspline	0.012	0.646	0.634	0.266
		Kernel	0.032	0.896	0.400	0.138
39	2-max	Lspline	0.000	0.228	0.306	0.070
		Kernel	0.000	0.148	0.344	0.222
	3-max	Lspline	0.000	0.230	0.294	0.034
		Kernel	0.000	0.136	0.356	0.248
	4-max	Lspline	0.000	0.184	0.254	0.048
		Kernel	0.000	0.134	0.394	0.278
	5-max	Lspline	0.000	0.178	0.266	0.092
		Kernel	0.000	0.140	0.278	0.180

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
40	2-max	Lspline	0.000	0.034	0.198	0.144
		Kernel	0.000	0.014	0.024	0.362
	3-max	Lspline	0.000	0.008	0.210	0.112
		Kernel	0.000	0.014	0.026	0.380
	4-max	Lspline	0.000	0.016	0.178	0.064
		Kernel	0.000	0.008	0.024	0.474
	5-max	Lspline	0.000	0.006	0.182	0.080
		Kernel	0.000	0.008	0.020	0.384
41	2-max	Lspline	0.000	0.424	0.962	0.920
		Kernel	0.000	0.276	1.000	0.984
	3-max	Lspline	0.000	0.452	0.974	0.944
		Kernel	0.000	0.260	1.000	0.998
	4-max	Lspline	0.000	0.420	0.954	0.984
		Kernel	0.000	0.272	0.994	0.982
	5-max	Lspline	0.000	0.338	0.924	0.860
		Kernel	0.000	0.216	NA	NA
42	2-max	Lspline	0.000	0.010	0.510	0.100
		Kernel	0.030	0.000	0.306	0.048
	3-max	Lspline	0.000	0.004	0.508	0.104
		Kernel	0.024	0.000	0.328	0.064
	4-max	Lspline	0.000	0.004	0.494	0.142
		Kernel	0.012	0.000	0.334	0.236
	5-max	Lspline	0.000	0.004	0.460	0.482
		Kernel	0.002	0.000	0.330	0.302

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
43	2-max	Lspline	0.000	0.228	0.046	0.858
		Kernel	0.000	0.134	0.010	0.778
	3-max	Lspline	0.000	0.218	0.080	0.902
		Kernel	0.000	0.130	0.012	0.728
	4-max	Lspline	0.000	0.232	0.036	0.944
		Kernel	0.000	0.148	0.014	0.668
	5-max	Lspline	0.000	0.164	0.074	0.870
		Kernel	0.000	0.188	0.024	0.730
44	2-max	Lspline	0.000	0.172	0.958	0.992
		Kernel	0.000	0.106	0.980	0.918
	3-max	Lspline	0.000	0.148	0.984	0.970
		Kernel	0.000	0.082	0.984	0.888
	4-max	Lspline	0.000	0.122	0.962	0.942
		Kernel	0.000	0.086	0.966	0.724
	5-max	Lspline	0.000	0.100	0.994	0.896
		Kernel	0.000	0.100	0.914	0.864
45	2-max	Lspline	0.000	0.384	0.076	0.010
		Kernel	0.052	0.320	0.122	0.108
	3-max	Lspline	0.000	0.340	0.092	0.002
		Kernel	0.036	0.320	0.116	0.116
	4-max	Lspline	0.000	0.290	0.098	0.026
		Kernel	0.032	0.322	0.102	0.138
	5-max	Lspline	0.000	0.290	0.102	0.066
		Kernel	0.032	0.298	0.096	0.160

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
46	2-max	Lspline	0.000	0.088	0.044	0.774
		Kernel	0.026	0.004	0.002	0.192
	3-max	Lspline	0.000	0.028	0.016	0.724
		Kernel	0.020	0.012	0.000	0.348
	4-max	Lspline	0.000	0.010	0.012	0.718
		Kernel	0.016	0.024	0.000	0.626
	5-max	Lspline	0.000	0.008	0.004	0.640
		Kernel	0.012	0.014	0.000	0.870
47	2-max	Lspline	0.000	0.628	0.628	0.924
		Kernel	0.000	0.466	0.462	0.134
	3-max	Lspline	0.000	0.634	0.548	0.898
		Kernel	0.000	0.468	0.436	0.342
	4-max	Lspline	0.000	0.588	0.504	0.938
		Kernel	0.000	0.426	0.392	0.732
	5-max	Lspline	0.000	0.538	0.438	0.868
		Kernel	0.000	0.432	0.412	0.860
48	2-max	Lspline	0.000	0.102	0.804	0.968
		Kernel	0.000	0.098	0.652	0.682
	3-max	Lspline	0.000	0.086	0.778	0.958
		Kernel	0.000	0.092	0.606	0.870
	4-max	Lspline	0.000	0.080	0.722	0.974
		Kernel	0.000	0.080	0.622	0.876
	5-max	Lspline	0.000	0.042	0.646	0.872
		Kernel	0.000	0.068	0.462	NA

Table B.1: *(continued)*

Female	Width	Method	$k = 0$	$k = 1$	$k = 2$	$k = 3$
49	2-max	Lspline	0.030	0.748	0.252	0.808
		Kernel	0.070	0.612	0.490	0.294
	3-max	Lspline	0.010	0.728	0.242	0.796
		Kernel	0.058	0.602	0.714	0.352
	4-max	Lspline	0.000	0.710	0.240	0.776
		Kernel	0.058	0.582	0.660	0.762
	5-max	Lspline	0.000	0.694	0.186	0.950
		Kernel	0.058	0.590	0.666	0.978
50	2-max	Lspline	0.000	0.842	0.708	0.924
		Kernel	0.000	0.488	0.414	0.238
	3-max	Lspline	0.000	0.804	0.652	0.938
		Kernel	0.000	0.440	0.536	0.358
	4-max	Lspline	0.000	0.740	0.550	0.926
		Kernel	0.000	0.480	0.536	0.602
	5-max	Lspline	0.000	0.690	0.372	0.860
		Kernel	0.000	0.398	0.428	0.894

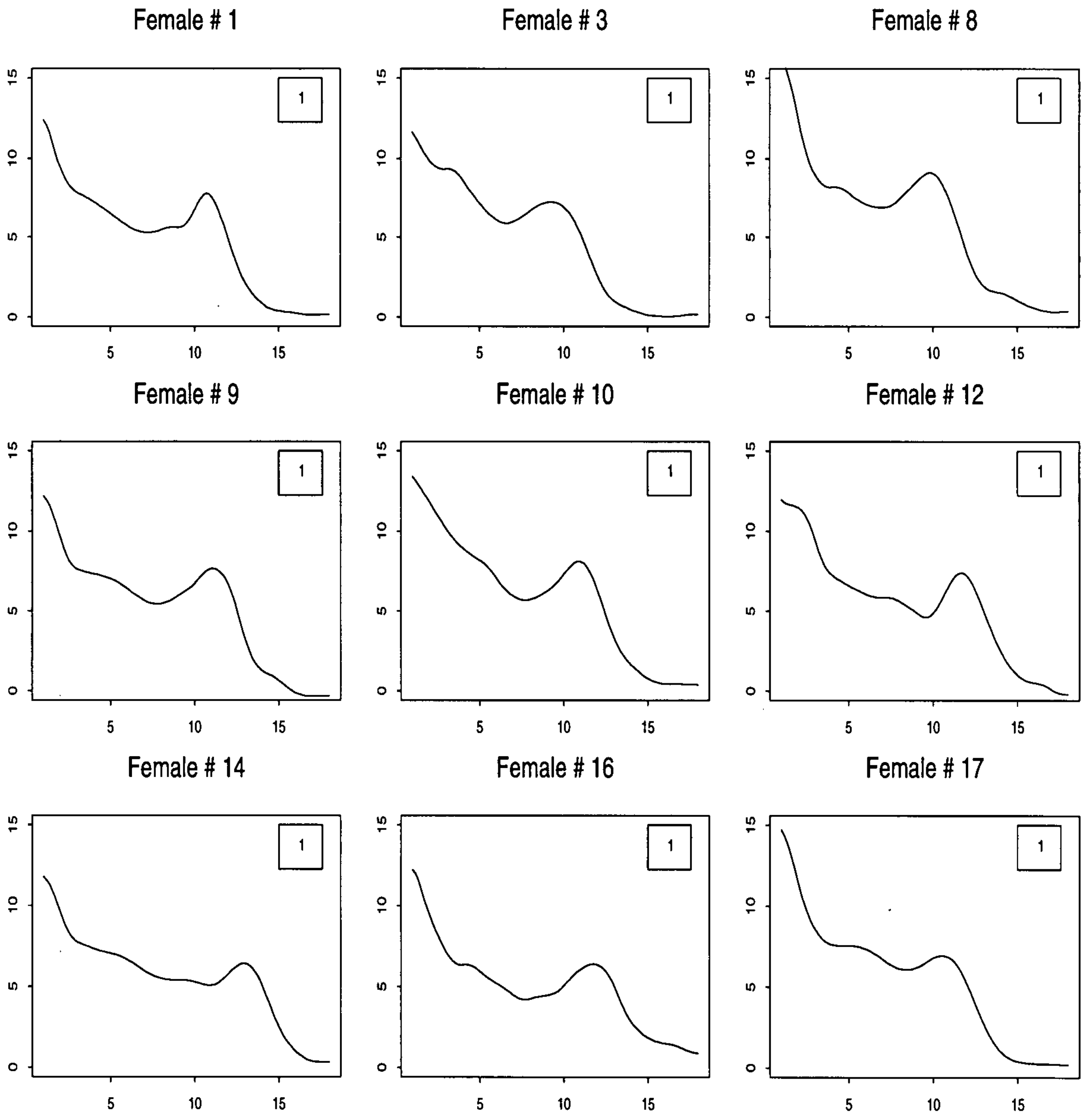


Figure B.1: Plots of female speed curves with 1 bump.

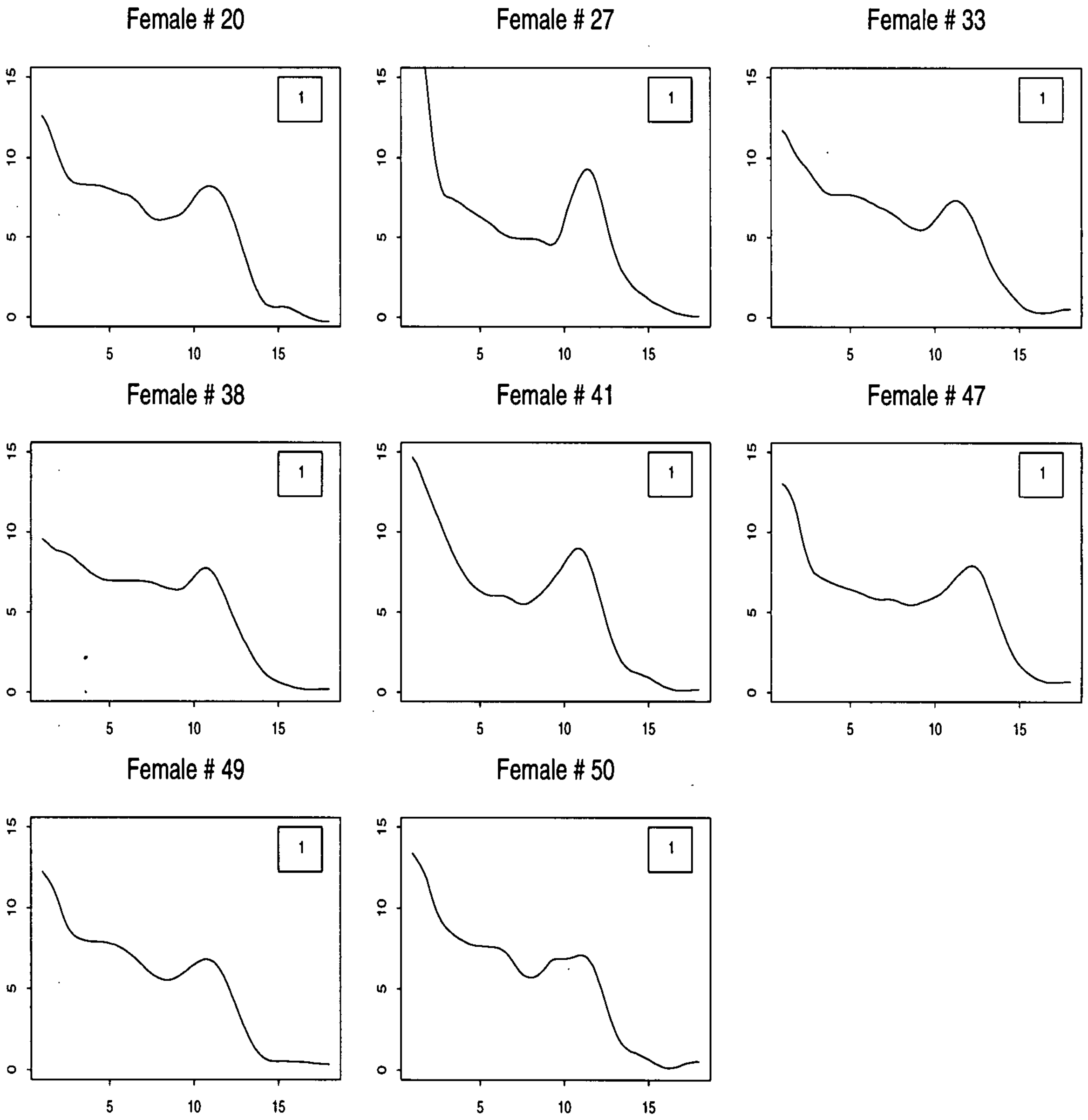
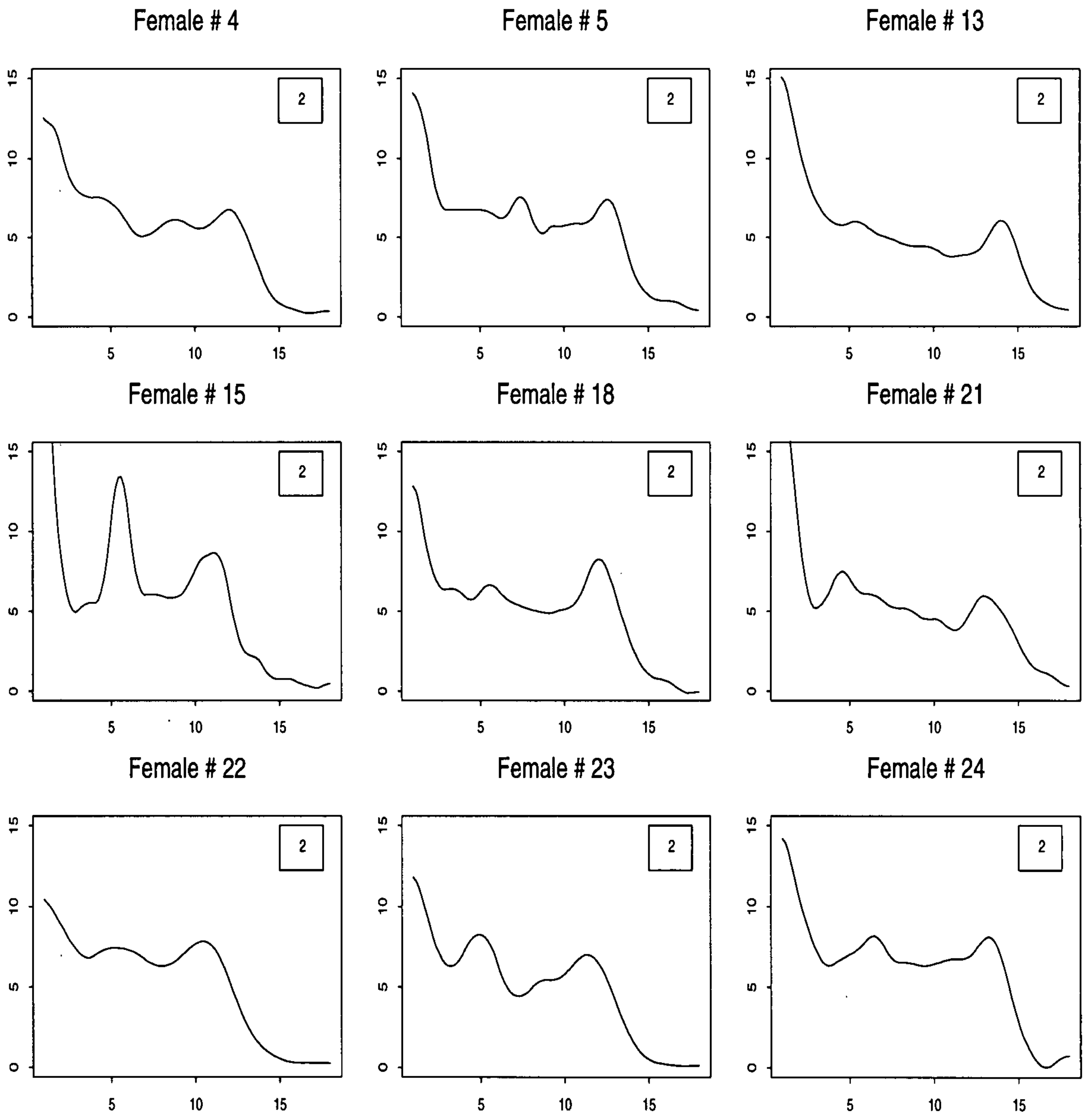


Figure B.2: Plots of female speed curves with 1 bump (continued).



Female # 4

Female # 5

Female # 13

Female # 15

Female # 18

Female # 21

Female # 22

Female # 23

Female # 24

Figure B.3: Plots of female speed curves with 2 bumps.

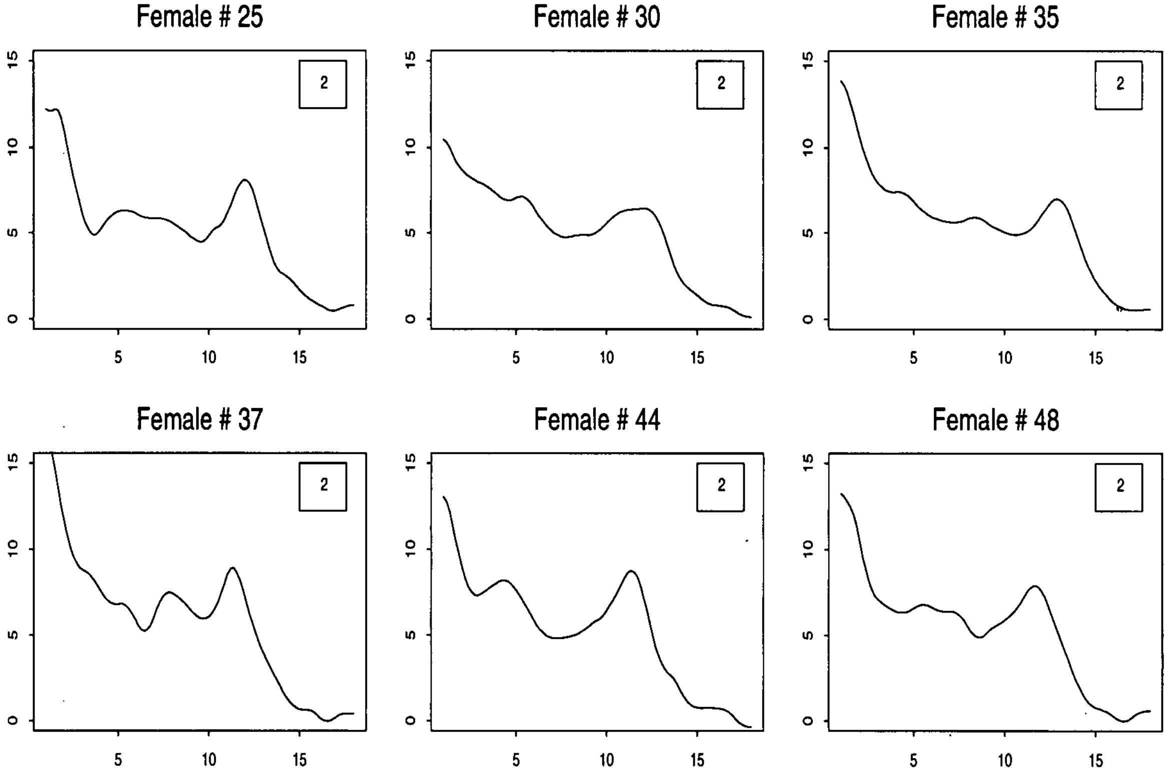


Figure B.4: Plots of female speed curves with 2 bumps (continued).

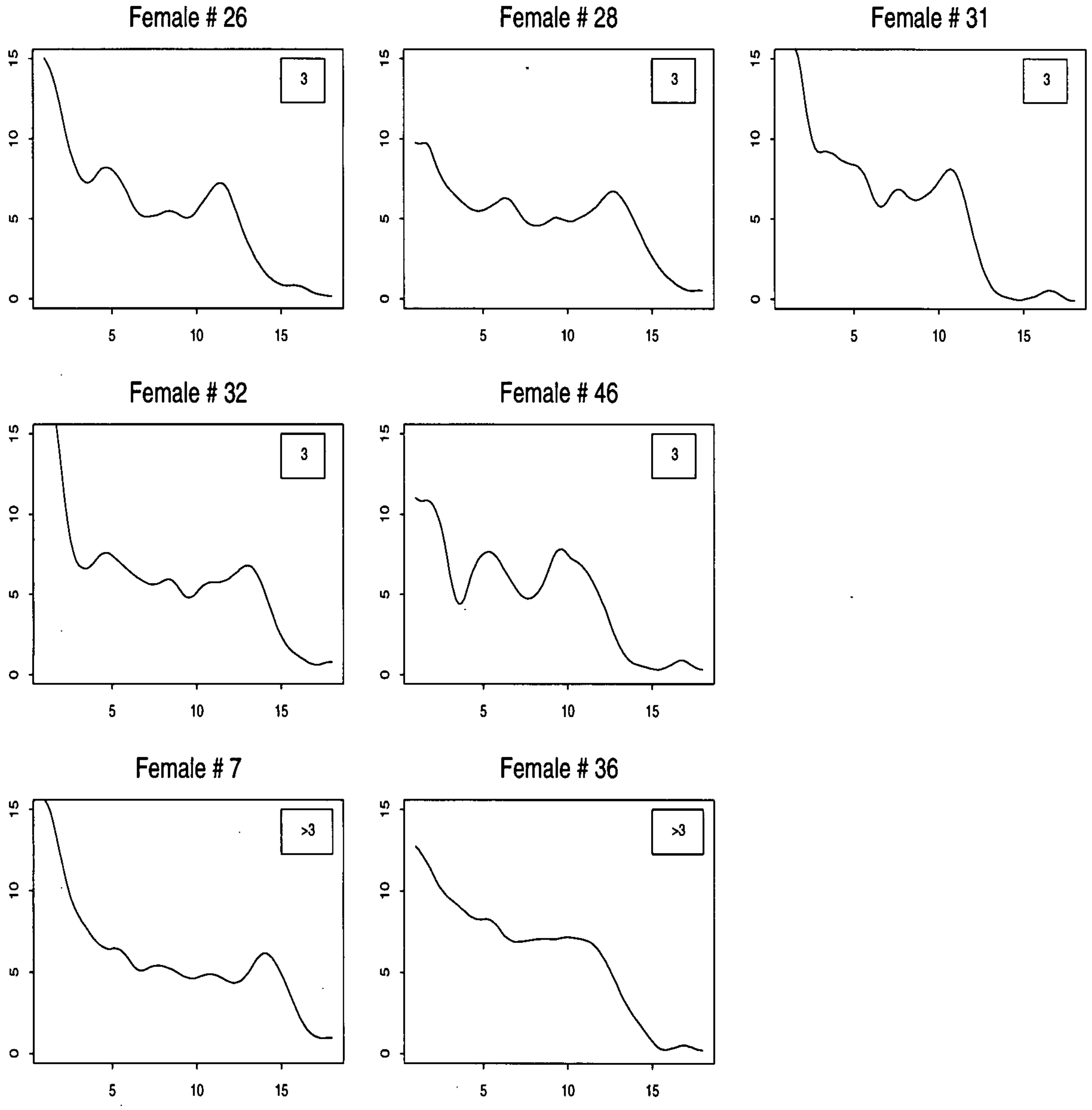


Figure B.5: Plots of female speed curves with 3 or more than 3 bumps.

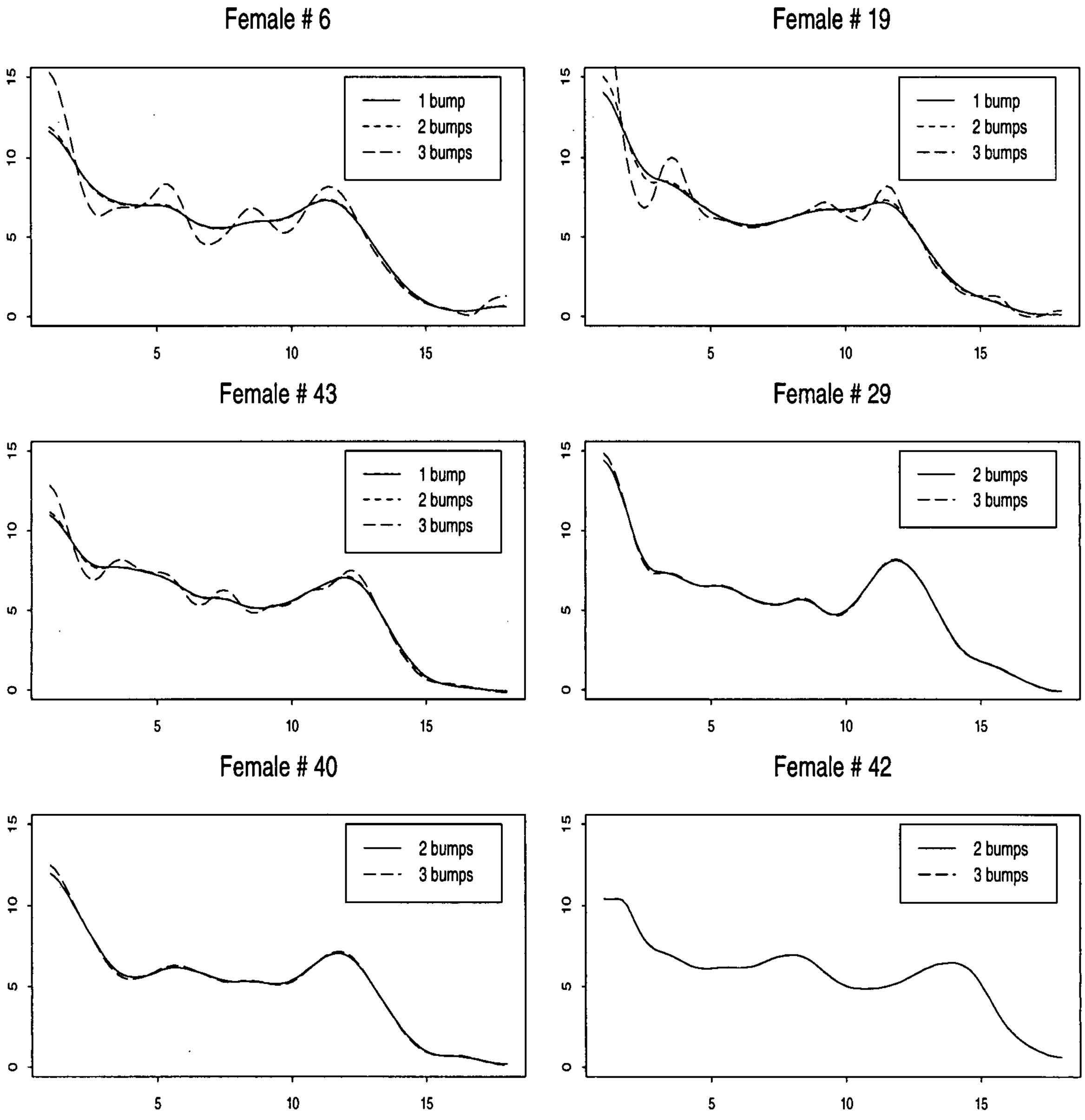


Figure B.6: Plots of the cases with no decision reached by the "strict" criterion.

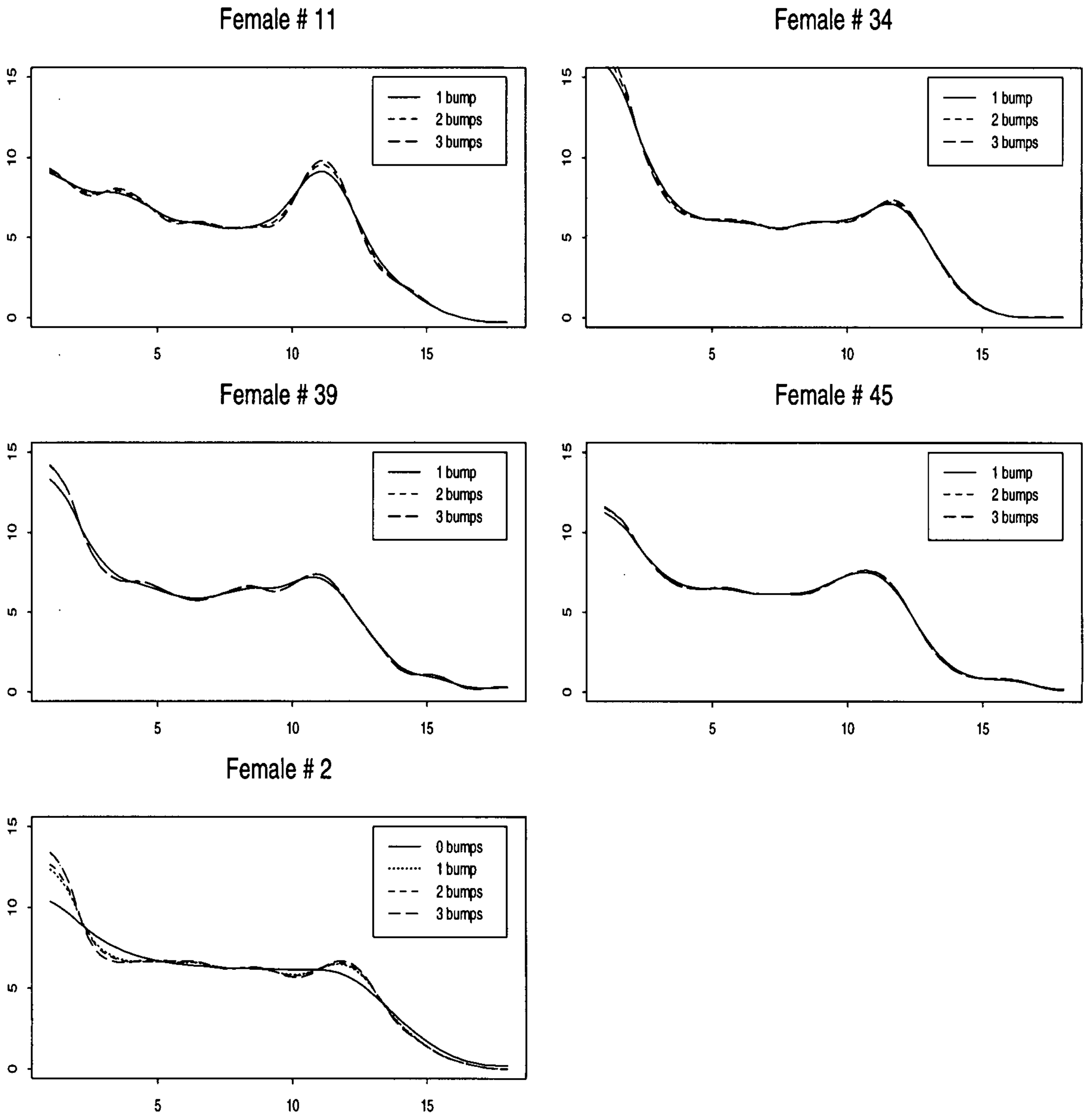


Figure B.7: Plots of the cases with no decision reached by the "strict" criterion (continued).