Springer Series in Statistics

J.O. Ramsay
B.W. Silverman

Functional Data Analysis

Second Edition

With 151 Illustrations

Springer

J.O. Ramsay
Department of Psychology
University of Montreal
Montreal, Quebec H3A 1B1
Canada
ramsay@psych.mcgill.ca

B.W. Silverman
St. Peter's College
Oxford OX1 2DL
United Kingdom
bernard.silverman@spc.ox.ac.uk

Library of Congress Control Number: 2005923773

ISBN-10: 0-387-40080-X Printed on acid-free paper.
ISBN-13: 978-0387-40080-8

Printed in the United States of America. (MP)

9 8 7 6 5 4 3 2 1

springeronline.com

J.O. Ramsay
Department of Psychology
McGill University
Montreal, Quebec H4A 1B1
Canada
ramsay@psych.mcgill.ca

B.W. Silverman
St. Peter's College
Oxford OX1 2DL
United Kingdom
bernard.silverman@spc.ox.ac.uk

Library of Congress Control Number: 2005923773

ISBN-10: 0-387-40080-X Printed on acid-free paper.
ISBN-13: 978-0387-40080-8

Printed in the United States of America. (MP)

9 8 7 6 5 4 3 2 1

springeronline.com

Preface to the Second Edition

This book continues in the footsteps of the First Edition in being a snapshot of a highly social, and therefore decidedly unpredictable, process. The combined personal view of functional data analysis that it presents has emerged over a number of years of research and contact, and has been greatly nourished by delightful collaborations with many friends. We hope that readers will enjoy the book as much as we have enjoyed writing it, whether they are our colleagues as researchers or applied data analysts reading the book as a research monograph, or students using it as a course text.

As in the First Edition, live data are used throughout for both motivation and illustration, showing how functional approaches allow us to see new things, especially by exploiting the smoothness of the processes generating the data. The data sets exemplify the wide scope of functional data analysis; they are drawn from growth analysis, meteorology, biomechanics, equine science, economics and medicine.

"Back to the data" was the heading to the last section of the First Edition. We did not know then how well those words would predict the next eight years. Since then we have seen functional data applications in more scientific and industrial settings than we could have imagined, and so we wanted an opportunity to make this new field accessible to a wider readership than the the first volume seemed to permit. Our book of case studies, Ramsay and Silverman (2002), was our first response, but we have known for some time that a new edition of our original volume was also required.

We have added a considerable amount of new material, and considered carefully how the original material should be presented. One main objec-

tive has been, especially when introducing the various concepts, to provide more intuitive discussion and to postpone needless mathematical terminology where possible. In addition we wanted to offer more practical advice on the processing of functional data. To this end, we have added a more extended account of spline basis functions, provided new material on data smoothing, and extended the range of ways in which data can be used to estimate functions. In response to many requests, we have added some proposals for estimating confidence regions, highlighting local features, and even testing hypotheses. Nevertheless, the emphasis in the revision remains more exploratory and confirmatory.

Our treatment of the functional linear model in the First Edition was only preliminary, and since then a great deal of work has been done on this topic by many investigators. A complete overhaul of this material was called for, and the chapters on linear modelling have been completely reworked. On the other hand, our coverage of principal components analysis and canonical correlation still seems appropriate, and not much has been changed. Readers reacted to the later chapters on differential equations as being difficult, and so we have tried to make them a friendlier place to be.

In some places we have opted for an 'intuitive' rather than 'rigorous' approach. This is not merely because we want our book to be widely accessible; in our view the theoretical underpinnings of functional data analysis still require rather more study before a treatment can be written that will please theoreticians. We hope that the next decade will see some exciting progress in this regard.

We both believe that a good monograph is a personal view rather than a dry encyclopedia. The average of two personal views is inevitably going to be less 'personal' than either of the two individual views, just as the average of a set of functions may omit detail present in the original functions. To counteract this tendency, we have ensured that everything we say in our informal and intuitive discussion of certain issues is the view of at least one of us, but we have not always pressed for unanimous agreement!

We owe so much to those who helped us to go here. We would like to repeat our thanks to those who helped with the First Edition: Michal Abrahamowicz, Philippe Besse, Darrell Bock, Catherine Dalzell, Shelly Feran, Randy Flanagan, Rowena Fowler, Theo Gasser, Mary Gauthier, Vince Gracco, Nancy Heckman, Anouk Hoedeman, Steve Hunka, Iain Johnstone, Alois Kneip, Wojtek Krzanowski, Xiaochun Li, Kevin Munhall, Guy Nason, Richard Olshen, David Ostry, Tim Ramsay, John Rice and Xiaohui Wang. We also continue our grateful acknowledgement of financial support from the Natural Science and Engineering Research Council of Canada, the National Science Foundation and the National Institute of Health of the USA, and the British Engineering and Physical Sciences Research Council. The seed for the First Edition, and therefore for the Revised Edition as well, was planted at a discussion meeting of the Royal Statistical Society

Research Section, where one of us read a paper and the other proposed the vote of thanks, not always an occasion that leads to a meeting of minds!

Turning to the Second Edition, Sofia Mosesova and Yoshio Takane read the entire manuscript with an eye to the technical correctness as well as the readability of what they saw, and caught us on many points. David Campbell helped with the literature review that supported our "Further readings and notes" sections. Time spent at the University of British Columbia made possible many stimulating conversations with Nancy Heckman and her colleagues. A discussion of many issues with Alois Kneip as well his hospitality for the first author at the University of Mainz was invaluable. The opportunity for us to spend time together afforded by St Peter's College and the Department of Statistics at Oxford University was essential to the project.

April 2005 *Jim Ramsay & Bernard Silverman*

Contents

1
Introduction

1.1 What are functional data?

Figure 1.1 provides a prototype for the type of data that we shall consider. It shows the heights of 10 girls measured at a set of 31 ages in the Berkeley Growth Study (Tuddenham and Snyder, 1954). The ages are not equally spaced; there are four measurements while the child is one year old, annual measurements from two to eight years, followed by heights measured biannually. Although great care was taken in the measurement process, there is an uncertainty or noise in height values that has a standard deviation of about three millimeters. Even though each record involves only discrete values, these values reflect a smooth variation in height that could be assessed, in principle, as often as desired, and is therefore a height *function*. Thus, the data consist of a sample of 10 *functional* observations $\texttt{Height}_i(t)$.

There are features in this data too subtle to see in this type of plot. Figure 1.2 displays the acceleration curves $D^2\texttt{Height}_i$ estimated from these data by Ramsay, Bock and Gasser (1995) using a technique discussed in Chapter 5. We use the notation D for differentiation, as in

$$D^2\texttt{Height} = \frac{d^2\texttt{Height}}{dt^2}.$$

In Figure 1.2 the pubertal growth spurt shows up as a pulse of strong positive acceleration followed by sharp negative deceleration. But most records also show a bump at around six years that is termed the mid-spurt. We therefore conclude that some of the variation from curve to curve can be explained at the level of certain derivatives. The fact that derivatives

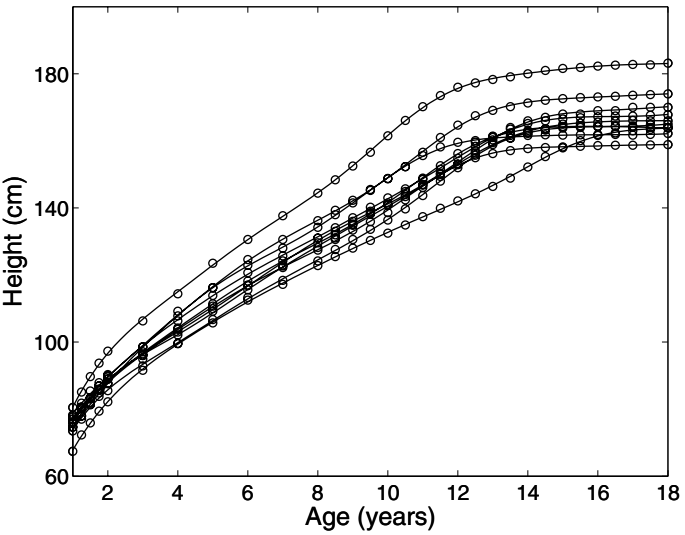

Figure 1.1. The heights of 10 girls measured at 31 ages. The circles indicate the unequally spaced ages of measurement.

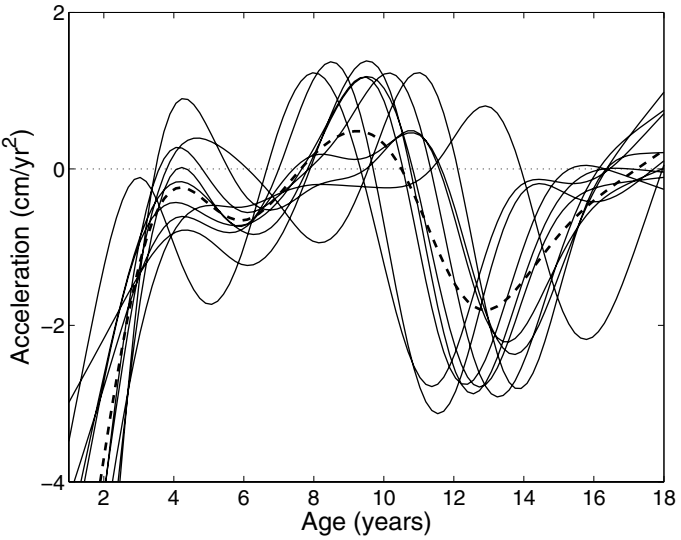

Figure 1.2. The estimated accelerations of height for 10 girls, measured in centimeters per year. The heavy dashed line is the cross-sectional mean, and is a rather poor summary of the curves.

are of interest is further reason to think of the records as functions, rather than vectors of observations in discrete time.

The ages themselves must also play an explicit role in our analysis, because they are not equally spaced. Although it might be mildly interesting to correlate heights at ages 9, 10 and 10.5, this would not take account of the fact that we expect the correlation for two ages separated by only half a year to be higher than that for a separation of one year. Indeed, although in this particular example the ages at which the observations are taken are nominally the same for each girl, there is no real need for this to be so; in general, the points at which the functions are observed may well vary from one record to another.

The replication of these height curves invites an exploration of the ways in which the curves vary. This is potentially complex. For example, the rapid growth during puberty is visible in all curves, but both the timing and the intensity of pubertal growth differ from girl to girl. Some type of principal components analysis would undoubtedly be helpful, but we must adapt the procedure to take account of the unequal age spacing and the smoothness of the underlying height functions. One objective might be to separate variation in timing of significant growth events, such as the pubertal growth spurt, from variation in the intensity of growth.

Not all functional data involves independent replications; we often have to work with a single long record. Figure 1.3 shows an important economic indicator, the nondurable goods manufacturing index for the United States. Data like these often show variation as multiple levels. There is a tendency for the index to show geometric or exponential increase over the whole century. But at a finer scale, we see departures from this trend due to the depression, World War II, the end of the Vietnam War and other more localized events. Moreover, at an even finer scale, there is a marked annual variation, and we can wonder whether this *seasonal trend* itself shows some longer term changes. Although there are no independent replications here, there is still a lot of repetition of information that we can exploit to obtain stable estimates of interesting curve features.

Functional data also arise as input/output pairs, such as in the data in Figure 1.4 collected at an oil refinery in Texas. The amount of a petroleum product at a certain level in a distillation column or cracking tower, shown in the top panel, reacts to the change in the flow of a vapor into the tray, shown in the bottom panel, at that level. How can we characterize this dependency?

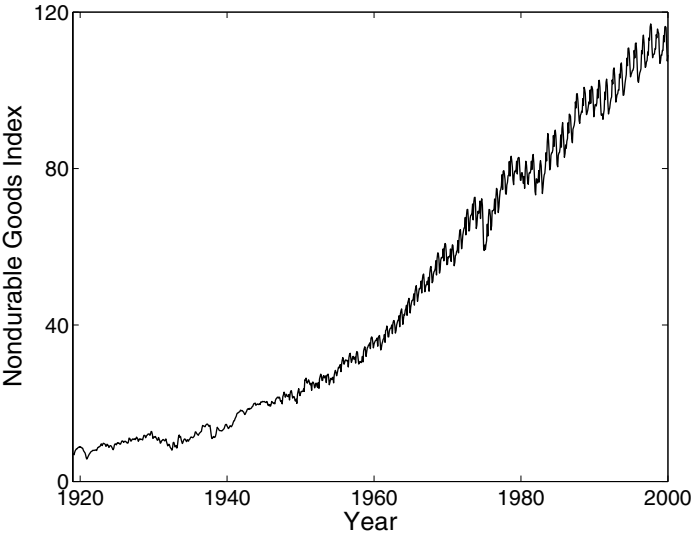

Figure 1.3. The nondurable goods manufacturing index for the United States.

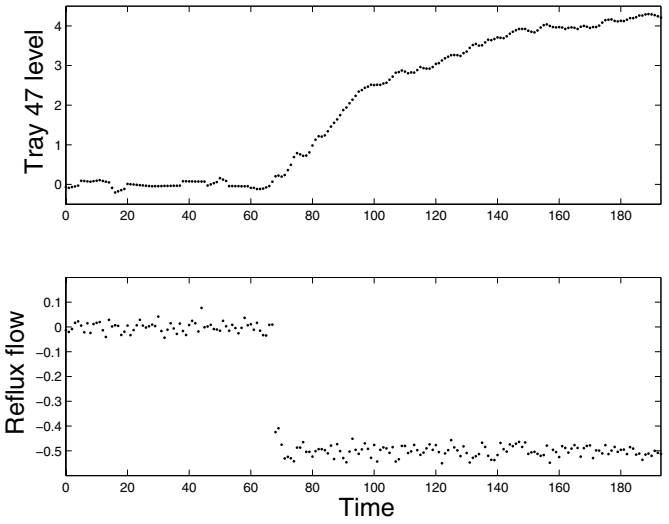

Figure 1.4. The top panel shows 193 measurements of tray level in a distillation column in an oil refinery. The bottom panel shows the flow of a vapor into the tray during an experiment.

1.2 Functional models for nonfunctional data

The data examples above seem to deserve the label "functional" since they so clearly reflect the smooth curves that we assume generated them. But not all data subject to a functional data analysis are themselves functional.

Consider the problem of estimating a probability density function p to describe the distribution of a sample of observations $x_1, \ldots, x_n$. The classic approach to this problem is to propose, after considering basic principles and closely studying the data, a *parametric model* with values $p(x|\boldsymbol{\theta})$ defined by a fixed and usually small number of parameters in the vector $\boldsymbol{\theta}$. For example, we might consider the normal distribution as appropriate for the data, so that $\boldsymbol{\theta} = (\mu, \sigma^2)'$. The parameters themselves are usually chosen to be descriptors of the shape of the density, as in location and spread for the normal density, and are therefore the focus of the analysis.

But suppose that we do not want to assume in advance one of the many textbook density functions because, perhaps, none of them seem to capture features of the behavior of the data that we can see in histograms and other graphical displays. *Nonparametric density* estimation methods assume only smoothness, and permit as much flexibility in the estimated $p(x)$ as the data require. To be sure, parameters are often involved, as in the density estimation method of Chapter 6, but the number of parameters is not fixed in advance of the data analysis, and our attention is focussed on the function p itself rather than on the estimated parameter values. Much of the technology for estimation of smooth *functional parameters* was originally developed and honed in the density estimation context, and Silverman (1986) can be consulted for further details.

Psychometrics or mental test theory also relies heavily on functional models for seemingly nonfunctional data. The data are usually zeros and ones indicating unsuccessful and correct answers to test items, but the model consists of a set of *item response functions*, one per test item, displaying the smooth relationship between the probability of success on an item and a presumed latent ability continuum. Figure 1.5 shows three such functional parameters for a test of mathematics estimated by the functional data analytic methods reported in Rossi, Wang and Ramsay (2002).

1.3 Some functional data analyses

Data in many fields come to us through a process naturally described as functional. To turn to a completely different context, consider Figure 1.6, where the mean monthly temperatures for four Canadian weather stations are plotted. It also shows estimates of the corresponding smooth temperature functions presumed to generate the observations. Montreal, with the warmest summer temperature, has a temperature pattern that appears to

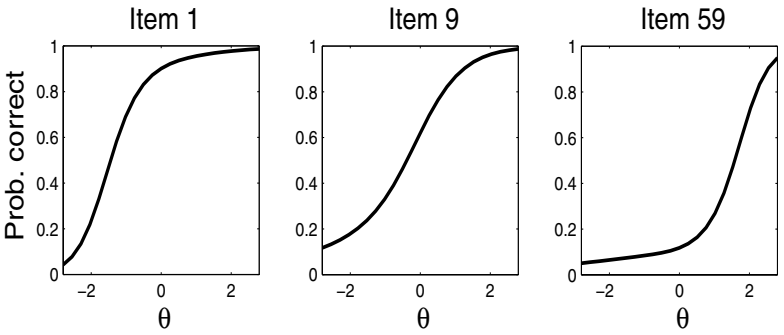

Figure 1.5. Each panel shows an item response function relating an examinee's position θ on a latent ability continuum to the probability of getting a test item in a mathematics test correct.

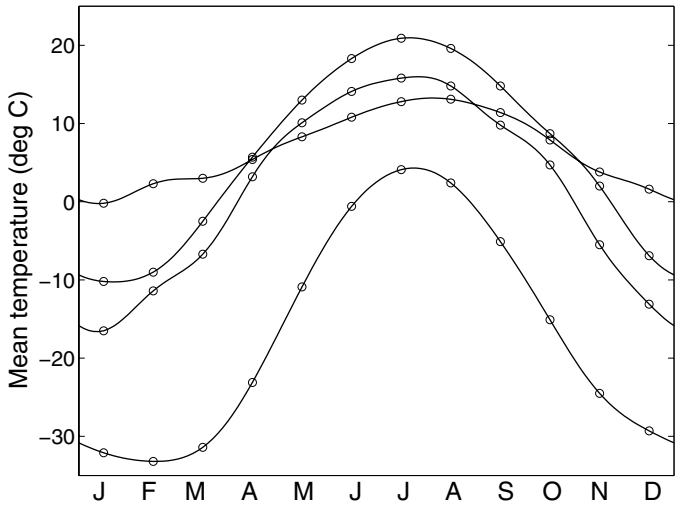

Figure 1.6. Mean monthly temperatures for the Canadian weather stations. In descending order of the temperatures at the start of the year, the stations are Prince Rupert, Montreal, Edmonton, and Resolute.

be nicely sinusoidal. Edmonton, with the next warmest summer temperature, seems to have some distinctive departures from sinusoidal variation that might call for explanation. The marine climate of Prince Rupert is evident in the small amount of annual variation in temperature, and Resolute has bitterly cold but strongly sinusoidal temperature.

One expects temperature to be primarily sinusoidal in character, and certainly periodic over the annual cycle. There is some variation in phase,

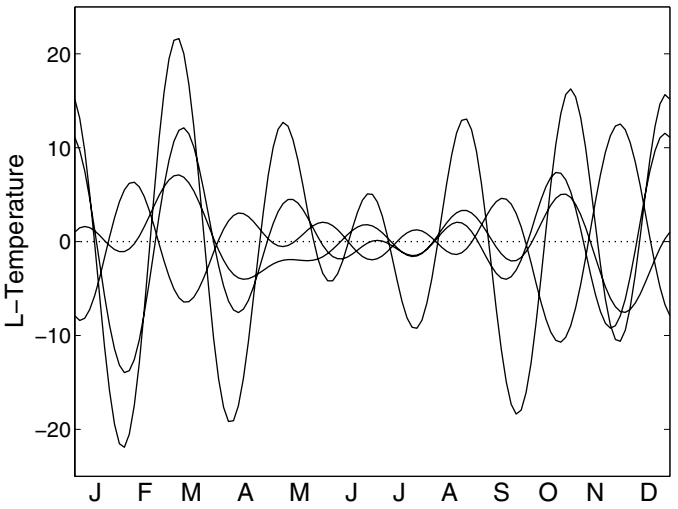

Figure 1.7. The result of applying the differential operator $L = (\pi/6)^2 D + D^3$ to the estimated temperature functions in Figure 1.6. If the variation in temperature were purely sinusoidal, these curves would be exactly zero.

because the coldest day of the year seems to be later in Montreal and Resolute than in Edmonton and Prince Rupert. Consequently, a model of the form

$$\texttt{Temp}_i(t) \approx c_{i1} + c_{i2}\sin(\pi t/6) + c_{i3}\cos(\pi t/6) \tag{1.1}$$

should do rather nicely for these data, where $\texttt{Temp}_i$ is the temperature function for the ith weather station, and (c_{i1}, c_{i2}, c_{i3}) is a vector of three parameters associated with that station.

In fact, there are clear departures from sinusoidal or simple harmonic behavior. One way to see this is to compute the function

$$L\texttt{Temp} = (\pi/6)^2 D\texttt{Temp} + D^3\texttt{Temp}. \tag{1.2}$$

As we have already noted in Section 1.1, the notation $D^m\texttt{Temp}$ means "take the mth derivative of function Temp," and the notation $L\texttt{Temp}$ stands for the function which results from applying the linear differential operator $L = (\pi/6)^2 D + D^3$ to the function Temp. The resulting function, $L\texttt{Temp}$, is often called a *forcing function.* Now, if a temperature function is truly sinusoidal, then $L\texttt{Temp}$ should be exactly zero, as it would be for any function of the form (1.1). That is, it would conform to the *differential equation*

$$D^3\texttt{Temp} = -(\pi/6)^2 D\texttt{Temp}.$$

But Figure 1.7 indicates that the functions $L\texttt{Temp}_i$ display systematic features that are especially strong in the spring and autumn months. Put

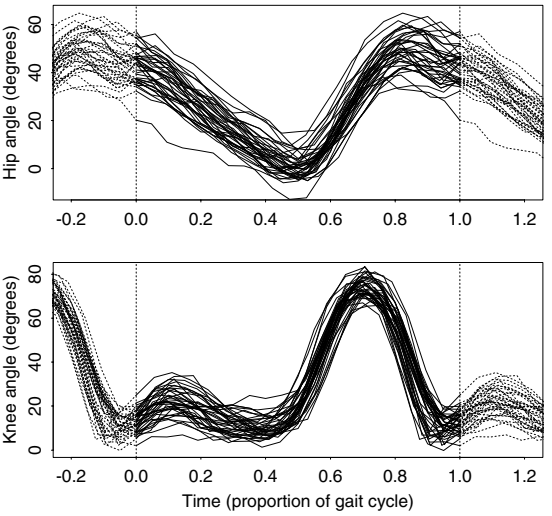

Figure 1.8. The angles in the sagittal plane formed by the hip and by the knee as 39 children go through a gait cycle. The interval $[0, 1]$ is a single cycle, and the dotted curves show the periodic extension of the data beyond either end of the cycle.

another way, temperature at a particular weather station can be described as the solution of the *nonhomogeneous* differential equation corresponding to $L\texttt{Temp} = u$, where the forcing function u can be viewed as input from outside of the system, or an exogenous influence. Meteorologists suggest, for example, that these spring and autumn effects are partly due to the change in the reflectance of land when snow or ice melts, and this would be consistent with the fact that the least sinusoidal records are associated with continental stations well separated from large bodies of water.

Here, the point is that we may often find it interesting to remove effects of a simple character by applying a differential operator, rather than by simply subtracting them. This exploits the intrinsic smoothness in the process, and long experience in the natural and engineering sciences suggests that this may get closer to the underlying driving forces at work than just adding and subtracting effects, as one routinely does in multivariate data analysis. We will consider this idea in depth beginning with Chapter 18.

Functional data are often multivariate in a different sense. Our third example is in Figure 1.8. The Motion Analysis Laboratory at Children's Hospital, San Diego, collected these data, which consist of the angles formed by the hip and knee of each of 39 children over each child's gait cycle. See Olshen et al. (1989) for full details. Time is measured in terms of the

individual gait cycle, so that every curve is given for values of t in $[0, 1]$. The cycle begins and ends at the point where the heel of the limb under observation strikes the ground. Both sets of functions are periodic, and are plotted as dotted curves somewhat beyond the interval for clarity. We see that the knee shows a two-phase process, while the hip motion is single-phase. What is harder to see is how the two joints interact; of course the figure does not indicate which hip curve is paired with which knee curve, and among many other things this example demonstrates the need for graphical ingenuity in functional data analysis.

Figure 1.9 shows the gait cycle for a single child by plotting knee angle against hip angle as time progresses round the cycle. The periodic nature of the process implies that this forms a closed curve. Also shown for reference purposes is the same relationship for the average across the 39 children. Now we see an interesting feature: a cusp occurring at the heel strike. The angular velocity is clearly visible in terms of the spacing between numbers, and it varies considerably as the cycle proceeds. The child whose gait is represented by the solid curve differs from the average in two principal ways. First, the portion of the gait pattern in the C–D part of the cycle shows an exaggeration of movement relative to the average, and second, in the part of the cycle where the hip is most bent, the amount by which the hip is bent is markedly less than average; interestingly, this is not accompanied by any strong effect on the knee angle. The overall shape of the cycle for the particular child is rather different from the average. The exploration of variability in these functional data must focus on features such as these.

Finally, in this introduction to types of functional data, we must not forget that they may come to our attention as full-blown functions, so that each record may consist of functions observed, for all practical purposes, everywhere. Sophisticated on-line sensing and monitoring equipment is now routinely used in research in medicine, seismology, meteorology, physiology, and many other fields.

1.4 The goals of functional data analysis

The goals of functional data analysis are essentially the same as those of any other branch of statistics. They include the following aims:

- to represent the data in ways that aid further analysis
- to display the data so as to highlight various characteristics
- to study important sources of pattern and variation among the data
- to explain variation in an outcome or dependent variable by using input or independent variable information

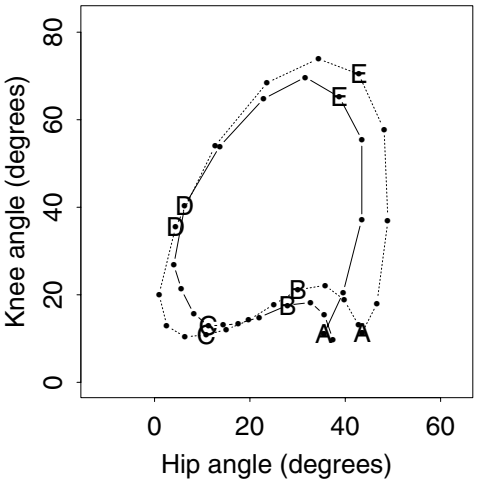

Figure 1.9. Solid line: The angles in the sagittal plane formed by the hip and by the knee for a single child plotted against each other. Dotted line: The corresponding plot for the average across children. The points indicate 20 equally spaced time points in the gait cycle, and the letters are plotted at intervals of one-fifth of the cycle, with A marking the heel strike.

- to compare two or more sets of data with respect to certain types of variation, where two sets of data can contain different sets of replicates of the same functions, or different functions for a common set of replicates.

Subsequent chapters explore each of these themes, and they are introduced only briefly here.

Each of these activities can be conducted with techniques appropriate to certain goals. Another way to characterize the strategy in a data analysis is as *exploratory*, *confirmatory*, or *predictive*. In exploratory mode, the questions put to the data tend to be rather open-ended in the sense that one expects the right technique to reveal new and interesting aspects of the data, as well as to shed light on known and obvious features. Exploratory investigations tend to consider only the data at hand, with less concern for statements about larger issues such as characteristics of populations or events not observed in the data. Confirmatory analyses, on the other hand, tend to be inferential and to be determined by specific questions about the data. Some type of structure is assumed to be present in the data, and one wants to know whether certain specific statements or hypotheses can be considered confirmed by the data. The dividing line between exploratory

and confirmatory analyses tends to be the extent to which probability theory is used, in the sense that most confirmatory analyses are summarized by one or more probability statements. Predictive studies are somewhat less common, and focus on using the data at hand to make a statement about unobserved states, such as the future.

Functional principal components and canonical correlation analyses are mainly exploratory methods, and are covered in Chapters 8 to 11. Functional linear models, on the other hand, are often used in a confirmatory way, and in 12 to 17 we introduce confidence interval estimation. In general, prediction is beyond our scope, and is only considered here and there.

1.5 The first steps in a functional data analysis

1.5.1 Data representation: smoothing and interpolation

Assuming that a functional datum for replication i arrives as a set of discrete measured values, $y_{i1}, \ldots, y_{in}$, the first task is to convert these values to a function x_i with values $x_i(t)$ computable for any desired argument value t. If the discrete values are assumed to be errorless, then the process is *interpolation*, but if they have some observational error that needs removing, then the conversion from discrete data to functions may involve *smoothing*.

Chapters 3 to 6 offer a survey of these procedures. The *roughness penalty* smoothing method discussed in Chapter 5 will be used much more broadly in many contexts throughout the book, not merely for the purpose of estimating a function from a set of observed values. The daily precipitation data for Prince Rupert, one of the wettest places on the continent, is shown in Figure 1.10. The curve in the figure, which seems to capture the smooth variation in precipitation, was estimated using a penalty on the harmonic acceleration as measured by the differential operator (1.2).

The gait data in Figure 1.8 were converted to functions by the simplest of interpolation schemes: joining each pair of adjacent observations by a straight line segment. This approach would be inadequate if we require derivative information. However, one might perform a certain amount of smoothing while still respecting the periodicity of the data by fitting a Fourier series to each record: A constant plus three pairs of sine and cosine terms does a reasonable job for these data. The growth data in Figure 1.1 and the temperature data in Figure 1.6 were smoothed using smoothing splines, and this more sophisticated technique also provides high quality derivative information.

There are often conceptual constraints on the functions that we estimate. For example, a smooth of precipitation such as that in Figure 1.10 should logically never be negative. There is no danger of this happening for a station as moist as this, but a smooth of the data in Resolute, the driest

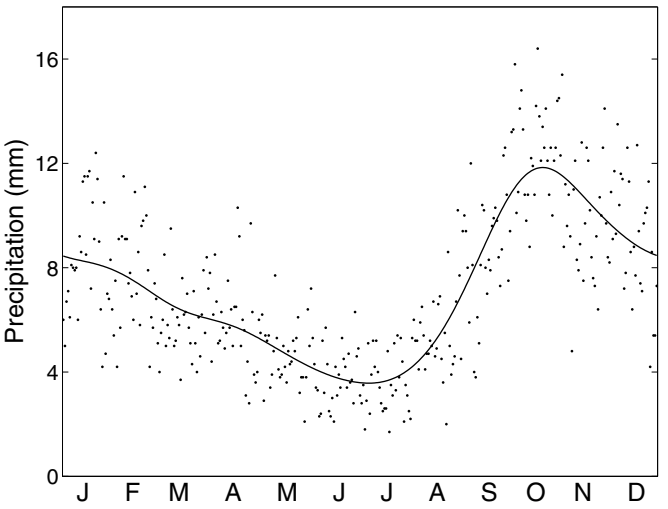

Figure 1.10. The points indicate average daily rainfall at Prince Rupert on the northern coast of British Columbia. The curve was fit to these data using a roughness penalty method.

place that we have data for, can easily violate this constraint. The growth curve fits should be strictly increasing, and we shall see that imposing this constraint results in a rather better estimate of the acceleration curves that we saw in Figure 1.2. Chapter 6 shows how to fit a variety of constrained functions to data.

1.5.2 Data registration or feature alignment

Figure 1.11 shows some biomechanical data. The curves in the figure are twenty records of the force exerted on a meter during a brief pinch by the thumb and forefinger. The subject was required to maintain a certain background force on a force meter and then to squeeze the meter aiming at a specified maximum value, returning afterwards to the background level. The purpose of the experiment was to study the neurophysiology of the thumb–forefinger muscle group. The data were collected at the MRC Applied Psychology Unit, Cambridge, by R. Flanagan; see Ramsay, Wang and Flanagan (1995).

These data illustrate a common problem in functional data analysis. The start of the pinch is located arbitrarily in time, and a first step is to align the records by some shift of the time axis. In Chapter 7 we take up the question of how to estimate this shift, and how to go further if necessary to estimate record-specific linear transformations of the argument, or even nonlinear transformations.

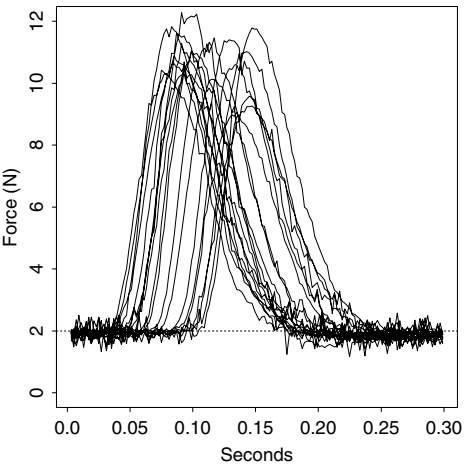

Figure 1.11. Twenty recordings of the force exerted by the thumb and forefinger where a constant background force of two newtons was maintained prior to a brief impulse targeted to reach 10 newtons. Force was sampled 500 times per second.

1.5.3 Data display

Displaying the results of a functional data analysis can be a challenge. With the gait data in Figures 1.8 and 1.9, we have already seen that different displays of data can bring out different features of interest, and that the standard plot of $x(t)$ against t is not necessarily the most informative. It is impossible to be prescriptive about the best type of plot for a given set of data or procedure, but we shall give illustrations of various ways of plotting the results. These are intended to stimulate the reader's imagination rather than to lay down rigid rules.

1.5.4 Plotting pairs of derivatives

Helpful clues to the processes giving rise to functional data can often be found in the *relationships* between derivatives. For example, two functions exhibiting simple derivative relationships are frequently found as strong influences in functional data: the exponential function, $f(t) = C_1 + C_2 e^{\alpha t}$, satisfies the differential equation

$$Df = -\alpha(f - C_1)$$

and the sinusoid $f(t) = C_1 + C_2 \sin[\omega(t-\tau)]$ with phase constant τ satisfies

$$D^2 f = -\omega^2(f - C_1).$$

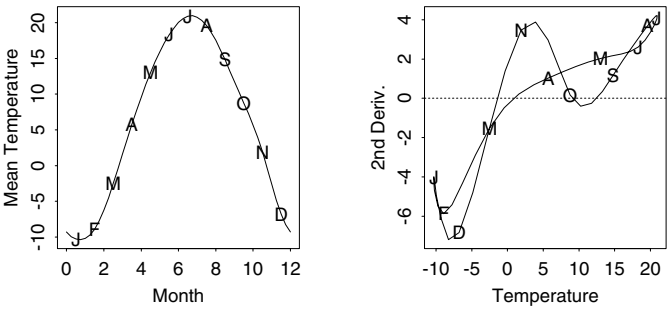

Figure 1.12. The left panel gives the annual variation in mean temperature at Montreal. The times of the mid-months are indicated by the first letters of the months. The right panel displays the relationship between the second derivative of temperature and temperature less its annual mean. Strictly sinusoidal or harmonic variation in temperature would imply a linear relationship.

Plotting the first or second derivative against the function value explores the possibility of demonstrating a linear relationship corresponding to one of these differential equations. Of course, it is usually not difficult to spot these types of functional variation by plotting the data themselves. However, plotting the higher derivative against the lower is often more informative, partly because it is easier to detect departures from linearity than from other functional forms, and partly because the differentiation may expose effects not easily seen in the original functions.

Consider, for example, the variation in mean temperature `Temp` at Montreal displayed in the left panel of Figure 1.12. Casual inspection does indeed suggest a strongly sinusoidal relationship between temperature and month, but the right panel shows that things are not so simple. Although there is a broadly linear relationship between $-D^2$`Temp` and `Temp` after subtracting the mean annual temperature, there is obviously an additional systematic trend, which is more evident in the summer through winter months than in the spring. This plot greatly enhances the small departures from sinusoidal behavior, and invites further attention.

Figure 1.13 plots the estimated derivatives for the logarithm of the U. S. nondurable goods index shown in Figure 1.3 for the year 1964. The second derivative or acceleration on the vertical axis is plotted against the first derivative or velocity on the horizontal axis in what is called a *phase plane plot.* The plot focuses attention on the interplay between Dx and D^2x by eliminating the explicit role of argument t, and reveals a fascinating cyclic structure that we will learn how to interpret in Chapter 2. Plotting derivatives as well as curve values is an essential part of functional data analysis.

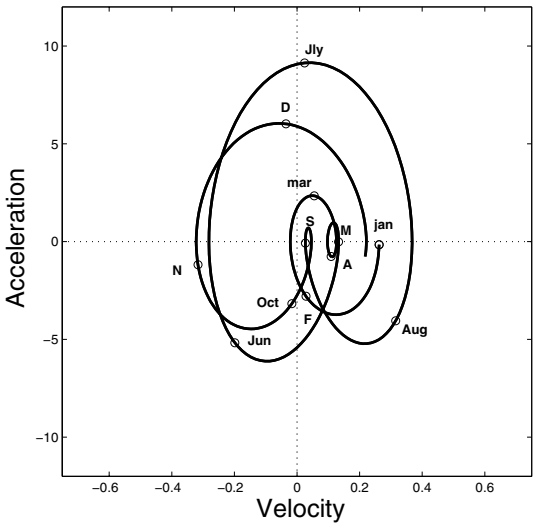

Figure 1.13. A phase plane plot of the first two derivatives of the logarithm of the U. S. nondurable goods manufacturing index in Figure 1.3 over 1964.

1.6 Exploring variability in functional data

The examples considered so far offer a glimpse of ways in which the variability of a set of functional data can be interesting, but there is a need for more detailed and sophisticated ways of investigating variability, and these are a major theme of this book.

1.6.1 Functional descriptive statistics

Any data analysis begins with the basics: Estimating means and standard deviations. Functional versions of these elementary statistics are given in Chapter 2. But what is elementary for univariate and multivariate data turns out to be not always so simple for functional data. Chapter 7 returns to the functional data summary problem, and shows that *curve registration* or feature alignment may have to be applied in order to separate *amplitude variation* from *phase variation* before these statistics are used.

1.6.2 Functional principal components analysis

Most sets of data display a small number of dominant or substantial modes of variation, even after subtracting the mean function from each observation. An approach to identifying and exploring these, set out in Chapter 8, is to adapt the classical multivariate procedure of principal components

analysis to functional data. In Chapter 9, techniques of smoothing or regularization are incorporated into the functional principal components analysis itself, thereby demonstrating that smoothing methods have a far wider rôle in functional data analysis than merely in the initial step of converting discrete observations to functional form. In Chapter 10, we show that functional principal components analysis can be made more selective and informative by considering specific types of variation in a special way. For example, we shall see that estimating a small shift of time for each temperature record and studying its variation will give a clearer understanding of record-to-record temperature variability.

1.6.3 Functional canonical correlation

How do two or more sets of records covary or depend on one another? As we saw in the cross-correlation plots, this is a question to pose for gait data, because relationships between record-to-record variation in hip angle and knee angle seem likely.

The functional linear modelling framework approaches this question by considering one of the sets of functional observations as a covariate and the other as a response variable, but in many cases, such as the gait data, it does not seem reasonable to impose this kind of asymmetry, and we shall develop two rather different methods that treat both sets of variables in an even-handed way. One method, described in Section 8.5, essentially treats the pair $(\texttt{Hip}_i, \texttt{Knee}_i)$ as a single vector-valued function, and then extends the functional principal components approach to perform an analysis. Chapter 11 takes another approach, a functional version of canonical correlation analysis, identifying components of variability in each of the two sets of observations which are highly correlated with one another.

For many of the methods we discuss, a naïve approach extending the classical multivariate method will usually give reasonable results, though regularization will often improve these. However, when a linear predictor is based on a functional observation, and also in functional canonical correlation analysis, regularization is not an optional extra but is an intrinsic and necessary part of the analysis; the reasons are discussed in Chapters 11, 15 and 16.

1.7 Functional linear models

The classical techniques of linear regression, analysis of variance, and linear modelling all investigate the way in which variability in observed data can be accounted for by other known or observed variables. They can all be placed within the framework of the general linear model

$$y = \mathbf{Z}\beta + \epsilon \tag{1.3}$$

where, in the simplest case, y is typically a vector of observations, β is a parameter vector, $\mathbf{Z}$ is a matrix that defines a linear transformation from parameter space to observation space, and ϵ is an error vector with mean zero. The design matrix $\mathbf{Z}$ incorporates observed covariates or independent variables.

To extend these ideas to the functional context, we retain the basic structure (1.3) but allow more general interpretations of the symbols within it. For example, we might ask of the Canadian weather data:

- If each weather station is broadly categorized as being Atlantic, Pacific, Continental or Arctic, in what way does the geographical category characterize the detailed temperature profile `Temp` and account for the different profiles observed? In Chapter 12 we introduce a functional analysis of variance methodology, where both the parameters and the observations become functions, but the matrix $\mathbf{Z}$ remains the same as in the classical multivariate case.
- Could a temperature record `Temp` be used to predict the logarithm of total annual precipitation? In Chapter 15 we extend the idea of linear regression to the case where the independent variable, or covariate, is a function, but the response variable (log total annual precipitation in this case) is not.
- Can the temperature record `Temp` be used as a predictor of the entire precipitation profile, not merely the total precipitation? This requires a fully functional linear model, where all the terms in the model have more general form than in the classical case. This topic is considered in Chapters 14 and 16.
- We considered earlier the many roles that derivatives play in functional data analysis. In the functional linear model, we may use derivatives as dependent and independent variables. Chapter 17 is a first look at this idea, and sets the stage for the following chapters on differential equations.

1.8 Using derivatives in functional data analysis

In Section 1.3 we have already had a taste of the ways in which derivatives and linear differential operators are useful in functional data analysis. The use of derivatives is important both in extending the range of simple graphical exploratory methods, and in the development of more detailed methodology. This is a theme that will be explored in much more detail in Chapters 18, 19 and 21, but some preliminary discussion is appropriate here.

Chapter 19 takes up the question, novel in functional data analysis, of how to use derivative information in studying components of variation. An

approach called *principal differential analysis* identifies important variance components by estimating a linear differential operator that will annihilate them. Linear differential operators, whether estimated from data or constructed from external modelling considerations, also play an important part in developing regularization methods more general than those in common use. Some of their aspects and advantages will be discussed in Chapter 21.

1.9 Concluding remarks

The last chapter of the book, Chapter 22, includes a discussion of some historical perspectives and bibliographic references not included in the main part of our development.

In the course of the book, we shall describe a considerable number of techniques and algorithms, to explain how the methodology we develop can actually be used in practice. We shall also illustrate our methodology on a variety of data sets drawn from various fields, including where appropriate the examples we have already introduced in this chapter. However, it is not our intention to provide a cook-book for functional data analysis.

In broad terms, we have a grander aim: to encourage readers to think about and understand functional data in a new way. The methods we set out are hardly the last word in approaching the particular problems, and we believe that readers will gain more benefit by using the principles we have laid down than by following our own suggestions to the letter.

For those who would like access to the software we have used ourselves, a selection is available on the website:

`http://www.functionaldata.org`

This website will also be used to publicize related and future work by the authors and others, and to make available the data sets referred to in the book that we are permitted to release publicly.

2
Tools for exploring functional data

2.1 Introduction

This chapter reviews topics that are notational and conceptual background to our main development of functional data analysis beginning in Chapter 3.

Our notation will already be familiar to many readers, but some will welcome a review, and others will encounter the notation that we use here for the first time. We have tried hard to avoid using notation other than what is familiar to statisticians and routine in calculus courses.

We will draw rather heavily on your expertise in matrix analysis and multivariate statistics, and you may want to consult Section A.7, which reviews some matrix algebra tools that we will need within framework of the multivariate linear model. This brief account is relevant here because, in fact, most of our functional data analyses and models will be converted to equivalent matrix formulations through the device of representing functions by basis function expansions, a topic that comes up in the next chapter. Also discussed in the Appendix are matrix decompositions, projections, and the constrained maximization of quadratic forms.

After some remarks on notation in Section 2.2, we consider the basic anatomy of a function in Section 2.4. What features in a function might be of interest? How are functions different from vectors? How do we quantify the amount of information that is needed to specify a function? What does it mean to say that a function is "smooth"?

2.2 Some notation

2.2.1 Scalars, vectors, functions and matrices

The reader should be warned that we try to use notation that brings out the basic structure of what is being done, and that this may entail the use of conventions that are at first sight a little unfamiliar. For example, we do not usually bother to distinguish in our notation between scalar quantities (numbers) and functions. This means that a single symbol x can refer to a scalar or to a function. The nervous reader should be assured that this convention is only used to clarify, rather than confuse, the discussion! In general, the context should always make clear when a symbol refers to a scalar or function. This emphasizes our guiding intuition that a function is to be considered as single unitary entity. The perhaps more familiar notation $x(t)$ refers to the *value* of function x at argument value t rather to the entire function.

On the other hand, in this edition we adhere to the usual practice of showing vectors as boldface lower case letters such as $\mathbf{x}$, and matrices in boldface upper case. We always use the notation $\mathbf{x}'$ for the transpose of a vector $\mathbf{x}$. We need matrix algebra at every turn, and it seems better not to ask readers used to bold symbols to do without this device.

If x is a vector or function, its elements or values x_i or $x(t)$ are usually scalars, but sometimes it is appropriate for the individual x_i or $x(t)$ to be a vector, and then we use boldface. Also, it is handy to use the notation $x(\mathbf{t})$ to denote the vector containing the values of function x at each of the argument values in vector $\mathbf{t}$.

It is often clearer to use longer strings of letters in a distinctive font to denote quantities more evocatively than standard notation allows. For example, we use names such as

- `Temp` for a temperature record,
- `Knee` for a knee angle
- `LMSSE` for a squared error fitting criterion for a linear model, and
- `RSQ` for a squared correlation measure.

2.2.2 Derivatives and integrals

Our notation for the derivative of order m of a function x is $D^m x$; this produces cleaner formulas than $d^m x/dt^m$. It stresses that differentiation is an *operator* that acts on a function x to produce another function Dx. Of course, $D^0 x$ refers to x itself. The superscript method works neatly when we consider derivatives of derivatives, and also when we use $D^{-1}x$ to refer to the indefinite integral of x, since $D^1 D^{-1} x = D^0 x = x$ as expected. We

also use operators that act on functions in other ways, and it is convenient to use a consistent notation.

The definite integral $\int_a^b x(t)\,dt$ will often be shortened to $\int x$ when the context makes clear both the limits of integration a and b and the variable t over which the integration takes place.

2.2.3 Inner products

Inner product notation for functions, as in

$$\langle x, y\rangle = \int x(t)y(t)\,dt, \tag{2.1}$$

was used much more frequently in the first edition than in this. We found that many readers had difficulty coping with the notation, and we also found that we could do without it nearly everywhere. Nevertheless, inner product notation is a powerful tool, and if a reader wishes to learn more, the Appendix offers a summary and some illustrations. We will use rather more frequently the notation $\|x\|$ for the *norm* of x, a measure of its size. The most common type of norm, called the L_2 norm, is related to the inner product through the relation

$$\|x\|^2 = \langle x, x\rangle = \int x^2(t)\,dt\ .$$

The Appendix contains additional material on inner product notation.

2.2.4 Functions of functions

Functions are often themselves arguments for other functions. For example, in Chapter 7 we will consider a nonlinear transformation $h(t)$ of argument t that maps t on to the same interval that it occupies. That is, for example, time is transformed nonlinearly into time. We then need the function whose values are $x[h(t)]$, which we can indicate by x^*. In this case, we use the *functional composition* notation $x^* = x \circ h$. The function value $x^*(t)$ is indicated by $(x \circ h)(t)$.

Moreover, in the same chapter, we will use the *inverse* function which results from solving the relation $h(g) = t$ for g given t. This function, having values $g(t)$, is denoted by h^{-1}. This does not mean, of course, the reciprocal of h, which we simply indicate as $1/h$ on the rare occasion that we need it. In fact, the functional compositions $h \circ h^{-1}$ and $h^{-1} \circ h$ satisfy

$$(h \circ h^{-1})(t) = (h^{-1} \circ h)(t) = t$$

and, in functional composition sense, therefore h and h^{-1} cancel one another.

Another type of function transforms one function into another; that is, takes an entire function as its argument rather than a function value. The

most important example is the transform D that transforms function x into its derivative Dx. The indefinite integral is another example, and as are the arithmetic operations applied to functions. We call such functional transformations *operations* or *operators*.

2.3 Summary statistics for functional data

2.3.1 Functional means and variances

The classical summary statistics for univariate data familiar to students in introductory statistics classes apply equally to functional data. The mean function with values

$$\bar{x}(t) = N^{-1} \sum_{i=1}^{N} x_i(t)$$

is the average of the functions point-wise across replications. Similarly the variance function `var` has values

$$\mathtt{var}_X(t) = (N-1)^{-1} \sum_{i=1}^{N} [x_i(t) - \bar{x}(t)]^2,$$

and the standard deviation function is the square root of the variance function.

Figure 2.1 displays the mean and standard deviation functions for the aligned pinch force data. We see that the mean force looks remarkably like a number of probability density functions well known to statisticians, and in fact the relationship to the lognormal distribution has been explored by Ramsay, Wang and Flanagan (1995). The standard deviation of force seems to be about 8% of the mean force over most of the range of the data.

2.3.2 Covariance and correlation functions

The *covariance function* summarizes the dependence of records across different argument values, and is computed for all t_1 and t_2 by

$$\mathtt{cov}_X(t_1, t_2) = (N-1)^{-1} \sum_{i=1}^{N} \{x_i(t_1) - \bar{x}(t_1)\}\{x_i(t_2) - \bar{x}(t_2)\}.$$

The associated *correlation function* is

$$\mathtt{corr}_X(t_1, t_2) = \frac{\mathtt{cov}_X(t_1, t_2)}{\sqrt{\mathtt{var}_X(t_1)\mathtt{var}_X(t_2)}}.$$

These are the functional analogues of the variance–covariance and correlation matrices, respectively, in multivariate data analysis.

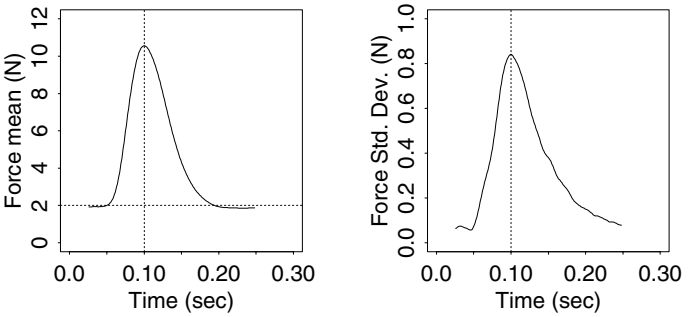

Figure 2.1. The mean and standard deviation functions for the 20 pinch force observations in Figure 1.11 after they were aligned or registered.

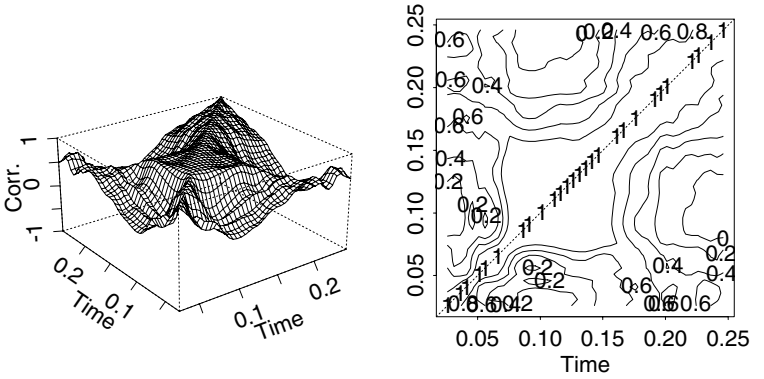

Figure 2.2. The left panel is a perspective plot of the bivariate correlation function values $r(t_1, t_2)$ for the pinch force data. The right panel shows the same surface by contour plotting. Time is measured in seconds.

Figure 2.2 displays the correlation function of the pinch force data, both as a surface over the plane of possible pairs of times (t_1, t_2) and also as a set of level contours.

Our experience with perspective and contour displays of correlation suggests that not everyone encountering them for the first time finds them easy to understand. Here is one strategy: The diagonal running from lower left to upper right in the contour or from front to back in the perspective plot of the surface contains the unit values that are the correlations between identical or very close time values. Directions perpendicular to this ridge of unit correlation indicate how rapidly the correlation falls off as two argument values separate. For example, one might locate a position along the unit ridge associated with argument value t, and then moving perpendicularly from this point shows what happens to the correlation between

values at time pair $(t-\delta, t+\delta)$ as the perpendicular distance δ increases. In the case of the pinch force data, we note that the correlation falls off slowly for values on either side of the time 0.1 of maximum force, but declines much more rapidly in the periods before and after the impulse. This suggests a two-phase system, with fairly erratic uncoupled forces in the constant background force phase, but with tightly connected forces during the actual impulse. In fact, it is common to observe low correlations or rapid fall-off when a system is in a resting or ballistic state free from any outside input, but to show strong correlations, either positive and negative, when exogenous influences apply.

2.3.3 Cross-covariance and cross-correlation functions

In the case of the gait data discussed in Section 1.3, we had both hip and knee angles measured through time. In general, if we have pairs of observed functions (x_i, y_i), the way in which these depend on one another can be quantified by the *cross-covariance* function

$$\mathtt{cov}_{X,Y}(t_1,t_2) = (N-1)^{-1}\sum_{i=1}^{N}\{x_i(t_1)-\bar{x}(t_1)\}\{y_i(t_2)-\bar{y}(t_2)\}.$$

or the *cross-correlation* function

$$\mathtt{corr}_{X,Y}(t_1,t_2) = \frac{\mathtt{cov}_{X,Y}(t_1,t_2)}{\sqrt{\mathtt{var}_X(t_1)\mathtt{var}_Y(t_2)}}.$$

Figure 2.3 displays the correlation and cross-correlation functions for the gait data. In each of the four panels, t_1 is plotted along the horizontal axis and t_2 along the vertical axis. The top left panel shows a contour plot of the correlation function $\mathtt{corr}_{\mathtt{Hip}}(t_1,t_2)$ for the hip angles alone, and the bottom right panel shows the corresponding plot for the knee angles. The cross-correlation functions $\mathtt{corr}_{\mathtt{Hip},\mathtt{Knee}}$ and $\mathtt{corr}_{\mathtt{Knee},\mathtt{Hip}}$ are plotted in the top right and bottom left panels respectively; since, in general, $\mathtt{corr}_{X,Y}(t_1,t_2) = \mathtt{corr}_{Y,X}(t_2,t_1)$, these are transposes of one another, in that each is the reflection of the other about the main diagonal $t_1 = t_2$. Note that each axis is labelled by the generic name of relevant data function, `Hip` or `Knee`, rather than by the argument value t_1 or t_2.

In this figure, different patterns of variability are demonstrated by the individual correlation functions $\mathtt{corr}_{\mathtt{Hip}}$ and $\mathtt{corr}_{\mathtt{Knee}}$ for the hip and knee angles considered separately. The hips show positive correlation throughout, so that if the hip angle is larger than average at one point in the cycle it will have a tendency to be larger than average everywhere. The contours on this plot are more or less parallel to the main diagonal, implying that the correlation is approximately a function of $t_1 - t_2$ and that the variation of the hip angles can be considered as an approximately stationary process.

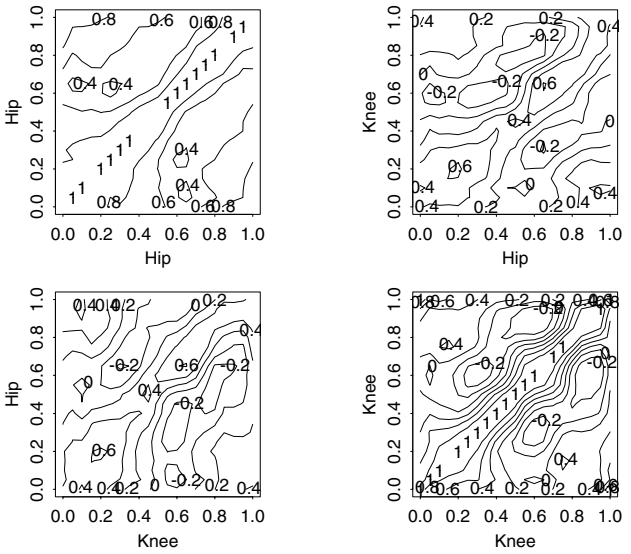

Figure 2.3. Contour plots of the correlation and cross-correlation functions for the gait data. In each panel t_1 is plotted on one axis and t_2 on the other; the legends indicate which observations are being correlated against each other.

On the other hand, the knee angles show behavior that is clearly nonstationary; the correlation between the angle at time 0.0 and time 0.3 is about 0.4, while that between times 0.3 and 0.6 is actually negative. In the middle of the cycle the correlation falls away rapidly as one moves away from the main diagonal, while at the ends of the cycle there is much longer range correlation. The hip angles show a slight, but much less marked, departure from stationarity of the same kind. These features may be related to the greater effect on the knee of external factors such as the heel strike and the associated weight placed on the joint, whereas the hip acts under much more even muscular control throughout the cycle.

The ridge along the main diagonal of the cross-correlation plots indicates that $\texttt{Hip}(t_1)$ and $\texttt{Knee}(t_2)$ are most strongly correlated when t_1 and t_2 are approximately equal, though the main ridge shows a slight reverse S shape (in the orientation of the top right panel). The analysis developed in Chapter 11 will elucidate the delays in the dependence of one joint on the other. Apart from this, there are differences in the way that the cross-correlations behave at different points of the cycle, but the cross-correlation function does not make it clear what these mean in terms of dependence between the functions.

Another example is provided by the Canadian weather data. Contour plotting in Figure 2.4 shows the correlation functions between tempera-

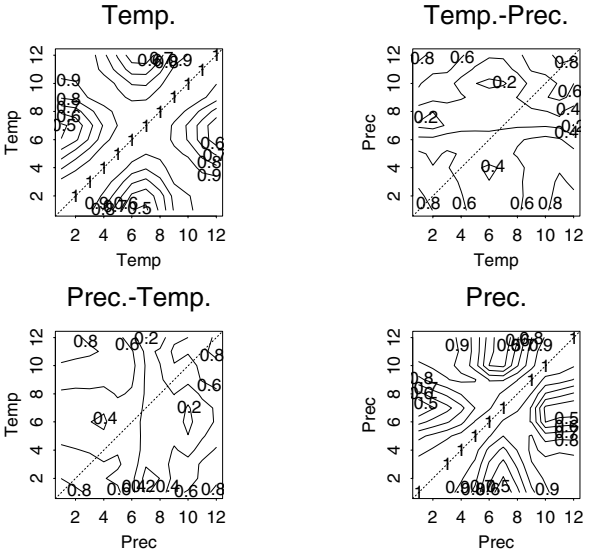

Figure 2.4. Contour plots of the correlation and cross-correlation functions for 35 Canadian weather stations for temperature and log precipitation. The cross-correlation functions are those in the upper right and lower left panels.

ture and log precipitation based on monthly data. The correlation is high for both temperature and precipitation on either side of the midsummer period, so that autumn weather tends to be highly correlated with spring weather. By contrast, winter and summer weather have a weaker correlation of around 0.5. The cross-correlations show that midsummer precipitation has a near zero correlation with temperature at any point in the year, but that midwinter temperature and midwinter precipitation are highly correlated. This is due to the fact that, in continental weather stations, both measures tend to be especially low in midwinter, whereas in marine stations, the tendency is for both temperature and precipitation to be higher.

2.4 The anatomy of a function

2.4.1 Functional features

What interests us when we consider functions such as the height acceleration curves in Figure 1.2? Certainly the *peak* and *valley* defining the pubertal growth spurt, as well as the smaller peaks at age 6 for most girls. *Crossings* of specified levels can also be important markers, such as the

age at which acceleration is zero in the middle of the pubertal growth spurt, marking out the point of peak growth velocity. *Levels* are function values that we consider significant, such as the zero level that a growth acceleration reaches after growth has stopped.

We can consider each of these functional features as events that are associated with a specific value of the argument t. That is, most features are characterized by a *location*. Many are also defined by *amplitude*, a measure of their size. For example, the height or depth of a peak or valley, respectively, is a matter of amplitude, as is the steepness with which a line crosses a specified level. Finally, events like peaks and valleys are also characterized by *widths*; the first peak in the knee angle curves in Figure 1.8 is narrower than the second peak.

In this sense, levels are one-dimensional events, crossings are two-dimensional, and peaks and valleys are three-dimensional. That is, in ideal errorless circumstances, we would need three pieces of information to fully define a peak, namely location, amplitude, and width. This corresponds to the fact that peaks look somewhat like parabolas, which are defined by three parameters; crossings look like lines, requiring two parameters; and levels are like points.

The *dimensionality* of a functional feature tells us a great deal about how much information we will need to estimate it. For example, even a tiny bit of observational error in the data will force us to provide five rather than three function values at locations within a peak, and for data with error levels common in functional data analyses, seven to eleven values per peak would be wise.

2.4.2 Data resolution and functional dimensionality

This suggests the notation of the *resolving power* or *resolution* of a set of data. This is inversely related to the width of the narrowest event that can be estimated to our satisfaction. We mean by the phrase "high resolution data" that they can pin down small events. The resolution of a set of data can be a rather more useful concept than simply the number of observations taken.

Resolution leads in turn to the notion of the *dimensionality* of a function. Expertise in the mathematical area of functional analysis is necessary to understand this concept in depth, but it is easy to say some common sensical things about the dimension of a curve. Roughly speaking, it is the sum across functional "features" of the numbers of pieces of information that are required to define each feature or event.

We can say that the *practical dimensionality* of a function is the total amount of information required to define it to some required level. This notion inevitably depends on the goals of the functional data analysis, since it supposes that we ignore error and other sources of high frequency

variation that would increase the actual dimensionality of the function greatly.

Functions are potentially infinite dimensional. That is, a complete specification of a function x could conceivably require us to know its value $x(t)$ at each possible argument value t, and since there are an infinity of these, the dimensionality of a function can be arbitrarily large. Or, put another way, if a function can pack an infinite number of peaks and valleys within any interval, no matter how small, we will need infinite resolving power in any set of data concerning this curve. For example, the terms like "Brownian motion" and "white noise" are used to describe functions so erratic that no information is contained in $x(t)$ about the value $x(t+\delta)$, no matter how small δ is. This is somewhat depressing, because it implies that we can never collect enough data to estimate functions like these exactly.

However, in practice we work with functions that do not display so much complexity. It is more or less accepted, for example, that from 12 to 16 pieces of information, in a sense to be made precise in the next chapter, are required to describe growth curves like those in Figure 1.1. Almost always there are several ways in which we can use this much information to get about the same result, and in the growth curve literature there are several competing parametric models. But what matters is that all of the successful growth curve models seem to need at least this much information.

2.4.3 The size of a function

Something like *energy* tends govern the behavior of many functional variables, just as it does in physics. By this we mean that change requires effort or work, and typically the systems that we study can only muster a limited amount of whatever brings change per unit time. For example, even a process as seemingly chaotic as the stock market reflects, on a time scale small enough, the effort required to move money and information from one place to another. Biological systems like growing children likewise cannot make very rapid changes to their status due to the need to burn calories to bring this change about. Because on a short time scale the energy available in a system is essentially conserved, we can expect to see smooth changes, just as we will not see extremely large accelerations in mechanical systems with substantial mass.

Consequently, the dimensionality of a function is actually a measure of its *size* in the same way that its amplitude is. That is, both amplitude and dimensionality require energy to produce. For example, white noise is an infinitely large function, even if its values are always within specified limits such as $[-1, 1]$, because it would take an infinite amount of energy to produce this much variability. Similarly, what mathematicians refer to as Brownian motion is an abstraction inspired by the seemingly chaotic but actually limited movements of small particles due to collision with molecules in the medium in which they are suspended. One learns in

functional analysis, for example, that an infinite dimensional hyper-sphere of radius one is infinitely large. Statisticians are referring to something like this by the colorful phrase "the curse of dimensionality."

Dimensionality matters a great deal as a size indicator in functional data analysis. We will return to this important theme in the next chapter when we consider what the terms "noise" and "observational error" might mean in a functional sense, and when we take up the notion *multi-resolution analysis.*

2.5 Phase-plane plots of periodic effects

The two concepts of energy and of functional data having variation on more than one time scale lead to the graphical technique of plotting one derivative against another, something that we will call *phase-plane plotting.* We saw an example in Figure 1.13, and we now return to the U.S. nondurable goods manufacturing index to illustrate these ideas.

Like most economic indicators, the nondurable goods index tends to exhibit exponential increase, corresponding to percentage increases over fixed time periods. Moreover, the index tends to increase in size and volatility at the same time, so that the large relative effects surrounding the Second World War seem to be small relative to the large changes in the 1970s and 1980s, and seasonal variation in recent years dwarfs that in early years.

2.5.1 The log nondurable goods index

We prefer, therefore, to study the logarithm of this index, displayed in Figure 2.5. The log index has a linear trend with a slope of 0.016, corresponding to an annual rate of increase of 1.6%, and the sizes of the seasonal cycles are also more comparable across time. We now see that the changes in the depression and war periods are now much more substantial and abrupt than those in recent times. The growth rate is especially high from 1960 to 1975, when the baby boom was in the years of peak consumption; but in subsequent years seems to be substantially lower, perhaps because middle-aged "boomers" consume less, or possibly because the nature of the index itself has changed.

The goods index exhibits variation on four time scales:

- The longest scale is the century-long nearly linear increase in the log index, or exponential trend in the index itself.
- There are events that last a decade or more, such as the depression, the unusually rapid growth in the 1960s, and the slower growth in the last two decades.

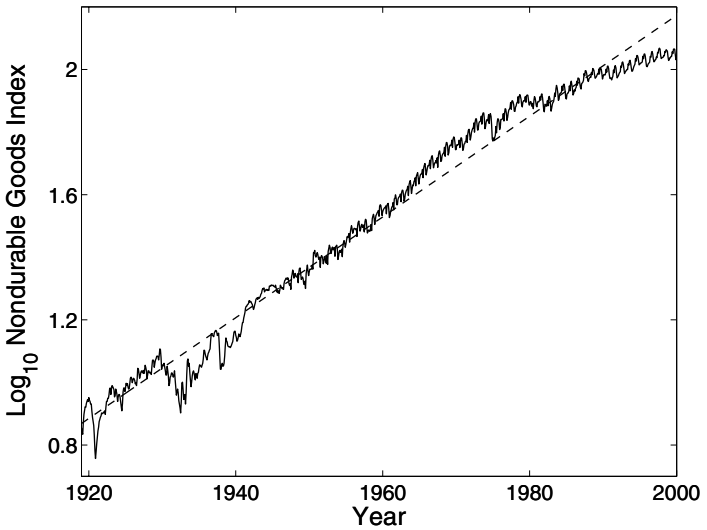

Figure 2.5. The monthly nondurable goods production of the United States shown in Figure 1.3 plotted on a logarithmic scale. The dotted straight line is estimated by least squares regression, and has a slope of 0.016, corresponding to a 1.6% increase in the index per year.

- Shorter term perturbations are also visible, such as World War II and the end of the Vietnam War in 1974.
- On the shortest scale there is seasonal variation over an annual cycle that tends to repeat itself.

A closer look at a comparatively stable period, 1964 to 1967 shown in Figure 2.6, suggests that the index varies fairly smoothly and regularly within each year. The solid line is a smooth of these data using the roughness penalty method described in Chapter 5. We now see that the variation within this year is more complex than Figure 2.5 can possibly reveal. This curve oscillates three times during the year, with the size of the oscillation being smallest in spring, larger in the summer, and largest in the autumn. In fact each year shows smooth variation with a similar amount of detail, and we now consider how we can explore these within-year patterns.

2.5.2 Phase–plane plots show energy transfer

Now that we have derivatives at our disposal, we can learn new things by studying how derivatives relate to each other. Our tool is a plot of acceleration against velocity. To see how this might be useful, consider the phase-plane plot of the function $\sin(2\pi t)$, shown in Figure 2.7. This simple function describes a basic *harmonic process*, such as the vertical position

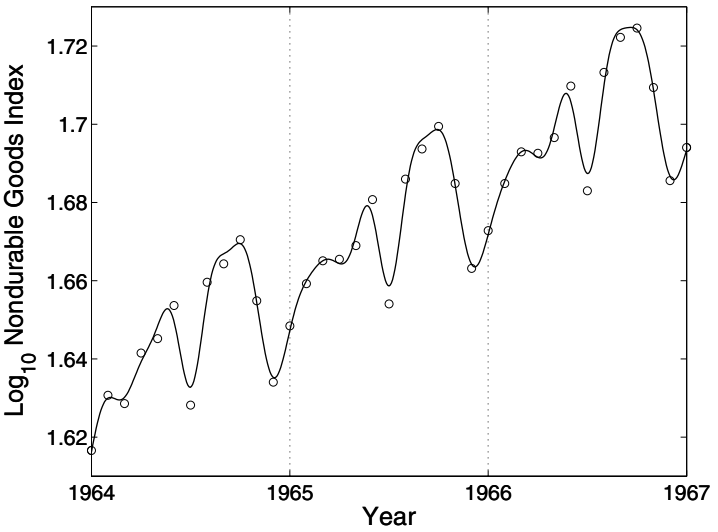

Figure 2.6. The log nondurable goods index for 1964 to 1967, a period of comparative stability. The solid line is a fit to the data using a polynomial smoothing spline. The circles indicate the value of the log index at the first of the month.

of the end of a suspended spring bouncing with a period of one time unit and starting at position zero at time $t = 0$.

The spring oscillates because energy is exchanged between two states: *potential* and *kinetic*. At times $\pi, 3\pi, \ldots$ the spring is at one or the other end of its trajectory, and the restorative force due to its stretching has brought it to a standstill. At that point, its potential energy is maximized, and so is the force, which is acting either upward (positively) or downward. Since force is proportional to acceleration, the second derivative of the spring position, $-(2\pi)^2 \sin(2\pi t)$, is also at its highest absolute value, in this case about ± 40. On the other hand, when the spring is passing through the position 0, its velocity, $2\pi \cos(2\pi t)$, is at its greatest, about ± 8, but its acceleration is zero. Since kinetic energy is proportional to the square of velocity, this is the point of highest kinetic energy. The phase-plane plot shows this energy exchange nicely, with potential energy being maximized at the extremes of Y and kinetic energy at the extremes of X.

Now harmonic processes and energy exchange are found in many situations besides mechanics. In economics, potential energy corresponds to available capital, human resources, raw material, and other resources that are at hand to bring about some economic activity, in this case the manufacture of nondurable goods. Kinetic energy corresponds to the manufacturing process in full swing, when these resources are moving along the assembly line, and the goods are being shipped out the factory door.

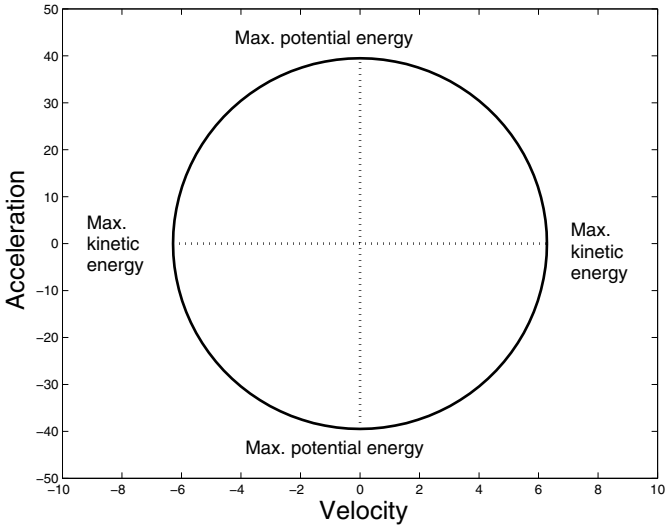

Figure 2.7. A phase-plane plot of the simple harmonic function $\sin(2\pi t)$. Kinetic energy is maximized when acceleration is 0, and potential energy is maximized when velocity is 0.

The process moves from strong kinetic to strong potential energy when the rate of change in production goes to zero. We see this, for example, after a period of rapid increase in production when labor supply and raw material stocks become depleted, and consequently potential energy is actually in a negative state. Or it happens when management winds down production because targets have been achieved, so that personnel and material resources are piling up and waiting to be used anew.

After a period of intense production, or at certain periods of crisis that we examine shortly, we may see that both potential and kinetic energy are low. This corresponds to a period when the phase-plane curve is closer to zero than is otherwise the case.

To summarize, here's what we are looking for:

- a substantial cycle;
- the size of the radius: the larger it is, the more energy transfer there is in the event;
- the horizontal location of the center: if it is to the right, there is net positive velocity, and if to the left, there is net negative velocity;
- the vertical location of the center: if it is above zero, there is a net velocity increase; if below zero, there is velocity decrease; and
- changes in the shapes of the cycles from year to year.

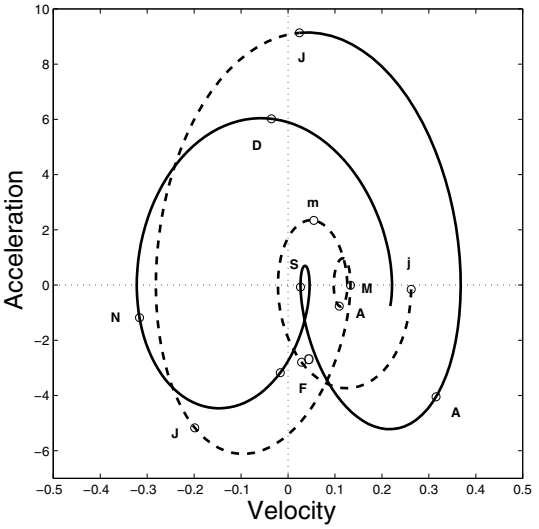

Figure 2.8. A phase-plane plot of the first derivative or velocity and the second derivative or acceleration of the smoothed log nondurable goods index for 1964. Letters indicate mid–months, with lowercase letters used for January and March. For clarity, the first half of the year is plotted as a dashed line, and the second half as a solid line.

2.5.3 The nondurable goods cycles

We use the phase-plane plot, therefore, to study the energy transfer within the economic system. We can examine the cycle within individual years, and also see more clearly how the structure of the transfer has changed throughout the twentieth century. Figure 2.8, a reproduction here of Figure 1.13, phase-plane plots the year 1964, a year in a relatively stable period for the index. To read the plot, find the lower-case "j" in the middle right of the plot, and move around the diagram clockwise, noting the letters indicating the months as you go. You will see that there are two large cycles surrounding zero, plus some small cycles that are much closer to the origin.

The largest cycle begins in mid-May (M), with positive velocity but near zero acceleration. Production is increasing linearly or steadily at this point. The cycle moves clockwise through June (first J) and passes the horizontal zero acceleration line at the end of the month, when production is now decreasing linearly. By mid-July (second J) kinetic energy or velocity is near zero because vacation season is in full swing. But potential energy or acceleration is high, and production returns to the positive kinetic/zero potential phase in early August (A), and finally concludes with a cusp at summer's end (S). At this point the process looks like it has run out of both potential and kinetic energy.

The cusp, near where both derivatives are zero, corresponds to the start of school in September, and to the beginning of the next big production cycle passing through the autumn months of October through November. Again this large cycle terminates in a small cycle with little potential and kinetic energy. This takes up the months of February and March (F and m). The tiny subcycle during April and May seems to be due to the spring holidays, since the summer and fall cycles, as well as the cusp, don't change much over the next two years, but the spring cycle cusp moves around, reflecting the variability in the timings of Easter and Passover.

To summarize, the production year in the 1960s has two large cycles swinging widely around zero, each terminating in a small cusp–like cycle. This suggests that each large cycle is like a balloon that runs out of air, the first at the beginning of school, and the second at the end of winter. At the end of each cycle, it may be that new resources must be marshalled before the next production cycle can begin.

2.6 Further reading and notes

These notes on other sources of information are intended only if you have some need to go beyond what is in this book. Otherwise, please push on to the following chapters, where we have tried to provide introductions to any concepts that you need to deal with at least the core topics for functional data analysis.

We find that inner product notation is appearing more and more often in statistics, and that it is already routinely used in engineering in fields such as signal analysis. Moore (1985) is an example of a reference oriented to applications of functional analysis that can be consulted for further information on many topics in this and subsequent chapters.

There have been many books that have used the notation of functional analysis to describe multivariate statistics, with a view to generalizing that methodology and synthesizing results within a common notational framework, but unfortunately not many that would be readable by anyone except mathematics specialists. Two references, however, have landmark qualities. Cailliez and Pagès (1976) attempted to write a text that combined high mathematics with an applied data analysis orientation, and the result was a unique and exciting approach that still merits attention for those able to read French. Our treatment of summary statistics in Section 2.3 is extended in many ways in their work. Grenander (1980) is a much more advanced book that we think of as dealing with many of the topics covered in this volume.

To see more of phase–plane plotting in action, consult Ramsay and Silverman (2002), where the method is used to show changes in the seasonal trend over longer time scales. The idea is taken directly from elementary

physics, where conservation of energy is used in so many ways. This graphical tool links naturally to differential equation models that are considered Chapter 17 and subsequently.

Since observed curves are often complex objects requiring large numbers of parameters to describe adequately, as we shall see in the next three chapters, finding ways to summarize their distribution can be a challenge. In fact, it is relatively routine to have the number of curves N rather less than the number of parameters n that must be estimated per curve. We will use principal components analysis in Chapters 8 to 10 to capture at least a few dimensions of the variation across curves. Hall and Heckman (2002) propose an ingenious technique using what they call *density ascent lines* to provide interesting summaries of the probability density function for curve data.

3

From functional data to smooth functions

3.1 Introduction

This chapter serves to introduce some ideas that are central to the next two chapters, where we will develop methods for turning raw discrete data into smooth functions.

Our goals in this chapter are:

- To understand what we mean when we refer to data as "functional".
- To explore the concept of "smoothness" of a function, and relate smoothness to the function's derivatives.
- To consider how noise or error of measurement combines with smooth functional variation to produce the observed data.

We will use linear combinations of basis functions as our main method for representing functions. The use of basis functions is a computational device well adapted to storing information about functions, and gives us the flexibility that we need combined with the computational power to fit even hundreds of thousands of data points. Moreover, it permits us to express the required calculations within the familiar context of matrix algebra.

Most of the functional analyses that we discuss can be expressed directly in terms of functional parameters using more advanced methods such as the calculus of variations and functional analysis, but we consider these approaches to be too technical to be useful to the readers that we have in

mind. Moreover, the basis function approach has not, in our experience, imposed any practical limitations on what we have needed to do.

We will consider in detail two basis function systems: The Fourier basis and the B-spline basis. The former tends to be used to describe periodic data, and the latter for functional information without any strongly cyclic variation. We will not neglect, however, several other types of basis systems, each having its own merits in certain contexts.

3.2 Some properties of functional data

The basic philosophy of functional data analysis is to think of observed data functions as single entities, rather than merely as a sequence of individual observations. The term *functional* in reference to observed data refers to the intrinsic structure of the data rather than to their explicit form. In practice, functional data are usually observed and recorded discretely as n pairs (t_j, y_j), and y_j is a snapshot of the function at time t_j, possibly blurred by measurement error. Time is so often the continuum over which functional data are recorded that we may slip into the habit of referring to t_j as such, but certainly other continua may be involved, such as spatial position, frequency, weight, and so forth.

3.2.1 What makes discrete data functional?

What would it mean for a functional observation to be known in functional form x? We do not mean that x is actually recorded for every value of t, because that would involve storing an uncountable number of values! Rather, we mean that we assume the existence of a function x giving rise to the observed data.

In addition, we usually want to declare that the underlying function x is *smooth*, so that a pair of adjacent data values, y_j and y_{j+1} are necessarily linked together to some extent and unlikely to be too different from each other. If this smoothness property did not apply, there would be nothing much to be gained by treating the data as functional rather than just multivariate.

By smooth, we usually mean that function x possesses one or more derivatives, which we indicate by Dx, D^2x, and so on, so that $D^m x$ refers to the derivative of order m, and $D^m x(t)$ is the value of that derivative at argument t. We will usually want to use the discrete data $y_j, j = 1, \ldots, n$ to estimate the function x and at the same time a certain number of its derivatives. For example, if we are tracking the position x of a moving object such as a rocket, we will want, also, to estimate its velocity Dx and its acceleration D^2x. The modelling of a system's rates of change is often

called the analysis of a system's *dynamics*. The many uses of derivatives will be a central theme of this book.

The actual observed data, however, may not be at all smooth due to the presence of what we like to call noise or measurement error. Some of this extraneous variation may indeed have all the characteristics of noise, that is, be formless and unpredictable, or it may be high-frequency variation that we could in principle model, but for practical reasons choose to ignore. Sometimes this noise level is a tiny fraction of the size of the function that it reflects, and then we say that the *signal-to-noise ratio* (S/N ratio) is high. However, higher levels of variation of the y_j's around the corresponding $x(t_j)$'s can make extracting a stable estimate of the the function and some of its derivatives a real challenge.

Most of this chapter and the next are given over to how to estimate x and some of its derivatives from noisy data.

3.2.2 Samples of functional data

In general, we are concerned with a collection or sample of functional data, rather than just a single function x. Specifically, the record or observation of the function x_i might consist of n_i pairs (t_{ij}, y_{ij}), $j = 1, \ldots, n_i$. It may be that the argument values t_{ij} are the same for each record, but they may also vary from record to record. It may be that the interval $\mathcal{T}$ over which data are collected also varies from record to record.

Normally, the construction of the functional observations x_i using the discrete data y_{ij} takes place separately or independently for each record i. Therefore, in this chapter, we will usually simplify notation by assuming that a single function x is being estimated. However, where the signal-to-noise ratio is low, or the data are sparsely sampled or few in number, it can be essential to use information in neighboring or similar curves to get more stable estimates of a specific curve.

Sometimes time t is considered cyclically, for instance when t is the time of year, and this means that the functions satisfy *periodic boundary conditions*, where the function x at the beginning of the interval $\mathcal{T}$ picks up smoothly from the values of x at the end. Data for functions which do not naturally wrap around in this way are called *non-periodic.*

Finally, a lot of functional data are distributed over multidimensional argument domains. We may have data observed over one or more dimensions of space as well as over time, for example. A photograph or a brain image is a functional observation where the intensity and possibly color composition is a function of spatial location.

3.2.3 The interplay between smooth and noisy variation

Smoothness, in the sense of possessing a certain number of derivatives, is a property of the latent function x, and may not be at all obvious in the raw

data vector $y = (y_1, \ldots, y_n)$ owing to the presence of observational error or noise that is superimposed on the underlying signal by aspects of the measurement process. We express this in notation as

$$y_j = x(t_j) + \epsilon_j, \tag{3.1}$$

where the noise, disturbance, error, perturbation or otherwise exogenous term ϵ_j contributes a roughness to the raw data. One of the tasks in representing the raw data as functions may be to attempt to filter out this noise as efficiently as possible. However, in other cases we may pursue the alternative strategy of leaving the noise in the estimated function; and instead require smoothness of the results of our analysis, rather than of the data that are analyzed.

Vector notation leads to much cleaner and simpler expressions, and so we express the "signal plus noise" model (3.1) as

$$\mathbf{y} = x(\mathbf{t}) + \mathbf{e} \tag{3.2}$$

where $\mathbf{y}, x(\mathbf{t}), \mathbf{t}$ and $\mathbf{e}$ are all column vectors of length n.

The variance-covariance matrix for the vector of observed values $\mathbf{y}$ is equal to the variance-covariance matrix for the corresponding vector $\boldsymbol{\epsilon}$ of residual values since the values $x(t_j)$ are here considered fixed effects with variance 0. Let $\boldsymbol{\Sigma}_e$ be our notation for residual variance-covariance matrix, which expresses how the residuals vary over repeated samples that are identical in every respect except for noise or error variation.

3.2.4 The standard model for error and its limitations

The standard or textbook statistical model for the distribution of the ϵ_j's is to assume that they are independently distributed with mean zero and constant variance σ^2. Consequently, according to the standard model,

$$\mathtt{Var}(\mathbf{y}) = \boldsymbol{\Sigma}_e = \sigma^2 \mathbf{I} \tag{3.3}$$

where the identity matrix $\mathbf{I}$ is of order n.

These assumptions in the standard model, in spite of being routinely made, are almost surely too simple for most functional data. Rather, for example, we must often recognize that the variance of the residuals will itself vary over argument t. We will see in Chapter 5, for example, that the standard error of measurement of the height of children is about eight millimeters in infancy, but declines to around five millimeters by age six.

We may also have to take into account a correlation among neighboring ϵ_j's. The *autocorrelation* that we often see in functional residuals reflects the fact that the functional variation that we choose to ignore is itself probably smooth at a finer scale of resolution.

In fact, the concept of independently distributed error in the standard model, which, as n increases, becomes what is called *white noise,* is not realistic or realizable in nature because white noise would require infinite

energy to achieve. For example, fluctuations in a large stock market are often treated as having white noise properties, but in reality only a limited number of stocks can be traded within a short time interval such as a millisecond, and consequently scale stock prices will exhibit some structure within a time scale that is small enough.

This does not necessarily mean that it will be always essential to model the variable variance or autocorrelation structure in the residuals or errors. Such models for $\mathbf{\Sigma}_e$ can burn up precious degrees of freedom, slow down computation significantly, and finally result in estimates of functions that are virtually indistinguishable from what is achieved by assuming independence in residuals. Nevertheless, a model specifically for variance heterogeneity and/or autocorrelation can pay off in terms of better estimation, and this type of structure may be in itself interesting. A thoughtful application of functional data analysis will always be open to these possibilities.

We should also keep in mind the possibility that errors or disturbances might multiply rather than add when the data are intrinsically positive, in which case it is more sensible to work with the logarithms of the data. We will do this, for example, with the precipitation data for Canadian weather stations in Chapter 14.

3.2.5 The resolving power of data

The *sampling rate* or *resolution* of the raw data is a key determinant of what is possible in the way of functional data analysis. This is essentially a local property of the data, and can be described as the density of the argument values t_j relative to the amount of curvature in the data, rather than simply the number n of argument values. The *curvature* of a function x at argument t is usually measured by the size of the second derivative, as reflected in either $|D^2x(t)|$ or $[D^2x(t)]^2$.

Where curvature is high, it is essential to have enough points to estimate the function effectively. What is enough? This depends on the amount of error ϵ_j; when the error level is small and the curvature is mild, we can get away with a low sampling rate. The gait data in Figure 1.8 exhibit little error and only mild curvature, and thus the sampling rate of 20 values per cycle is enough for our purposes. The human growth data in Figure 1.1 have moderately low error levels, amounting to about 0.3% of adult height, but the curvature in the second derivative functions is fairly severe, so that a sampling rate of measurements every six months for these data is barely sufficient for making inferences about growth acceleration.

3.2.6 Data resolution and derivative estimation

Figure 3.1 provides an interesting example of functional data. The letters "fda" were written on a flat surface by one of the authors. The pen positions

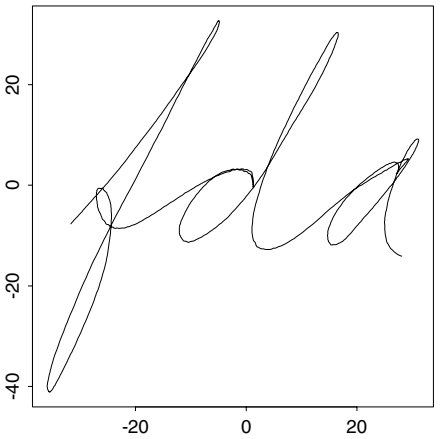

Figure 3.1. A sample of handwriting in which the X-Y coordinates are recorded 600 times per second.

were recorded by an Optotrak system that gives the position of an infrared emitting diode in three-dimensional space 600 times per second with an error level of about 0.5 millimeters. The X and Y position functions `ScriptX` and `ScriptY` are plotted separately in Figure 3.2, and we can see that the error level is too small to be visible. The total event took about 2.3 seconds, and the plotted functions each have 1401 discrete values. This is certainly a lot of resolution, but the curvature is rather high in places, and it turns out that even with the small error level involved, this level of resolution is not excessive.

Because the observed function looks reasonably smooth, the sampling rate is high, and the error level is low, one might be tempted to use the first *forward difference* $(y_{j+1} - y_j)/(t_{j+1} - t_j)$, or the *central difference* $(y_{j+1} - y_{j-1})/[(t_{j+1} - t_{j-1})]$, to estimate $Dx(t_j)$, but Figure 3.3 shows that the resulting derivative estimate for `ScriptX` is rather noisy. The second central difference estimate of D^2`ScriptX`

$$D^2x(t_j) \approx (y_{j+1} + y_{j-1} - 2y_j)/(\Delta t)^2$$

is shown in Figure 3.3 to be a disaster. The reason for this failure is precisely the high sampling rate for the data; taking differences between extremely close values magnifies the influence of error enormously. Press et al. (1999) comment on how simple differencing to estimate derivatives can go wrong even when functions are available analytically.

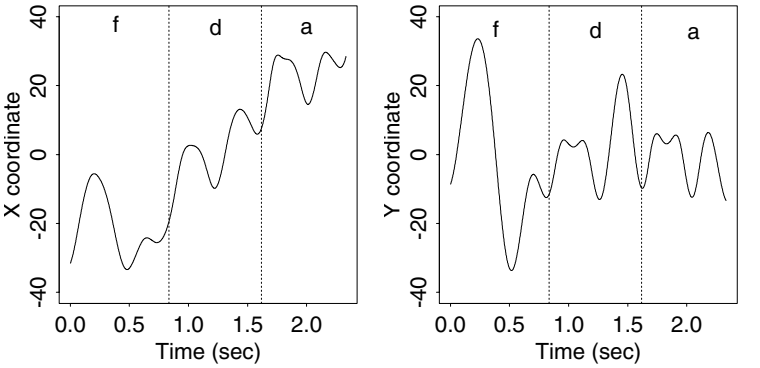

Figure 3.2. The X and Y coordinates for the handwriting sample plotted separately. Note the strongly periodic component with roughly two cycles per second.

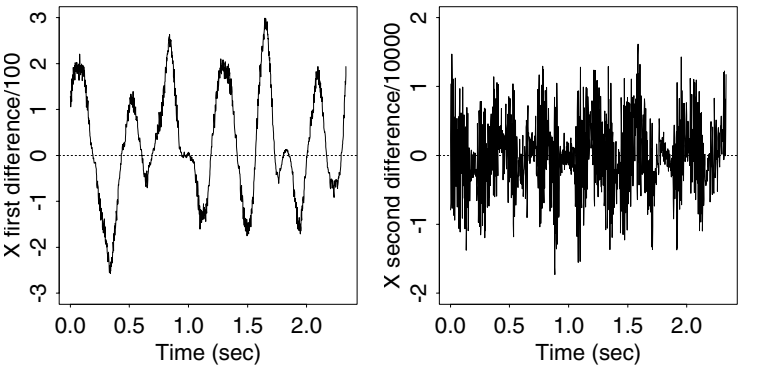

Figure 3.3. The first and second central differences for the X coordinate for the handwriting sample. The high sampling rate causes differencing to greatly magnify the influence of noise.

We will give a lot of attention to derivative estimation in this and the next chapter, including methods for estimating confidence intervals for derivative estimates. Many challenges remain, however, and there is plenty of room for improvement in existing techniques.

3.3 Representing functions by basis functions

A basis function system is a set of known functions ϕ_k that are mathematically independent of each other and that have the property that we can approximate arbitrarily well any function by taking a weighted sum or *linear combination* of a sufficiently large number K of these functions. The

most familiar basis function system is the collection of *monomials* that are used to construct power series,

$$1, t, t^2, t^3, \ldots, t^k, \ldots \ .$$

Right behind the power series in our list of golden oldies is the *Fourier series* system,

$$1,\ \sin(\omega t), \cos(\omega t),\ \sin(2\omega t), \cos(2\omega t),\ \sin(3\omega t), \cos(3\omega t), \ldots,$$

$$\sin(k\omega t), \cos(k\omega t), \ldots$$

Basis function procedures represent a function x by a linear expansion

$$x(t) = \sum_{k=1}^{K} c_k \phi_k(t) \tag{3.4}$$

in terms of K known basis functions ϕ_k.

By letting $\mathbf{c}$ indicate the vector of length K of the coefficients c_k and $\boldsymbol{\phi}$ as the functional vector whose elements are the basis functions ϕ_k, we can also express (3.4) in matrix notation as

$$x = \mathbf{c}'\boldsymbol{\phi} = \boldsymbol{\phi}'\mathbf{c} \ . \tag{3.5}$$

In effect, basis expansion methods represent the potentially infinite-dimensional world of functions within the finite-dimensional framework of vectors like c. The *dimension* of the expansion is therefore K. It would be a mistake, though, to conclude that functional data analysis in this case simply reduces to multivariate data analysis; a great deal also depends on how the basis system, $\boldsymbol{\phi}$, is chosen.

An exact representation or *interpolation* is achieved when $K = n$, in the sense that we can choose the coefficients c_k to yield $x(t_j) = y_j$ for each j. Therefore the degree to which the data y_j are *smoothed* as opposed to interpolated is determined by the number K of basis functions. Consequently, we do not view a basis system as defined by a fixed number K of parameters, but rather we see K as itself a parameter that we choose according to the characteristics of the data.

Ideally, basis functions should have features that match those known to belong to the functions being estimated. This makes it easier to achieve a satisfactory approximation using a comparatively small number K of basis functions. The smaller K is and the better the basis functions reflect certain characteristics of the data,

- the more degrees of freedom we have to test hypotheses and compute accurate confidence intervals,
- the less computation is required, and
- the more likely it is that the coefficients themselves can become interesting descriptors of the data from a substantive perspective.

Consequently, certain classic off-the-rack bases such as polynomials and Fourier series may be ill-advised in some applications; there is no such thing as a universally good basis.

The choice of basis is particularly important for a derivative estimate

$$D\hat{x}(t) = \sum_{k}^{K} \hat{c}_k D\phi_k(t) = \hat{\mathbf{c}}' D\boldsymbol{\phi}(t). \tag{3.6}$$

Bases that work well for function estimation may give rather poor derivative estimates. This is because an accurate representation of the observations may force $\hat{x}$ to have small but high-frequency oscillations that have dreadful consequences for its derivatives. Put more positively, one of the criteria for choosing a basis may be whether or not one or more of the derivatives of the approximation behave reasonably.

Chapter 21 touches on tailoring a basis to fit a particular problem. For now, we discuss some popular bases that are widely used in practice and when to use them. To summarize what follows, most functional data analyses these days involve either a Fourier basis for periodic data, or a B-spline basis for non-periodic data. Where derivatives are not required, wavelet bases are seeing more and more applications. Poor old polynomials are now the senior citizens of the basis world, relegated to only the simplest of functional problems.

3.4 The Fourier basis system for periodic data

Perhaps the best known basis expansion is provided by the Fourier series:

$$\hat{x}(t) = c_0 + c_1 \sin \omega t + c_2 \cos \omega t + c_3 \sin 2\omega t + c_4 \cos 2\omega t + \ldots \tag{3.7}$$

defined by the basis $\phi_0(t) = 1, \phi_{2r-1}(t) = \sin r\omega t$, and $\phi_{2r}(t) = \cos r\omega t$. This basis is periodic, and the parameter ω determines the period $2\pi/\omega$. If the values of t_j are equally spaced on $\mathcal{T}$ and the period is equal to the length of interval $\mathcal{T}$, then the basis is *orthogonal* in the sense that the cross product matrix $\boldsymbol{\Phi}'\boldsymbol{\Phi}$ is diagonal, and can be made equal to the identity by dividing the basis functions by suitable constants, $\sqrt{n}$ for $j = 0$ and $\sqrt{n/2}$ for all other j.

The Fast Fourier transform (FFT) makes it possible to find all the coefficients extremely efficiently when n is a power of 2 and the arguments are equally spaced, and in this case we can find both the coefficients c_k and all n smooth values at $x(t_j)$ in $O(n \log n)$ operations. This is one of the features that has made Fourier series the traditional basis of choice for long time series, but newer techniques such as B-splines and wavelets can match and even exceed this computational efficiency.

Derivative estimation in a Fourier basis is simple since

$$\begin{aligned} D\sin r\omega t &= r\omega\cos r\omega t \\ D\cos r\omega t &= -r\omega\sin r\omega t \end{aligned} \tag{3.8}$$

This implies that the Fourier expansion of Dx has coefficients

$$(0, c_1, -\omega c_2, 2\omega c_3, -2\omega c_4, \ldots)$$

and of D^2x has coefficients

$$(0, -\omega^2 c_1, -\omega^2 c_2, -4\omega^2 c_3, -4\omega^2 c_4, \ldots).$$

Similarly, we can find the Fourier expansions of higher derivatives by multiplying individual coefficients by suitable powers of $r\omega$, with sign changes and interchange of sine and cosine coefficients as appropriate.

The Fourier series is so familiar to statisticians, engineers and applied mathematicians that it is worth stressing its limitations. Invaluable though it may often be, neither it nor any other basis should be used uncritically. A Fourier series is especially useful for extremely stable functions, meaning functions where there are no strong local features and where the curvature tends to be of the same order everywhere. Ideally, the periodicity of the Fourier series should be reflected to some degree in the data, as is certainly the case for the temperature and gait data. Fourier series generally yield expansions which are uniformly smooth. But they are inappropriate to some degree for data known or suspected to reflect discontinuities in the function itself or in low order derivatives. A Fourier series is like margarine: It's cheap and you can spread it on practically anything, but don't expect that the result will be exciting eating. Nevertheless, we find many applications for Fourier series expansion in this book.

3.5 The spline basis system for open-ended data

Spline functions are the most common choice of approximation system for non-periodic functional data or parameters. They have more or less replaced polynomials, which in any case they contain within the system. Splines combine the fast computation of polynomials with substantially greater flexibility, often achieved with only a modest number of basis functions. Moreover, basis systems have been developed for spline functions that require an amount of computation that is proportional to n, a vital property since many applications involve thousands or millions of observations. In this section we first examine the structure of a spline function, and then describe the usual basis system used to construct it, the B-spline system.

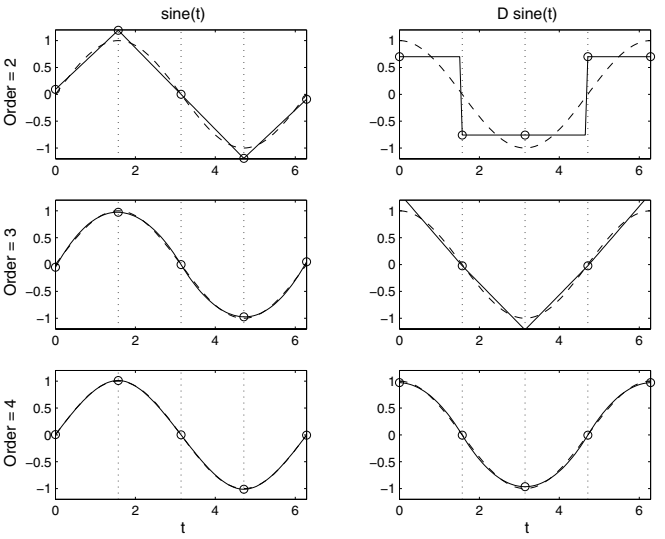

Figure 3.4. In the left panels the solid line indicates spline function of a particular order that fits the sine function shown as a dashed line. In the right panels the corresponding fits to its derivative, a cosine function, are shown. The vertical dotted lines are the interior breakpoints or knots defining the spline fits.

3.5.1 Spline functions and degrees of freedom

The anatomy of a spline is illustrated in Figure 3.4, where three spline functions are fit to $\sin(t)$ over the interval $[0, 2\pi]$ in the left panels, and where we also see the fit to its derivative, $\cos(t)$, in the right panels.

The first step in defining a spline is to divide the interval over which a function is to be approximated into L subintervals separated by values $\tau_\ell, \ell = 1, \ldots, L-1$ that are called *breakpoints* or *knots.* The former term is, strictly speaking, more correct for reasons that will be indicated shortly. We see in the figure that three breakpoints divide the interval into four subintervals. If we include the endpoints 0 and 2π as breakpoints, we may number them $\tau_0, \ldots, \tau_L$, where $L = 4$.

Over each interval, a spline is a polynomial of specified order m. The *order* of a polynomial is the number of constants required to define it, and is one more than its *degree*, its highest power. Thus, the spline in the top left of Figure 3.4 is piecewise linear, the center left spline is piecewise quadratic, and the bottom left piecewise cubic, corresponding to orders 2, 3 and 4, respectively. An order one spline can be seen in the top right panel, and this is a step function of degree zero.

Adjacent polynomials join up smoothly at the breakpoint which separates them for splines of order greater than one, so that the function values are constrained to be equal at their junction. Moreover, derivatives up to

order $m-2$ must also match up at these junctions. For example, for the commonly used order four cubic spline, the second derivative is a polygonal line and the third derivative is a step function. See a few paragraphs further on in this section, however, for an account of the possibility of reducing these smoothness constraints by using multiple knots at junction points.

We see in the top left panel of Figure 3.4, where an order two spline is fit to the sine curve, that only the function values join. Thus that there is one constraint on adjacent lines. Since there are two degrees of freedom in a line, and we have four lines, the total number of degrees of freedom in this line is calculated as follows. We count a total of 2×4 coefficients to define the four line segments, but we subtract one degree of freedom for each of the continuity constraints at each of the three junctions. This makes five in all.

Similarly, in the center left panel, the piecewise polynomials are quadratic, giving $3 \times 4 = 12$ coefficients, but this time both the function value and the first derivative join smoothly, so that we subtract six to get six remaining degrees of freedom. Finally, in the third row, where the polynomials are cubic, and where the function values, first derivatives and second derivatives must join, the accounting gives $4 \times 4 = 16$ less $3 \times 3 = 9$ constraints, leaving us with seven degrees of freedom. The rule is simple:

> The total number of degrees of freedom in the fit equals the order of the polynomials plus the the number of *interior* breakpoints.

If there are no interior knots, the spline reverts to being a simple polynomial.

We see that with increasing order comes a better and better approximation to both the sine and its derivative, and that by order four the fit is very good indeed. In fact, if we were to increase the order to five or beyond, we would also get a fine fit to the second derivative as well.

The main way to gain flexibility in a spline is to increase the number of breakpoints. Here we have made them equally spaced, but in general, we want more breakpoints over regions where the function exhibits the most complex variation, and fewer where the function is only mildly nonlinear. A subsidiary consideration is that we certainly do not want intervals that do not contain data, but then this seems reasonable since we cannot expect to capture a function's features without data.

We mentioned above that breakpoints are not quite the same thing as knots. This is because we can have two or more breakpoints that move together to coalesce or be coincident. When this happens, there is a loss of continuity condition for each additional coincident breakpoint. In this way, we can engineer abrupt changes in a derivative or even a function value at pre-specified breakpoints. The interested reader should consult de Boor (2001) for further details.

Thus, the term *breakpoint*, strictly speaking, refers to the number of unique knot values, while the term *knot* refers to the sequence of values at breakpoints, where some breakpoints can be associated with multiple knots. The knots are all distinct in most applications, and consequently breakpoints and knots are then the same thing. But we will encounter data input/output systems where the inputs are varied in a discrete step-wise way, and these will require coincident knots to model these sharp changes in level.

To review, a spline function is determined by two things: The order of the polynomial segments, and the knot sequence τ. The number of parameters required to define a spline function in the usual situation of one knot per breakpoint is the order plus the number of interior knots, $m + L - 1$.

3.5.2 The B-spline basis for spline functions

We have now defined a spline function, but have given no clue as to how to actually construct one. For this, we specify a system of basis functions $\phi_k(t)$, and these will have the following essential properties:

- Each basis function $\phi_k(t)$ is itself a spline function as defined by an order m and a knot sequence τ.
- Since a multiple of a spline function is still a spline function, and since sums and differences of splines are also splines, any linear combination of these basis functions is a spline function.
- Any spline function defined by m and τ can be expressed as a linear combination of these basis functions.

Although there are many ways that such systems can be constructed, the *B-spline* basis system developed by de Boor (2001) is the most popular, and code for working with B-splines is available in a wide range of programming languages, including R, S-PLUS and MATLAB®. Other spline basis systems are truncated power functions, M-splines and natural splines, and these and others are discussed by de Boor (2001) and Schumaker (1981).

Figure 3.5 shows the thirteen B-spline basis functions for an order four spline defined by the nine equally spaced interior breakpoints, which are also shown in this figure. Notice that each of the seven basis functions in the center only is positive over four adjacent sub-intervals. Because cubic splines have two continuous derivatives, each basis function makes a smooth transition to the regions over which it is zero. These central basis splines have the same shape because of the equal spacing of breakpoints; unequally spaced breakpoints would define splines varying in shape. The left-most three basis functions and their three right counterparts do differ in shape, but nevertheless are also positive over no more than four adjacent sub-intervals.

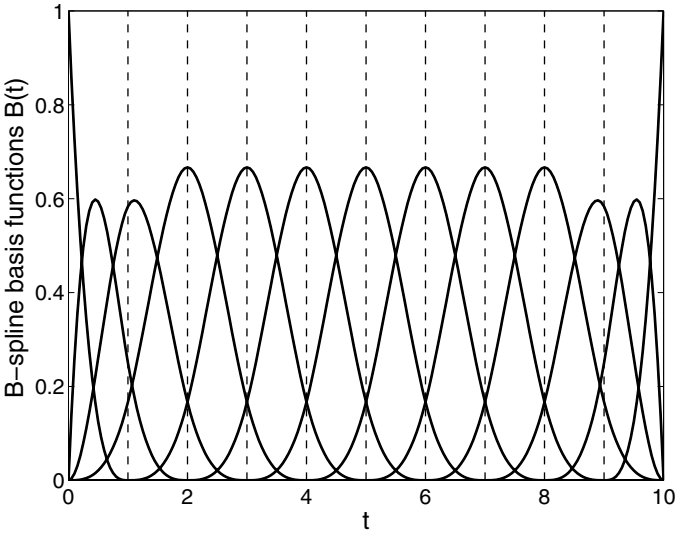

Figure 3.5. The thirteen basis functions defining an order four spline with nine interior knots, shown as vertical dashed lines.

The property that an order m B-spline basis function is positive over no more than m intervals, and that these are adjacent, is called the *compact support* property, and is of the greatest importance for efficient computation. If there are K B-spline basis functions, then the order K matrix of inner products of these functions will be band-structured, with only $m-1$ sub-diagonals above and below the main diagonal containing nonzero values. This means that no matter how large K is, and we will be dealing with values in the thousands, the computation of spline function can be organized so as to increase only linearly with K. Thus splines share the computational advantages of potentially *orthogonal* basis systems such as as Fourier and wavelet bases.

The three basis functions on the left and the three on the right are different. As we move from the left boundary towards the center, the intervals over which the basis functions are positive increase from one to four, but always make the same smooth twice-differentiable transition to the zero region. On the other hand, their transition to the left boundary varies in smoothness, with the left-most spline being discontinuous, the next being continuous only, and the third being once-differentiable. The same thing happens on the right side, but in reverse order. That we lose differentiability at the boundaries makes good sense, since we normally have no information about what the function we are estimating is doing beyond the interval on which we collect data. We therefore are allowing for the possibility that the function may be discontinuous beyond the boundaries.

This boundary behavior of B-spline basis functions is achieved by placing, in effect, m knots at the boundaries. That is, when B-splines are actually computed, the knot sequence τ is extended at each end to add an additional $m-1$ replicates of the boundary knot value. As we noted before, there are some applications where we do not want $m-2$ continuous derivatives at certain fixed points in the interior of the interval. This can be readily accommodated by B-splines. We place a knot at such fixed points, and then for each reduction in differentiability an additional knot is placed at that location as well. For example, if were were working with order four splines, and wanted the derivative to be able to change abruptly at a certain value of t but still wanted the fitted function to be continuous, we would place three knots at that value.

The notation $B_k(t,\tau)$ is often used to indicate the value at t of the B-spline basis function defined by the breakpoint sequence τ. Here k refers to the number of the largest knot at or to the immediate left of value t. The $m-1$ knots added to the initial breakpoint are also counted in this scheme, and this is consistent with the fact that the first m B-spline basis functions all have supports all beginning at the left boundary. This notation gives us $m+L-1$ basis functions, as required in the usual case where all interior knots are discrete. According to this notation, a spline function $S(t)$ with discrete interior knots is defined as

$$S(t) = \sum_{k=1}^{m+L-1} c_k B_k(t,\tau) \ . \tag{3.9}$$

It remains to give some guidance as to where the interior breakpoints or knots τ_ℓ should be positioned. Many applications default to equal spacing, which is fine as long as the data are relatively equally spaced. If they are not, it may be wiser to place a knot at every jth data point, were j is a number fixed in advance. This amounts to placing interior knots at the *quantiles* of the argument distribution. A special case is that of *smoothing splines* that we will take up in the next chapter, where a breakpoint is placed at each argument value. Finally, one can depart from either of these simple procedures to place more knots in regions known to contain high curvature, and fewer where there is less.

Figure 3.6 shows an example of using coincident knots to measurements of the level of a fluid in a tray in a oil refinery distillation column, previously shown in Figure 1.4. At time 67 a valve was turned and the flow of fluid into the tray changed abruptly, whereupon the fluid level increases rapidly at first, and then more and more slowly as it approaches its final value. It is clear that the first derivative should be discontinuous at time 67, but that the fluid level is essentially smooth elsewhere. These data were fit with B-splines of order four, with a single knot mid-way between times 0 and 67, three equally spaced knots between times 67 and 193, and three coincident knots at time 67. Now an order four spline has a third derivative that is

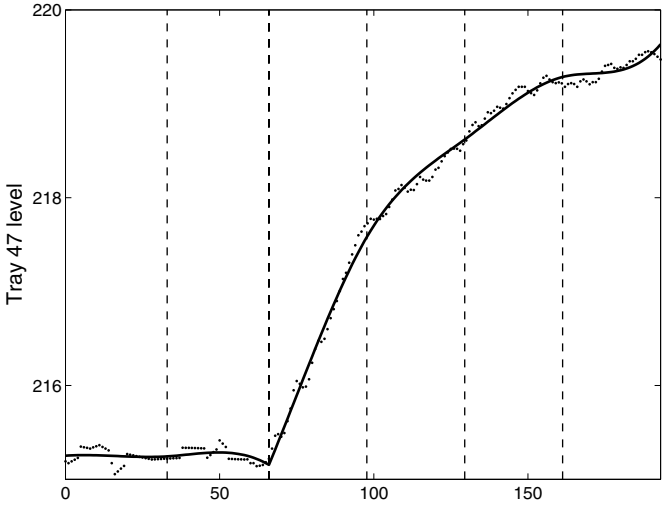

Figure 3.6. The oil refinery tray 47 level shown in Figure 1.4. The heavy smooth line is a fit to the data using B-spline basis functions with knots located as shown by the vertical dashed lines. There are three coincident knots at time 67 in order to achieve the discontinuity in the first derivative of the fit.

discontinuous at single knot locations, and, recalling that each additional coincident knot decreases the order of continuity by one, we achieve first derivative discontinuity at time 67. This can be seen in the smooth line fit to the data by the methods described in the next chapter. Go to Chapter 17 for further analyses of these data.

One possibly disconcerting feature of spline bases is that increasing K does not always improve certain aspects of the fit to the data. This is because, when the order of a spline is fixed, the function space defined by K B-splines is not necessarily contained within that defined by $K + 1$ B-splines. Complicated effects due to knot spacing relative to sampling points can result in a lower-dimensional B-spline system actually producing better results than a higher-dimensional system. However, if K is increased by either adding a new breakpoint to the current τ, or by increasing the order and leaving τ unchanged, then the K-space is contained within the $(K + 1)$-space.

There are data-driven methods for breakpoint positioning. Some approaches begin with a dense set of breakpoints, and then eliminate unneeded ones by an algorithmic procedure similar to variable selection techniques used in multiple regression. See, for example, Friedman and Silverman (1989). Alternatively, one can optimize the fitting criterion with respect to knot placement at the same time that one estimates the coefficients of the expansion. However, this can lead to computational problems,

since fitting criteria can vary in highly complex ways as a function of knot placement. Some useful techniques for improving knot placement are discussed by de Boor (2001).

It is not easy to find a readable introduction to splines, but the functional data analysis website, `www.functionaldata.org`, offers a beginner's treatment. The most comprehensive reference is de Boor (2001), which contains a wealth of information on computational as well as theoretical issues. But it is for advanced readers only, whereas Eubank (1999) and Green and Silverman (1994) are at a more intermediate level.

3.6 Other useful basis systems

We must not, however, forget about a number of other potentially important basis systems. In fact, two contrasting developments in recent years are having a large impact on data analysis. On the side of great mathematical sophistication we have wavelets that combine the frequency-specific approximating power of the Fourier series with the time- or spatially-localized features of splines. On the other hand, we have seen a fascinating resurgence of interest in exceedingly simple bases such as step functions (order one splines in effect) and polygons (order two splines) (Hastie, et al. 2001).

3.6.1 Wavelets

We can construct a basis for all functions on $(-\infty, \infty)$ that are square-integrable by choosing a suitable *mother wavelet* function ψ and then considering all dilations and translations of the form

$$\psi_{jk}(t) = 2^{j/2}\psi(2^j t - k)$$

for integers j and k. We construct the mother wavelet to ensure that the basis is orthogonal, in the sense that the integral of the product of any two distinct basis functions is zero. Typically, the mother wavelet has compact support, and hence so do all the basis functions. The wavelet basis idea is easily adapted to deal with functions defined on a bounded interval, most simply if periodic boundary conditions are imposed.

The wavelet expansion of a function f gives a *multiresolution analysis* in the sense that the coefficient of ψ_{jk} yields information about f near position $2^{-j}k$ on scale 2^{-j}, i.e., at frequencies near $c2^j$ for some constant c. Thus wavelets provide a systematic sequence of degrees of locality. In contrast to Fourier series, wavelet expansions cope well with discontinuities or rapid changes in behavior; only those basis functions whose support includes the region of discontinuity or other bad behavior are affected. This property, as well as a number of more technical mathematical results, means that it is often reasonable to assume that an observed function is well approximated

by an economical wavelet expansion with few non-zero coefficients, even if it displays sharp local features.

Suppose a function x is observed without error at $n = 2^M$ regularly spaced points on an interval $\mathcal{T}$. just as with the Fourier transformation, there is a discrete wavelet transform (DWT) which provides n coefficients closely related to the wavelet coefficients of the function x. We can calculate the DWT and its inverse in $O(n)$ operations, even faster than the $O(n \log n)$ of the FFT. As a consequence, most estimators based on wavelets can be computed extremely quickly, many of them in $O(n)$ operations.

Now suppose that the observations of x are subject to noise. The fact that many intuitively attractive classes of functions have economical wavelet expansions leads to a simple *nonlinear* smoothing approach: Construct the DWT of the noisy observations, and threshold it by throwing away the small coefficients in the expansion and possibly shrinking the large ones. The basic motivation of thresholding is the notion that any coefficient that is small is entirely noise and does not reflect any signal at all. This nonlinear thresholding has attractive and promising theoretical properties (see, for example, Donoho, Johnstone, Kerkyacharian and Picard, 1995), indicating that thresholded wavelet estimators should adapt well to different degrees of smoothness and regularity in the function being estimated.

3.6.2 Exponential and power bases

Exponential basis systems consist of a series of exponential functions,

$$e^{\lambda_1 t}, e^{\lambda_2 t}, \ldots, e^{\lambda_k t}, \ldots$$

where the *rate parameters* λ_k are all distinct, and often $\lambda_1 = 0$. Linear differential equations with constant coefficients have as solutions expansions in terms of exponential bases.

Power bases,

$$t^{\lambda_1}, t^{\lambda_2}, \ldots, t^{\lambda_k}, \ldots$$

likewise are important from time to time, often when t is strictly positive so that negative powers are possible.

3.6.3 Polynomial bases

The monomial basis $\phi_k(t) = (t - \omega)^k, k = 0, \ldots, K$ is also classic, where ω is a shift parameter that is usually chosen to be in the center of the interval of approximation. Care must be taken to avoid rounding error in the computations, since monomial values are more and more highly correlated as the degree increases. However, if the argument values t_j are equally spaced or can be chosen to exhibit a few standard patterns, orthogonal polynomial expansions can be obtained, implying $O((n+m)K)$ operations for all

smooth values. Otherwise we are condemned to contemplate $O((n+m)K^2)$ operations.

Like the Fourier series expansion, polynomials cannot exhibit very local features without using a large K. Moreover, polynomials tend to fit well in the center of the data but exhibit rather unattractive behavior in the tails. They are usually a poor basis for extrapolation or forecasting, for example.

Although derivatives of polynomial expansions are simple to compute, they are seldom satisfactory as estimators of the true derivative because of the rapid localized oscillation typical of high order polynomial fits.

3.6.4 The polygonal basis

Smoothing the observed rough data is not always necessary, and especially if our interest is not in the fit to the data itself, but rather in some functional parameter not directly connected to the data. In the chapters on the functional linear model, we will see that we can interpolate the data with a simple basis, and move the smoothing issue to where it belongs, namely the estimation of the functional parameter. In fact, polygonal or piecewise linear data fits have much to recommend them, and can even offer a crude estimate of the first derivative.

3.6.5 The step-function basis

Data mining problems often involve huge numbers of variables combined with phenomenal sample sizes. Because computational constraints can become critical, and we look for the simplest methods that will work. One of the great success stories in modern statistics has the usefulness of simple splits of variables into two categories, and the construction of tree-based representations of relationships or classification schemes. A split of variable values can be viewed as a functional transformation with a basis consisting of two step functions. This is, in effect, an order one B-spline system with a single interior knot. A recent reference on data mining in general that has considerable material on tree-based classification is Hastie, Tibshirani and Friedman (2001). It is somehow refreshing that we return to basics from time to time and rediscover that "effective" is not always the same thing as "sophisticated".

3.6.6 The constant basis

We shouldn't neglect the most humble of bases, the single basis function $1(t)$ whose value is one everywhere. It comes in handy surprisingly often. Firstly, it provides a useful point of reference or null hypothesis when we estimate regression coefficient functions for the functional linear model and elsewhere. Secondly, it is explicitly in systems like the Fourier, and

implicitly into B-spline bases. Finally, we can view a scalar observation as a functional datum whose value is the same everywhere, and consequently its value becomes the coefficient for the constant basis. Using this device we can, in effect, include most multivariate statistical techniques into functional data analysis in a seamless manner.

3.6.7 Empirical and designer bases

If choosing a basis that matches the characteristics of the data is important, can't we design our own basis systems? The answer is positive in two ways. First, we will discuss basis systems associated with differential equation models for functional data in Chapter 21, and the chapter preceding this will show how such models can be fit empirically to the data.

Designer bases can also be constructed empirically using *functional principal components analysis*, the subject of Chapters 8 to 10. Such bases have the property of optimizing the amount of variance in the data explained by basis systems of size K. If one wants the most compact basis possible with the sole objective of fitting the data, principal components analysis is usually the method that is first considered.

3.7 Choosing a scale for t

From the perspective of mathematics, the choice of unit of measurement for argument t may appear to be of no great consequence. But the implications for computation can be dramatic, and especially when we work with derivatives.

The two main bases that we intend to work with, the B-spline and Fourier systems, have *normalized* basis function, meaning that basis function values are bounded. In the case of B-splines, the bounds are zero and one, and at any point t the sum of B-spline basis functions that are nonzero at that point is exactly one. The only B-splines that attain the upper limit of one are those at the extreme ends of the interval. In the Fourier series case, function values are found within [-1,1].

As a result, if large number of basis functions are packed into a small interval, their derivatives are bound to be large. This is particularly easy to see in the Fourier series case, where the mth derivative of $\sin k\omega t$ will attain limits $\pm(k\omega)^m$. For example, if we opt to define the unit of time for the daily weather data to be one year, we decide to work with a saturated basis containing 365 basis functions, then we will be looking at values of the fourth derivative oscillating between $\pm 1.7 \times 10^{12}$, as opposed to about ± 1560 if we use the day as the unit of time.

The same applies to B-splines. For example, the handwriting data has 1401 sampling points equally spaced between zero and 2.3 seconds. On

this time scale, if we use 1405 basis functions of order six with knots at the time points, which is not an unreasonable proposal, we will see fourth derivative values of about 2^{13}. On the other hand, if we use a time scale of milliseconds, then we see the same derivatives reaching values of only about 20.

Why does this matter? We will at many points in our investigation want to combine derivatives of various orders. For example, in Chapter 5, we will use the fourth derivative to stabilize or smooth estimates of the second derivative, and will do this by combining within the same fitting criterion B-spline basis functions values with their fourth derivatives. When you try to add together quantities of the order of one with quantities of the order of 10^{12}, it is easy to run into prodigious rounding error problems if you are not careful. All this trouble can be avoided by using using a unit for t which is roughly equal to the period of oscillation of the most rapidly varying basis function that you will use. In any case, we tend to find that our clients do not take well to seeing plots or tables of quantities far beyond magnitudes that they can imagine.

3.8 Further reading and notes

We imagine that the Fourier series needs little introduction for most of our readers. Most introductory calculus texts cover the topic, and many branches of statistics apply it.

Spline functions are another thing entirely. We have not found many treatments that are for beginners, and have often been brought up short when asked for something to read. This is why we have supplied the rather lengthy account that this chapter contains, at the risk of boring spline experts. The introduction to splines in Hastie and Tibshirani (1990) has proven helpful, and Green and Silverman (1994) is useful those with more intermediate exposures to mathematics and statistics. Even after a revision, de Boor (2001) remains a challenging book, but is unequalled in its coverage of splines. Texts devoted to smoothing and nonparametric regression such as Eubank (1999) and Simonoff (1996) are also useful references. Schumaker (1981) is an important but more advanced treatment of splines. Wahba (1990) is often cited, but if you can understand that book, you shouldn't be reading these early chapters!

Wavelet bases are comparatively recent, and they have considerable promise in many functional data analysis contexts. For further reading, see Chui (1992), Daubechies (1992), Press et al. (1992), Nason and Silverman (1994), Donoho et al. (1995) and Johnstone and Silverman (1997), as well as the many references contained in these books and papers. An entire issue in 1999 of the *Philosophical Transactions of the Royal Society of London, Series A,* was devoted to wavelet applications and theory, and

the papers there by Silverman (1999) and Silverman and Vassilicos (1999), as well as Silverman (2000) are to be recommended to newcomers to this exciting field.

Polynomial and power bases appear often under other titles. Power series, treated in all calculus texts, and the Taylor and Maclaurin expansions found there are specialized methods for estimating polynomial expansions. Later in Chapter 21 we will consider ways of generalizing these important tools.

4

Smoothing functional data by least squares

4.1 Introduction

In this chapter and the next we turn to a discussion of specific smoothing methods. Our goal is to give enough information to those new to the topic of smoothing to launch a functional data analysis. Here we focus on the more familiar technique of fitting models to data by minimizing the sum of squared errors, or *least squares estimation*. This approach ties in functional data analysis with the machinery of multiple regression analysis. A number of tools taken from this classical field are reviewed here, and especially those that arise because least squares fitting defines a model whose estimate is a linear transformation of the data.

The treatment is far from comprehensive, however, and primarily because we will tend to favor the more powerful methods using roughness penalties to be taken up in the next chapter. Rather, notions such as degrees of freedom, sampling variance, and confidence intervals are introduced here as a first exposure to topics that will be developed in greater detail in Chapter 5.

4.2 Fitting data using a basis system by least squares

Recall that our goal is to fit the discrete observations $y_j, j = 1, \ldots, n$ using the model $y_j = x(t_j) + \epsilon_j$, and that we are using a basis function expansion

for $x(t)$ of the form

$$x(t) = \sum_{k}^{K} c_k \phi_k(t) = \mathbf{c}'\boldsymbol{\phi}.$$

The vector $\mathbf{c}$ of length K contains the coefficients c_k. Let us define the n by K matrix $\boldsymbol{\Phi}$ as containing the values $\phi_k(t_j)$.

4.2.1 Ordinary or unweighted least squares fits

A simple linear smoother is obtained if we determine the coefficients of the expansion c_k by minimizing the least squares criterion

$$\texttt{SMSSE}(\mathbf{y}|\mathbf{c}) = \sum_{j=1}^{n} [y_j - \sum_{k}^{K} c_k \phi_k(t_j)]^2. \tag{4.1}$$

The criterion is expressed more cleanly in matrix terms as

$$\texttt{SMSSE}(\mathbf{y}|\mathbf{c}) = (\mathbf{y} - \boldsymbol{\Phi}\mathbf{c})'(\mathbf{y} - \boldsymbol{\Phi}\mathbf{c}) \ . \tag{4.2}$$

The right side is also often written in functional notation as $\|\mathbf{y} - \boldsymbol{\Phi}\mathbf{c}\|^2$.

Taking the derivative of criterion $\texttt{SMSSE}(\mathbf{y}|\mathbf{c})$ with respect to $\mathbf{c}$ yields the equation

$$2\boldsymbol{\Phi}\boldsymbol{\Phi}'\mathbf{c} - 2\boldsymbol{\Phi}'\mathbf{y} = 0$$

and solving this for $\mathbf{c}$ provides the estimate $\hat{\mathbf{c}}$ that minimizes the least squares solution,

$$\hat{\mathbf{c}} = (\boldsymbol{\Phi}'\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{y} \ . \tag{4.3}$$

The vector $\hat{\mathbf{y}}$ of fitted values is

$$\hat{\mathbf{y}} = \boldsymbol{\Phi}\hat{\hat{\mathbf{c}}} = \boldsymbol{\Phi}(\boldsymbol{\Phi}'\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{y} \ . \tag{4.4}$$

Simple least squares approximation is appropriate in situations where we assume that the residuals ϵ_j about the true curve are independently and identically distributed with mean zero and constant variance σ^2. That is, we prefer this approach when we assume the *standard model for error* discussed in Section 3.2.4.

As an example, Figure 4.1 shows the daily temperatures in Montreal averaged over 34 years, 1960–1994, for 101 days in the summer and 101 days in the winter. There is some higher frequency variation that seems to require fitting in addition to the smooth quasi-sinusoidal long-term trend. For example, there is a notable warming period from about January 16 to January 31 that is present in the majority of Canadian weather stations. The smooth fit shown in the figure was obtained with 109 Fourier basis functions, which would permit $108/2 = 54$ cycles per year, or roughly one per week. The curve seems to track nicely these shorter-term variations in temperature.

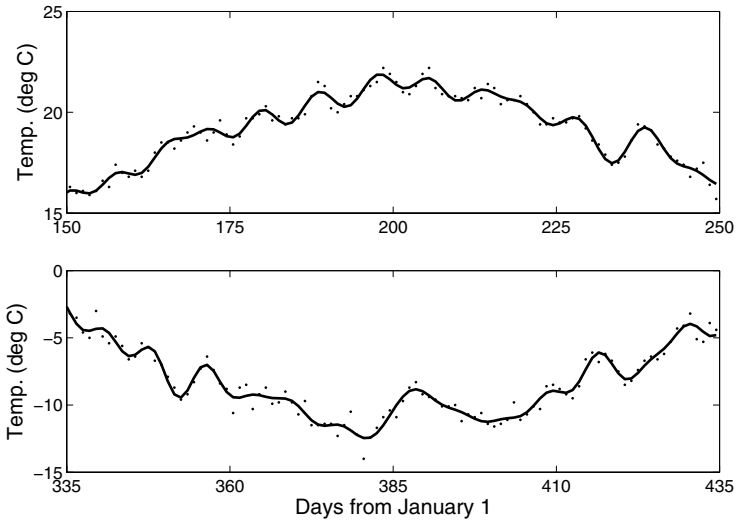

Figure 4.1. The upper panel shows the average daily temperatures for 101 days over the summer in Montreal, and the lower panel covers 101 winter days, with the day values extended into the following year. The solid curves are unweighted least squares smooths of the data using 109 Fourier basis functions.

4.2.2 Weighted least squares fits

As we noted in Section 3.2.4, the standard model for error will often not be realistic. To deal with nonstationary and/or autocorrelated errors, we may need to bring in a differential weighting of residuals by extending the least squares criterion to the form

$$\texttt{SMSSE}(\mathbf{y}|\mathbf{c}) = (\mathbf{y} - \mathbf{\Phi c})'\mathbf{W}(\mathbf{y} - \mathbf{\Phi c}) \tag{4.5}$$

where $\mathbf{W}$ is a symmetric positive definite matrix that allows for unequal weighting of squares and products of residuals.

Where do we get $\mathbf{W}$? If the variance-covariance matrix $\mathbf{\Sigma}_e$ for the residuals ϵ_j is known, then

$$\mathbf{W} = \mathbf{\Sigma}_e^{-1} \ .$$

In applications where an estimate of the complete $\mathbf{\Sigma}_e$ is not feasible, the covariances among errors are often assumed to be zero, and then $\mathbf{W}$ is diagonal with, preferably, reciprocals of the error variance associated with the y_j's in the diagonal. We will consider various ways of estimating $\mathbf{\Sigma}_e$ in Section 4.6.2. But in the meantime, we will not lose anything if we always include the weight matrix $\mathbf{W}$ in results derived from least squares estimation; we can always set it to $\mathbf{I}$ if the standard model is assumed.

The weighted least squares estimate $\hat{\mathbf{c}}$ of the coefficient vector $\mathbf{c}$ is

$$\hat{\mathbf{c}} = (\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}'\mathbf{W}\mathbf{y} \ . \tag{4.6}$$

Whether the approximation is by simple least squares or by weighted least squares, we can express what is to be minimized in the more universal functional notation $\texttt{SMSSE}(\mathbf{y}|\mathbf{c}) = \|\mathbf{y} - \mathbf{\Phi}\mathbf{c}\|^2$.

4.3 A performance assessment of least squares smoothing

It may be helpful to see what happens when we apply least squares smoothing to a situation where we know what the right answer is, and can therefore check the quality of various aspects of the fit to the data, as well as the accuracy of data-driven bandwidth selection methods.

We turn now to the growth data, where a central issue was obtaining a good estimate of the acceleration or second derivative of the height function. For example, can we trust the acceleration curves displayed in Figure 1.1?

The parametric growth curve proposed by Jolicoeur (1992) has the following form:

$$h(t) = a\frac{\sum_{\ell=1}^{3}[b_\ell(t+e)]^{c_\ell}}{1+\sum_{\ell=1}^{3}[b_\ell(t+e)]^{c_\ell}}. \tag{4.7}$$

Jolicoeur's model is now known to be a bit too smooth, and especially in the period before the pubertal growth spurt, but it does offer a reasonable account of most growth records for the comparatively modest investment of estimating eight parameters, namely a, e and $(b_\ell, c_\ell), \ell = 1, 2, 3$. The model has been fit to the Fels growth data (Roche, 1992) by R. D. Bock (2000), and from these fits it has been possible to summarize the variation of parameter values for both genders reasonably well using a multivariate normal distribution. The average parameter values are $a = 164.7, e = 1.474, \mathbf{b} = (0.3071, 0.1106, 0.0816)', \mathbf{c} = (3.683, 16.665, 1.474)'$. By sampling from this distribution, we can simulate the smooth part of as many records as we choose.

The standard error of measurement has also been estimated for the Fels data as a function of age by one of the authors, and Figure 4.2 summarizes this relation. We see height measurements are noisier during infancy, where the standard error is about eight millimeters, but by age six or so, the error settles down to about five millimeters. Simulated noisy data were generated from the smooth curves by adding independent random errors having a mean of zero and standard deviation defined by this curve to the smooth values at the sampling points. The reciprocal of the square of this function was used to define the entries of the weight matrix $\mathbf{W}$, which in this case was diagonal. The sampling ages were those of the Berkeley data, namely

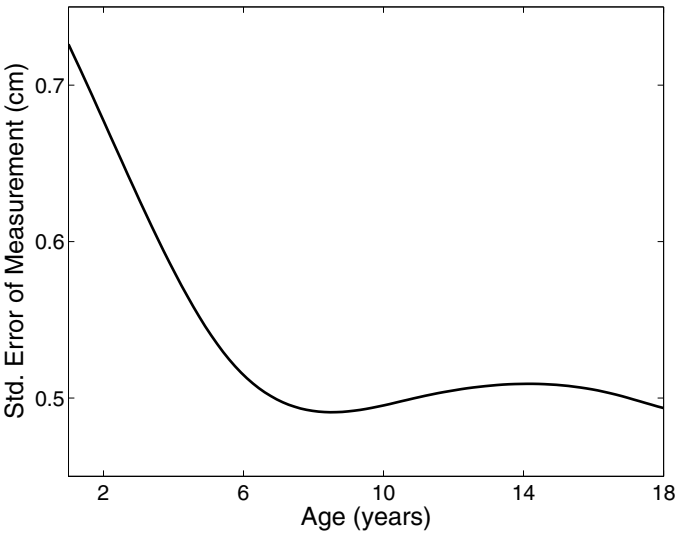

Figure 4.2. The estimated relation between the standard error of height measurements and age for females based on the Fels growth data.

quarterly between one and two years, annually between two to eight years, and twice a year after that to eighteen years of age.

We estimated the growth acceleration function by fitting a single set of data for a female. For the analysis, a set of 12 B-spline basis functions were used of order six and with equally spaced knots. We chose order six splines so that the acceleration estimate would be a cubic spline and hence smooth. A weighted least squares analysis was used with $\mathbf{W}$ being diagonal and with diagonal entries being the reciprocals of the squares of the standard errors shown in Figure 4.2.

Figure 4.3 shows how well we did. The maximum and minimum for the pubertal growth spurt are a little underestimated, and there are some peaks and valleys during childhood that aren't in the true curve. However, the estimate is much less successful at the lower and upper boundaries, and this example is a warning that we will have to look for ways to get better performance in these regions. On the whole, though, the important features in the true acceleration curve are reasonably reflected in the estimate.

4.4 Least squares fits as linear transformations of the data

The smoothing methods described in this chapter all have the property of being *linear*. Linearity simplifies computational issues considerably, and

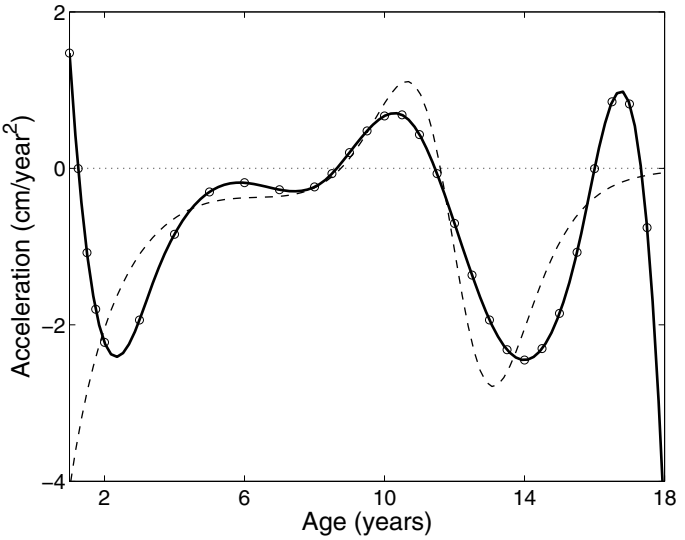

Figure 4.3. The solid curve is the estimated growth acceleration for a single set of simulated data, and the dashed curve is the errorless curve. The circles indicate the ages at which simulated observations were generated.

is convenient in a number of other ways. Most smoothing in practice gets done by linear procedures. Consequently, before we turn to other smoothing methods, we need to consider what linearity in a smoothing procedure means.

4.4.1 How linear smoothers work

A *linear smoother* estimates the function value $\hat{y}_j = \hat{x}(t_j)$ by a linear combination of the discrete observations

$$\hat{x}(t_j) = \sum_{\ell=1}^{n} S_j(t_\ell) y_\ell \ , \tag{4.8}$$

where $S_j(t_\ell)$ weights the ℓth discrete data value in order to generate the fit to y_j.

In matrix terms,

$$\hat{x}(\mathbf{t}) = \mathbf{S}\mathbf{y} \ , \tag{4.9}$$

where $\hat{x}(\mathbf{t})$ is a column vector containing the values of the estimate of function x at each sampling point t_j.

In the unweighted least squares case, for example, we see in (4.4) that

$$\mathbf{S} = \mathbf{\Phi}(\mathbf{\Phi}'\mathbf{\Phi})^{-1}\mathbf{\Phi}'. \tag{4.10}$$

In regression analysis, this matrix is often called the "hat matrix" because it converts the dependent variable vector $\mathbf{y}$ into its fit $\hat{\mathbf{y}}$.

In the context of least squares estimation, the smoothing matrix has the property of being a *projection matrix*. This means that it creates an image of data vector y on the space spanned by the columns of matrix $\mathbf{\Phi}$ such that the residual vector $\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}}$ is orthogonal to the fit vector $\hat{y}$,

$$(\mathbf{y} - \hat{\mathbf{y}})'\hat{\mathbf{y}} = 0 \ .$$

This in turn implies that the smoothing matrix has the property $\mathbf{SS} = \mathbf{S}$, a relation called *idempotency*. In the next chapter on roughness-penalized least squares smoothing, we shall see that property does not hold.

The corresponding smoothing matrix for weighted least squares smoothing is

$$\mathbf{S} = \mathbf{\Phi}(\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi})^{-1}\mathbf{\Phi}'\mathbf{W} \ . \tag{4.11}$$

Matrix $\mathbf{S}$ is still an orthogonal projection matrix, except that now the residual and fit vectors are orthogonal in the sense that

$$(\mathbf{y} - \hat{\mathbf{y}})'\mathbf{W}\hat{\mathbf{y}} = 0 \ .$$

In this case $\hat{\mathbf{y}} = \mathbf{S}\mathbf{y}$ is often said to be a *projection in the metric* $\mathbf{W}$.

Figure 4.4 shows the weights associated with estimating the growth acceleration curve in Figure 4.3 for ages six, twelve, and eighteen. For ages away from the boundaries, the weights have a positive peak centered on the age being estimated, and two negative side-lobes. For age twelve in the middle of the pubertal growth spurt for females, the observations receiving substantial weight, of either sign, range from ages seven to seventeen. This is in marked contrast to second difference estimates

$$D^2x(t_j) \approx \left(\frac{y_{j+1} - y_j}{t_{j+1} - t_j} - \frac{y_j - y_{j-1}}{t_j - t_{j+1}}\right)/(t_{j+1} - t_{j-1}),$$

which would only use three adjacent ages.

At the upper boundary, we see why there is likely to be considerable instability in the estimate. The final observation receives much more weight than any other value, and only observations back to age fifteen are used at all. The boundary estimate pools much less information than do interior estimates, and is especially sensitive to the boundary observations.

Many widely used smoothers are linear. The linearity of a smoother is a desirable feature for various reasons: The linearity property

$$\mathbf{S}(a\mathbf{y} + b\mathbf{z}) = a\mathbf{S}\mathbf{y} + b\mathbf{S}\mathbf{z}$$

is important for working out various properties of the smooth representation, and the simplicity of the smoother implies relatively fast computation. On the other hand, some nonlinear smoothers may be more adaptive to different behavior in different parts of the range of observation, and may be robust to outlying observations. Smoothing by the thresholded

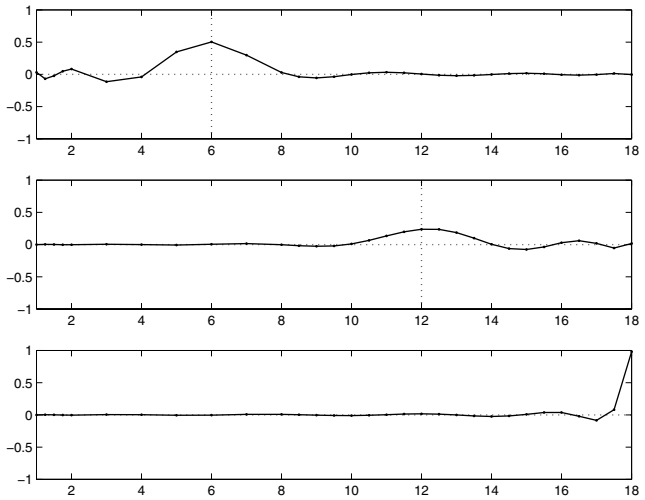

Figure 4.4. The top panel indicates how observations are weighted in order to estimate growth acceleration at age six in figure 4.3. The central panel shows the weights for age twelve, and the bottom for age eighteen. The dots indicate the ages at which simulated observations were generated.

wavelet transform, discussed in Section 3.6.1, is an important example of a nonlinear smoothing method.

Speed of computation can be critical; a smoother that is useful for a few hundred data points can be completely impractical for thousands. Smoothers that require a number of operations that is proportional to n to compute n smoothed values $\hat{x}(s_j)$, abbreviated $O(n)$ operations, are virtually essential for large n. If $\mathbf{S}$ is band-structured, meaning that only a small number K of values on either side of its diagonal in any row are nonzero, then $O(n)$ computation is assured.

4.4.2 The degrees of freedom of a linear smooth

We are familiar with the idea that the model for observed data offers an image of the data that has fewer *degrees of freedom* than are present in the original data. In most textbook situations, the concept of the degrees of freedom of a fit means simply the number of parameters estimated from the data that are required to define the model.

The notion of degrees of freedom applies without modification to data smoothing using least squares, where the number of parameters is the length K of the coefficient vector $\mathbf{c}$. The number of *degrees of freedom for error* is therefore $n - K$.

When we begin to use roughness penalty methods in Chapter 5, however, things will not be so simple, and we will need a more general way of computing the effective degrees of freedom of a smooth fit to the data, and consequently the corresponding degrees of freedom for error. We do this by using the "hat" matrix $\mathbf{S}$ by defining the degrees of freedom of the smooth fit to be

$$df = \text{trace}\,\mathbf{S} \tag{4.12}$$

where the trace of a square matrix means the sum of its diagonal elements. This more general definition yields exactly K for least squares fits, and therefore does not represent anything new. But this definition will prove invaluable in our later chapters.

There are also situations in which it may be more appropriate to use the alternative definition

$$df = \text{trace}\,(\mathbf{S}\mathbf{S}') \tag{4.13}$$

but most of the time (4.12) is employed. In any case, the two definitions give the same answer for least squares estimation.

4.5 Choosing the number K of basis functions

How do we choose the order of the expansion K? The larger K, the better the fit to the data, but of course we then risk also fitting noise or variation that we wish to ignore. On the other hand, if we make K too small, we may miss some important aspects of the smooth function x that we are trying to estimate.

4.5.1 The bias/variance trade-off

This trade-off can be expressed in another way. For large values of K, n the *bias* in estimating $x(t)$, that is

$$\texttt{Bias}[\hat{x}(t)] = x(t) - \text{E}[\hat{x}(t)], \tag{4.14}$$

is small. In fact, if the notion of additive errors having expectation zero expressed in (3.1) holds, then we know that the bias will be zero for $K = n$.

But of course, that is only half of the story. One of the main reasons that we do smoothing is to reduce the influence of noise or ignorable variation on the estimate $\hat{x}$. Consequently we are also interested in the *variance of estimate*

$$\texttt{Var}[\hat{x}(t)] = \text{E}[\{\hat{x}(t) - \text{E}[\hat{x}(t)]\}^2]\ . \tag{4.15}$$

If $K = n$, this is almost certainly going to be unacceptably high. Reducing variance leads us to look for smaller values of K, but of course not so small

as to make the bias unacceptable. The worse the signal-to-noise ratio in the data, the more reducing sampling variance will outweigh controlling bias.

One way of expressing what we really want to achieve is *mean-squared error*

$$\texttt{MSE}[\hat{x}(t)] = \mathrm{E}[\{\hat{x}(t) - x(t)\}^2] \ , \tag{4.16}$$

also called the $\mathcal{L}^2$ *loss function*. In most applications we can't actually minimize this since we have no way of knowing what $x(t)$ is without using the data. However, one of the most important equations in statistics links mean squared error to bias and sampling variance by the simple additive decomposition

$$\texttt{MSE}[\hat{x}(t)] = \texttt{Bias}^2[\hat{x}(t)] + \texttt{Var}[\hat{x}(t)] \ . \tag{4.17}$$

What this relation tells us is that it would be worthwhile to tolerate a little bias if the result is a big reduction in sampling variance. In fact, this is almost always the case, and is the fundamental reason for smoothing data in order to estimate functions. We will return to this matter in Chapter 5.

Figure 4.5 shows some total squared error measures as a function of various numbers of basis functions. The measures were computed by summing mean squared error, sampling variance and squared bias across the ages ranging from three to sixteen. This range was used to avoid ages near the boundaries, where the curve estimates tend to have much greater error levels. The results are based on smoothing 10,000 random samples constructed in the same manner as that in Figure 4.3.

Notice that the measures for sampling variance and squared bias sum to those for mean squared error, as in (4.17). Sampling variance increases rapidly when we use too many basis functions, but squared bias tends to decay more gently to zero at the same time. We see there that the best results for totaled mean squared error are obtained with ten and twelve basis functions, and we broke the tie by opting for the result with the least bias.

It may see surprising that increasing K does not always decrease bias. If so, recall that, when the order of a spline is fixed and knots are equally spaced, K B-splines do not span a space that lies within that defined by $K+1$ B-splines. Complicated effects due to knot spacing relative to sampling points can result in a lower-dimensional B-spline system actually producing better results than a higher-dimensional system.

Although the decomposition mean squared error (4.17) is helpful for expressing the bias/variance tradeoff in a neat way, the principle applies more widely. In fact, there are many situations where it is preferable to use other loss functions. For example, minimizing $\mathrm{E}[|\hat{x}(t) - x(t)|]$, called the $\mathcal{L}^1$ norm, is more effective if the data contain outliers. For this and nearly any fitting criterion or loss function for smoothing, we can assume that when bias goes down, sampling variance goes up, and some bias must be tolerated to achieve a stable estimate of the smooth trend in the data.

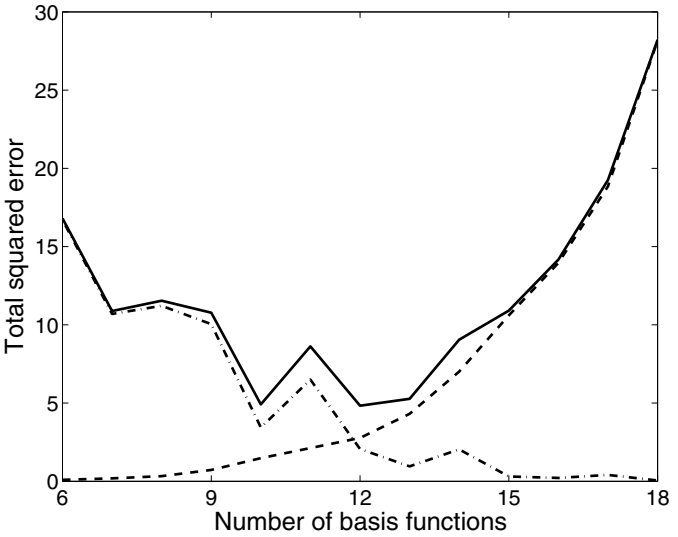

Figure 4.5. The heavy solid line indicates mean squared error totaled across the ages of observation between three and sixteen. The dashed line shows the totaled sampling variance, and the dotted-dashed line shows the totaled squared bias.

4.5.2 Algorithms for choosing K

The vast literature on multiple regression contains many ideas for deciding how many basis functions to use. For example, *stepwise variable selection* would proceed in a step-up fashion by adding basis functions one after another, testing at each step whether the added function significantly improves fit, and also checking that the functions already added continue to play a significant role. Conversely, *variable-pruning* methods are often used for high-dimensional models, and work by starting with a generous choice of K and dropping a basis function on each step that seems to not account for a substantial amount of variation.

These methods all have their limitations, and are often abused by users who do not appreciate these problems. The fact that there is no one gold standard method for the variable selection problem should warn us at this point that we face a difficult task in attempting to fix model dimensionality. The discrete character of the K-choice problem is partly to blame, and the methods described in Chapter 5 providing a continuum of smoothing levels will prove helpful.

4.6 Computing sampling variances and confidence limits

4.6.1 Sampling variance estimates

The estimation of the coefficient vector $\mathbf{c}$ of the basis function expansion $x = \mathbf{c}'\boldsymbol{\phi}$ by minimizing least squares defines a linear mapping (4.6) from the raw data vector $\mathbf{y}$ to the estimate. With this mapping in hand, it is a relatively simple matter to compute the sampling variance of the coefficient vector, and of anything that is linearly related to it.

We begin with the fact that if a random variable y is normally distributed with a variance-covariance matrix $\boldsymbol{\Sigma}_y$, then the random variable $\mathbf{Ay}$ defined by any matrix $\mathbf{A}$ has the variance-covariance matrix

$$\mathtt{Var}[\mathbf{Ay}] = \mathbf{A}\boldsymbol{\Sigma}_y\mathbf{A}' \ . \tag{4.18}$$

Now in this and other linear modelling situations that we will encounter, the model for the data vector $\mathbf{y}$, in this case $x(\mathbf{t})$, is regarded as a fixed effect having zero variance. Consequently, the variance-covariance matrix of y using the model $\mathbf{y} = x(\mathbf{t}) + \boldsymbol{\epsilon}$ is the variance-covariance matrix $\boldsymbol{\Sigma}_e$ of the residual vector $\boldsymbol{\epsilon}$. We must in some way use the information in the actual residuals to replace the population quantity $\boldsymbol{\Sigma}_e$ by a reasonable sample estimate $\hat{\boldsymbol{\Sigma}}_e$.

For example, to compute the sampling variances and covariances of the coefficients themselves in $\mathbf{c}$, we use that fact that in this instance

$$\mathbf{A} = (\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{W} \ .$$

to obtain

$$\mathtt{Var}[\mathbf{c}] = (\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Sigma}_e\mathbf{W}\boldsymbol{\Phi}(\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi})^{-1} \ . \tag{4.19}$$

When the standard model is assumed, $\boldsymbol{\Sigma}_e = \sigma^2\mathbf{I}$, and if unweighted least squares is used, then we obtain the simpler result that appears in textbooks on regression analysis

$$\mathtt{Var}[\mathbf{c}] = \sigma^2(\boldsymbol{\Phi}'\boldsymbol{\Phi})^{-1} \ . \tag{4.20}$$

However, in our functional data analysis context there will seldom be much interest in interpreting the coefficient vector $\mathbf{c}$ itself. Rather, we will want to know the sampling variance of some quantity computed from these coefficients. For example, we might want to know the sampling variance of the the fit to the data defined by $x(t) = \boldsymbol{\phi}(t)'\mathbf{c}$. Since we now have in hand the sampling variance of $\mathbf{c}$ through (4.19) or (4.20), we can simply apply result (4.18) again to get

$$\mathtt{Var}[\hat{x}(t)] = \boldsymbol{\phi}(t)'\mathtt{Var}[\mathbf{c}]\boldsymbol{\phi}(t) \tag{4.21}$$

and the variances of all the fitted values corresponding to the sampling values t_j are in the diagonal of the matrix

$$\mathtt{Var}[\hat{\mathbf{y}}] = \mathbf{\Phi}\mathtt{Var}[\mathbf{c}]\mathbf{\Phi}'$$

which, in the standard model/unweighted least squares case, and using (4.10), reduces to

$$\mathtt{Var}[\hat{\mathbf{y}}] = \sigma^2\mathbf{\Phi}(\mathbf{\Phi}'\mathbf{\Phi})^{-1}\mathbf{\Phi}' = \sigma^2\mathbf{S} \ .$$

4.6.2 Estimating $\mathbf{\Sigma}_e$

Clearly our estimates of sampling variances are only as good as our estimates of the variances and covariances among the residuals ϵ_j.

When we are smoothing a single curve, the total amount of information involved is insufficient for much more than estimating either a single constant variance σ^2 assuming the standard model for error, or at most a variance function with values $\sigma^2(t)$, that has fairly mild variation over t. It is important to use methods which produce relatively unbiased estimate of variance in order to avoid underestimating sampling variance. For example, if the standard model for error is accepted,

$$s^2 = \frac{1}{n-K}\sum_j^n (y_j - \hat{y}_j)^2 \tag{4.22}$$

is much preferred as an estimate of σ^2 than the maximum likelihood estimate that involves dividing by n. In fact, we shall see in the next chapter that this estimate is related to a popular more general method for choosing smoothing level called *generalized cross-validation.*

One reasonable strategy for choosing K is to add basis functions until s^2 fails to decrease substantially. Figure 4.6 shows how s decreases to a value of about 0.56 degrees Celsius by the time we use 109 Fourier basis functions for smoothing the Montreal temperature data shown in Figure 4.1. There are places where s^2 is even lower, but we worried that the minimum at 240 basis functions corresponded to over-fitting the data.

A common strategy for estimating at least a limited number of covariances in $\mathbf{\Sigma}_e$ given a small N, or even $N = 1$, is to assume an autoregressive (AR) structure for the residuals. This is often realistic, since adjacent residuals are frequently correlated because they are mutually influenced by unobserved variables. For example, the weather on one day is naturally likely to be related to the weather on the previous day because of the influence of large slow-moving low or high pressure zones. An intermediate level text on regression analysis such as Draper and Smith (1998) can be consulted for details on how to estimate AR structures among residuals.

When a substantial number N of replicated curves are available, as in the growth curve data and Canadian weather data, we can attempt more sophisticated and detailed estimates of $\mathbf{\Sigma}_e$. For example, we may opt for

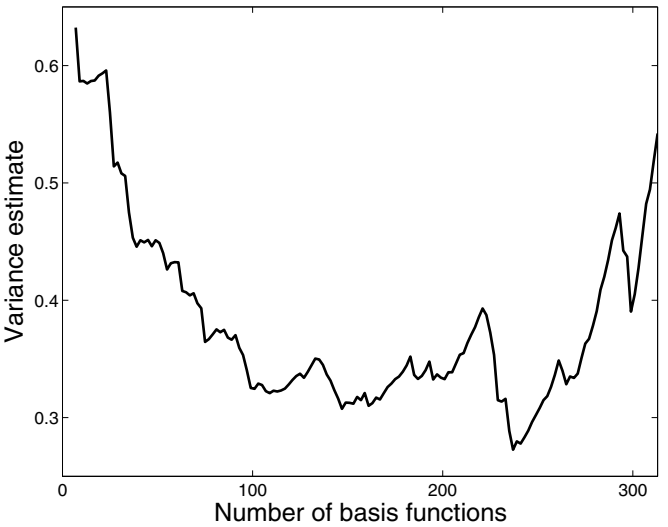

Figure 4.6. The relation between the number of Fourier basis functions and the unbiased estimate of the residual variance (4.22) in fitting the Montreal temperature data.

estimating the entire variance-covariance matrix from the N by n matrix $\mathbf{E}$ of residuals by

$$\hat{\mathbf{\Sigma}}_e = (N-1)^{-1}\mathbf{E}'\mathbf{E}.$$

However, even then, an estimate of a completely unrestricted $\mathbf{\Sigma}_e$ requires the estimation of $n(n-1)/2$ variances and covariances from N replications, and it is unlikely that data with the complexity of the daily weather records would ever have N sufficiently large to do this accurately.

4.6.3 Confidence limits

Confidence limits are typically computed by adding and subtracting a multiple of the standard errors, that is, the square root of the sampling variances, to the actual fit. For example, 95% limits correspond to about two standard errors up and down from a smooth fit. These standard errors are estimated using (4.21). Confidence limits on fits computed in this way are called *point-wise* because they reflect confidence regions for *fixed* values of t rather than regions for the curve as a whole.

Figure 4.7 shows the temperatures during the 16 days over which the January thaw takes place in Montreal, along with the smooth to the data and 95% point-wise confidence limits on the fit. The standard error of the estimated fit was 0.26 degrees Celsius.

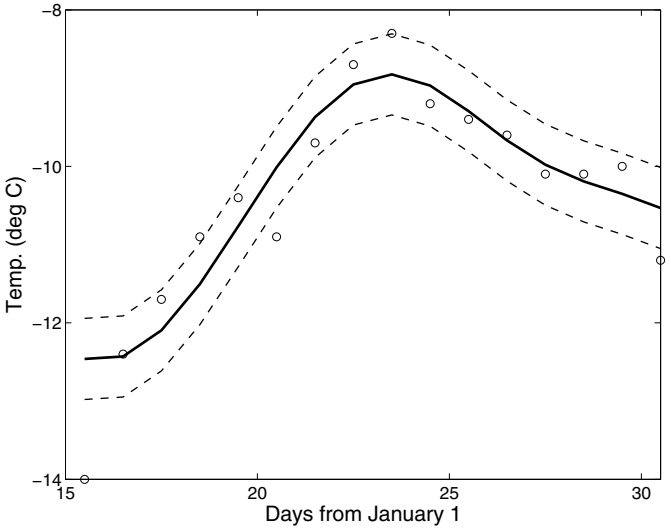

Figure 4.7. The temperatures over the mid-winter thaw for the Montreal temperature data. The solid line is the smooth curve estimated in Figure 4.1 and the lower and upper dashed lines are estimated 95% point-wise confidence limits for this fit.

We will have much to say in the next chapter and elsewhere about the hazards of placing too much faith in sampling variances and confidence limits estimated in these ways. But we should at least note two important ways in which confidence limits computed in this way may be problematic. First, it is implicitly assumed that K is a fixed constant, but the reality is that K for smoothing problems is more like a parameter estimated from the data, and consequently the size of these confidence limits does not reflect the uncertainty in our knowledge of K. Secondly, the smooth curve to which we add and subtract multiples of the standard error to get point-wise limits is itself subject to bias, and especially in regions of high curvature. We can bet, for example, that the solid curve in Figure 4.7 is too low on January 23rd, the center of the January thaw. Thus, the confidence limits calculated in this way are themselves biassed, and the region covered by them may not be quite as advertised.

4.7 Fitting data by localized least squares

For a smoothing method to make any sense at all, the value of the function estimate at a point t must be influenced mostly by the observations near t. This feature is an implicit property of the estimators we have considered

so far. In this section, we consider estimators where the local dependence is made more explicit by means of local weight functions.

Keeping within the domain of linear smoothing means that our estimate of the value of function x at argument t_j is of the form

$$x(t_j) = \sum_{\ell}^{n} w_\ell y_\ell \ .$$

It seems intuitively reasonable that the weights w_ℓ will only be relatively large for sampling values t_ℓ fairly close to the target value t_j. And, indeed, this tends to hold for the basis function smoothers (4.10) and (4.11).

We now look at smoothing methods that make this *localized weighting principle* explicit. The localizing weights w_j are simply constructed by a location and scale change of a *kernel* function with values $\texttt{Kern}(u)$. This kernel function is designed to have most of its mass concentrated close to 0, and to either decay rapidly or disappear entirely for $|u| \geq 1$. Three commonly used kernels are

Uniform:	$\texttt{Kern}(u) = 0.5$ for $\lvert u\rvert \leq 1$,	0 otherwise
Quadratic:	$\texttt{Kern}(u) = 0.75(1-u^2)$ for $\lvert u\rvert \leq 1$,	0 otherwise
Gaussian:	$\texttt{Kern}(u) = (2\pi)^{-1/2}\exp(-u^2/2)$.	

If we then define weight values to be

$$w_\ell(t) = \texttt{Kern}\left(\frac{t_\ell - t_j}{h}\right) \ , \tag{4.23}$$

then substantially large values $w_\ell(t)$ as a function of ℓ are now concentrated for t_ℓ in the vicinity of t_j. The degree of concentration is controlled by the size of h. The concentration parameter h is usually called the *bandwidth* parameter, and small values imply that only observations close to t receive any weight, while large h means that a wide-sweeping average uses values that are a considerable distance from t.

4.7.1 Kernel smoothing

The simplest and classic case of an estimator that makes use of local weights is the *kernel estimator*. The estimate at a given point is a linear combination of local observations,

$$\hat{x}(t) = \sum_{j}^{n} S_j(t) y_j \tag{4.24}$$

for some suitably defined weight functions S_j. Probably the most popular kernel estimator the Nadaraya-Watson estimator (Nadaraya, 1964; Watson,

1964) is constructed by using the weights

$$S_j(t) = \frac{\texttt{Kern}[(t_j - t)/h]}{\sum_r \texttt{Kern}[(t_r - t)/h]}. \tag{4.25}$$

Although the weight values $w_j(t)$ for the Nadaraya-Watson method are normalized to have a unit sum, this is not essential. The weights developed by Gasser and Müller (1979, 1984) are constructed as follows:

$$S_j(t) = \frac{1}{h} \int_{\bar{t}_{j-1}}^{\bar{t}_j} \texttt{Kern}(\frac{u - t}{h})\ du, \tag{4.26}$$

where $\bar{t}_j = (t_{j+1} + t_j)/2, 1 < j < n, \bar{t}_0 = t_1$ and $\bar{t}_n = t_n$. These weights are faster to compute, deal more sensibly with unequally spaced arguments, and have good asymptotic properties.

The need for fast computation favors the compact support uniform and quadratic kernels, and the latter is the most efficient when only function values are required and the true underlying function x is twice-differentiable. The Gasser-Müller weights using the quadratic kernel are

$$S_j(t) = \frac{1}{4}[\{3r_{j-1}(t) - r_{j-1}^3(t)\} - \{3r_j(t) - r_j^3(t)\}]$$

for $|t_j - t| \leq h$ and 0 otherwise, and where

$$r_j(t) = \frac{t - \bar{t}_j}{h}. \tag{4.27}$$

We need to take special steps if t is within h units of either t_1 or t_n. These measures can consist of simply extending the data beyond this range in some reasonable way, making h progressively smaller as these limits are approached, or sophisticated modifications of the basic kernel function `Kern`. The problem that all kernel smoothing algorithms have of what to do near the limits of the data is one of their major weaknesses, and especially when h is large relative to the sampling rate.

Estimating the derivative just by taking the derivative of the kernel smooth is not usually a good idea, and in any case kernels such as the uniform and quadratic are not differentiable. However, kernels specifically designed to estimate a derivative of fixed order can be constructed by altering the nature of kernel function `Kern`. For example, a kernel $\texttt{Kern}(u)$ suitable for estimating the first derivative must be zero near $u = 0$, positive above zero, and negative below, so that it is a sort of smeared-out version of the first central difference. The Gasser-Müller weights for the estimation of the first derivative are

$$S_j(t) = \frac{15}{16h}[\{r_{j-1}^4(t) - 2r_{j-1}^2(t)\} - \{r_j^4(t) - 2r_j^2(t)\}] \tag{4.28}$$

and for the second derivative are

$$S_j(t) = \frac{105}{16h^2}[\{2r_{j-1}^3(t) - r_{j-1}^5(t) - r_{j-1}(t)\} - \{2r_j^3(t) - r_j^5(t) - r_j(t)\}] \tag{4.29}$$

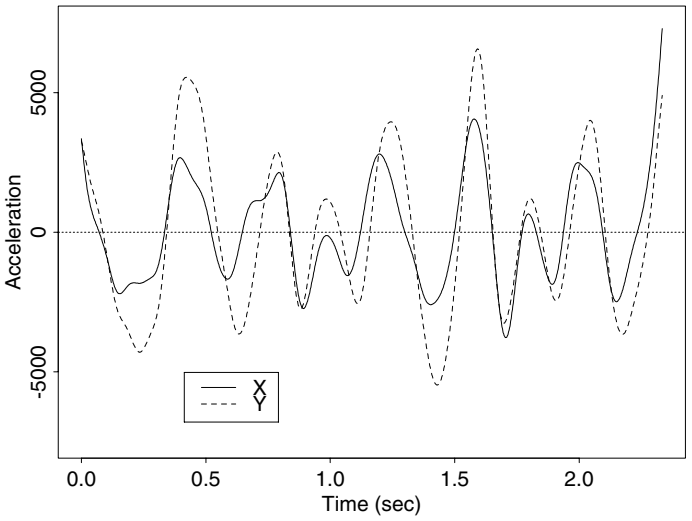

Figure 4.8. The second derivative or acceleration of the coordinate functions for the handwriting data. Kernel smoothing was used with a bandwidth $h = 0.075$.

for $|t_j - t| \leq h$ and 0 otherwise. It is usual to need a somewhat larger value of bandwidth h to estimate derivatives than is required for estimating the function.

Figure 4.8 shows the estimated second derivative or acceleration for the two handwriting coordinate functions. After inspection of the results produced by a range of bandwidths, we settled on $h = 0.075$. This implies that any smoothed acceleration value is based on about 150 milliseconds of data and about 90 values of y_j.

4.7.2 Localized basis function estimators

The ideas of kernel estimators and basis function estimators can, in a sense, be combined to yield *localized basis function estimators*, which encompass a large class of function and derivative estimators. The basic idea is to extend the least squares criterion (4.1) to give a local measure of error as follows:

$$\texttt{SMSSE}_t(\mathbf{y}|\mathbf{c}) = \sum_{j=1}^{n} w_j(t)[y_j - \sum_{k=1}^{K} c_k \phi_k(t_j)]^2, \tag{4.30}$$

where the weight functions w_j are constructed from the kernel function using (4.23).

In matrix terms,

$$\texttt{SMSSE}_t(\mathbf{y}|\mathbf{c}) = (\mathbf{y} - \mathbf{\Phi c})' \mathbf{W}(t)(\mathbf{y} - \mathbf{\Phi c}), \tag{4.31}$$

where $\mathbf{W}(t)$ is a diagonal matrix containing the weight values $w_j(t)$ in its diagonal. Don't be confused by the formal similarity of this expression with (4.5); the matrix $\mathbf{W}(t)$ plays a very different role here.

Choosing the coefficients $\mathbf{c}(t)$ to minimize $\mathtt{SMSSE}_t$ yields

$$\hat{\mathbf{c}}(t) = [\mathbf{\Phi}'\mathbf{W}(t)\mathbf{\Phi}]^{-1}\mathbf{\Phi}'\mathbf{W}(t)\mathbf{y},$$

and substituting back into the expansion $\hat{x}(t) = \sum_{k=1}^{K} \hat{c}_k\phi_k(t)$ gives a linear smoothing estimator of the form (4.8) with smoothing weight values $S_j(t)$ being the elements of the vector

$$S(t) = \mathbf{W}(t)\mathbf{\Phi}[\mathbf{\Phi}'\mathbf{W}(t)\mathbf{\Phi}]^{-1}\boldsymbol{\phi}(t), \tag{4.32}$$

where $\boldsymbol{\phi}(t)$ is the vector with elements $\phi_k(t)$.

The weight values $w_j(t)$ in (4.30) are designed to have substantially nonzero values only for observations located close to the evaluation argument t at which the function is to be estimated. This implies that only the elements in $S(t)$ in (4.32) associated with data arguments values t_j close to evaluation argument t are substantially different from zero, and consequently that $\hat{x}(t)$ is essentially a linear combination of only the observations y_j in the neighborhood of t.

Since the basis has only to approximate a limited segment of the data surrounding t, the basis can do a better job of approximating the local features of the data and, at the same time, we can expect to do well with only a small number K of basis functions. The computational overhead for a single t depends on the number of data argument values t_j for which $w_j(t)$ is nonzero, as well as on K. Both of these are typically small. However, the price we pay for this flexibility is that the expansion must essentially be carried out anew for each evaluation point t.

4.7.3 Local polynomial smoothing

It is interesting to note that the Nadaraya-Watson kernel estimate can be obtained as a special case of the localized basis expansion method by setting $K = 1$ and $\phi_i(t) = 1$. A popular class of methods is obtained by extending from a single basis function to a low order polynomial basis. Thus we choose the estimated curve value $\hat{x}(t)$ to minimize the localized least squares criterion

$$\mathtt{SMSSE}_t(\mathbf{y}|\mathbf{c}) = \sum_{j=1}^{n} \mathtt{Kern}_h(t_j, t)[y_j - \sum_{\ell=0}^{L} c_\ell(t - t_j)^\ell]^2 \ . \tag{4.33}$$

Setting $L = 0$, we recover the Nadaraya-Watson estimate. For values of $L \geq 1$, the function value and L of its derivatives can be estimated by the corresponding derivatives of the locally fitted polynomial at t. In general, the value of L should be at least one and preferably two higher than the highest order derivative required.

Local polynomial smoothing has a strong appeal; see, for example, the detailed discussion provided by Fan and Gijbels (1996). Its performance is superior in the region of the boundaries, and it adapts well to unequally spaced argument values. Local linear expansions give good results when we require only an estimate of the function value. They can easily be adapted in various ways to suit special requirements, such as robustness, monotonicity and adaptive bandwidth selection.

4.7.4 Choosing the bandwidth h

In all the localized basis expansion methods we have considered, the primary determinant of the degree of smoothing is the bandwidth h, rather than the number of basis functions used. The bandwidth controls the balance between two considerations: bias and variance in the estimate. Small values of h imply that the expected value of the estimate $\hat{x}(t)$ must be close to the true value $x(t)$, but the price we pay is in terms of the high variability of the estimate, since it is be based on comparatively few observations. On the other hand, variability can always be decreased by increasing h, although this is inevitably at the expense of higher bias, since the values used cover a region in which the function's shape varies substantially. Mean squared error at t, which is the sum of squared bias and variance, provides a composite measure of performance.

There is a variety of data-driven automatic techniques for choosing an appropriate value of h, usually motivated by the need to minimize mean squared error across the estimated function. Unfortunately, none of these can always be trusted, and the problem of designing a reliable data-driven bandwidth selection algorithm continues to be a subject of active research and considerable controversy. Our own view is that trying out a variety of values of h and inspecting the consequences graphically remains a suitable means of resolving the bandwidth selection problem for most practical problems.

4.7.5 Summary of localized basis methods

Explicitly localized smoothing methods such as kernel smoothing and local polynomial smoothing are easy to understand and have excellent computational characteristics. The role of the bandwidth parameter h is obvious, and as a consequence it is even possible to allow h to adapt to curvature variation. On the negative side, however, is the instability of these methods near the boundaries of the interval, although local polynomial smoothing performs substantially better than kernel smoothing in this regard. As with unweighted basis function expansions, it is well worthwhile to consider matching the choice of basis functions to known characteristics of the data, especially in regions where the data are sparse, or where they are asymmetrically placed around the point t of interest, for example near

the boundaries. The next chapter on the roughness penalty approach looks at the main competitor to kernel and local polynomial methods: spline smoothing.

4.8 Further reading and notes

This chapter and the next are so tightly related that you may prefer to read on, and then consider these notes along with those found there.

Much of the material in this chapter is an application of multiple regression, and references such as Draper and Smith (1998) are useful supplements, and especially on other ways of estimating residual covariance structures.

For more complete treatments of data smoothing, we refer the reader to sources such as Eubank (1999), Green and Silverman (1994), Härdle (1990) and Simonoff (1996). Fan and Gijbels (1996) and Wand and Jones (1995) focus more on kernel smoothing and local polynomial methods. Hastie and Tibshirani (1990) use smoothing methods in the context of estimating the generalized additive or GAM model, but their account of smoothing is especially accessible. Data smoothing also plays a large role in data mining and machine learning, and Hastie, Tibshirani and Friedman (2001) is a recent reference on these topics.

We use spline expansions by fixing the knot locations in advance of the analysis, and optimizing fit with respect to the coefficients multiplying the spline basis functions defined by this fixed knot sequence. The main argument for regarding knots as fixed is computational convenience, but there is also a large literature on using the data to estimate knot locations. Such splines are often called *free-knot splines*. The least squares fitting criterion is highly nonlinear in knot locations, and the computational challenges are severe. Nevertheless, in certain applications where strong curvature is localized in regions not known in advance, this is the more natural approach. For recent contributions to free-knot spline model estimation, see Lindstrom (2002), Lindstrom and Kotz (2004) and Mao and Zhao (2003).

We hope that we have not left the reader with the impression that least squares estimation is the only way to do smoothing. One of the most important developments in statistics in recent years has been the development of *quantile regression* methods by R. Koenker and S. Portnoy, where the model estimates a quantile of the conditional distribution of the dependent variable. Least squares methods, by contrast, attempt to estimate the mean of this distribution. Quantile regression minimizes the sum of absolute values of residuals rather than their sum of squares. Koenker and Portnoy (1994) applied quantile regression to the spline smoothing problem.

5

Smoothing functional data with a roughness penalty

5.1 Introduction

We saw in Chapter 4 that basis expansions can provide good approximations to functional data provided that the basis functions have the same essential characteristics as the process generating the data. Thus, a Fourier basis is useful if the functions we observe are periodic and do not exhibit fluctuations in any particular interval that are much more rapid than those elsewhere. However, fitting basis expansions by least squares implies clumsy discontinuous control over the degree of smoothing, and we wonder if it is not possible to get better results with other methods.

Kernel smoothing and local polynomial fitting techniques, on the other hand, are based on appealing, efficient and easily understood algorithms that are fairly simple modifications of classic statistical techniques. They offer continuous control of the smoothness of the approximation, but they are seldom optimal solutions to an explicit statistical problem, such as minimizing a measure of total squared error, and their rather heuristic character makes extending them to other smoothing situations difficult.

In this chapter we introduce a more powerful option for approximating discrete data by a function. The *roughness penalty* or *regularization* approach retains the advantages of the basis function and local expansion smoothing techniques developed in Chapter 4, but circumvents some of their limitations. More importantly, it adapts gracefully to the more general functional data analysis problems that we consider in subsequent chapters. Finally, it often produces better results, and especially in the estimation of

derivatives. This roughness penalty approach will be our smoothing method of choice throughout this book.

Like the least squares methods of Chapter 4, roughness penalty methods are based on optimizing a fitting criterion that defines what a smooth of the data is trying to achieve. But here the precise meaning of "smooth" is expressed explicitly at the level of the criterion being optimized, rather than implicitly in terms of the number of basis functions being used. Moreover, roughness penalty approaches can be applied to a much wider range of smoothing problems than simply estimating a curve x from observations of $x(t_j)$ for certain points t_j. Green and Silverman (1994) discuss a variety of statistical problems that can be approached using roughness penalties, including those where the data's dependence on the underlying curve is akin to the dependence on parameters in generalized linear models. Here we extend still further the scope of roughness penalty methods by discussing various functional data analysis contexts where roughness penalties are an elegant way to introduce smoothing into the analysis.

Figure 5.1 shows what we are trying to achieve. The refinery data from the top panel of Figure 1.4 show measurements that seem flat up to time 67, followed by a sharp upward turn and then an smooth approach toward a new level. In Chapter 17 we will want to model the change or derivative of this trend. A good estimate should show near zero derivative to time 64, an abrupt increase to a maximum value, and then an approximately exponential decay thereafter. Three estimates of this derivative computed by penalizing the roughness of the derivative are shown in the Figure. The best of these seems to be the heavy line, which combines a near zero value on the left with the abrupt upward turn, high peak value, and fairly smooth decay that we want. The smoother of the other two curves fails at both the upward turn and at the peak, and the other is too wild below time 50.

5.2 Spline smoothing

Let us consider how regularization works in the simplest functional case when the goal is to estimate a non-periodic function x on the basis of discrete and noisy observations in a vector $\mathbf{y}$. We continue with the data-smoothing problem described in Chapter 4. However, we will reserve the term "spline smoothing" for using roughness penalties in the way described in this section. By contrast, the smoothing literature often refers to the least squares fitting of B-spline expansions that we described in Chapter 4 as "regression spline smoothing."

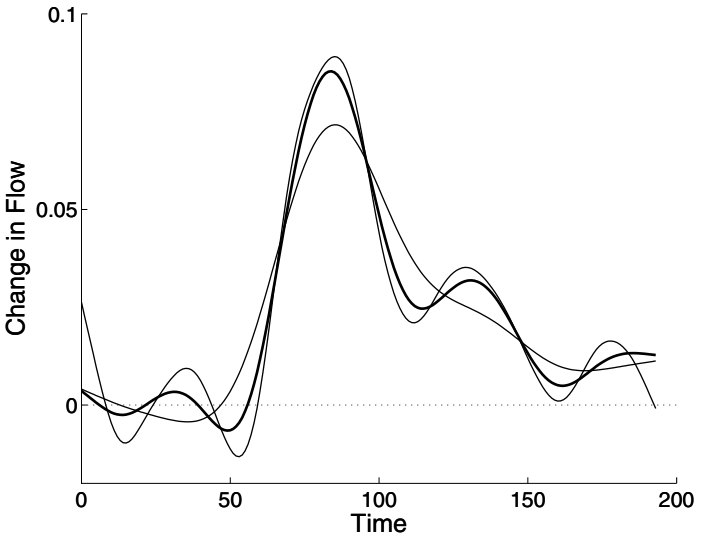

Figure 5.1. Three estimates of the rate of change or first derivative of the data shown in the top panel of Figure 1.4. Each curve has its roughness penalized.

5.2.1 Two competing objectives in function estimation

The spline smoothing method estimates a curve x from observations $y_j = x(t_j) + \epsilon_j$ by making explicit two conflicting goals in curve estimation. On the one hand, we wish to ensure that the estimated curve gives a good fit to the data, for example in terms of the residual sum of squares $\sum[y_j - x(t_j)]^2$. On the other hand, we do not wish the fit to be too good if this results in a curve x that is excessively "wiggly" or locally variable.

These competing aims correspond to the elements of the basic principle of statistics, discussed in Section 4.5,

$$\text{Mean squared error} = \text{Bias}^2 + \text{Sampling variance},$$

where bias, sampling variance and mean squared error were defined in Section 4.5.1. A completely unbiased estimate of the function value $x(t_j)$ can be produced by a curve fitting y_j exactly, since this observed value is itself an unbiased estimate of $x(t_j)$ according to our error model. But any such curve must have high variance, manifested in the rapid local variation of the curve.

In spline smoothing, as in other smoothing methods, the mean squared error, usually abbreviated MSE, is one way of capturing what we usually mean by the quality of estimate. We noted in Section 4.5.1 that other loss functions may be preferable in certain situations, but that the notion of a trade-off between bias and sampling variance applies more widely to these situations as well, although not with this exact decomposition.

MSE can often be dramatically reduced by sacrificing some bias in order to reduce sampling variance, and this is a key reason for imposing smoothness on the estimated curve. By requiring that the estimate vary only gently from one value to another, we are effectively "borrowing information" from neighboring data values, thereby expressing our faith in the regularity of the underlying function x that we are trying to estimate. This pooling of information is what makes our estimated curve more stable, at the cost of some increase in bias. The roughness penalty makes explicit what we sacrifice in bias to achieve an improvement MSE or some other loss function.

5.2.2 Quantifying roughness

Here is popular way to quantify the notion "roughness" of a function. The square of the second derivative $[D^2x(t)]^2$ of a function at t is often called its *curvature* at t, since a straight line, which we all agree has no curvature, also has a zero second derivative. Consequently, a natural measure of a function's roughness is the integrated squared second derivative

$$\texttt{PEN}_2(x) = \int [D^2x(s)]^2\, ds\ . \tag{5.1}$$

Highly variable functions can be expected to yield high values of $\texttt{PEN}_2(x)$ because their second derivatives are large over at least some of the range of interest. For example, consider the two acceleration curves displayed in Figure 4.3, the estimated and actual growth acceleration according to the Jolicoeur model. The values of $\texttt{PEN}_2(x)$ for these curves are 0.22 and 1.42, respectively, indicating the the estimated acceleration curve is substantially rougher than the true curve.

Of course, since these curves are themselves second derivatives, these values are actually the values of

$$\texttt{PEN}_4(x) = \int [D^4x(s)]^2\, ds,$$

where x is a height function. This suggests that we may need to generalize the roughness penalty (5.1) by allowing a derivative $D^m x$ of arbitrary order so as to work with the penalty

$$\texttt{PEN}_m(x) = \int [D^m x(s)]^2\, ds\ . \tag{5.2}$$

5.2.3 The penalized sum of squared errors fitting criterion

We now need to modify the last squares fitting criterion (4.5), defined in Chapter 4, so as to allow the roughness penalty $\texttt{PEN}_2(x)$ to play a role in defining $x(s)$. Let $x(\mathbf{t})$ be the vector resulting from function x being

evaluated at the vector $\mathbf{t}$ of argument values. We define a compromise that explicitly trades off smoothness against data fit by defining the *penalized* residual sum of squares as

$$\texttt{PENSSE}_\lambda(x|\mathbf{y}) = [\mathbf{y} - x(\mathbf{t})]'\mathbf{W}[\mathbf{y} - x(\mathbf{t})]^2 + \lambda \times \texttt{PEN}_2(x) \ , \tag{5.3}$$

Our estimate of the function is obtained by finding the function x that minimizes $\texttt{PENSSE}_\lambda(x)$ over the space of functions x for which $\texttt{PEN}_2(x)$ is defined.

The parameter λ is a *smoothing parameter* that measures the rate of exchange between fit to the data, as measured by the residual sum of squares in the first term, and variability of the function x, as quantified by $\texttt{PEN}_2(x)$ in the second term. As λ becomes larger and larger, functions which are not linear must incur a more and more substantial roughness penalty through the term $\texttt{PEN}_2(x)$, and consequently the composite criterion $\texttt{PENSSE}_\lambda(x)$ must place more and more emphasis on the smoothness of x and less and less on fitting the data. For this reason, as $\lambda \to \infty$ the fitted curve x must approach the standard linear regression to the observed data, where $\texttt{PEN}_2(x) = 0$.

On the other hand, for small λ the curve tends to become more and more variable since there is less and less penalty placed on its roughness, and as $\lambda \to 0$ the curve x approaches an *interpolant* to the data, satisfying $x(t_j) = y_j$ for all j. However, even in this limiting case the interpolating curve is not arbitrarily variable; instead, it is the smoothest twice-differentiable curve that exactly fits the data.

5.2.4 The structure of a smoothing spline

Suppose for the moment that we make no assumptions about function x except that it has a second derivative.[1] We also assume here that sampling points $t_j, j = 1, \ldots, n$ are distinct. What kind of function minimizes this penalized error sum of squares?

A remarkable theorem, found in de Boor (2002) and other more advanced texts on smoothing, states that the curve x that minimizes $\texttt{PENSSE}_\lambda(x|y)$ is a cubic spline with knots at the data points t_j. Note that we have not here assumed anything about how x is constructed; the spline structure of x is a consequence of this theorem, in which an objective function is optimized with respect to an entire function. Solutions to problems that involve optimizing with respect to functions rather than with respect to parameters are called *variational problems.*

Placing knots at data points eliminates one of the issues in the use of regression splines: where to place the knots. Smoothing splines adapt nat-

[1] More technically, a slightly weaker assumption is required: that the integral of the squared second derivative is finite.

urally to unequal spacing of sampling points, and thus automatically take advantage of regions where data density is high, and at the same time are especially smooth over regions where there are few observations.

The most common computational technique for spline smoothing is to use an order four B-spline basis function expansion with knots at the sampling points, and to minimize criterion (5.3) with respect to the coefficients of the expansion. In this case, the fitting function is piece-wise cubic, and the method is often referred to as *cubic spline smoothing*.

Recalling the relation between number of knots, the order of the spline and the number of basis functions that was described in Chapter 3, using order four B-splines in this way implies that we have $n+2$ basis functions, which are obviously enough to fit n data points exactly if $\lambda = 0$.

5.2.5 How spline smooths are computed

Reminding ourselves of expressions and relations drawn from Chapter 4 will help us to see how the use of a roughness penalty changes the smoothing process from a projection to something that generalizes the idea of a projection.

Recall that, without a roughness penalty, the coefficient vector $\mathbf{c}$ in the expansion

$$x(t) = \sum_{k}^{K} c_k \phi_k(t) = \mathbf{c}'\boldsymbol{\phi}(t) = \boldsymbol{\phi}'(t)\mathbf{c},$$

where $\mathbf{c}$ is the K-vector of coefficients and $\boldsymbol{\phi}$ is the K-vector of basis functions, has the solution

$$\hat{\mathbf{c}} = (\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{W}'\mathbf{y} \tag{5.4}$$

where n by K matrix $\boldsymbol{\Phi}$ contains the values of the K basis functions at the n sampling points, $\mathbf{W}$ is a weight matrix to allow for possible covariance structure among residuals, and where $\mathbf{y}$ is the vector of discrete data to be smoothed. The corresponding expression for the vector of fits to the data is

$$\hat{\mathbf{y}} = \boldsymbol{\Phi}(\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{W}\mathbf{y} = \mathbf{S}_\phi\mathbf{y}, \tag{5.5}$$

where $\mathbf{S}_\phi$ is the projection operator

$$\mathbf{S}_\phi = \boldsymbol{\Phi}(\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}'\mathbf{W} \tag{5.6}$$

corresponding to the basis system $\boldsymbol{\phi}$.

We can re-express the roughness penalty $\texttt{PEN}_m(x)$ in matrix terms as follows.

$$\texttt{PEN}_m(x) \quad = \quad \int [D^m x(s)]^2 \, ds$$

$$
\begin{aligned}
&= \int [D^m \mathbf{c}'\boldsymbol{\phi}(s)]^2 \, ds \\
&= \int \mathbf{c}' D^m \boldsymbol{\phi}(s) D^m \boldsymbol{\phi}'(s) \mathbf{c} \, ds \\
&= \mathbf{c}' [\int D^m \boldsymbol{\phi}(s) D^m \boldsymbol{\phi}'(s) \, ds] \mathbf{c} \\
&= \mathbf{c}' \mathbf{R} c \ , \qquad (5.7)
\end{aligned}
$$

where

$$\mathbf{R} = \int D^m \boldsymbol{\phi}(s) D^m \boldsymbol{\phi}'(s) \, ds \ . \qquad (5.8)$$

Note that we will often encounter matrices like $\mathbf{R}$ that contain integrals of *outer products* of vectors of functions, and it will keep the notation cleaner if we can suppress the argument s and ds and use the notation

$$\mathbf{R} = \int D^m \boldsymbol{\phi} D^m \boldsymbol{\phi}' \ .$$

By adding the error sum of squares $\mathtt{SSE}(y|c)$ and $\mathtt{PEN}_m(x)$ multiplied by a smoothing parameter λ, we obtain

$$\mathtt{PENSSE}_m(\mathbf{y}|\mathbf{c}) = (\mathbf{y} - \boldsymbol{\Phi}\mathbf{c})' \mathbf{W} (\mathbf{y} - \boldsymbol{\Phi}\mathbf{c}) + \lambda \mathbf{c}' \mathbf{R} \mathbf{c} \ . \qquad (5.9)$$

Taking the derivative with respect to parameter vector $\mathbf{c}$, we obtain

$$-2\boldsymbol{\Phi}'\mathbf{W}\mathbf{y} + \boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi}\mathbf{c} + \lambda \mathbf{R}\mathbf{c} = 0,$$

from which we obtain the expression for the estimated coefficient vector

$$\hat{\mathbf{c}} = (\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi} + \lambda \mathbf{R})^{-1} \boldsymbol{\Phi}'\mathbf{W}\mathbf{y} \ . \qquad (5.10)$$

5.2.6 Spline smoothing as a linear operation

The expression for the data-fitting vector $\hat{\mathbf{y}}$ is

$$\hat{\mathbf{y}} = \boldsymbol{\Phi}(\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi} + \lambda \mathbf{R})^{-1} \boldsymbol{\Phi}'\mathbf{W}\mathbf{y} = \mathbf{S}_{\phi,\lambda}\mathbf{y} \ , \qquad (5.11)$$

where the order n symmetric "hat" matrix is

$$\mathbf{S}_{\phi,\lambda} = \boldsymbol{\Phi}(\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi} + \lambda \mathbf{R})^{-1} \boldsymbol{\Phi}'\mathbf{W} \ . \qquad (5.12)$$

Comparing this expression with (5.6) shows us that the only change is the addition of $\lambda\mathbf{R}$ to the cross-product matrix $\boldsymbol{\Phi}'\mathbf{W}\boldsymbol{\Phi}$ prior to its inversion, and that the two operators become identical when $\lambda = 0$. The more general operator (5.12) can be called a *sub-projection* operator because, unlike the projection operator, the sub-projection does not satisfy the *idempotency* relation, since

$$\mathbf{S}_{\phi,\lambda}\mathbf{S}_{\phi,\lambda} \neq \mathbf{S}_{\phi,\lambda}.$$

In plain language, this says that the spline smooth of a spline smooth is even smoother.

Expressions of the form (5.10) occur often in statistics where linear models are used. For example, in multilevel or random coefficient models, a similar expression arises when information about within-level regression coefficients is borrowed across levels. In Bayesian versions of multiple regression, the variance-covariance matrix $\mathbf{\Sigma}_0$ of the prior density for the regression coefficient matrix shows up where we have $\mathbf{R}$. Indeed, we can think of the roughness penalty as analogous to the logarithm of a prior density, just as the error sum of squares term is, except for a scale factor, the logarithm of a likelihood. An early example of regularization, *ridge regression*, also used this operator.

Computing the matrix $\mathbf{R}$ will generally require approximating the integral in (5.8) by a numerical quadrature scheme, although exact expressions are possible where both B-spline and Fourier bases are involved. In fact, it is seldom necessary to have very high accuracy in the approximation. An illustration of this point is that replacing $\mathbf{R}$ by a matrix of mth order difference operators applied to the coefficients themselves appears to work very well as a smoothing technique for equally spaced sampling points (Eilers and Marx, 1996).

It is also useful to plot the *linear filter* defined by the smoothing process for estimating acceleration. Let $\mathbf{\Phi}^{(2)}$ contain the values of the second derivatives of the basis functions evaluated at the sampling points, that is, $D^2\phi_k(t_j)$, and let $\hat{\mathbf{y}}^{(2)}$ be the vector of acceleration estimates at the sampling points. Then

$$\hat{\mathbf{y}}^{(2)} = \mathbf{\Phi}^{(2)}(\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi} + \lambda\mathbf{R})^{-1}\mathbf{\Phi}'\mathbf{W}\mathbf{y} = \mathbf{S}^{(2)}_{\phi,\lambda}\mathbf{y} \ ,$$

where $\mathbf{S}^{(2)}_{\phi,\lambda}$ is the matrix mapping the data vector into the vector of acceleration estimates. Rows in this matrix corresponding to acceleration estimates at ages one, ten and eighteen years are displayed in Figure 5.2. Notice that the acceleration estimate at

- age one requires some weighting of data all the way up to eight years,
- age ten, in the middle of the pubertal growth spurt, uses data from ages seven to thirteen, and
- age eighteen uses data back to age thirteen.

If the widths of these age ranges seems surprising, recall, firstly, that acceleration is intrinsically much harder to estimate than height; and, secondly, that the sparse sampling of function values forces us to borrow information from widely dispersed sampling points. Put another way, acceleration at age ten is a composite of a peak, a valley, and a peak, and uses about twelve data points, which works out to four per feature, and this in turn is close to the minimum possible of three per feature.

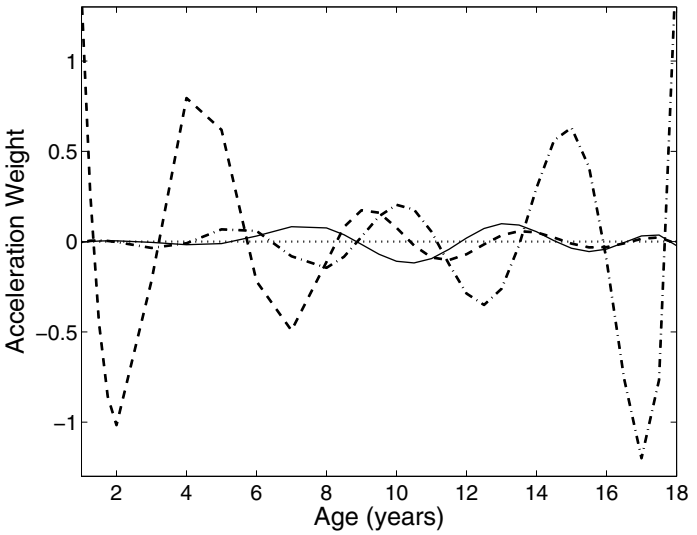

Figure 5.2. The solid curve indicates the weights placed on observations in the growth data for estimating acceleration at age ten. The dashed line corresponds to weights for height acceleration at age one, and the dashed-dotted line for age eighteen.

Another useful application of $\mathbf{S}_{\phi,\lambda}$ is in computing a *degrees of freedom* value for a spline smooth,

$$df(\lambda) = \operatorname{trace} \mathbf{S}_{\phi,\lambda} \ . \tag{5.13}$$

Hastie and Tibshirani (1990) discuss this and other ways of assessing the degrees of freedom of a smoothing procedure and, more generally, any estimation procedure that maps the data vector linearly to a parameter vector. Zhang (2003) offers a more in-depth update of this issue.

5.2.7 Spline smoothing as an augmented least squares problem

Expression (5.9) can be interpreted as arising from an *augmented least squares problem.* First, since $\mathbf{R}$ is a positive semidefinite matrix because of its cross-product structure, we can express it as

$$\mathbf{R} = \mathbf{L}'\mathbf{L}$$

by applying, among other possibilities, the Choleski decomposition. Now let

$$\tilde{\mathbf{y}} = \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix} \ ,$$

where the zero vector is of the same length as $\mathbf{c}$. We can match this augmented response vector by the augmented design matrix

$$\tilde{\mathbf{\Phi}} = \begin{bmatrix} \mathbf{\Phi} \\ \sqrt{\lambda}\mathbf{L} \end{bmatrix} .$$

Finally, we augment the weight matrix $\mathbf{W}$ with the identity matrix $\mathbf{I}$ on the diagonal and zeros elsewhere to get the augmented weight matrix $\tilde{\mathbf{W}}$.

Now we can express coefficient vector $\mathbf{c}$ using the roughness penalty as the solution to the weighted least squares problem

$$\mathtt{SSE}(\tilde{\mathbf{y}}|\mathbf{c}) = (\tilde{\mathbf{y}} - \tilde{\mathbf{\Phi}}\mathbf{c})'\tilde{\mathbf{W}}(\tilde{\mathbf{y}} - \tilde{\mathbf{\Phi}}\mathbf{c}) \ . \tag{5.14}$$

This version of the roughness penalty problem makes clear that a roughness penalized least squares is a regular least squares where the data $\mathbf{y}$ are augmented by a vector of zeros, and the zeros are fit using the augmented portion of the design matrix $\sqrt{\lambda}\mathbf{L}$. Moreover, using the QR decomposition to minimize (5.14) rather than using (5.10) directly is preferable from the standpoint of rounding error in computing $\hat{\mathbf{c}}$.

5.2.8 Estimating derivatives by spline smoothing

Many functional data analyses call for the estimation of derivatives, either because these are of direct interest, or because they play a role in some other part of the analysis. The penalty (5.1) may not be suitable, since it controls curvature in x itself, and therefore only slope in the derivative Dx. It does not require the second derivative D^2x even to be continuous, let alone smooth in any sense.

If the derivative of order m is the highest required, one should actually penalize the derivatives of order $m + 2$ in order to control the curvature of the highest order derivative. For example, the estimate of acceleration is better if we use

$$\mathtt{PEN}_4(x) = \int [D^4x(s)]^2 ds = \|D^4x\|^2 \tag{5.15}$$

in (5.3) since this controls the curvature in D^2x.

We can use relation (5.13), for example, to compare the acceleration estimates by least squares and roughness penalized smoothing from the single simulated observation in Figure 4.3. By solving for the value of λ that produces a value of *df* of 12, we obtain $\lambda = 0.06$. Smoothing the data with this observation produces the acceleration estimate shown as a heavy line in Figure 5.3. This estimate does much better at the boundaries than the least squares estimate, which is also shown. Over the interior of the interval, however, the two estimates are rather similar, although the spline smooth does a slightly better job on the pubertal growth spurt.

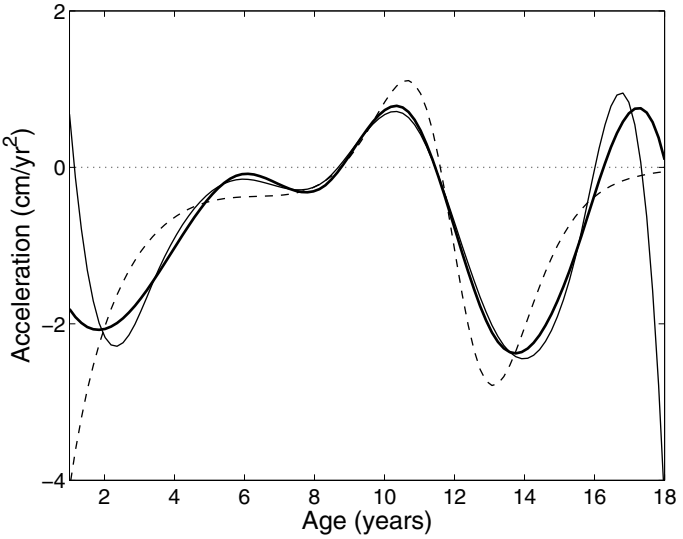

Figure 5.3. The heavy solid curve is the estimated growth acceleration for a single set of simulated data shown in Figure 4.3 computed by roughness penalized spline smoothing. The lighter solid line is for least squares smoothing, and the dashed curve is the errorless true curve.

5.3 Some extensions

The spline smoothing procedure given above can be extended in many ways, and many of these extensions are of great importance in applications. We summarize fairly briefly a number of these in this section.

5.3.1 Roughness penalties with fewer basis functions

In applications such as the study of handwriting and other forms of movement, we may use motion capture equipment that produce observations hundreds or thousands of times per second. Even the nondurable goods manufacturing index involves nearly a thousand sampling points. In these situations, placing a knot at every sampling point may imply a prohibitive amount of computation. Moreover, rounding errors may accumulate in the calculations to the point where the results are seriously inaccurate. See Section 5.4.3 below for more comments on this problem.

In these situations involving very large numbers of sampling points, it may entirely reasonable to use a lower-dimensional B-spline basis defined by some appropriate more limited knot sequence τ, provided that there remains sufficient flexibility to capture the features of interest. For example, handwriting data in Ramsay (2000) involved sampling pen position 400 times per second. The strokes making up the characters being produced

each lasted around 120 milliseconds, and thus included about 50 argument values, but was found that only about ten equally-spaced knots per stroke was more than sufficient to capture the shape of any stroke as well as three of its derivatives.

As a further example, 34 years of daily weather measurements represents about 12,500 observations, and it is a heavy chore to use that many basis functions. Instead, a system of 500 spline basis functions was considered sufficient in Ramsay and Silverman (2002) to capture all the variation of interest, and a roughness penalty was then used with this system to impose further smoothness on the result.

None of the mathematics outlined above changes when we use fewer knots than sampling points, and yet roughness penalization can remain an effective way to ensure a smooth fit and stable derivative estimates.

5.3.2 More general measures of data fit

There are aspects of the roughness penalty method that are really useful in our treatment of functional data analysis. For example, instead of quantifying fit to the data by the residual sum of squares, we can penalize *any* criterion of fit by a roughness penalty. For instance, we might have a model for the observed y_j for which the log likelihood of x can be written down. Subtracting $\lambda \times \texttt{PEN}_2(x)$ from the log likelihood and then finding the maximum allows smoothing to be introduced in a wide range of statistical problems, not merely those in which error is appropriately measured by a residual sum of squares. These extensions of the roughness penalty method are a major theme of Green and Silverman (1994).

In the functional data analysis context, we adopt this philosophy in considering functional versions of several multivariate techniques. The function estimated by these methods is expressed as the solution of a maximization (or minimization) problem based on the given data. For example, principal components are chosen to have maximum possible variance subject to certain constraints. By penalizing this variance using a roughness penalty term appropriately, the original aim of the analysis can be traded off against the need to control the roughness of the estimate. There are different ways of incorporating the roughness penalty according to the context, but the overall idea remains the same: Penalize whatever is the appropriate measure of goodness-of-fit to the data for the problem under consideration.

5.3.3 More general roughness penalties

The second extension of the roughness penalty method uses measures of roughness other than $\|D^2x\|^2$. We have already seen one reason for this in Section 5.2.8, where the estimation of derivatives of x was considered. However, even if the function itself is of primary interest, there are two related reasons for considering more general roughness penalties. On the

one hand, we may wish that the class of functions with zero roughness were wider than, or otherwise different from, those that are of the form $a + bt$. On the other hand, we may have in mind that, locally at least, curves x should ideally satisfy a particular differential equation, and we may wish to penalize departure from this.

For example, if we are analyzing periodic data, it would be more natural to use the *harmonic acceleration* operator

$$Lx = D^3x + \omega^2 Dx \tag{5.16}$$

since zero roughness implies that x is of the form

$$x(t) = c_1 + c_2 \sin \omega t + c_3 \cos \omega t,$$

where ω is the period.

We can achieve both of these goals by replacing the second derivative operator D^2 with a more general linear differential operator L, defined as

$$Lx = w_0 x + w_1 Dx + \ldots + w_{m-1} D^{m-1} x + D^m x,$$

where the weights w_j may be either constants or functions $w_j(t)$. Then we can define

$$\texttt{PEN}_L(x) = \int [(Lx)^2](t)\, dt = \|Lx\|^2, \tag{5.17}$$

the integral of the square of Lx.

As an alternative to pre-specifying the differential operator, we can use observed functional data to *estimate* the operator L. These ideas are developed further in Chapters 19 and 21.

5.3.4 Computing the roughness penalty matrix

The roughness penalty matrix $\mathbf{R}$ defined in (5.8) is composed of the integrals of products of a derivative D^m of basis functions. For B-spline bases, Fourier bases, and most of the basis systems that we are likely to work with in practice, these integrals can be computed analytically. In the B-spline case, however, the details (see de Boor, 2002) are intricate, and few users of FDA will want to write programming code for this problem. There are functions in the MATLAB®, R and S-PLUS languages that can do this work for you.

When more general roughness penalties are involved of the kind defined in (5.17) above, it will be necessarily to resort to numerical approximation of the integrals in (5.8) for matrix $\mathbf{R}$. There are two main strategies in this case.

The safer approach is to use a numerical method that iteratively improves its estimate of an integral until a test for its accuracy is satisfied. A classic approach is to use a simple method such as the trapezoidal rule and to double the number of points at which the integrand is evaluated

until an estimate of the integral is judged to have converged. We have had good experience with Romberg integration, also called Richardson's extrapolation, and have used variants of the algorithm described in Press et al. (1999). However, there are more modern methods that may well perform even better.

However, these adaptive methods can be too slow for applications where $\mathbf{R}$ must be evaluated many times during the course of a calculation. In this case, a lower accuracy non-iterative approach that is still considered to be sufficiently accurate may be preferable. For example, the integrals in (5.8) can be converted to matrix products using a fine mesh of values of t and a numerical quadrature method such as Simpson's Rule (Stoer and Bulirsch, 2002). As a rough guideline, we have found that about 21 evaluation points per interval when working with B-spline basis functions gives a level of accuracy that has sufficed for our purposes.

If multiple knots at the same location are used in order to allow for discontinuity in a derivative or function value, be careful not to evaluate the discontinuous quantity at the function value. Aside from the fact that the value is not defined mathematically, available software for evaluating spline basis functions can fail to warn you that you did something wrong, and cheerfully return a function value of large and unpredictable size, which will play havoc with your integral approximation. The better procedure is to carry out the integration piecewise over each interval, and integrate only up to a t-value separated from the knot location by a small constant.

5.4 Choosing the smoothing parameter

When we fit data using a roughness penalty instead of least squares, we switch from defining the smooth in terms of degrees of freedom K to defining the smooth in terms of the smoothing parameter λ. Nevertheless, strategies for selecting λ are rather similar to those that we used in Chapter 4 in that we use a "discounted" measure of fit that compensates for the degrees of freedom in the data used up by the fit.

5.4.1 Some limits imposed by computational issues

Although from a mathematical perspective we can contemplate any positive values of λ, the realities of floating point computation actually impose some severe limits. These limits are due to the need to solve a system of linear equations with the coefficient matrix

$$\mathbf{M}(\lambda) = \mathbf{\Phi}'\mathbf{W}\mathbf{\Phi} + \lambda\mathbf{R},$$

where $\mathbf{R}$ is defined in (5.8). The two matrices $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$ and $\mathbf{R}$ can have elements of radically different sizes. In particular, the size of

$$\|\mathbf{R}\| = \sqrt{\sum_k \sum_\ell r_{k\ell}^2}$$

increases by roughly an order of magnitude for each increase in the order m of derivative that is used to define the roughness penalty.

Now $\mathbf{R}$ itself has rank $K-m$, and so cannot itself be useful as a coefficient matrix. This implies that we cannot have $\lambda\mathbf{R}$ so large as to overwhelm $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$; otherwise, attempting to invert $\mathbf{M}(\lambda)$ will either produce an error message or, worse, a result that is so full of rounding error as to lead to seriously wrong results further on down the line. A rough rule of thumb is that the size of $\lambda\mathbf{R}$ should not be more than 10^{10} times the size of $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$.

Consider the handwriting data, for example. There are 1401 sampling points evenly spaced between 0 and 2.3 seconds. We will need to estimate the third derivative of the X and Y coordinates of pen position in Chapter 19, and consequently will need to penalize the size of the derivative of order $m = 5$. The minimal order of B-spline that will serve to define an integrable fifth derivative is 7. If we choose to use smoothing splines with a knot at each sampling value, this implies 1406 basis functions defining matrix $\mathbf{\Phi}$. The size $\|\mathbf{R}\|$ of $\mathbf{R}$ in this context is about 2×10^{31}! By contrast, $\|\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}\| \approx 20$. Hence, by our rule of thumb, we will be in trouble if $\lambda > 10^{-20}$ or so.

This illustrates the importance of some preliminary explorations along these lines before plunging into functional data analysis, and especially when high orders of derivatives are involved. In any case, the cure is simple; as we indicated in Section 3.7, these problems arise because the unit of measurement, 2.3 seconds, for t is far larger than the length of the interval over which a spline basis function is nonzero. Measuring time in milliseconds removes the problem.

On the lower limit side, we clearly cannot always use $\lambda = 0$; in this example, there are more basis functions than data points and consequently $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$ would not be invertible. Again, a rule of thumb can be proposed: Choose λ at least large enough to ensure that the size of $\lambda\mathbf{R}$ is at least with ten orders of magnitude of the size of $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$.

Now we turn to two strategies for choosing smoothing parameter somewhere between these broad limits.

These difficulties are actually a result of the way the penalized least squares criterion is defined in almost all the statistical literature. The application of the method of *dimensional analysis* used routinely in the physical sciences can be helpful here. The basic idea is that two quantities that are added should have the same units of measurement.

Now the error sum of squares $\|\mathbf{y} - \hat{\mathbf{y}}\|^2$ has the unit of measurement of x squared. In the handwriting data, this would be squared meters, for

example. We should probably also divide this criterion by n to allow for the number of sampling points.

Smoothing parameter λ can be made dimensionless by using its logarithm, which is consistent with the idea that it will be a positive quantity. Thus, we should multiply a roughness penalty such as $\texttt{PEN}_m(x)$ by 10^ν where $\nu = \log_{10}\lambda$. In fact, we basically do this already since we tend to vary λ by multiplying it by a fixed factor.

Finally, the units of $D^m x$ are those of x itself divided by the time unit, that we can indicate by τ, taken to the power m. This suggests that $\texttt{PEN}_m(x)$ should be multiplied by T^{2m}, where T is the length of time in τ units over which the integration takes places. This will cancel out the role of the time unit in the integrand. We might divide the integral, on the other hand, by T itself to allow for the summation over time that the integral represents. Putting this all together, it would be better to redefine the penalized least squares criterion as

$$\texttt{PENSSE}(x) = \frac{1}{n}\|\mathbf{y} - \hat{\mathbf{y}}\|^2 + 10^\nu T^{2m-1}\texttt{PEN}_m(x). \tag{5.18}$$

For the handwriting data, for example, if we use the time unit $\tau = 1$ second, so that the interval of integration is 2.3 seconds, along with $m = 5$ to control the curvature of the third derivative, then $T^9 \approx 1800$, but if we opt for milliseconds as the time unit, then $T^9 \approx 1.8 \times 10^{12}$. Now, of course, the fifth derivative takes on huge values in the time scale of seconds, but comparatively mild values on the time scale of milliseconds so that, finally, we will wind up using the same value of ν in either time scale.

5.4.2 The cross-validation or CV method

The basic idea behind *cross-validation* is to set part of the data to one side, calling it a *validation sample*, and fit the model to the balance of the data, called the *training sample*. In that way, we see how well the model fits data that were not used to estimate the model, thus avoiding the somewhat incestuous procedure of using the data to both fit the model and assess fit.

A versatile technique for choosing a smoothing parameter involves taking this notion to the extreme situation where we leave only one observation out as the validation sample, fitting the data to the rest, and then estimating the fitted value for the left out data value. If this procedure is repeated for each observation in turn, and the resulting error sum of squares summed over all values, the result is the *cross-validated* error sum of squares. We compute this criterion over a range of values of λ, and choose that value that yields its minimum.

Cross-validation can be used in a wide range of situations, and in effect rests only on the assumption that observations are relatively independent of one another. However, the method has two problems. First, it is usually computationally intensive, and not the sort of thing that would be feasible

for sample sizes in the thousands. However, there are specific situations in which some computational tricks can be used to reduce the computational burden. The second problem is that minimizing CV can lead to under-smoothing the data because the method tends too often to favor fitting noisy or high-frequency types of variation that we would prefer to ignore.

5.4.3 The generalized cross-validation or GCV method

A measure that is popular in the spline smoothing literature is the *generalized cross-validation* measure GCV developed by Craven and Wahba (1979). It was originally developed as a simpler version of the cross-validation procedure that avoided the need to re-smooth n times. But it also has been found to be rather more reliable than cross-validation in the sense of having less of a tendency to under-smooth. The criterion is usually expressed as

$$\texttt{GCV}(\lambda) = \frac{n^{-1}\,\texttt{SSE}}{[n^{-1}\text{trace}\,(\mathbf{I} - \mathbf{S}_{\phi,\lambda})]^2}\ ,$$

where df is the equivalent degrees of freedom measure (5.13) and $\mathbf{S}_\lambda$ is the smoothing operator defined in (5.12). But it can be more revealing to use the equivalent expression

$$\texttt{GCV}(\lambda) = \big(\frac{n}{n - df(\lambda)}\big)\big(\frac{\texttt{SSE}}{n - df(\lambda)}\big)\ . \tag{5.19}$$

Notice that this is a twice-discounted mean squared error measure. The right factor is the unbiased estimate of error variance σ^2 familiar in regression analysis, and thus represents some discounting by subtracting $df(\lambda)$ from n. The left factor further discounts this estimate by multiplying by $n/(n - df(\lambda))$.

As a practical matter, C. Gu (2002) reports that the remaining tendency for GCV to yield under-smoothing can be further reduced by multiplying df by factors such as 1.2 or 1.4 in (5.19). This is a third level of discounting, in effect. Apparently the additional discounting does not seriously increase the odds of over-smoothing the data.

The minimization of GCV with respect to λ will inevitably involve trying a large number of values of λ, whether grid-search or a numerical optimization algorithm is used. The computation of $\texttt{GCV}(\lambda)$ can be greatly speeded up by performing a preliminary generalized eigenanalysis. Criterion GCV can be expressed in terms of the n by N data matrix $\mathbf{Y}$, the $n \times K$ matrix $\mathbf{\Phi}$ of basis function values and the order K penalty matrix $\mathbf{R}$ as follows:

$$\texttt{GCV}(\lambda) = \frac{n\,\text{trace}\{\mathbf{Y}'[\mathbf{I} - \mathbf{S}_{\phi,\lambda}]^{-2}\mathbf{Y}\}}{\{\text{trace}[\mathbf{I} - \mathbf{S}_{\phi,\lambda}]\}^2}\ ,$$

where the "hat" matrix $\mathbf{S}_{\phi,\lambda}$ has the expression

$$\mathbf{S}_{\phi,\lambda} = \mathbf{\Phi}\mathbf{M}(\lambda)^{-1}\mathbf{\Phi}'\mathbf{W}$$

and where, in turn

$$\mathbf{M}(\lambda) = \mathbf{\Phi}'\mathbf{W}\mathbf{\Phi} + \lambda\mathbf{R}.$$

Note that we have dropped the weight matrix $\mathbf{W}$ from these expressions to keep the notation a little simpler.

We actually don't need to invert $\mathbf{M}(\lambda)$ each time we change λ, but we do need to solve a linear system of equations for which it is the coefficient matrix, and this is that we want to avoid. This can be achieved if we first solve the generalized eigenvalue problem

$$\mathbf{R}\mathbf{V} = \mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}\mathbf{V}\mathbf{D},$$

where $\mathbf{D}$ is the matrix of eigenvalues of $\mathbf{R}$ in the metric defined by $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$ and $\mathbf{V}$, the columns of which are the corresponding eigenvectors of $\mathbf{R}$, satisfy the orthogonality condition

$$\mathbf{V}'\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}\mathbf{V} = \mathbf{I}.$$

Note that the generalized eigenvalue problem has a solution only if $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$ is nonsingular. This will not be the case if knots are placed at every data point. However, a trick recommended by de Boor (2002) is to drop enough knots next to the boundary to make the number of basis functions equal to the number of sampling points. For example, if we are working with cubic smoothing splines of order four and we have 101 sampling points, then this implies 103 basis functions. But if we drop the knots associated with sampling points 2 and 100, the number of basis functions drops to 101, and $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$ will be nonsingular, at least if sampling points are reasonably well-spaced. It is, needless to say, always a good idea to check $\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}$ for singularity.

We now express, for any new value of λ, the required inverse very efficiently as

$$\mathbf{M}(\lambda)^{-1} = \mathbf{V}(\mathbf{I} + \lambda\mathbf{D})^{-1}\mathbf{V}',$$

since the matrix now being inverted is diagonal. Moreover, taking the derivative of $\texttt{GCV}(\lambda)$ involves calculating the matrix

$$\mathbf{M}(\lambda)^{-1}\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi}\mathbf{M}(\lambda)^{-1} = \mathbf{V}(\mathbf{I} + \lambda\mathbf{D})^{-2}\mathbf{V}'$$

so that providing a derivative value to a numerical optimization algorithm is also computationally efficient and likely to decrease the number of evaluations of $\texttt{GCV}(\lambda)$ substantially.

Gu (2002) offers a detailed and up to date discussion of theoretical and computational issues associated with the $\texttt{CV}(\lambda)$, $\texttt{GCV}(\lambda)$ and other methods for choosing λ.

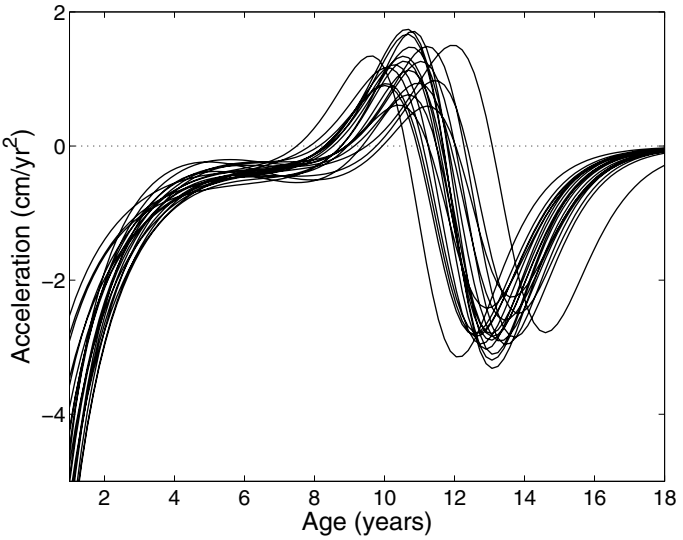

Figure 5.4. A sample of twenty height acceleration curves for females generated using the Jolicoeur model.

5.4.4 Spline smoothing the simulated growth data

We illustrate here some of the points made in this chapter by the analysis of 1000 simulated records for females using the Jolicoeur model described in Section 4.3. A random sample of twenty acceleration curves from this model are shown in Figure 5.4.

Figure 5.5 shows the variation of the generalized cross-validation statistic GCV over a range of $\log_{10}(\lambda)$ values in its top panel. We see that the minimum GCV is attained at $\lambda = -0.1$. At this smoothing level, the degrees of freedom measure has the value of 11.4, which is not far from the number twelve of basis functions that we used in least squares smoothing.

In the lower panel, we see the square root of the mean squared error ($RMSE$) of the acceleration curve values at ages eight, before puberty; twelve, mid-puberty for the average girl; and sixteen, post-puberty for most girls. These curves do not bottom out at the same value as the GCV curve, but they come close to doing so. It is not surprising that the curve for age twelve favors a lower value of λ; the curvature of the acceleration function is much sharper for the average girl at mid-puberty. The more stable curves typical for most girls at ages eight and sixteen favor higher values of λ. Nevertheless, the GCV-favored value gives nearly optimal values for RMSE.

Figure 5.6 indicates the variation in RMSE, bias, and sampling standard error over age for the smoothing level minimizing GCV. We see that the curve estimates are of limited value for ages less than three years or more than sixteen years. But they aren't bad at all in between these extremes,

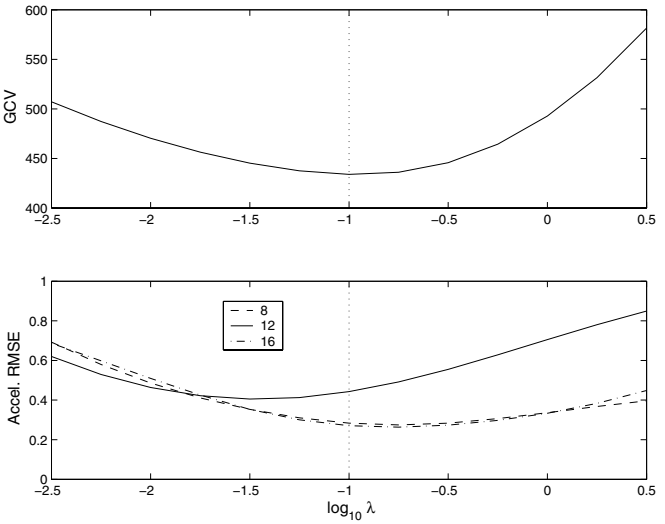

Figure 5.5. The top panel displays the relation between the GCV statistic and smoothing level for 1000 simulated female records. The bottom panel displays the root-mean-squared error in acceleration estimates at the selected ages of 8, 12 and 16 years of age.

and the bias in particular is small. Of course, we could do better if we had sampled height more often. It is also not surprising that sampling error is higher during the pubertal growth spurt when curvature is high.

Perhaps the main conclusion to be drawn here is that the spline smoothing method does a good job in this context, and especially given that there are only 31 observations in each record. Choosing λ using the GCV criterion gets us close to the best answer, on the average.

5.5 Confidence intervals for function values and functional probes

We now want to see how to compute confidence limits on some useful quantities that depend on an estimated function x that has, in turn, been computed by the smoothing with a roughness penalty a vector of discrete data $\mathbf{y}$.

For example, how precisely is the function value at t, $x(t)$, determined by our sample of data $\mathbf{y}$? Or, what sampling standard deviation can we expect if we re-sample the data over and over again, estimating $x(t)$ anew with each sample? Can we construct a pair of *confidence limits* such that the probability that the true value of $x(t)$ lies within these limits is a specified

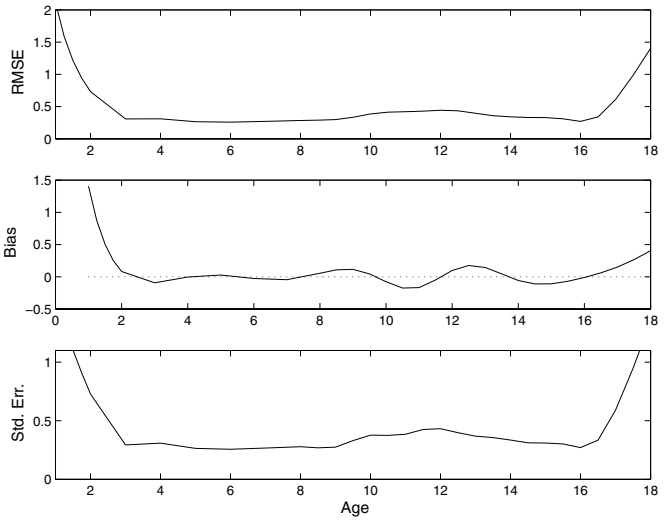

Figure 5.6. The root-mean-squared error for the GCV-optimal smoothing level as a function of age is shown in the top panel, and the corresponding values of bias and sampling standard error are shown in the middle and bottom panels, respectively.

value, such as 0.95? Displaying functions or their derivatives with point-wise confidence limits is a useful way of conveying how much information there is in the data used to estimate these functions. See Figure 5.7 below for an example.

However, do be aware of the distinction between these point-wise limits, which tell us only the precision at a fixed location, and global confidence limits, which would tell us a region of confidence for the entire function. Constructing an upper and a lower curve such that the probability that the *entire* true curve lies between these functional limits can be achieved by computationally intensive methods such as bootstrapping (Efron and Tibshirani, 1993).

5.5.1 Linear functional probes

More generally, we may wish to examine quantities of the form

$$\rho_\xi(x) = \int \xi(t)x(t)\,dt\ . \tag{5.20}$$

We use the term *functional probe* for the quantity $\rho_\xi(x)$ and *linear probe function* for the weighting function ξ that defines it. The probe function, in turn, is chosen so as to highlight some interesting feature, such as a

peak, valley, or difference between function values over two non-overlapping regions.

A probe is a generalization of the notion of a *contrast* in analysis of variance, used there to probe a set of treatment effects for specific types of variation. However, there is no need for the values of ξ to integrate to zero. If we have multiple probes, it may be helpful for pairs of probe functions to be orthogonal, but this is not essential.

For example, to highlight the behavior of x over an interval, an appropriate probe function ξ might be the box function, which takes the value 1 within the interval and 0 elsewhere. Or, to highlight the difference between x in two intervals A and B of equal length, one could use a probe function taking the value 1 on A, -1 on B, and 0 elsewhere. In cases like these, we will want to compute the sampling standard deviation of the scalar ρ_ξ in order to decide whether it differs significantly from some reference value like zero.

Functional probes ρ_ξ of this nature include the simpler situation of $x(t)$ as a special case, since $x(t)$ can be obtained by choosing ξ to be nonnegative and concentrating its nonzero values arbitrarily near t while preserving unit area under the its curve. We can denote such a probe by

$$\rho_t(x) = x(t) \ , \tag{5.21}$$

and it is called the *evaluation map* because it maps function x into its value $x(t)$ at t. Probes of this nature are taken up in detail in Section 20.3.

Probe ρ_ξ is a linear function of the estimated smoothing function x in the sense if that we multiply two such functions, x_1 and x_2 by the constants a and b, respectively, then

$$\rho_\xi(ax_1 + bx_2) = a\rho_\xi(x_1) + b\rho_\xi(x_2).$$

This linearity implies that there is a linear transformation of the coefficient vector $\mathbf{c}$ that defines x that yields the value $\rho_\xi(x)$. At the same time, we already worked out in this chapter the linear transformation that takes or *maps* the data vector $\mathbf{y}$ to the coefficient vector $\mathbf{c}$.

5.5.2 Two linear mappings defining a probe value

In order to study the sampling behavior of ρ_ξ, we need to compute these two linear mappings plus their composite. They are given names and described as follows:

1. Mapping `y2cMap`, which converts the raw data vector $\mathbf{y}$ to the coefficient vector $\mathbf{c}$ for the basis function expansion of x. If $\mathbf{y}$ and $\mathbf{c}$ are lengths n and K, respectively, the mapping is a K by n matrix $\mathbf{S}$ such that

$$\mathbf{c} = \mathbf{S}\mathbf{y} \ .$$

2. Mapping `c2rMap`, which converts the coefficient vector $\mathbf{c}$ to the scalar quantity $\rho_\xi(x)$. This mapping is a 1 by K row vector $\mathbf{L}$ such that

$$\rho_\xi(x) = \mathbf{Lc} \ .$$

3. The composite mapping called `y2rMap` defined by

$$\texttt{y2rMap} = \rho_\xi(x) = \texttt{c2rMap} \circ \texttt{y2cMap},$$

which takes data vector $\mathbf{y}$ directly to the probe value and is the 1 by n row vector $\mathbf{LS}$ that yields

$$\rho_\xi(x) = \mathbf{LSy} \ .$$

As an illustration, consider a conventional linear regression model with design matrix $\mathbf{Z}$

$$\mathbf{y} = \mathbf{Zc} + \mathbf{e},$$

where the regression coefficient vector $\mathbf{c}$ is estimated by ordinary least squares. Then, since $\mathbf{c} = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{y}$, the matrix corresponding to `y2cMap` is $\mathbf{S} = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'$. Now suppose that for some reason we want to estimate the difference between the first and second regression coefficients, possibly because we conjecture that they may be equal in the population. Then the probe function ξ is equivalent to the probe vector $\mathbf{L} = (1, -1, 0, \ldots)$, and this is the row vector corresponding to mapping `c2rMap`. Finally, the composite mapping `y2rMap` taking $\mathbf{y}$ directly into the value of this difference is simply the row vector $\mathbf{L}(\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'$.

Now the random behavior of the estimator of whatever we choose to estimate is ultimately tied to the random behavior of the data vector $\mathbf{y}$. Let us indicate the order n variance-covariance matrix of $\mathbf{y}$ as $\texttt{Var}(y) = \mathbf{\Sigma}_e$, as we did in Sections 4.6.1 and 4.6.2. Recall that we are operating in this chapter with the model

$$\mathbf{y} = x(\mathbf{t}) + \boldsymbol{\epsilon} \ ,$$

where $x(\mathbf{t})$ here means the n-vector of values of x at the n argument values t_j. In this model $x(\mathbf{t})$ is regarded as fixed, and as a consequence $\mathbf{\Sigma}_e = \texttt{Var}(\boldsymbol{\epsilon})$.

5.5.3 Computing confidence limits for function values

Now let's express these mappings in the context of estimating confidence limits specifically for a function value $x(t)$. Let n by K matrix $\mathbf{\Phi}$ contain the values $\phi_k(t_j)$, and let the matrices $\mathbf{R}$ and $\mathbf{W}$ be defined as before. Then the matrix corresponding to `y2cMap` is

$$\mathbf{S} = (\mathbf{\Phi}'\mathbf{W}\mathbf{\Phi} + \lambda\mathbf{R})^{-1}\mathbf{\Phi}'\mathbf{W}$$

for smoothing parameter λ.

Suppose that we are interested in the sampling behavior of the function value $\rho_t(x) = x(t)$. We know that

$$x(t) = \boldsymbol{\phi}'(t)\mathbf{c} = \boldsymbol{\phi}'(t)\mathbf{S}\mathbf{y},$$

and from this we can see that the 1 by K matrix $\mathbf{L}$ corresponding to `c2rMap` is simply $\boldsymbol{\phi}'$, the row vector resulting from evaluating each of the basis functions at t. And of course the composite mapping `y2rMap` corresponds to the matrix $\mathbf{LS}$. Consequently, using the expression for the variance of a linear transform of a random vector, we have that

$$\mathtt{Var}[\hat{x}(t)] = \mathbf{LS}\boldsymbol{\Sigma}_e\mathbf{S}'\mathbf{L}'. \tag{5.22}$$

The matrix $\mathbf{LS}$ used in (5.22) is also of interest in itself. Each row of this matrix indicates the profile of weights on the data used to define what is being estimated for that row. For example, if row j corresponds to the function evaluation $\rho_{t_j}(x)$ at time t_j, then a plot of the values in this row shows the entries in $\mathbf{y}$ that are used to define this estimate. A row of this matrix is often called a *linear filter* for estimating the quantity in question by engineers. See Figure 5.7 below for an example.

5.5.4 Confidence limits for growth acceleration

With this information in hand, we can gain an impression of how well the acceleration function can be estimated using the results in Section 5.5. If we use spline smoothing using order six B-splines as the basis for smoothing, a smoothing parameter $\lambda = 0.1$, and weight matrix $\mathbf{W}$ a diagonal matrix containing the values of the variances of estimate as derived from Figure 4.2, then Figure 5.7 shows the acceleration curve for the Jolicoeur model based on using the mean coefficients along with point-wise 95% confidence limits. The confidence limits balloon out at the extremes because of the difficulty of estimating derivatives in these regions.

5.6 A bi-resolution analysis with smoothing splines

We now turn to a more general approach, of which spline smoothing turns out to be a special case. So far we have used basis functions in two essentially different ways. In section 4.2 of Chapter 4, we forced the function x to lie in a relatively low dimensional space, defined in terms of a suitable basis. On the other hand, in Section 4.7, we did not assume that the whole function was in the span of a particular basis, but rather we considered a local basis expansion at any given point. In this section, we allow the function to have a higher-dimensional basis expansion, but use a roughness penalty in fitting the function to the observed data.

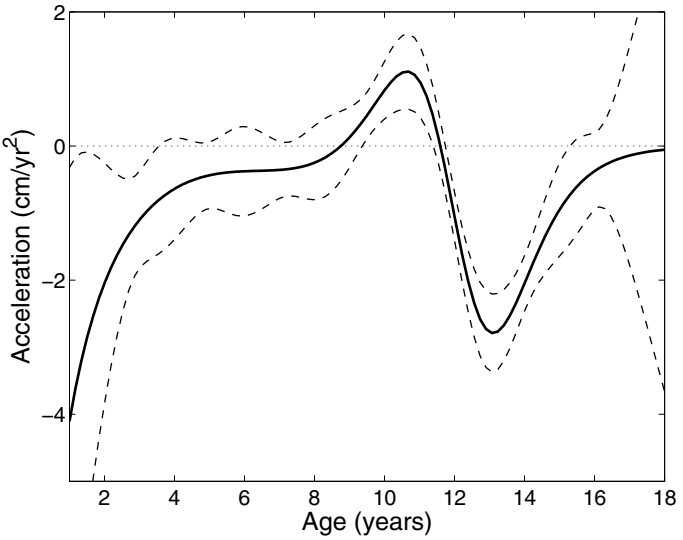

Figure 5.7. The solid curve is an acceleration curve derived from the Jolicoeur model. The dashed lines are 95% point-wise confidence limits on the curve based on a smoothing spline estimate from data having the standard error of measurement plotted in Figure 4.2.

5.6.1 Complementary bases

To develop our approach, suppose that we have two sets of basis functions, $\phi_j, j = 1, \ldots, J$ and $\psi_k, k = 1, \ldots, K$, that complement one another. Let functions ϕ_j be small in number and chosen to give reasonable account of the large-scale features of the data. The complementary basis functions ψ_k will generally be much larger in number, and are designed to catch local and other features not representable by the ϕ_j. Assume that any function x of interest can be expressed in terms of the two bases as

$$x(s) = \sum_{j=1}^{J} d_j \phi_j(s) + \sum_{k=1}^{K} c_k \psi_k(s). \tag{5.23}$$

For example, for the Canadian temperature data, the first three Fourier series functions with $\omega = \pi/6$ would be a natural choice for the ϕ_j, setting $J = 3$ and letting the $\boldsymbol{\phi}$ basis be the functions

$$1, \ \sin(\omega t), \ \cos(\omega t).$$

The appropriate choice for the ψ_k in this case would be the remaining K functions in an order $(J + K)$ Fourier series expansion. In the monthly temperature data case, they could be the remaining nine Fourier series terms needed to represent the data exactly. Usually, as in the Fourier case

above, the bases $[\phi_j]$ and $[\psi_k]$ are mutually linearly independent, and the expansion is unique, but this is not entirely essential to our method.

5.6.2 Specifying the roughness penalty

Let us now develop a roughness penalty for x so that linear combinations of the ϕ_j are in effect completely smooth, in that they contribute nothing to the roughness penalty. Then the roughness penalty must depend only on the coefficients of the ψ_k. One way of motivating this choice is by thinking of x as the sum of two parts, an "ultrasmooth" function $x_S = \sum_j d_j\phi_j$ and a function $x_R = \sum_k c_k\psi_k$. Therefore we seek a measure $\texttt{PEN}(x_R)$ of how rough, or in any other way important, we would consider the function x_R expressed solely in terms of the ψ_k. One possibility is simply to take the usual L_2 norm of x_R, defining

$$\texttt{PEN}_0(x_R) = \int x_R(s)^2\,ds = \int(\mathbf{c}'\boldsymbol{\psi})^2 = \int[\sum_{k=1}^{K} c_k\psi_k(s)]^2\,ds.$$

Another possibility is to take a certain order of derivative of the expansion prior to squaring and integrating, just as we did for the function x itself in Section 5.2. For example, we might use

$$\texttt{PEN}_2(x_R) = \int(D^2x_R)^2 = \int[\sum_{k=1}^{K} c_kD^2\psi_k(s)]^2\,ds$$

to assess the importance of x_R in terms of its total curvature, as measured by its squared second derivative, or $\texttt{PEN}_4(x_R) = \int(\mathbf{c}'D^4\boldsymbol{\psi})^2$ to assess the curvature of its second derivative. More generally, we can use any linear differential operator L, defining

$$\texttt{PEN}_L(x_R) = \int(Lx_R)^2 = \int[\sum_{k=1}^{K} c_kL\psi_k(s)]^2\,ds.$$

Of course, setting L as the identity operator or the second derivative operator yields $\texttt{PEN}_0$ and $\texttt{PEN}_2$ as special cases.

We can express these penalties in matrix terms as

$$\texttt{PEN}_L(x_R) = \mathbf{c}'\mathbf{R}\mathbf{c},$$

where the order K symmetric matrix $\mathbf{R}$ contains elements

$$\mathbf{R}_{kl} = \int L\psi_k(s)L\psi_l(s)ds.$$

If computing the integrals proves difficult, a simple numerical integration scheme, such as the trapezoidal rule applied to a fine mesh of argument values, usually suffices, and then we can also estimate derivatives numerically. Alternatively, we can specify $\mathbf{R}$ directly as any suitable symmetric

non-negative definite matrix, without explicit reference to the roughness of the function x_R.

Now we consider a general function x of the form (5.23), and simply define the roughness of x as

$$\texttt{PEN}(x) = \mathbf{c}'\mathbf{R}\mathbf{c}.$$

To express the penalized sum of squares, we need to express the residual sum of squares in terms of the coefficient vectors d and $\mathbf{c}$. Working just as in (4.1),

$$\sum_j [y_j - x(t_j)]^2 = \|\mathbf{y} - \mathbf{\Phi}\mathbf{d} - \mathbf{\Psi}\mathbf{c}\|^2,$$

where the $n \times K$ matrix $\mathbf{\Psi}$ has elements $\Psi_{ik} = \psi_k(t_j)$. We can now define the composite smoothing criterion

$$\texttt{PENSSE}_\lambda(x|y) = \|\mathbf{y} - \mathbf{\Phi}\mathbf{d} - \mathbf{\Psi}\mathbf{c}\|^2 + \lambda\mathbf{c}'\mathbf{R}\mathbf{c}. \tag{5.24}$$

We can minimize this quadratic form in $\mathbf{d}$ and $\mathbf{c}$ to find the fitted curve x in terms of its expansion (5.23) as follows. The solution for $\mathbf{d}$ for any fixed value of $\mathbf{c}$ is given by

$$\mathbf{d} = (\mathbf{\Phi}'\mathbf{\Phi})^{-1}\mathbf{\Phi}'(\mathbf{y} - \mathbf{\Psi}\mathbf{c}) \tag{5.25}$$

and, consequently,

$$\mathbf{\Phi}\mathbf{d} = \mathbf{S}_\phi(\mathbf{y} - \mathbf{\Psi}\mathbf{c}),$$

where the projection matrix $\mathbf{S}_\phi$ is

$$\mathbf{S}_\phi = \mathbf{\Phi}(\mathbf{\Phi}'\mathbf{\Phi})^{-1}\mathbf{\Phi}'.$$

In words, the ϕ basis component of the fit is the conventional basis expansion of the residual vector $\mathbf{y} - \mathbf{\Psi}\mathbf{c}$. Substitute this solution for $\mathbf{d}$ into $\texttt{PENSSE}_\lambda$ and define the complementary projection matrix $\mathbf{Q}_\phi$ by

$$\mathbf{Q}_\phi = \mathbf{I} - \mathbf{S}_\phi.$$

Recalling that because $\mathbf{Q}_\phi$ is a projection matrix, $\mathbf{Q}_\phi\mathbf{Q}_\phi = \mathbf{Q}_\phi$, we arrive at the equations

$$\begin{aligned} \hat{\mathbf{c}} &= (\mathbf{\Psi}'\mathbf{Q}_\phi\mathbf{\Psi} + \lambda\mathbf{R})^{-1}\mathbf{\Psi}'\mathbf{Q}_\phi\mathbf{y} \\ \hat{\mathbf{d}} &= (\mathbf{\Phi}'\mathbf{\Phi})^{-1}\mathbf{\Phi}'[\mathbf{I} - \mathbf{\Psi}(\mathbf{\Psi}'\mathbf{Q}_\phi\mathbf{\Psi} + \lambda\mathbf{R})^{-1}\mathbf{\Psi}']\mathbf{y}\ . \end{aligned} \tag{5.26}$$

5.6.3 Some properties of the estimates

The first term of (5.24) is identical in structure to the error sum of squares criterion $Q(\mathbf{c})$ defined in (4.1), except that both sets of basis functions are used in the expansion. The second term, however, modifies the basis function expansion problem by penalizing the roughness or size in some sense of the ψ-component of the expansion.

The size of the penalty on the ψ-component is controlled by the smoothing parameter λ. In the limit as $\lambda \to 0$, no penalty whatsoever is applied, and the estimates obtained by minimizing the criterion $\texttt{PENSSE}_\lambda$ revert to those obtained by an ordinary basis expansion in the combined basis of ϕ_j and ψ_k. At the other extreme, when $\lambda \to \infty$, the penalty is so severe that ψ-contribution to the roughness penalty is forced to zero; if $\mathbf{R}$ is strictly positive-definite, we obtain the basis function estimate corresponding to the basis $[\phi_j]$ alone. If $\mathbf{R}$ is not strictly positive-definite, then a contribution x_R from the $[\psi_k]$ basis is allowed, provided that it satisfies $Lx_R(s) = 0$ for all s.

It is instructive to study the minimizing values of the coefficient vectors $\mathbf{d}$ and $\mathbf{c}$. The smoothing matrix $\mathbf{S}$ then becomes

$$\mathbf{S}_\lambda = \mathbf{S}_\phi \mathbf{Q}_{\psi,\lambda} + \mathbf{S}_{\psi,\lambda}\mathbf{Q}_\phi,$$

where the smoothing operator $\mathbf{S}_{\psi,\lambda}$ is

$$\mathbf{S}_{\psi,\lambda} = \boldsymbol{\Psi}(\boldsymbol{\Psi}'\mathbf{Q}_\phi\boldsymbol{\Psi} + \lambda\mathbf{R})^{-1}\boldsymbol{\Psi}'. \tag{5.27}$$

and $\mathbf{Q}_{\psi,\lambda} = \mathbf{I} - \mathbf{S}_{\psi,\lambda}$. From this we can see that $\mathbf{S}_{\psi,\lambda}$ is a kind of "sub-projection"' matrix in the metric of the projection $\mathbf{Q}_\phi$ in that it has the structure of a true projection except for a perturbation of $\boldsymbol{\Psi}'\mathbf{Q}_\phi\boldsymbol{\Psi}$ by $\lambda\mathbf{R}$.

Moreover, the fit vector $\hat{\mathbf{y}}$ is now partitioned into two orthogonal parts, $\hat{\mathbf{y}} = \hat{\mathbf{y}}_0 + \hat{\mathbf{y}}_1$, where

$$\begin{aligned} \hat{\mathbf{y}}_0 &= \mathbf{S}_\phi \mathbf{Q}_{\psi,\lambda}\mathbf{y} \\ \hat{\mathbf{y}}_1 &= \mathbf{S}_{\psi,\lambda}\mathbf{Q}_\phi\mathbf{y}. \end{aligned} \tag{5.28}$$

The first "ultra-smooth" term comes from first smoothing $\mathbf{y}$ using rough basis $\boldsymbol{\psi}$, and then projecting the residual from that smooth onto the space spanned by smooth basis $\boldsymbol{\phi}$. The second "rough" term comes from first projecting $\mathbf{y}$ on to the orthogonal complement of the ϕ-space, and then applying the ψ-smoother to the result.

This elaborates the way in which the regularized basis approach provides a continuous range of choices between low-dimensional basis expansion in terms of the functions ϕ_j and a high-dimensional expansion also making use of the functions ψ_k.

5.6.4 Relationship to the roughness penalty approach

We conclude with some remarks about the connections between the regularized basis method and the method discussed in Section 5.3.3 above. Firstly, to minimize the residual sum of squares penalized by $\|Lx\|^2$, we need not specify any functions at all in the ϕ_j part of the basis, but merely ensure that $[\psi_k]$ is a suitable basis for the functions of interest. In the original spline smoothing context, with $L = D^2$, we can take the $[\psi_k]$ to be a B-spline basis with knots at the data points, and, by using suitable

methods of numerical linear algebra, we can obtain a stable $O(n)$ algorithm for spline smoothing; this is the approach of the S-PLUS function `smooth.spline`.

Secondly, if we wish to prescribe a particular ultrasmooth class $\mathcal{F}_0$, the regularized basis approach allows us to choose basis functions ϕ_j to span $\mathcal{F}_0$, and then allow $\mathbf{R}$ to be any appropriate strictly positive-definite matrix. In this way, the choice of the ultrasmooth class is decoupled from the way that roughness is measured.

5.7 Further reading and notes

We drew on the treatment of roughness penalties in Green and Silverman (1994) in preparing this chapter, but possibly the best current reference for fairly advanced readers is Gu (2002). Wahba (1990) reviews the many remarkable contributions of the author and her students to spline smoothing technology, but requires a background in functional analysis to read.

Although we have expressed roughness penalties in terms of integrated squared derivatives, many authors have used the simpler approach of summing squared first or second difference values instead. This only works if the sampling points t_j are equally spaced, but in this context, summing squared differences works well, and is discussed in Eilers and Marx (1996), and also by O'Sullivan (1986) and O'Sullivan, Yandell and Raynor (1986).

Two efforts stand out as path-breaking attempts to use derivative information in data analysis. The first of these is a series of papers on human growth data beginning with Largo et al. (1978) that focussed on the shape of the acceleration function. By careful and innovative use of smoothing techniques, spline and kernel, they were able to isolate a hitherto ignored phenomenon, the so-called mid-spurt, or hump in the acceleration curve that precedes the pubertal growth spurt and occurs at around seven to eight years in almost all children of either gender. These studies confirmed a principle that we have seen in many of our own functional data analyses: Exogenous influences and other interesting events are often much more apparent in some order of derivative than in the original curves.

On a somewhat more technical note, the thesis by Besse (1979) and his subsequent papers (Besse and Ramsay, 1986; Besse, 1980 & 1988) moved the French data analytic school into a new realm involving data that take values in spaces of functions possessing a certain number of derivatives. Besse's discussion of principal components analysis in the context of observations in Sobelev space was inspired by Dauxois and Pousse (1976), Dauxois, Pousse and Romain (1982) and the functional analytic approaches to spline smoothing by Atteia (1965). Ramsay and Dalzell (1991), who coined the term functional data analysis, extended this line of work to linear models.

6
Constrained functions

6.1 Introduction

Up to now we have only asked smoothness of our functions, but in many situations the function that we estimate must also satisfy important side conditions, such as being strictly increasing. Unfortunately, our central idea of using a basis expansion can get us into trouble here. We saw in Chapters 4 and 5 that smoothing the height data often produced curves that did not increase everywhere and consequently had negative velocities. It is, in general, difficult to force functions defined by linear expansions to satisfy constraints such as being everywhere positive, monotone, and so forth.

In this chapter we consider four constrained estimation situations: functions which must be positive, those which must be strictly monotone, those whose values are probabilities, and probability density functions, which must be both positive and integrate to one. We will in each case redefine the original problem so that the function to be estimated is unconstrained. The idea of defining a constrained function by a differential equation will be introduced. In the density estimation case, it will be necessary to use a fitting criterion other than least squares.

6.2 Fitting positive functions

Data are often collected on functions that are strictly positive. The data themselves may be zero, but these zero values indicate only that the func-

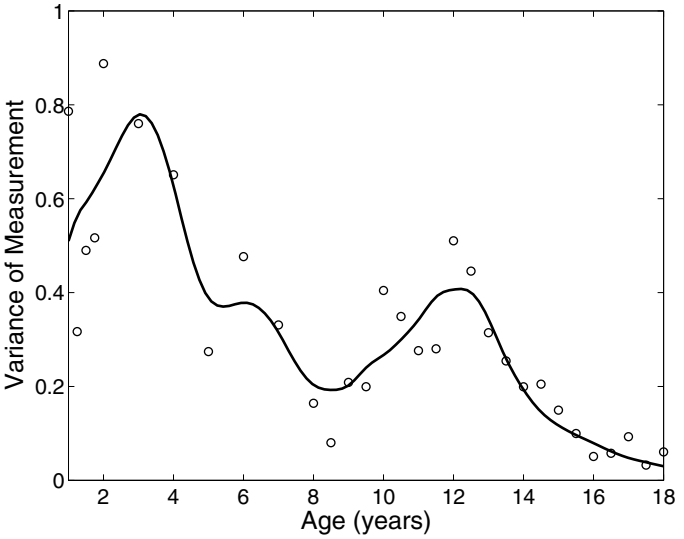

Figure 6.1. The circles indicate squared residuals averaged across the 54 girls in the Berkeley growth study for the ages of observation. The solid line is a positive smooth of these data.

tion being estimated is small at that point. For example, within any year we are apt to observe many days without any rainfall, and there can be days where no rain is recorded in thirty years. But eventually it rains, at least in Canada. Counts of errors, as another example, can easily be zero, but eventually everybody makes a mistake.

We often need to use the data to estimate variances $\sigma^2(t)$ that vary over argument t, and a zero or negative variance estimate can cause all kinds of problems. We don't want a string of zero observed variances to translate into a zero estimate.

We can estimate a nonstationary variance function for the 54 females in the Berkeley growth data as follows. First, we smooth the height values assuming a constant variance. We opted for smoothing splines with a penalty on $(D^4x)^2$ and $\lambda = 1$. The squared residuals from these fits were averaged over cases, and Figure 6.1 displays the resulting averages. The measurement error is elevated in infancy and around the pubertal growth spurt, but the error variance estimates near the boundaries are certainly too small, probably due to the over-fitting that happens in these regions due to the lack of data.

The solid line in Figure 6.1 shows the fit using a version of a smoothing spline, to be described below, that is constrained to be positive, so as to avoid any estimated value $\hat{\sigma}^2(t)$ that is zero or negative. The fit indicates that the standard error of measurement is about seven millimeters in infancy, but is more like three millimeters in later years. We then re-smoothed

the data using a diagonal weight matrix $\mathbf{W}$ containing its diagonal the reciprocals of the fitting variances $w_j = 1/\hat{\sigma^2}(t_j)$. We could go on to iterate this process by re-estimating average squared residuals, and so on.

6.2.1 A positive smoothing spline

A positive smoothing function x can always be defined as the exponential of an unconstrained function W:

$$x(t) = e^{W(t)} \ , \tag{6.1}$$

so that W is the logarithm of x. Other bases besides e for the logarithm are, of course, always appropriate; some of our clients may prefer the base 10 to aid interpretation.

Since $W(t)$ can be positive or negative and is not in any other way constrained, it is reasonable to expand W in terms of a set of basis functions,

$$W(t) = \sum_k c_k \phi_k(t) \ , \tag{6.2}$$

probably using a Fourier series for periodic data such as daily precipitation levels and B-spline expansions for non-periodic data such as the mean squared residuals for the growth data.

The roughness of a positive smoothing function x is defined as the roughness of its logarithm W, so that the roughness-penalized fitting criterion for positive smoothing, using the size of the second derivative, is

$$\texttt{PENSSE}_\lambda(W|\mathbf{y}) = [\mathbf{y} - e^{W(\mathbf{t})}]'\mathbf{W}[\mathbf{y} - e^{W(\mathbf{t})}]^2 + \lambda \int [D^2 W(t)]^2 \, dt \ . \tag{6.3}$$

A complication on the computational side is that we must now use numerical methods to minimize criterion (6.3) with respect to the coefficients c_k of the expansion. These methods iteratively decrease an initial estimate of $W(t)$ until convergence is reached. However, because the exponential transform is only mildly nonlinear, these iterative methods usually converge rapidly, even from initial estimates far away from the final value. In fact, we find that starting with $W = 0$ works just fine in most circumstances. Keep in mind, however, that if the data are mostly zero in a region, and especially at the boundaries, the values of $W(t)$ in that region will be poorly defined large negative numbers.

The positive smooth of the residuals in Figure 6.1 was obtained by using an order four B-spline expansion of $W(t)$ with a knot located at each age of observation. The fit was made smooth by using as a roughness penalty the integrated squared derivative $D^2\sigma$ multiplied by $\lambda = 0.0001$.

6.2.2 *Representing a positive function by a differential equation*

A differential equation expresses a relationship between a function and one or more of its derivatives, and is often an elegant way of describing functions with special structures.

For example, what does the notation e^{wt}, where w is here some fixed rate constant, really mean? We may say that the notation stands for a recipe for computing its value, namely the convergent infinite series

$$x(t) = \sum_{i=0}^{\infty} \frac{(wt)^i}{i} \ . \tag{6.4}$$

But a recipe is not the same as a taste, and a computer program is not the same as the mathematical concept whose value it calculates. We might prefer a definition that tells us directly about an important property of e^{wt}. Here it is:

$$Dx(t) = w(t)x(t). \tag{6.5}$$

This is easier to remember, and a positive w evokes the image of a graph that increases more and more rapidly as the function gets larger and larger, that is, an image of explosive growth. Or, if w is negative, that the slope of x goes to zero as the function value $x(t)$ goes to zero.

If $w(t)$ is a function, the solution function x for the differential equation (6.5) is

$$x(t) = C \exp[\int_{t_0}^{t} w(u)\, du] \tag{6.6}$$

for some nonzero constant C and lower limit of integration t_0. In the cleaner functional notation, $x = C \exp D^{-1} w$. If $C > 0$, then

$$x(t) = \exp[W(t)], \tag{6.7}$$

where

$$W(t) = \int_{t_0}^{t} w(u)\, du + \log C = D^{-1}w(t) + \log C.$$

In fact, our invocation of the infinite series (6.4) would not be correct in a wider functional sense; the recipe (6.4) only works when w is a constant and therefore $(D^{-1}w)(t) = wt$. Three lessons are therefore to be drawn:

- Going from scalar to functional notation can turn up some surprises.
- A differential equation can be a powerful and evocative way of defining a function.
- The solution to a differential equation is a *class of functions*, in this case corresponding to the arbtrary choice of constant C.

Perhaps the need for this little bit of mathematics will be less than apparent here. If so, ignore it, but do expect the differential equation theme to return again and again, and to become progressively more important.

6.3 Fitting strictly monotone functions

Now that we have the principle that smooth functions can result from transforming the constrained smoothing problem to one that is unconstrained, we are ready to look at the monotone smoothing of the growth data. A strictly monotone function has a strictly positive first derivative. The spline smoothing approach that has been used up to now has worked fine for ages up to about sixteen, but after that the estimated velocities have in many cases gone negative. We hope that preventing negative estimates of velocities can stabilize height, velocity and acceleration estimates at the adult end of the fitting interval.

6.3.1 Fitting the growth of a baby's tibia

Figure 6.2 displays a tough monotone smoothing problem. The data were collected by M. Hermanussen et al. (1998), who developed an instrument measuring bone lengths to within about 0.1 millimeters. They are the lengths of the tibia, the large bone in the lower leg, in a newborn baby measured daily for the first 40 days of its life. It is clear that growth at this age is not a smooth process; a few days of little growth are separated by the astonishing increase of two or more millimeters within twenty-four hours. The only way a conventional unconstrained smoother could avoid having negative slope would be to smooth so heavily that the data would be badly fit. We especially need to fit the data monotonically here in order to get a good estimate of the velocity of growth, displayed in Figure 6.3.

6.3.2 Expressing a strictly monotone function explicitly

The solution x to the strictly monotone smoothing problem is linked to positive function estimation in Section 6.2 since velocity Dx is now assumed to be positive. We can, therefore, use (6.1) by expressing Dx as the exponential of an unconstrained function W to obtain

$$Dx(t) = e^{W(t)} \ . \tag{6.8}$$

By integrating both sides of this equation, we obtain

$$x(t) = C + \int_{t_0}^{t} \exp[W(u)]\, du \ , \tag{6.9}$$

where C is a constant that will have to be estimated from the data.

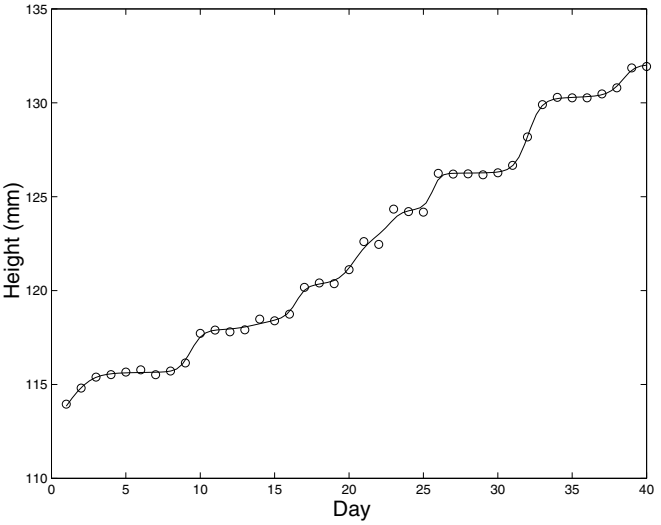

Figure 6.2. The length of the tibia of a newborn infant over the first 40 days of life. The solid line is the fit to the data using a monotone smoothing spline.

6.3.3 Expressing a strictly monotone function as a differential equation

Again we can pass directly from differential equation (6.5) to the corresponding equation for monotone functions by substituting Dx for x:

$$D^2x = wDx \ . \tag{6.10}$$

Here function $w = D^2x/Dx$, and is consequently the derivative of the logarithm of velocity, the log velocity always existing because velocity is positive.

This differential equation has the following general solution:

$$x(t) = C_0 + C_1 \int_{t_0}^{t} \exp[\int_{t_0}^{v} w(v)\, dv]\, du \ , \tag{6.11}$$

where C_1 is nonzero. This is the same equation as (6.9) if we substitute

$$W(u) = \int_{t_0}^{v} w(v)\, dv + \log C_1 = D^{-1}w(u) + \log C_1 \ .$$

Let us consider the role of function w. First, if $w(t) = 0$ for all t, we have the solution

$$x(t) = C_0 + C_1 t \ .$$

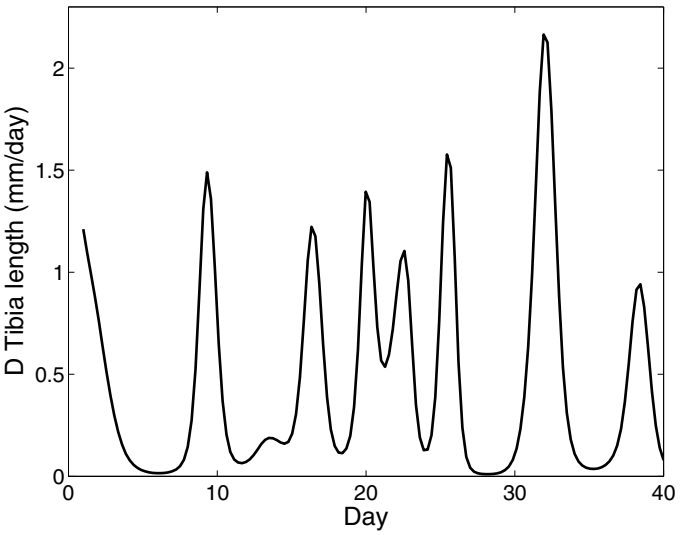

Figure 6.3. The estimated derivative or velocity of the data in Figure 6.2.

If w is a nonzero constant, then the solution becomes

$$x(t) = C_0 + C_1 e^{wt} \ .$$

Thus, linear functions are at the origin of a sort of one-dimensional functional coordinate system defined by varying function w, and exponential functions are contained within the same system. If w is a function, then the closer to zero it is at an argument value t, the more its local behavior around t will be linear. If C_1 is positive, then positive values of $w(t)$ imply local exponential increase, and negative values imply an exponential approach locally to some asymptote.

We will be especially interested in the next chapter in strictly monotone functions, called *warping functions*, that monotonically transform a time interval $[0, T]$ into itself. There the differential equation will reveal other neat properties.

6.4 The performance of spline smoothing revisited

In Chapter 5 we used direct spline smoothing to fit simulated growth data for girls. We now ask how monotone smoothing compares in performance with this direct smoothing. We simulated 1000 samples from the mean curve used in Section 5.5, but this time fit each curve with a monotone smooth, penalizing the variation in the third derivative of relative acceleration function W with a smoothing parameter of 0.1. This is roughly

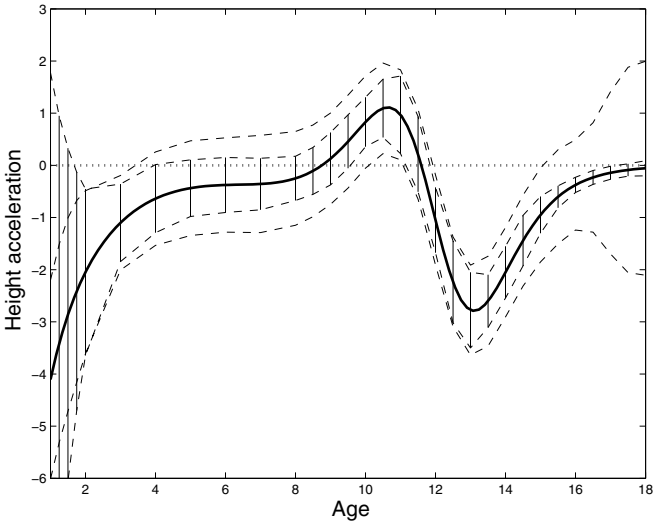

Figure 6.4. The cross-hatched area shows point-wise 95% confidence limits obtained from monotone smoothing, and the other dashed lines are the limits obtained with direct smoothing that are shown in Figure 5.7.

equivalent to the the nature and amount of roughness penalization that we used for the direct spline smoothing.

The results are shown in Figure 6.4 as point-wise confidence limits around the true curve, along with the standard error estimates that we obtained previously. There is a great improvement in precision of estimation at later ages, where the monotonicity constraint acts as a powerful smoothing principle in its own right. There is also much improvement in precision in the childhood ages as well, where we very much need the extra fitting power in order to study the smaller "mid-spurts" that are often found there. We lose, though, in early childhood, where the steep slope on the acceleration function leaves a lot of room for variation, and where violating monotonicity is no problem for the direct smoothing estimate.

6.5 Fitting probability functions

It is often necessary to estimate the probability of an event happening as a function of time or some other continuum. Does, for example, the probability that a non-smoking worker will get lung cancer depend on the number of cigarettes smoked per hour in his workspace? This *probability function* with values $p(n)$ is an example of a *dose response function* of the kind often estimated in pharmacokinetics and toxicology.

A function p taking values on the interior of the unit interval $(0, 1)$ can be neatly defined by a differential equation. The differential equation $Dx = wx$ worked for nonzero functions because, by definition, x is never zero. In this case, what is never zero is $p(t)[1 - p(t)]$. Consequently, we can propose the nonlinear differential equation

$$Dp(t) = w(t)p(t)[1 - p(t)] \ . \tag{6.12}$$

The equation implies that

$$w(t) = \frac{Dp(t)}{p(t)[1 - p(t)]} \ ,$$

so that function $w(t)$ is the slope of p at t relative to the variance of the binary variable with $p(t)$ as its parameter.

The explicit solution to this equation is

$$p(t) = \frac{\exp[\int_{t_0}^{t} w(u)\, du]}{1 + \exp[\int_{t_0}^{t} w(u)\, du]} \ , \tag{6.13}$$

and, defining

$$W(t) = \int_{t_0}^{t} w(u)\, du,$$

we have that

$$W(t) = \log \left[\frac{p(t)}{1 - p(t)}\right]$$

is the *log odds-ratio* function.

An example using of this formulation of the binomial smoothing problem can be found in Chapter 9 of Ramsay and Silverman (2002) and in Rossi, Wang and Ramsay (2002).

We would not normally choose to fit a set of frequency sample size pairs (f_j, N_j) by least squares estimation. Rather, we would use maximum likelihood estimation, or treat the model as a *general linear model* (GLM), which amounts to the same thing.

6.6 Estimating probability density functions

The estimation of a probability density function p describing the distribution of a set of sample values $t_1, \ldots, t_N$ is perhaps one of the oldest functional estimation problems in statistics. A probability density function is positive, and therefore is a special case of (6.6), and thus of the form

$$p(t) = Ce^{W(t)}$$

but with the additional restriction

$$\int p(t)\,dt = 1.$$

The constant C in (6.6) is therefore

$$C = 1/\Big(\int e^{W(t)}\,dt\Big).$$

Maximizing likelihood is the usual strategy for estimating a density function, Given N observed values t_k, we would in practice maximize the log likelihood

$$\begin{aligned}\ln L(W|\mathbf{c}) &= \sum_i^N \log p(t_i) \\ &= \sum_i^N W(t_i) - N \ln \int e^{W(t)}\,dt \\ &= \sum_i^N \mathbf{c}'\boldsymbol{\phi}(t_i) - N \ln \int \exp[\mathbf{c}'\boldsymbol{\phi}(t)]\,dt \qquad (6.14)\end{aligned}$$

where $W(t) = \mathbf{c}'\boldsymbol{\phi}(t)$ for a vector $\boldsymbol{\phi}$ of K basis functions.

The roughness of the estimated density can always be controlled by the number K of basis functions, but the versatility of roughness penalties that we have already encountered suggests they might work better here, too. If we maximize the log likelihood, we will have to subtract the roughness penalty to control roughness.

Using the penalty

$$\texttt{PEN}_3(W) = \int [D^3 W(t)]^2\,dt$$

implies that the heavier the penalty, the more W will approximate a quadratic function and, consequently, the more density p will approach a normal or Gaussian density function (Silverman, 1982, 1986). Later in Chapter 18 it will be shown that there is a linear differential operator L corresponding to most of the textbook density functions such that penalizing the size of LW can permit us to smooth toward one of a wide range of default densities.

Figure 6.5 shows the probability density function for the log of daily rainfall at Prince Rupert, British Columbia, one of the rainiest places in North America, over the years 1960 through 1994. Even there, however, about a third of the days have a precipitation of 0.1 mm or less, and we used only the 7697 days having precipitation in excess of 0.1 mm. Sixteen B-spline basis functions of order five and equally spaced knots were used to expand $W(t)$, and the size of the third derivative was penalized with $\lambda = 10^{-6}$. The distribution is rather negatively skewed, even after taking

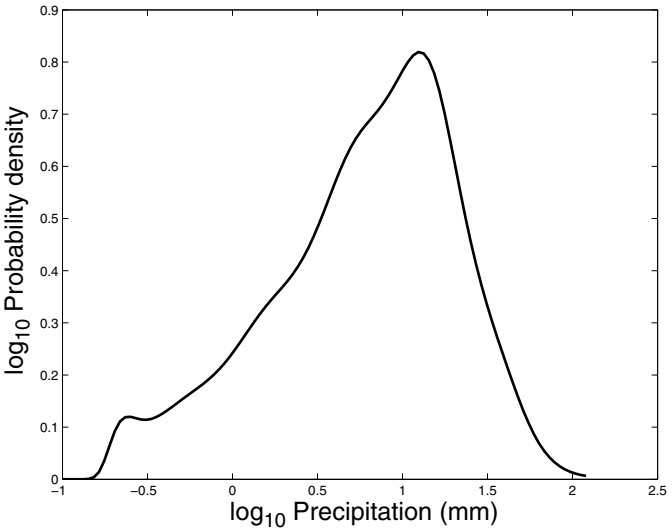

Figure 6.5. The estimated probability density function for the log (base 10) of daily precipitation at Prince Rupert, British Columbia. Only the 7697 precipitations in excess of 0.1 mm were used.

a log transformation. The sharp peak in the density suggests that rainfall comes in two forms: a steady drizzle that leaves up to a centimeter of rain, and large violent storms that can dump more than 10 centimeters of rain. It is possible to show point-wise confidence regions for density estimates such as these, but with this many observations the limits are too close to the estimated curve to be worth plotting.

6.7 Functional data analysis of point processes

A point process is a sample of N times of events, $t_1, \ldots, t_N$. These times are usually taken relative to some time t_0 at which recording begins, and this can be taken without losing any generality as $t_0 = 0$.

There are two questions that are central to point processes:

- Given that an event has already taken place at time t_i, how probable is it that the next event will take place at a time at or near $t \geq t_i$?
- What is the relation of this probability to the events already observed? For example, how does the probability that the next event will be at t depend on the time $t_i - t_{i-1}$ between the last two events?

The simplest of point processes, the *homogeneous Poisson process*, answers these two questions in this way. First, let there be no relationship

whatever between the time of the next event and the times of previous events. Second, the distribution of the time to the next event, that is $t - t_i$, is *exponential*. The distribution function and density function for an exponential distribution, respectively, are

$$F(t-t_i|\mu) = 1-\exp[-\mu(t-t_i)] \text{ and } p(t-t_i|\mu) = \mu\exp[-\mu(t-t_i)], \quad (6.15)$$

where the parameter μ is average number of events per unit time, and is often called the *intensity parameter* of the process. The exponential distribution is a model of perfect chaos in the sense that if you have waited already to time t for an event to occur, the distribution of how much longer you have to wait remains exponential. That is, you gain nothing by waiting. The larger μ, the shorter your average waiting time, which is $1/\mu$.

The likelihood $L(t_1, \ldots, t_N)$ of the sample of event times is, using $t_0 = 0$,

$$L(\mathbf{t}|\mu) = \prod_i^N p(t_i - t_{i-1}|\mu)$$

and the log likelihood is

$$\ln L(\mathbf{t}|\mu) = \sum_i^N [\ln\mu - \mu(t_i - t_{i-1})] = N\ln\mu - \mu t_N. \quad (6.16)$$

Consequently, the maximum likelihood estimate of μ is

$$\hat{\mu} = \frac{N}{t_N},$$

and it is interesting to note that the estimate depends only on the last observed time, and thus ignores previous event times.

Poisson processes, although well understood by statisticians, are usually much too simple to model real-life event times. For example, if you have waited twenty minutes for a bus, it is reasonable to assume that you won't have to wait much longer. Also, the probability of a particular waiting time is often not likely to remain constant, as (6.15) suggests; if you are waiting for a bus at 3 a.m., you can expect to wait longer than at 5 p.m.

We may decide to keep the assumption of independence of event times, but relax the assumption of a constant intensity parameter. Suppose, now, that intensity parameter μ is a function of time with values $\mu(t)$. The mathematically natural way to generalize (6.15) to this situation is

$$\begin{aligned} F(t - t_i|\mu) &= 1 - \exp[-\int_{t_i}^t \mu(s)\, ds] \\ p(t - t_i|\mu) &= \mu(t)\exp[-\int_{t_i}^t \mu(s)\, ds]. \end{aligned} \quad (6.17)$$

This more general model reduces to the Poisson process when $\mu(s)$ is a constant. The log likelihood now becomes

$$\ln L(\mathbf{t}|\mu) = \sum_i^N \ln \mu(t_i) - \int_0^{t_N} \mu(s)\, ds. \tag{6.18}$$

The fact that $\mu(t)$ is nonnegative suggests that we should take the approach in Section 6.2 and use the exponentiated basis function expansion

$$\mu(t) = \exp[\mathbf{c}'\boldsymbol{\phi}(t)], \tag{6.19}$$

where $\boldsymbol{\phi}$ is a functional vector of K basis functions, and vector $\mathbf{c}$ contains the coefficients of the expansion. Substituting this into 6.18 gives us

$$\ln L(\mathbf{t}|\mu) = \sum_i^N \mathbf{c}'\boldsymbol{\phi}(t_i) - \int_0^{t_N} \exp[\mathbf{c}'\boldsymbol{\phi}(s)]\, ds. \tag{6.20}$$

In order to compute the maximum likelihood estimate of the coefficients in $\mathbf{c}$, we need the derivative

$$D_{\mathbf{c}} \ln L(\mathbf{t}|\mu) = \sum_i^N \boldsymbol{\phi}(t_i) - \int_0^{t_N} \boldsymbol{\phi}(s) \exp[\mathbf{c}'\boldsymbol{\phi}(s)]\, ds. \tag{6.21}$$

Setting $D_{\mathbf{c}} \ln L$ to zero does not result in any simple solution for $\mathbf{c}$, and we must resort to numerical optimization methods to maximize (6.20).

If we compare the log likelihood in this situation with (6.14) for the log likelihood in the problem density estimation, the similarity is striking. The first term is the same, and the second term for densities involves multiplying by N, logging and then integrating to ∞ rather than just integrating to t_N. The problems are thus essentially the same except for relatively minor changes in the normalizing constraint.

Lupus is an autoimmune disease characterized by sudden flares in symptoms. Figure 6.6 shows the timings of 41 flares for a single patient over nearly 19 years, along with the estimated intensity function μ for these data. The intensity function reflects well the two periods when this patient was relatively free of flares, as well as the period of intense disease activity around year eight. The point-wise confidence limits, however, caution us that this amount of data does not pin down the intensity function especially well. These results were achieved using 13 order four B-splines with a roughness penalty on $D^2\mu$ multiplied by smoothing parameter 5.0.

6.8 Fitting a linear model with estimation of the density of residuals

Our default approach to fitting data has been to minimize the sum of squared residuals. This is tantamount to assuming that the residuals are

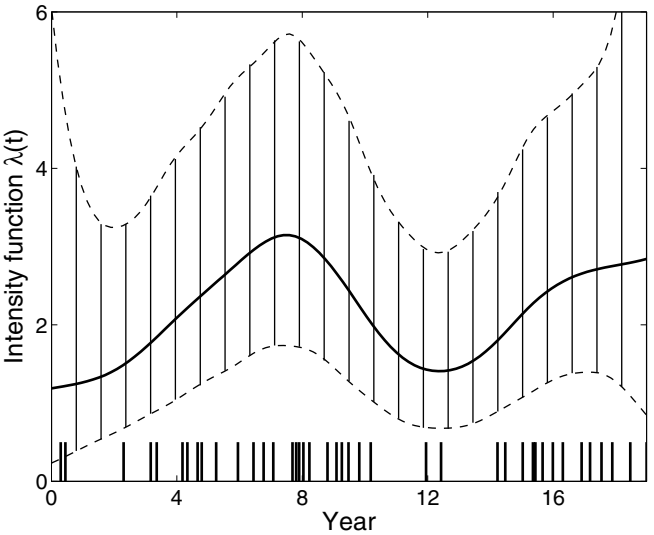

Figure 6.6. The times of flares in lupus symptoms for a single patient are indicated by vertical lines on the horizontal axis. The solid line is the intensity function μ for a nonhomogeneous Poisson process estimated from these data. The dashed lines indicate 95% point-wise confidence limits for the intensity function.

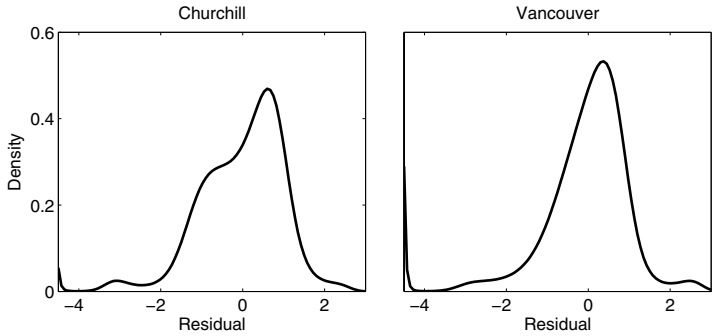

Figure 6.7. The estimated probability density functions of the residuals from fitting log (base 10) of daily precipitation at Churchill, Manitoba, and Vancouver, British Columbia.

normally distributed in the population of potential observations from which we have sampled.

Assuming normality can be risky, and especially if the true distribution has long tails. It would be safer if we could estimate the density of the residuals as well as the fit to the data by combining the methodology in

Chapters 4 and 5 with the density estimation approach of Section 6.6. For example, if there are residuals many standard deviations from the curve, then allowing for this in estimating the fitted curve will give a measure of *robustness* to the fit. In other situations, the density of the residuals might be interesting on its own, as we saw in Figure 6.5.

Suppose that we have a vector $\mathbf{y}$ of N dependent variable observations, and an N by p design matrix $\mathbf{Z}$ available as a basis for a linear model for $\mathbf{y}$. In the curve-fitting situation, for example, $\mathbf{Z}$ will contain the values $\phi_k(t_j)$ of the basis functions at the sampling points $t_i, i = 1 \ldots, N$. Let a vector $\mathbf{e}$ of N residuals be of the form

$$\mathbf{e} = \mathbf{y} - \mathbf{Z}\mathbf{c}, \tag{6.22}$$

where we expect to estimate the coefficients in p-vector $\mathbf{c}$.

In addition to estimating $\mathbf{c}$, however, we want an estimate of the density of the residuals $\mathbf{e}$, and we want to use that density in the estimation of $\mathbf{c}$. It will often be convenient to *standardize* the residuals prior to estimating the residuals by dividing them by a constant σ. If σ is a preliminary estimate of the standard deviation of the e_i's, for example, this will normalize the interval over which the residuals are distributed to something not too far from $[-3, 3]$. Consequently, defining $\mathbf{r}$ to be

$$\mathbf{r} = \mathbf{e}/\sigma, \tag{6.23}$$

we want an estimate p of the standardized residual density with values

$$\begin{aligned} p(r_i) &= \frac{e^{W(r_i)}}{\int e^{W(u)}\,du} \\ &= \frac{e^{W(\frac{y_i - \sum_j z_{ij}c_j}{\sigma})}}{\int e^{W(u)}\,du}, \end{aligned} \tag{6.24}$$

where, as with (6.2) but with a change of symbols,

$$W(r) = \sum_k b_k \psi_k(r).$$

The computational problem is now to maximize

$$\texttt{PENLIK}(W|\mathbf{b}, \mathbf{c}) = \sum_i^N \log p(t_i) = \sum_i^N W(t_i) - N \int e^{W(t)}\,dt - \int [LW(t)]^2\,dt \tag{6.25}$$

with respect to both $\mathbf{b}$ and $\mathbf{c}$. This criterion can, of course, be further augmented with a roughness penalty on the fit defined by $\mathbf{Z}\mathbf{c}$.

Figure 6.7 shows the residual density functions estimated in fitting log precipitation for Churchill high up on Hudson's Bay and Vancouver on the lower Pacific coast. In both cases, we see some strong departures from normality. Both densities have much heavier negative tails than the normal

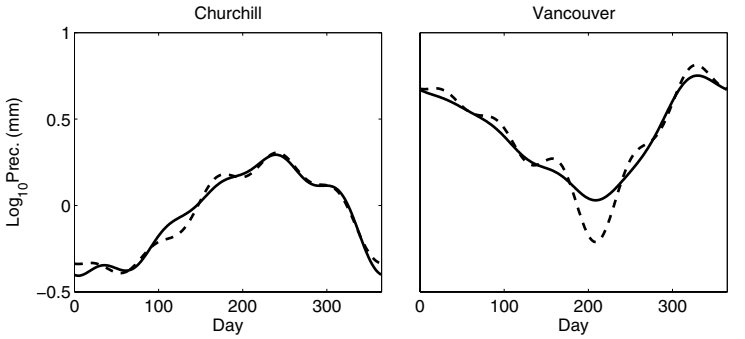

Figure 6.8. The estimated log (base 10) of daily precipitation at Churchill, Manitoba, and Vancouver, British Columbia. The heavy solid lines are the fits resulting from estimating the residual densities, and the lighter dashed lines were fits using least squares estimation.

distribution, and Vancouver in particular has a number of strong normalized residuals at about −4.5. Churchill has a shoulder on the negative side of its density, suggesting a rainy season and a dry season.

Figure 6.8 shows the two fitted precipitation curves, and in each case also the curve fit by least squares. Both curves fitted by the estimated density are smoother, and Vancouver's fit is also less perturbed by some large negative residuals in mid–summer, while the least squares fit was. Estimating the density in this case made the fit more robust.

6.9 Further notes and readings

For a general purpose introduction to nonparametric density estimation, see Silverman (1986). Scott (1992) also considers multivariate density estimation. Otherwise, there is a vast literature on this topic, and especially using kernel smoothing methods.

A thorough reference on point processes is Snyder and Miller (1991), but the older Cox and Lewis (1966) is rather more readable.

7

The registration and display of functional data

7.1 Introduction

We can now assume that our observations are in functional form, and want to proceed to consider methods for their analysis. We are not quite ready, however; a problem of critical importance to functional data needs a solution. We see often that variation in functional observations involves both phase and amplitude, and that confounding these two leads to many problems. Our main emphasis is on *registration* of the data, involving transformations of the argument t rather than the values $x(t)$.

Figure 1.2 illustrates a problem that can frustrate even the simplest analyses of replicated curves. Ten records of the acceleration in children's height show individually the salient features of growth: the large deceleration during infancy is followed by a rather complex but small-sized acceleration phase during late childhood. Then the dramatic acceleration-deceleration pulses of the pubertal growth spurt finally give way to zero acceleration in adulthood. But the timing of these salient features obviously varies from child to child, and ignoring this timing variation in computing a cross-sectional mean function, shown by the heavy dashed line in Figure 1.2, can result in a estimate of average acceleration that does not resemble any of the observed curves. In this case, the mean curve has less variation during the pubertal phase than any single curve, and the duration of the mean pubertal growth spurt is rather larger than that of any individual curve.

The problem is that the growth curves exhibit two types of variability. *Amplitude variability* pertains to the sizes of particular features such as the

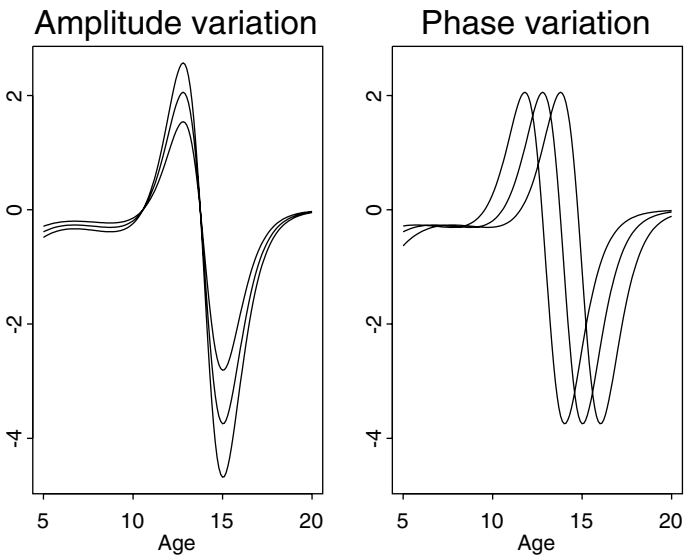

Figure 7.1. The left panel shows three height acceleration curves varying only in amplitude. The right panel shows three curves varying only in phase.

velocity peak in the pubertal growth spurt, ignoring their timings. *Phase variability* is variation in the timings of the features without considering their sizes. Before we can get a useful measure of a typical growth curve, we must separate these two types of variation, so that features such as the pubertal spurt occur at roughly the same "times" for all girls. The problem is expressed in schematic terms in Figure 7.1, where we see in the left panel two acceleration curves that differ only in amplitude, and in the right panel two curves with the same amplitude, but differing in phase.

The need to transform curves by transforming their arguments, which we call *curve registration*, can be motivated as follows. The rigid metric of physical time may not be directly relevant to the internal dynamics of many real-life systems. Rather, there can be a sort of biological or meteorological time scale that can be nonlinearly related to physical time, and can vary from case to case.

Human growth, for example, is the consequence of a complex sequence of hormonal events that do not happen at the same rate for every child. The intensity of the pubertal growth spurts of two children should be compared at their respective ages of peak velocity rather than at any fixed age. A colleague with a musical turn of mind refers to this as differences in the *tempo* of growth.

Similarly, weather is driven by ocean currents, reflectance changes for land surfaces, and other factors that are timed differently for different spatial locations and different years. Winter comes early in some years, and

late in others, and typically arrives later at some weather stations than others. We need to assess how cold the average winter is at the time the average temperature bottoms out rather than at any fixed time.

Put more abstractly, the values of two or more function values $x_i(t_i)$ can in principle differ because of two types of variation. The first is the more familiar vertical variation, or *amplitude variation*, due to the fact that $x_1(t)$ and $x_2(t)$ may simply differ at points of time t at which they are compared, but otherwise exhibit the same shape features at that time. But they may also exhibit *phase variation* in the sense that functions x_1 and x_2 should not be compared at the same time t because they are not exhibiting the same behavior. Instead, in order to compare the two functions, the time scale itself has to be distorted or transformed.

We now look at several types of curve registration problems, beginning first with the problem of simply translating or shifting the values of t by a constant amount δ. Then we discuss landmark registration, which involves transforming t nonlinearly in order to line up important features or landmarks for all curves. Finally, we look at a more general method for curve registration.

7.2 Shift registration

Many of the issues involved in registration can be illustrated by considering the simplest case, a simple shift in the time scale. The pinch force data illustrated in Figure 1.11 are an example of a set of functional observations that must be aligned by moving each curve horizontally before any meaningful cross-curve analysis is possible. This often happens because the time at which the recording process begins is arbitrary, and is unrelated to the beginning of the interesting segment of the data, in this case the period over which the measured squeeze actually takes place.

Let the interval $\mathcal{T}$ over which the functions are to be registered be $[T_1, T_2]$. We also need to assume that each sample function x_i is available for some region beyond each end of $\mathcal{T}$. The pinch force data, for example, are observed for substantial periods both before and after the force pulse that we wish to study. In the case of periodic data such as the Canadian temperature records, this requirement is easily met since one can wrap the function around by using the function's behavior at the opposing end of the interval.

We are actually interested in the values

$$x_i^*(t) = x_i(t + \delta_i),$$

where the shift parameter δ_i is chosen in order to appropriately align the curves. For the pinch force data, the size of δ_i is of no real interest, since it merely measures the gap between the initialization of recording and the beginning of a squeeze. Silverman (1995) refers to this situation, in which

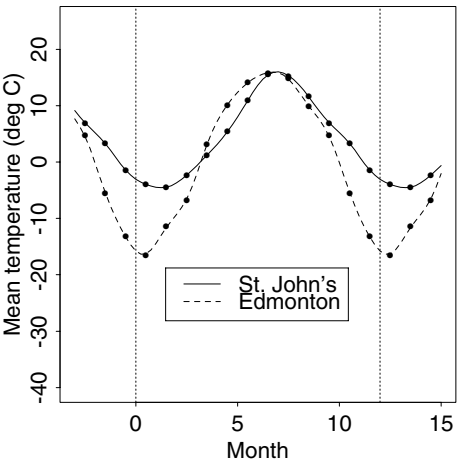

Figure 7.2. Temperature records for two weather stations where in the timing of the seasons differs by a roughly constant shift.

a shift parameter must be accounted for but is of no real interest, as a *nuisance effects* problem.

The Canadian temperature data present a curve alignment problem of a somewhat different nature. As Figure 7.2 indicates, two temperature records, such as those for St. John's, Newfoundland, and Edmonton, Alberta, can differ noticeably in terms of the phase or timing of key events, such as the lowest mean temperature and the timing of spring and autumn. In this case, the shifts that would align these two curves vertically are of intrinsic interest, and should be viewed as a component of variation that needs careful description. It turns out that continental stations such as Edmonton have earlier seasons than marine stations such as St. John's, because of the capacity of oceans to store heat and to release it slowly. In fact, either station's weather would have to be shifted by about three weeks to align the two.

When, as in the temperature data case, the shift is an important feature of each curve, we characterize its estimation as a *random effects* problem. Silverman (1995) also distinguishes a third and intermediate *fixed effects* case in which the shift must be carried out initially, and while not being discarded completely once the functions x_i^* have been constructed, is nevertheless only of tangential interest.

7.2.1 The least squares criterion for shift alignment

The basic mechanics of estimating the shifts δ_i are the same, whether they are considered as nuisance or random effects. The differences become important when we consider the analysis in subsequent chapters, because in the random effects case (and, to some extent, the fixed effects case) the δ_i enter the analysis. However, for present purposes we concentrate on the pinch force data as an example.

The estimation of a shift or an alignment requires a criterion that defines when several curves are properly registered. One possibility is to identify a specific feature or *landmark* for a curve, and shift each curve so that this feature occurs at a fixed point in time. The time of the maximum of the smoothed pinch force is an obvious landmark. Note that this might also be expressed as the time at which the first derivative crosses zero with negative slope, and landmarks are often more easily identifiable at the level of some derivative.

However, the registration by landmark or feature alignment has some potentially undesirable aspects: The location of the feature may be ambiguous for certain curves, and if the alignment is only of a single point, variations in other regions may be ignored. If, for example, we were to register the two temperature curves by aligning the midsummers, the midwinters might still remain seriously out of phase.

Instead, we can define a global registration criterion for identifying a shift δ_i for curve i as follows. First we estimate an overall mean function $\hat{\mu}(t)$ for t in $\mathcal{T}$. If the individual functional observations x_i are smooth, it usually suffices to estimate $\hat{\mu}$ by the sample average $\bar{x}$. However, we wish to be able to evaluate derivatives of $\hat{\mu}$, and so more generally we want to smooth the overall estimate using one of the methods described in Chapters 4 and 5. We can now define our global registration criterion by

$$\begin{aligned} \texttt{REGSSE} &= \sum_{i=1}^{N} \int_{\mathcal{T}} [x_i(t+\delta_i) - \hat{\mu}(t)]^2 \, ds \\ &= \sum_{i=1}^{N} \int_{\mathcal{T}} [x_i^*(t) - \hat{\mu}(t)]^2 \, ds. \end{aligned} \tag{7.1}$$

Thus, our measure of curve alignment is the integrated or global sum of squared vertical discrepancies between the shifted curves and the sample mean curve.

The target function for transformation in (7.1) is the unregistered cross-sectional estimated mean $\hat{\mu}$. But of course one of the goals of registration is to produce a better estimate of this same mean function. We therefore expect to proceed iteratively: beginning with the unregistered cross-sectional estimated mean, argument values for each curve are shifted so as to minimize `REGSSE`, then the estimated mean $\hat{\mu}$ is updated by re-estimating it from the *registered* curves x_i^*, and a new iteration is then undertaken us-

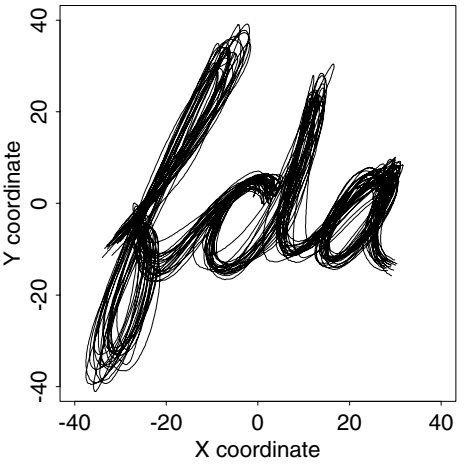

Figure 7.3. Twenty replications of "fda" written by one of the authors.

ing this revised target. This procedure of estimating a transformation by transforming to an iteratively updated average is often referred to as the *Procrustes method*. In practice, we have found that the process usually converges within one or two iterations.

7.3 Feature or landmark registration

A landmark or a feature of a curve is some characteristic that one can associate with a specific argument value t. These are typically maxima, minima, or zero crossings of curves, and may be identified at the level of some derivatives as well as at the level of the curves themselves.

We now turn to the more general problem of estimating a possibly nonlinear transformation h_i of t, and indicate how we can use landmarks to estimate this transformation. Coincidentally, the illustrative example we use shows how vector-valued functional data can be handled by obvious extensions of methods for scalar-valued functions.

The landmark registration process requires for each curve x_i the identification of the argument values $t_{if}, f = 1, \ldots, F$ associated with each of F features. The goal is to construct a transformation h_i for each curve such that the registered curves with values

$$x^*(t) = x_i[h_i(t)]$$

have more or less identical argument values for any given landmark.

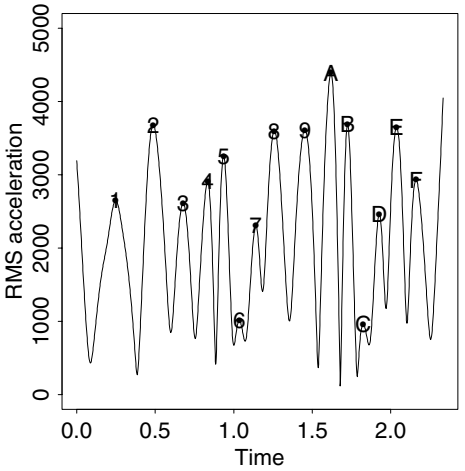

Figure 7.4. The average length of the acceleration vector for the 20 handwriting samples. The characters identify the 15 features used for landmark registration.

Consider, for example, the 20 replications of the letters "fda" in Figure 7.3. Each sample of handwriting was obtained by recording the position of a pen at a sampling rate of 600 times per second. There was some preprocessing to make each script begin and end at times 0 and 2.3 seconds, and to compute coordinates at the same 1,401 equally-spaced time-values. Each curve x_i in this situation is vector-valued, since two spatial coordinates are involved, and we use $\texttt{ScriptX}_i$ and $\texttt{ScriptY}_i$ to designate the X- and Y-coordinates, respectively.

Not surprisingly, there is some variation from observation to observation, and one goal is to explore the nature of this variation. But we want to take into account that, for example, variation in the "f" can be of two sorts. There is temporal variation due to the fact that timing of the top of the upper loop, for example, is variable. While this type of variation would not show up in the plots in Figure 7.3, it may still be an important aspect of how these curves vary. On the other hand, there is variation in the way the shape of each letter is formed, and this is obvious in the figure.

We estimated the accelerations or second derivatives of the two coordinate functions $D^2\texttt{ScriptX}_i$ and $D^2\texttt{ScriptY}_i$ by the local polynomial method described in Chapter 4. Figure 7.4 displays the average length of the acceleration vector

$$\sqrt{(D^2\texttt{ScriptX}_i)^2 + (D^2\texttt{ScriptY}_i)^2}$$

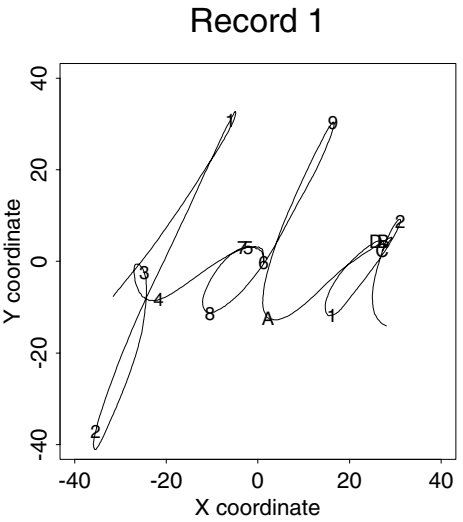

Figure 7.5. The first handwriting curve with the location of the 15 landmarks indicated by the characters used in Figure 7.4.

and we note that there are 15 clearly identified maxima, indicating points where the pen is changing direction. We also found that these maxima were easily identifiable in each record, and we were able to determine the values of t_{if} corresponding to them by just clicking on the appropriate points in a plot. Figure 7.5 shows the first curve with these 15 features labelled, and we can see that landmarks labelled "4" and "A" mark the boundaries between letters. Figure 7.6 plots the values of the landmark timings t_{if} against the corresponding timings for the mean function, t_{0f}. We were interested to see that the variability of the landmark timings was rather larger for the initial landmarks than for the later ones, and we were surprised by how small the variability was for all of them.

The identification of landmarks enabled us to compare the X- and Y-coordinate values for the 20 curves at the landmark times, but of course we also wanted to make comparisons at arbitrary points between landmarks. This required the computation of a function h_i for each curve, called a *time-warping function* in the engineering literature, with the properties

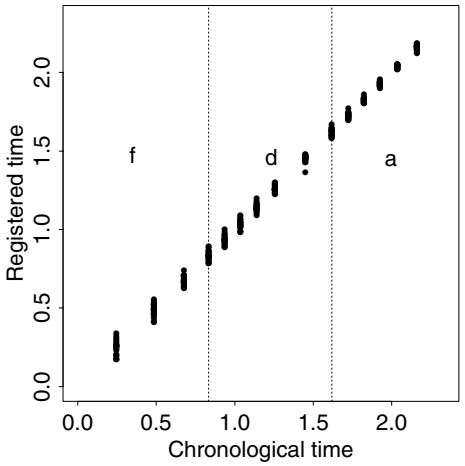

Figure 7.6. The timings of the landmarks for all 20 scripts plotted against the corresponding timings for the mean curve.

- $h_i(0) = 0$
- $h_i(2.3) = 2.3$
- $h_i(t_{0f}) = t_{if}, f = 1, \ldots, 15$
- h_i is strictly monotonic: $s < t$ implies that $h_i(t) < h_i(t)$.

The values of the adjusted curves at time t are $\texttt{ScriptX}[h_i(t)]$ and $\texttt{ScriptY}[h_i(t)]$. In all the adjusted curves, the landmarks each occur at the same time as in the mean function. In addition, the adjusted curves are also more or less aligned between landmarks. In this application, we merely used linear interpolation for time values between the points (t_{0f}, t_{if}) (as well as (0,0) and (2.3,2.3)) to define the time warping function h_i for each curve. We introduce more sophisticated notions in the next section. Figure 7.7 shows the warping function computed in this manner for the first script record. Because h_1 is below the diagonal line in the region of "f," the aligned time $h_1(t)$ is earlier than the actual time of features, and hence the actual times for curve 1 are retarded with respect to the mean curve.

We can now re-compute the mean curve by averaging the registered curves. The result is in Figure 7.8, shown along with the mean for the unregistered data. Although the differences are not dramatic, as we might expect given the mild curvature in h_1, we do see that the upper and lower loops of the "f" are now more pronounced, and in fact do represent the original curves substantially better.

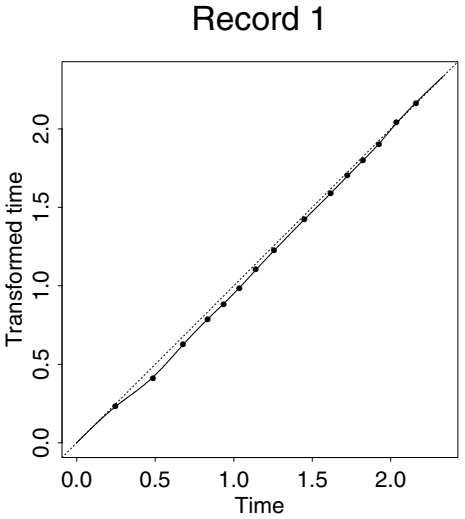

Figure 7.7. The time warping function h_1 estimated for the first record that registers its features with respect to the mean curve.

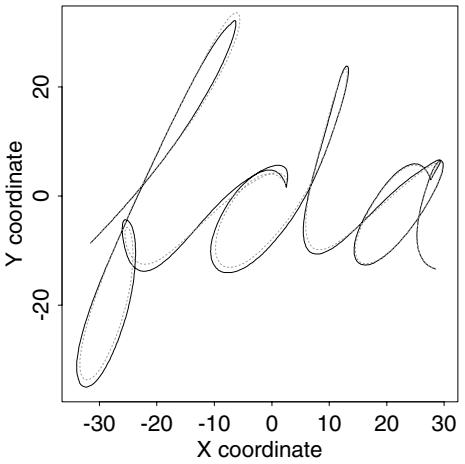

Figure 7.8. The solid line is the mean of the registered "fda" curves, and the dashed line is the mean of the unregistered curves.

7.4 Using the warping function h to register x

Now that a warping function h has been estimated from landmark registration, or by using the continuous method described in a later section, you will want to calculate the registered function values $x^*(t) = x[h(t)]$. This requires two steps.

First, estimate the *inverse warping function* $h^{-1}(t)$ with the property $h^{-1}[h(t)] = t$. Note that this is not an inverse in the sense of the reciprocal. Instead, $h^{-1}(t)$ is computed by smoothing or interpolating the relationship between $h(t)$ plotted on the horizontal axis and t plotted on the vertical axis. You can then use simple interpolation to get the values of this inverse function at an equally spaced set of values of t if required. Note that it will be essential that this smoothing or interpolation function be strictly monotonic, so you may have to use lots of values of t and/or employ monotone smoothing described in Chapter 6.

The second step is to smooth or interpolate the relationship between $h^{-1}(t)$ plotted on the abscissa and $x(t)$ plotted on the ordinate. You can then use simple interpolation to get the values of this registered function at an equally spaced set of values of t if required.

7.5 A more general warping function h

The linear interpolation scheme that we used on the handwriting data to estimate the time-warping function h has two limitations. First, if we want to compute higher order derivatives of the curves with respect to warped time, the warping function must also be differentiable to the same order, a linear interpolation would not carry us beyond the first derivative. Secondly, we will shortly consider *continuous registration* methods that do not use landmarks and where the idea of interpolating a sequence of points will not be helpful.

Time is itself a growth process, and thus can be linked to our discussion in Chapter 6 on how to model the children's growth curves. That is, we can use the formulation

$$h(t) = C_0 + C_1 \int_0^t \exp W(u)\, du \tag{7.2}$$

that we used in (6.9). Here the constants C_0 and C_1 are fixed by the requirement that $h(t) = t$ at the lower and upper limits of the interval over which we model the data. Or, if shift registration is a possibility, the constant term C_0 can be allowed to pick any constant phase shift that is required.

Physical or clock time grows linearly, of course, and thus corresponds to $W(u) = 0$. If $W(u)$ is *positive*, then $h(t) > t$, warped time is growing faster than clock time, and this is what we want if our observed process

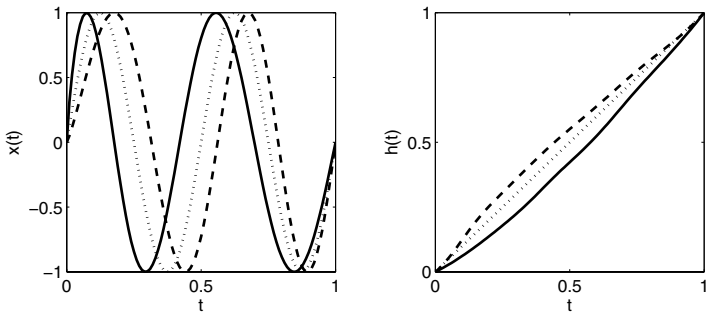

Figure 7.9. The left panel shows the target function, $x_0(t) = \sin(4\pi t)$, as a dotted line; an early function, $x_E(t) = \sin(4\pi t^{0.8})$, as a solid line; and a late function, $x_L(t) = \sin(4\pi t^{1.2})$, as a dashed line. The corresponding warping functions that register the early and late curves to the target are shown in the right panel.

is running *late*. Similarly, for negative values of $W(u)$, $h(t) < t$, and clock time is being slowed down for a process that is running ahead of some target.

The left panel of Figure 7.9 displays two examples. Here the target or standard function is $x_0(t) = \sin(4\pi t)$, the early function is $x_E(t) = \sin(4\pi t^{0.8})$ and the late function is $x_L(t) = \sin(4\pi t^{1.2})$. Warping $h_E(t) = t^{0.125}$ will register the first example since $\sin[4\pi(t^{0.8})^{1.25}] = \sin(4\pi t)$, and similarly $h_L(t) = t^{0.833}$. Approximations to the two warping functions by a method to be described below are presented in the right panel, and we can see there how early functions are associated with time-decelerating warpings, and late functions with time-accelerating warpings.

The use of (7.2) as a representation of a warping function has a very handy bonus. Providing that the warp h is reasonably smooth and mild, the inverse warp h^{-1} is achieved to a close approximation by merely replacing W in the equation by $-W$.

7.6 A continuous fitting criterion for registration

The least squares criterion (7.1) worked well for simple shift registration, but gets us into trouble for more general warping functions. The lower panel in Figure 7.10 shows why. When two functions differ in terms of amplitude as well as phase, the least squares criterion uses time warping to also minimize amplitude differences by trying to squeeze out of existence regions where amplitudes differ. Put another way, the least squares fitting criterion is intrinsically designed to assess differences in amplitude rather

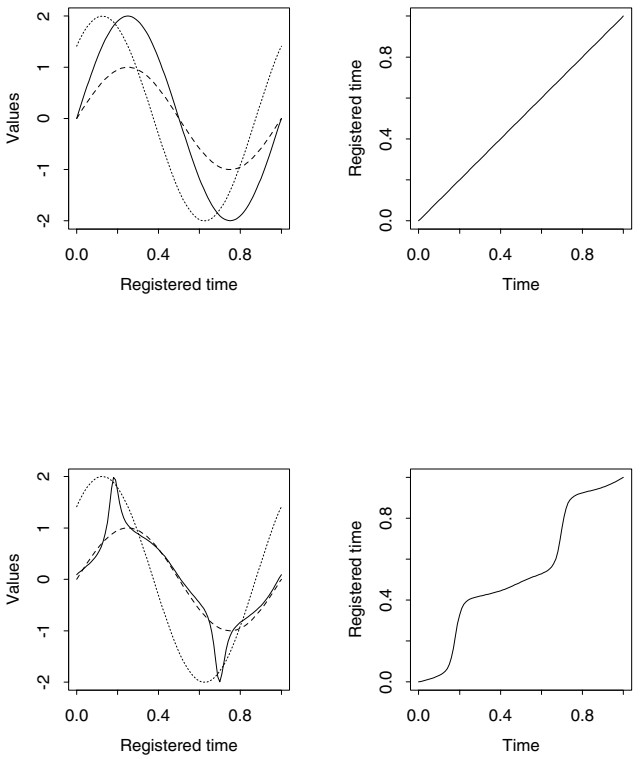

Figure 7.10. The upper two panels show results for an artificial registration problem using the minimum eigenvalue criterion. The dotted curve in the upper-left panel is the curve to be registered to the curve indicated by the dashed line. The solid line is the registered curve. The upper-right panel contains the warping function for this case, $h(t) = t$. The lower panels show the same results using the least squares criterion.

than phase. This wasn't a problem when only time shifts were involved since such simple time warps cannot affect amplitude differences.

Suppose two curves x_0 and x_1 differ only in amplitude but not in phase, such as in the left panel of Figure 7.10. Then, if we plot the function values $x_0(t)$ and $x_1(t)$ against each other, we will see a straight line. Amplitude differences will then be reflected in the slope of the line, a line at 45^o corresponding to no amplitude differences.

Now thinking about a line as a one-dimensional set of points on a plane, we can turn to principal components analysis as just the right technique for assessing how many dimensions are required to represent the distribution of these points. This technique will yield only one positive eigenvalue if the

point spread is, in fact, one-dimensional. That is, the size of the smallest eigenvalue measures departures from unidimensionality.

Let us consider now evaluating both the target function x_0 and the registered function x^* at a fine mesh of n values of t to obtain the pairs of values $(x_0(t), x[h(t)])$. Let the n by two matrix $\mathbf{X}$ contain these pairs of values. Then the two-by-two cross-product matrix $\mathbf{X}'\mathbf{X}$ would be what we would analyze by principal components.

The following order two matrix is the functional analogue of the cross-product matrix $\mathbf{X}'\mathbf{X}$.

$$\mathbf{T}(h) = \begin{bmatrix} \int \{x_0(t)\}^2\, dt & \int x_0(t)x[h(t)]\, dt \\ \int x_0(t)x[h(t)]\, dt & \int \{x[h(t)]\}^2\, dt \end{bmatrix} \tag{7.3}$$

We see that the summations over points implied by the expression $\mathbf{X}'\mathbf{X}$ have here been replaced by integrals. Otherwise this is the same matrix. We have expressed the matrix as a function of warping function h to remind ourselves that it does depend on h.

Consequently, we can now express our fitting criterion for assessing the degree to which two functions are registered as follows:

$$\texttt{MINEIG}(h) = \mu_2[\mathbf{T}(h)], \tag{7.4}$$

where the function μ_2 is the size of the second eigenvalue of its argument, which is an order two symmetric matrix. When $\texttt{MINEIG}(h) = 0$, we have achieved registration, and h is the warping function that does the job.

As is now routine, we will want to apply some regularization now and then to impose smoothness on h, so we extend our criterion to

$$\texttt{MINEIG}_\lambda(h) = \texttt{MINEIG}(h) + \lambda \int \{W^{(m)}(t)\}^2\, dt. \tag{7.5}$$

Here we are assuming that h is of the form (7.2), and that we achieve smoothness in h by smoothing the function W that defines it.

The results in Figure 7.9 were achieved by expanding W in terms of 13 B-splines with equally spaced knots, and penalizing the size of its second derivative using a smoothing parameter of $\lambda = 10^6$.

7.7 Registering the height acceleration curves

The 10 acceleration functions in Figure 1.2 were registered by the Procrustes method and the regularized basis expansion method set out in Section 7.6. The interval $\mathcal{T}$ was taken to be $[4, 18]$ with time measured in years. The break-values τ_k defining the monotone transformation family (7.2) were 4, 7, 10, 12, 14, 16 and 18 years, and the curves were registered over the interval [4,18] using criterion (7.5) with $\lambda = 0.001$. A single Procrustes iteration produced the results displayed in Figure 7.11. The left panel displays the 10 warping functions h_i, and the right panel shows

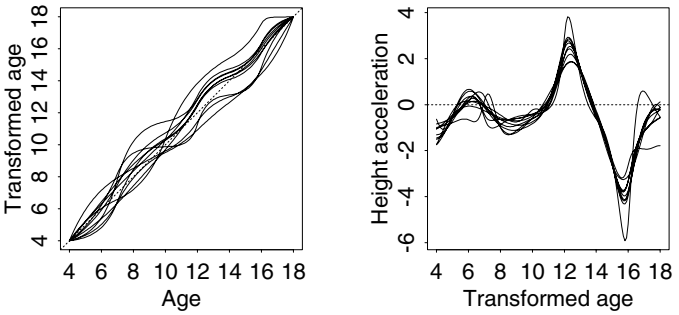

Figure 7.11. The left panel contains the estimated time warping functions h_i for the 10 height acceleration curves in Figure 1.2. The right panel displays the registered curves.

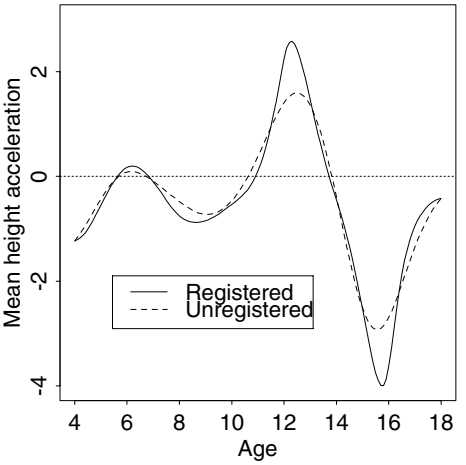

Figure 7.12. The cross-sectional means of the registered and unregistered height acceleration curves displayed in Figure 1.2.

the curve values $x_i[h_i(t)]$. Figure 7.12 compares the unregistered and registered cross-sectional means. We see that the differences are substantial, and moreover that the mean of the registered function tends to resemble much more closely most of the sample curves.

7.8 Some practical advice

Before registration, remove amplitude effects that can be accounted for by vertical shifts or scale changes, by centering and possibly rescaling the curves. This is standard advice in data analysis; deal with obvious effects in a simple way before moving on to more sophisticated procedures.

In general it is not clear that variation in the amplitude of curves can be cleanly separated from the variation that the registration process aims to account for. It is easy to construct examples where a registration function h that is allowed to be highly nonlinear can remove variation that is clearly of an amplitude nature, and the lower panels of Figure 7.10.

This problem of lack of identifiability of the two types of variation, horizontal and vertical, is perhaps less of a concern if only linear transformations are permitted, and is also not acute for landmark registration, where the role of the transformation is to only align curve features.

However, there is one situation that implies relatively unambiguous separation of the two types of variation. This happens with curves that cross zero at a number of points. At and near these zero crossings, only phase variation is possible. In effect, zero crossings are landmarks that should be aligned. Consequently, it may be wise to consider registering a derivative of a curve rather than the curve itself, since derivatives often cross zero. This is why we registered the acceleration curves above rather than the height or velocity curves.

If flexible families of monotone transformations such as those described above are used in conjunction with a global fitting criterion such as `MINEIG`, allow transformations to differ from linear only with caution by careful application of regularization.

In general, we have found it wise to first register on any landmarks that are clearly identifiable before using the continuous registration procedure. For example, in our work with the growth data we first register the curves using the zero-crossing in the middle of the pubertal growth spurt as a single landmark. Then we use the curves resulting from this preliminary registration as inputs to a continuous registration. If we use the notation h_L and $h_{C|L}$ to refer to the landmark warps and the continuous warps after landmark registration, respectively, then the final composite warping function is $h(t) = h_{C|L}[h_L(t)]$ or $h = h_{C|L} \circ h_L$.

7.9 Computational details

7.9.1 Shift registration by the Newton-Raphson algorithm

We can estimate a specific shift parameter δ_i iteratively by using a modified Newton-Raphson algorithm for minimizing `REGSSE`. This procedure requires derivatives of `REGSSE` with respect to the δ_i. If we assume that the

differences between x_i^* and $\hat{\mu}$ at the ends of the interval can be ignored (this is exactly true in the periodic case, and often approximately true in the non-periodic case if the effects of real interest are concentrated in the middle of the interval), then we have

$$\begin{aligned}\frac{\partial}{\partial\delta_i}\texttt{REGSSE} &= 2\int_{\mathcal{T}}\{x_i(t+\delta_i)-\hat{\mu}(t)\}Dx_i(t)\,dt \\ \frac{\partial^2}{\partial\delta_i^2}\texttt{REGSSE} &= 2\int_{\mathcal{T}}\{x_i(t+\delta_i)-\hat{\mu}(t)\}D^2x_i(t)\,dt \\ &+ 2\int_{\mathcal{T}}\{Dx_i(t)\}^2\,dt. \qquad (7.6)\end{aligned}$$

The modified Newton-Raphson algorithm works as follows:

Step 0: Begin with some initial shift estimates $\delta_i^{(0)}$, perhaps by aligning with respect to some feature, or even $\delta_i^{(0)} = 0$. But the better the initial estimate, the faster and more reliably the algorithm converges. Complete this step by estimating the average $\hat{\mu}$ of the shifted curves, using a method that allows the first two derivatives of $\hat{\mu}$ to give good estimates of the corresponding derivatives of the population mean, such as local polynomial regression of degree 4, or roughness penalty smoothing with an integrated squared fourth derivative penalty.

Step ν, for $\nu = 1, 2, \ldots$: Modify the estimate $\delta_i^{(\nu-1)}$ on the previous iteration by

$$\delta_i^{(\nu)} = \delta_i^{(\nu-1)} - \alpha\frac{(\partial/\partial\delta_i)\texttt{REGSSE}}{(\partial^2/\partial\delta_i^2)\texttt{REGSSE}},$$

where α is a step-size parameter that can sometimes simply be set to one. It is usual to drop the first term (7.6) in the second derivative of REGSSE since it vanishes at the minimizing values, and convergence without this term tends to be more reliable when current estimates are substantially far from the minimizing values. Once the new shifts are estimated, recompute the estimated average $\hat{\mu}$ of the shifted curves.

Although the algorithm can in principle be iterated to convergence, and although convergence is generally fast, we have found that a single iteration is often sufficient with reasonable initial estimates. For the pinch force data, we began by aligning the smoothed curves by setting the location of the maximum of each curve at 0.1 seconds. The shifts involved ranged from -20 to 50 milliseconds. We then carried out a single Newton-Raphson update ($\nu = 1$ above) where the range $\mathcal{T}$ of integration was from 23 to 251 milliseconds. The changes in the δ_i ranged from -3 to 2 milliseconds, and after this update, a second iteration did not yield any changes larger than a millisecond. The aligned curves are shown in Figure 7.13.

As part of a technique that they call *self-modelling nonlinear regression*, which attempts to estimate both parametric and nonparametric components of variation among several curves, Kneip and Gasser (1988) use linear

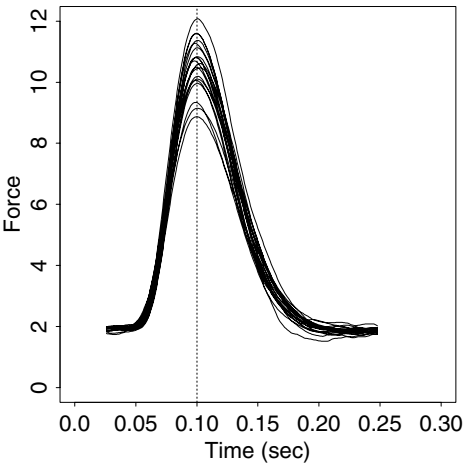

Figure 7.13. The pinch force curves aligned by minimizing the Procrustes criterion `REGSSE`.

transformations of t, that is both shift and scale changes. Kneip and Engel (1995) use such shift-scale transformations to identify "shape invariant features" of curves, which remain unaltered by these changes in t.

7.10 Further reading and notes

The classic paper on the estimation of time warping functions is Sakoe and Chiba (1978), who used dynamic programming to estimate the warping function in a context where there was no need for the warping function to be smooth.

Landmark registration has been studied in depth by Kneip and Gasser (1992) and Gasser and Kneip (1995), who refer to a landmark as a *structural feature*, its location as a *structural point*, to the distribution of landmark locations along the t axis as *structural intensity*, and to the process of averaging a set of curves after registration as *structural averaging*. Their papers contain various technical details on the asymptotic behavior of landmark estimates and warping functions estimated from them. Their papers on growth curves (Gasser et al., 1990, 1991a,b) are applications of this process. Another source of much information on the study of landmarks and their use in registration is Bookstein (1991).

Ramsay (1996b) and Ramsay and Li (1996) developed the fitting of a general and flexible family of warping functions h_i making use of a regular-

ization technique. Their work used a piecewise linear basis for function W in order to avoid numerical integration, but our subsequent work has found numerical integration to be easy to apply here as well as elsewhere, and consequently W may now be expanded in terms of any basis. Kneip, Li, MacGibbon and Ramsay (2000) developed a method that is rather analogous to local polynomial smoothing for identifying warping functions that register a sample of curves.

Wang and Gasser (1997, 1998, 1999) and Gervini and Gasser (2004) have evolved registration technology that does not use landmarks in a number of useful ways, and consider some important theoretical issues. Liu and Müller (2004) advanced their theoretical framework by discussing curve registration in the context of a model for random or stochastic functions where time is itself transformed in a random manner. They propose the operation of taking a *functional convex sum* as a way of computing convex sums of unregistered functions. This operation defines a type of mean that preserves the locations and shapes of features. See also Rønn (2001) for a model-based approach to shift registration.

The functional two-sample functional testing problem considered by Munoz, Maldonado, Staniswalis, Irwin and Byers (2002) uses landmark registration of some image density curves as a pre-processing step.

8

Principal components analysis for functional data

8.1 Introduction

For many reasons, principal components analysis (PCA) of functional data is a key technique to consider. First, our own experience is that, after the preliminary steps of registering and displaying the data, the user wants to explore that data to see the features characterizing typical functions. Some of these features are expected to be there, for example the sinusoidal nature of temperature curves, but other aspects may be surprising. Some indication of the complexity of the data is also required, in the sense of how many types of curves and characteristics are to be found. Principal components analysis serves these ends admirably, and it is perhaps also for these reasons that it was the first method to be considered in the early literature on FDA.

Just as for the corresponding matrices in the classical multivariate case, the variance-covariance and correlation functions can be difficult to interpret, and do not always give a fully comprehensible presentation of the structure of the variability in the observed data directly. The same is true, of course, for variance-covariance and correlation matrices in classical multivariate analysis. A principal components analysis provides a way of looking at covariance structure that can be much more informative and can complement, or even replace altogether, a direct examination of the variance-covariance function.

PCA also offers an opportunity to consider some issues that reappear in subsequent chapters. For example, we consider immediately how PCA is

defined by the notion of a linear combination of function values, and why this notion, at the heart of most of multivariate data analysis, requires some care in a functional context. A second issue is that of *regularization*; for many data sets, PCA of functional data is more revealing if some type of smoothness is required of the principal components themselves. We consider this topic in detail in Chapter 9.

8.2 Defining functional PCA

8.2.1 PCA for multivariate data

The central concept exploited over and over again in multivariate statistics is that of taking a linear combination of variable values,

$$f_i = \sum_{j=1}^{p} \beta_j x_{ij},\ i = 1, \ldots, N, \tag{8.1}$$

where β_j is a weighting coefficient applied to the observed values x_{ij} of the jth variable. We can express (8.1) as

$$f_i = \beta' x_i,\ i = 1, \ldots, N, \tag{8.2}$$

where β is the vector $(\beta_1, \ldots, \beta_p)'$ and x_i is the vector $(x_{i1}, \ldots, x_{ip})'$.

In the multivariate situation, we choose the weights so as to highlight or display types of variation that are very strongly represented in the data. Principal components analysis can be defined in terms of the following stepwise procedure, which defines sets of normalized weights that maximize variation in the f_i's:

1. Find the weight vector $\boldsymbol{\xi}_1 = (\xi_{11}, \ldots, \xi_{p1})'$ for which the linear combination values
$$f_{i1} = \sum_j \xi_{j1} x_{ij} = \boldsymbol{\xi}_1' x_i$$
have the largest possible mean square $N^{-1} \sum_i f_{i1}^2$ subject to the constraint
$$\sum_j \xi_{j1}^2 = \|\boldsymbol{\xi}_1\|^2 = 1.$$

2. Carry out second and subsequent steps, possibly up to a limit of the number of variables p. On the mth step, compute a new weight vector $\boldsymbol{\xi}_m$ with components ξ_{jm} and new values $f_{im} = \boldsymbol{\xi}_m' x_i$. Thus, the values f_{im} have maximum mean square, subject to the constraint $\|\boldsymbol{\xi}_m\|^2 = 1$ and the $m - 1$ additional constraint(s)
$$\sum_j \xi_{jk} \xi_{jm} = \boldsymbol{\xi}_k' \boldsymbol{\xi}_m = 0,\ k < m.$$

The motivation for the first step is that by maximizing the mean square, we are identifying the strongest and most important mode of variation in the variables. The unit sum of squares constraint on the weights is essential to make the problem well defined; without it, the mean squares of the linear combination values could be made arbitrarily large. On second and subsequent steps, we seek the most important modes of variation again, but require the weights defining them to be orthogonal to those identified previously, so that they are indicating something new. Of course, the amount of variation measured in terms of $N^{-1}\sum_i f_{im}^2$ will decline on each step. At some point, usually well short of the maximum index p, we expect to lose interest in modes of variation thus defined.

The definition of principal components analysis does not actually specify the weights uniquely; for example, it is always possible to change the signs of all the values in any vector $\boldsymbol{\xi}_m$ without changing the value of the variance that it defines.

The values of the linear combinations f_{im} are called *principal component scores* and are often of great help in describing what these important components of variation mean in terms of the characteristics of specific cases or replicates.

To be sure, the mean is a very important aspect of the data, but we already have an easy technique for identifying it. Therefore, we usually subtract the mean for each variable from corresponding variable values before doing PCA. When this is done, maximizing the mean square of the principal component scores corresponds to maximizing their sample variance.

8.2.2 Defining PCA for functional data

How does PCA work in the functional context? The counterparts of variable values are function values $x_i(s)$, so that the discrete index j in the multivariate context has been replaced by the continuous index s. When we were considering vectors, the appropriate way of combining a weight vector β with a data vector x was to calculate the inner product

$$\beta' x = \sum_j \beta_j x_j.$$

When β and x are functions $\beta(s)$ and $x(s)$, summations over j are replaced by integrations over s to define the inner product

$$\int \beta x = \int \beta(s)x(s)\,ds. \tag{8.3}$$

Within the principal components analysis, the weights β_j now become functions with values $\beta_j(s)$. Using the notation (8.3), the principal

component scores corresponding to weight β are now

$$f_i = \int \beta x_i = \int \beta(s) x_i(s)\, ds. \tag{8.4}$$

For the rest of our discussion, we will often use the short form $\int \beta x_i$ for integrals in order to minimize notational clutter.

In the first functional PCA step, the weight function $\xi_1(s)$ is chosen to maximize $N^{-1}\sum_i f_{i1}^2 = N^{-1}\sum_i (\int \xi_1 x_i)^2$ subject to the continuous analogue $\int \xi_1(s)^2 ds = 1$ of the unit sum of squares constraint. This time, the notation $\|\xi_1\|^2$ is used to mean the squared norm $\int \xi_1(s)^2 ds = \int \xi_1^2$ of the *function* ξ_1.

Postponing computational details until Section 8.4, now consider as an illustration in the upper left panel in Figure 8.1. This displays the weight function ξ_1 for the Canadian temperature data after the mean across all 35 weather stations has been removed from each station's monthly temperature record. Although ξ_1 is positive throughout the year, the weight placed on the winter temperatures is about four times that placed on summer temperatures. This means that the greatest variability between weather stations will be found by heavily weighting winter temperatures, with only a light contribution from the summer months; Canadian weather is most variable in the wintertime, in short. Moreover, the percentage 89.3% at the top of the panel indicates that this type of variation strongly dominates all other types of variation. Weather stations for which the score f_{i1} is high will have much warmer than average winters combined with warm summers, and the two highest scores are in fact assigned to Vancouver and Victoria on the Pacific Coast. To no one's surprise, the largest negative score goes to Resolute in the High Arctic.

As for multivariate PCA, the weight function ξ_m is also required to satisfy the orthogonality constraint(s) $\int \xi_k \xi_m = 0,\ k < m$ on subsequent steps. Each weight function has the task of defining the most important mode of variation in the curves subject to each mode being orthogonal to all modes defined on previous steps. Note again that the weight functions are defined only to within a sign change.

The weight function ξ_2 for the temperature data is displayed in the upper right panel of Figure 8.1. Because it must be orthogonal to ξ_1, we cannot expect that it will define a mode of variation in the temperature functions that will be as important as the first. In fact, this second mode accounts for only 8.3% of the total variation, and consists of a positive contribution for the winter months and a negative contribution for the summer months, therefore corresponding to a measure of uniformity of temperature through the year. On this component, one of the highest scores f_{i2} goes to Prince Rupert, also on the Pacific coast, for which there is comparatively low discrepancy between winter and summer. Prairie stations such as Winnipeg, on the other hand, have hot summers and very cold winters, and receive large negative second component scores.

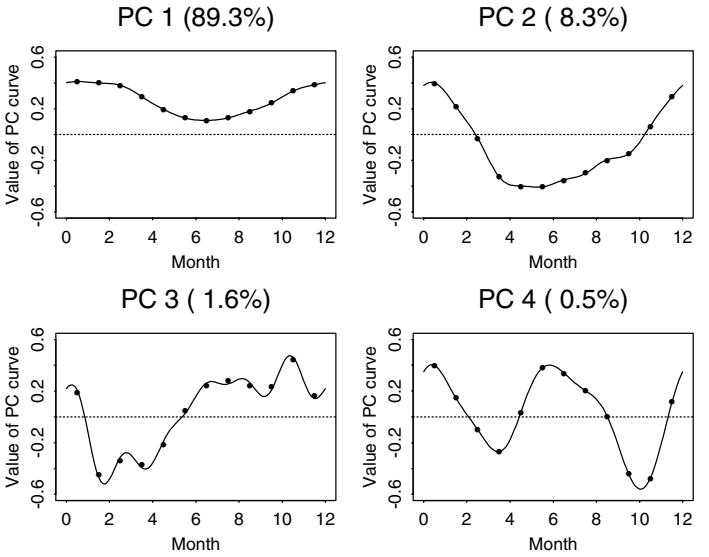

Figure 8.1. The first four principal component curves of the Canadian temperature data estimated by two techniques. The points are the estimates from the discretization approach, and the curves are the estimates from the expansion of the data in terms of a 12-term Fourier series. The percentages indicate the amount of total variation accounted for by each principal component.

The third and fourth components account for small proportions of the variation, since they are required to be orthogonal to the first two as well as to each other. At this point they are difficult to interpret, but we look at techniques for understanding them in Section 8.3.

Displays such as Figure 8.1 can remind one of the diagrams of modes of vibration in a string fixed at both ends always found in introductory physics texts. The first and dominant type is simple in structure and resembles a single cycle of a sine wave. Subdominant or higher order components are also roughly sinusoidal, but with more and more cycles. With this analogy in mind, we find the term *harmonics* evocative in referring to principal components of variation in curves in general.

8.2.3 Defining an optimal empirical orthonormal basis

There are several other ways to motivate PCA, and one is to define the following problem: We want to find a set of exactly K orthonormal functions ξ_m so that the expansion of each curve in terms of these basis functions approximates the curve as closely as possible. Since these basis functions

are orthonormal, it follows that the expansion will be of the form

$$\hat{x}_i(t) = \sum_{k=1}^{K} f_{ik}\xi_k(t),$$

where f_{ik} is the principal component value $\int x_i\xi_k$. As a fitting criterion for an individual curve, consider the integrated squared error

$$\|x_i - \hat{x}_i\|^2 = \int [x(s) - \hat{x}(s)]^2\, ds$$

and as a global measure of approximation,

$$\texttt{PCASSE} = \sum_{i=1}^{N} \|x_i - \hat{x}_i\|^2. \tag{8.5}$$

The problem is then, more precisely, what choice of basis will minimize the error criterion (8.5)?

The answer, it turns out, is precisely the same set of principal component weight functions that maximize variance components as defined above. For this reason, these functions ξ_m are referred to in some fields as *empirical orthonormal functions*, because they are determined by the data they are used to expand.

8.2.4 PCA and eigenanalysis

In this section, we investigate another characterization of PCA, in terms of the eigenanalysis of the variance-covariance function or operator.

Assume for this section that our observed values, x_{ij} in the multivariate context and $x_i(t)$ in the functional situation, result from subtracting the mean variable or function values, so that their sample means $N^{-1}\sum_i x_{ij}$, or cross-sectional means $N^{-1}\sum_i x_i(t)$, respectively, are zero.

Texts on multivariate data analysis tend to define principal components analysis as the task of finding the eigenvalues and eigenvectors of the covariance or correlation matrix. The logic for this is as follows. Let the $N \times p$ matrix $\mathbf{X}$ contain the values x_{ij} and the vector $\boldsymbol{\xi}$ of length p contain the weights for a linear combination. Then the mean square criterion for finding the first principal component weight vector can be written as

$$\max_{\boldsymbol{\xi}'\boldsymbol{\xi}=1} N^{-1}\boldsymbol{\xi}'\mathbf{X}'\mathbf{X}\boldsymbol{\xi} \tag{8.6}$$

since the vector of principal component scores f_i can be written as $\mathbf{X}\boldsymbol{\xi}$.

Use the $p \times p$ matrix $\mathbf{V}$ to indicate the sample variance-covariance matrix $\mathbf{V} = N^{-1}\mathbf{X}'\mathbf{X}$. (One may prefer to use a divisor of $N-1$ to N since the means have been estimated, but it makes no essential difference to the principal components analysis.) The criterion (8.6) can now be expressed

as

$$\max_{\boldsymbol{\xi}'\boldsymbol{\xi}=1} \boldsymbol{\xi}'\mathbf{V}\boldsymbol{\xi}.$$

As explained in Section A.5, this maximization problem is now solved by finding the solution with largest eigenvalue ρ of the eigenvector problem or *eigenequation*

$$\mathbf{V}\boldsymbol{\xi} = \rho\boldsymbol{\xi}. \tag{8.7}$$

There is a sequence of different eigenvalue-eigenvector pairs $(\rho_j, \boldsymbol{\xi}_j)$ satisfying this equation, and the eigenvectors $\boldsymbol{\xi}_j$ are orthogonal. Because the mean of each column of $\mathbf{X}$ is usually subtracted from all values in that column as a preliminary to principal components analysis, the rank of $\mathbf{X}$ is $N-1$ at most, and hence the $p \times p$ matrix $\mathbf{V}$ has, at most, $\min\{p, N-1\}$ nonzero eigenvalues ρ_j. For each j, the eigenvector $\boldsymbol{\xi}_j$ satisfies the maximization problem (8.6) subject to the additional constraint of being orthogonal to all the eigenvectors $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_{j-1}$ found so far. This is precisely what was required of the principal components in the second step laid out in Section 8.2.1. Therefore, as we have defined it, the multivariate PCA problem is equivalent to the algebraic and numerical problem of solving the eigenequation (8.7). Of course, there are standard computer algorithms for doing this.

Now consider the functional version of PCA. Define the covariance function $v(s,t)$ by

$$v(s,t) = N^{-1}\sum_{i=1}^{N} x_i(s)x_i(t). \tag{8.8}$$

Again, note that we may prefer to use $N-1$ to define the variance-covariance function v; nothing discussed here changes in any essential way.

The more general results set out in Section A.5.2 can be applied, to find the principal component weight functions $\xi_j(s)$. Each of these satisfies the equation

$$\int v(s,t)\xi(t)\,dt = \rho\xi(s) \tag{8.9}$$

for an appropriate eigenvalue ρ. The left side of (8.9) is an *integral transform* V of the weight function ξ defined by

$$V\xi = \int v(\cdot,t)\xi(t)\,dt. \tag{8.10}$$

This integral transform is called the *covariance operator* V. Therefore we may also express the eigenequation directly as

$$V\xi = \rho\xi, \tag{8.11}$$

where ξ is now an eigenfunction rather than an eigenvector. By suitable choice of notation, the equation (8.11) for functional PCA now looks the same as the eigenequation (8.7) relevant to conventional PCA.

There is an important difference between the multivariate and functional eigenanalysis problems, concerning the maximum number of different eigenvalue-eigenfunction pairs. The counterpart of the number of variables p in the multivariate case is the number of function values in the functional case, and thus infinity. However, provided the functions x_i are not linearly dependent, the operator V will have rank $N-1$, and there will be only $N-1$ nonzero eigenvalues.

To summarize, in this section we find that principal components analysis is defined as the search for a set of mutually orthogonal and normalized weight functions ξ_m. Functional PCA can be expressed as the problem of the eigenanalysis of the covariance operator V. By suitable choice of notation, the formal steps to be carried out are the same, whether the data are multivariate or functional.

In Section 8.4 we discuss practical methods for actually computing the eigenfunctions ξ_m, but first we consider some aspects of the display of principal components once they have been found.

8.3 Visualizing the results

The fact that interpreting the components is not always an entirely straightforward matter is common to most functional PCA problems. We now consider some techniques that may aid their interpretation.

8.3.1 Plotting components as perturbations of the mean

A method found to be helpful is to examine plots of the overall mean function and the functions obtained by adding and subtracting a suitable multiple of the principal component function in question. Figure 8.2 shows such a plot for the temperature data. In each case, the solid curve is the overall mean temperature, and the dotted and dashed curves show the effects of adding and subtracting a multiple of each principal component curve. This considerably clarifies the effects of the first two components. We can now see that the third principal component corresponds to a time shift effect combined with an overall increase in temperature and in range between winter and summer. The fourth corresponds to an effect whereby the onset of spring is later and autumn ends earlier.

In constructing this plot, it is necessary to choose which multiple of the principal component function to use. Define a constant C to be the root-mean-square difference between $\hat{\mu}$ and its overall time average,

$$C^2 = T^{-1}\|\hat{\mu} - \bar{\mu}\|^2, \tag{8.12}$$

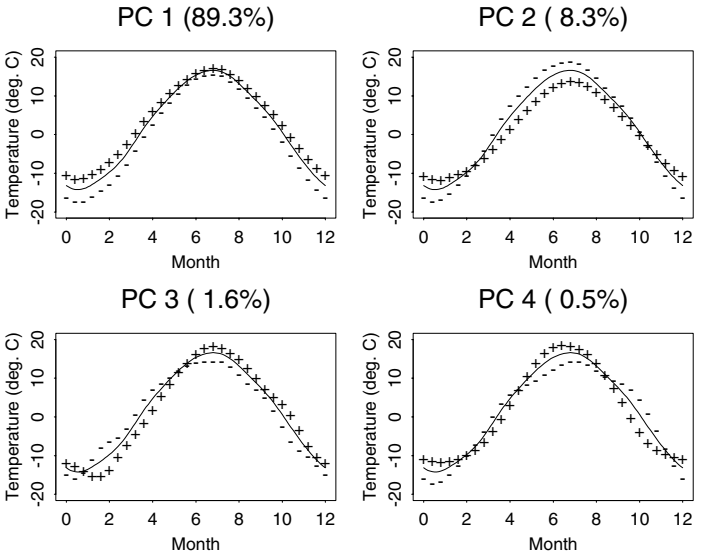

Figure 8.2. The mean temperature curves and the effects of adding (+) and subtracting (−) a suitable multiple of each PC curve.

where

$$\bar{\mu} = T^{-1} \int \hat{\mu}(t)\, dt.$$

It is then appropriate to plot $\hat{\mu}$ and $\hat{\mu} \pm 0.2C\hat{\gamma}_j$, where we have chosen the constant 0.2 to give easily interpretable results. Depending on the overall behavior of $\hat{\mu}$, it may be helpful to adjust the value 0.2 subjectively. But for ease of comparison between the various modes of variability, it is best to use the same constant for all the principal component functions plotted in any particular case.

In Figure 8.3, we consider the hip angles observed during the gait of 39 children, as plotted in Figure 1.8. The angles for a single cycle are shown, along with the results of a functional PCA of these data. The effect of the first principal component of variation is approximately to add or subtract a constant to the angle throughout the gait cycle. The second component corresponds roughly to a time shift effect, which is not constant through the cycle. The third component corresponds to a variation in the overall amplitude of the angle traced out during the cycle.

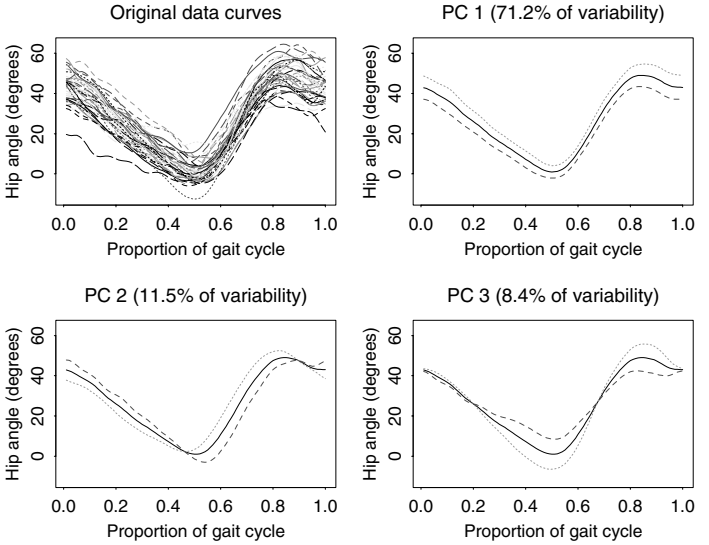

Figure 8.3. The hip angle observed in the gait cycles of 39 children, and the effect on the overall mean of adding and subtracting a suitable multiple of each of the first three principal component functions.

8.3.2 Plotting principal component scores

An important aspect of PCA is the examination of the scores f_{im} of each curve on each component. In Figure 8.4, each weather station is identified by a four-letter abbreviation of its name given in Table 8.1. The strings are positioned roughly according to the scores on the first two principal components; some positions have been adjusted slightly to improve legibility. The West Coast stations Vancouver (VANC), Victoria (VICT) and Prince Rupert (PRUP) are in the upper right corner because they have warmer winters than most stations (high on PC 1) and less summer-winter temperature variation (high on PC 2). Resolute (RESO), on the other hand, has an extremely cold winter, but does resemble the Pacific weather stations in having less summer/winter variation than some Arctic cousins, such as Inuvik (INUV).

8.3.3 Rotating principal components

In Section 8.2 we observed that the weight functions ξ_m can be viewed as defining an orthonormal set of K functions for expanding the curves to minimize a summed integrated squared error criterion (8.5). For the

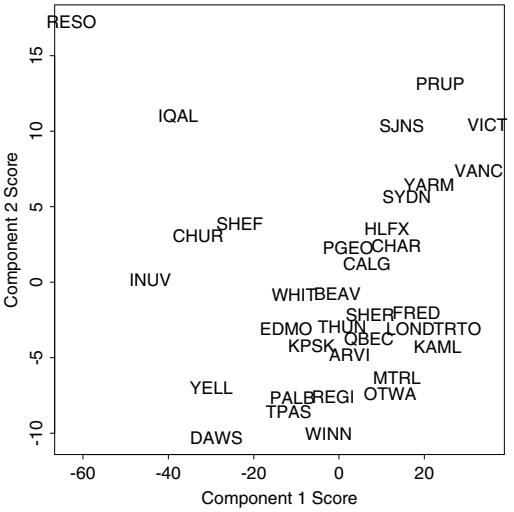

Figure 8.4. The scores of the weather stations on the first two principal components of temperature variation. The location of each weather station is shown by the four-letter abbreviation of its name assigned in Table 8.1.

Table 8.1. The Canadian Weather Stations

Arvida, Que.	Kapuskasing, Ont.	St. John's, Nfld
Beaverlodge, B.C.	London, Ont.	Sydney, N.S.
Calgary, Alta.	Montreal, Que.	The Pas, Man.
Charlottetown, P.E.I.	Ottawa, Ont.	Thunder Bay, Ont.
Churchill, Man.	Prince Albert, Sask.	Toronto, Ont.
Dawson, Yukon	Prince George, B.C.	Vancouver, B.C.
Edmonton, Alta.	Prince Rupert, B.C.	Victoria, B.C.
Fredericton, N.B.	Quebec City, Que.	Whitehorse, Yukon
Halifax, N.S.	Regina, Sask.	Winnipeg, Man.
Inuvik, N.W.T.	Resolute, N.W.T.	Yarmouth, N.S.
Iqualuit, N.W.T.	Schefferville, Que.	Yellowknife, N.W.T.
Kamloops, B.C.	Sherbrooke, Que.	

temperature data, for example, no set of four orthonormal functions will do a better job of approximating the curves than those displayed in Figure 8.1.

This does not mean, however, that there aren't other orthonormal sets that will do just as well. In fact, if we now use ξ to refer to the vector-valued

function $(\xi_1, \ldots, \xi_K)'$, then an equally good orthonormal set is defined by

$$\psi = \mathbf{T}\xi, \tag{8.13}$$

where $\mathbf{T}$ is any orthonormal matrix of order K, meaning that $\mathbf{T}'\mathbf{T} = \mathbf{T}\mathbf{T}' = \mathbf{I}$. From a geometrical perspective, the vector of functions ψ is a rigid rotation of ξ. Of course, after rotation, we can no longer expect that ψ_1 will define the largest component of variation. But the point is that the orthonormal basis functions $\psi_1, \ldots, \psi_K$ are just as effective at approximating the original curves in K dimensions as their unrotated counterparts.

Can we find some rotated functions that are perhaps a little easier to interpret? Here again, we can borrow a tool that has been invaluable in multivariate analysis, VARIMAX rotation. Let $\mathbf{B}$ be a $K \times n$ matrix representing the first K principal component functions $\xi_1, \ldots, \xi_K$. For the moment, suppose that $\mathbf{B}$ has, as row m, the values $\xi_m(t_1), \ldots, \xi_m(t_n)$ for n equally spaced argument values in the interval $\mathcal{T}$. The corresponding matrix $\mathbf{A}$ of values of the rotated basis functions $\psi = \mathbf{T}\xi$ will be given by

$$\mathbf{A} = \mathbf{T}\mathbf{B}. \tag{8.14}$$

The VARIMAX strategy for choosing the orthonormal rotation matrix $\mathbf{T}$ is to maximize the variation in the values a_{mj}^2 strung out as a single vector. Since $\mathbf{T}$ is a rotation matrix, the overall sum of these squared values will remain the same no matter what rotation we perform. In algebraic terms,

$$\sum_m \sum_j a_{mj}^2 = \operatorname{trace} \mathbf{A}'\mathbf{A} = \operatorname{trace} \mathbf{B}'\mathbf{T}'\mathbf{T}\mathbf{B} = \operatorname{trace} \mathbf{B}'\mathbf{B}.$$

Therefore, maximizing the variance of the a_{mj}^2 can happen only if these values tend either to be relatively large or relatively near zero. The values a_{mj} themselves are encouraged to be either strongly positive, near zero, or strongly negative; in-between values are suppressed. This clustering of information tends to make the components of variation easier to interpret.

There are fast and stable computational techniques for computing the rotation matrix $\mathbf{T}$ that maximizes the VARIMAX criterion. A C function for computing the VARIMAX rotation can be found through the book's world-wide web page described in Section 1.9.

Figure 8.5 displays the VARIMAX rotation of the four principal components for the temperature data. There, $n = 12$ equally spaced time points t_j were used, and the variance of the squared values $\psi_m^2(t_j)$ was maximized with respect to $\mathbf{T}$. The resulting rotated functions ψ_m, along with the percentages of variances that they account for, are now quite different.

Collectively, the rotated functions ψ_m still account for a total of 99.7% of the variation, but they divide this variation in different proportions. The VARIMAX rotation has suppressed medium-sized values of ψ_m while preserving orthonormality. (Note that the rotated component scores are no longer uncorrelated; however, the sum of their variances is still the same, because $\mathbf{T}$ is a rotation matrix, and so they may still be considered to

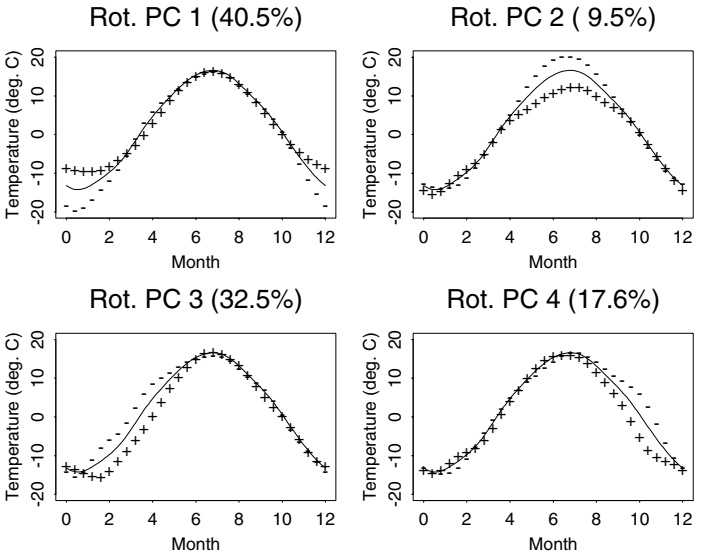

Figure 8.5. Weight functions rotated by applying the VARIMAX rotation criterion to weight function values, and plotted as positive and negative perturbations of the mean function.

partition the variability in the original data.) The result is four functions that account for local variation in the winter, summer, spring and autumn, respectively. Not only are these functions much easier to interpret, but we see something new: although winter variation remains extremely important, now spring variation is clearly almost as important, about twice as important as autumn variation and over three times as important as summer variation.

Another way of using the VARIMAX idea is to let $\mathbf{B}$ contain the coefficients for the expansion of each ξ_m in terms of a basis ϕ of n functions. Thus we rotate the coefficients of the basis expansion of each ξ_m rather than rotating the values of the ξ_m themselves. Figure 8.6 shows the results using a Fourier series expansion of the principal components. The results are much more similar to the original principal components displayed in Figure 8.2. The main difference is in the first two components. The first rotated component function in Figure 8.6 is much more constant than the original first principal component, and corresponds almost entirely to a constant temperature effect throughout the year. The general shape of the second component is not changed very much, but it accounts for more of the variability, having essentially taken on part of the variability in the first unrotated component. Because the first component originally accounted for such a large proportion, 89.3%, of the variability, it is not surprising that a

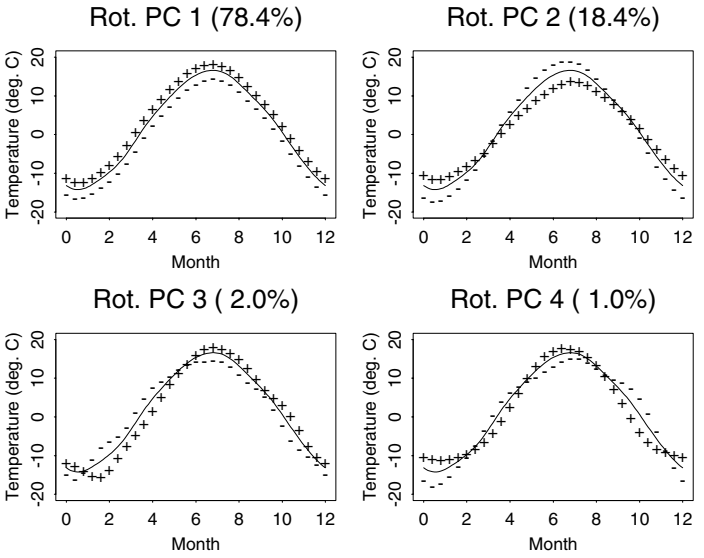

Figure 8.6. Weight functions rotated by applying the VARIMAX rotation criterion to weight function coefficients, and plotted as positive and negative perturbations of the mean function.

fairly small change in the shape of the second component results in moving about 10% of the total variability from the first to the second component. The third and fourth components are not enormously affected by the VARIMAX rotation in the Fourier domain.

By no means is the VARIMAX criterion the only rotation criterion available. References on factor analysis and multivariate statistics such as Basilevsky (1994), Johnson and Wichern (1988), Mulaik (1972) and Seber (1984) offer a number of other possibilities. Even from the relatively brief discussion in this section, it is clear that much research remains to be done on rotation schemes tailored more directly to the functional context.

8.4 Computational methods for functional PCA

Now suppose that we have a set of N curves x_i, and that preliminary steps such as curve registration and the possible subtraction of the mean curve from each (curve centering) have been completed. Let $v(s,t)$ be the sample covariance function of the observed data. In this section, we consider possible strategies for approaching the eigenanalysis problem in (8.9). In all cases, we convert the continuous functional eigenanalysis problem to an approximately equivalent matrix eigenanalysis task.

8.4.1 Discretizing the functions

A simple approach is to discretize the observed functions x_i to a fine grid of n equally spaced values s_j that span the interval $\mathcal{T}$. This yields an $N \times n$ data matrix $\mathbf{X}$ that can be fed into a standard multivariate principal components analysis program such as the S-PLUS routine `prcomp`. This produces eigenvalues and eigenvectors satisfying

$$\mathbf{V}\mathbf{u} = \lambda\mathbf{u} \tag{8.15}$$

for n-vectors $\mathbf{u}$.

Notice that we may well have n much larger than N. Rather than working with the $n \times n$ matrix $\mathbf{V}$, one possible approach to finding the solutions of the eigenequation (8.15) is to work in terms of the SVD $\mathbf{UDW}'$ of $\mathbf{X}$. The variance matrix satisfies $N\mathbf{V} = \mathbf{WD}^2\mathbf{W}'$, and hence the nonzero eigenvalues of $\mathbf{V}$ are the squares of the singular values of $\mathbf{X}$, and the corresponding eigenvectors are the columns of $\mathbf{U}$. If we use a standard PCA package, these steps, or corresponding ones, will be carried out automatically in any case.

How do we transform the vector principal components back into functional terms? The sample variance-covariance matrix $\mathbf{V} = N^{-1}\mathbf{X}'\mathbf{X}$ will have elements $v(s_j, s_k)$ where $v(s,t)$ is the sample covariance function. Given any function ξ, let $\tilde{\boldsymbol{\xi}}$ be the n-vector of values $\xi(s_j)$. Let $w = T/n$ where T is the length of the interval $\mathcal{T}$. Then, for each s_j,

$$V\xi(s_j) = \int v(s_j, s)\xi(s)ds \approx w \sum v(s_j, s_k)\tilde{\boldsymbol{\xi}}_k,$$

so the functional eigenequation $V\xi = \rho\xi$ has the approximate discrete form

$$w\mathbf{V}\tilde{\boldsymbol{\xi}} = \rho\tilde{\boldsymbol{\xi}}.$$

The solutions of this equation will correspond to those of (8.15), with eigenvalues $\rho = w\lambda$. The discrete approximation to the normalization $\int \xi(s)^2 ds = 1$ is $w\|\tilde{\boldsymbol{\xi}}\|^2 = 1$, so that we set $\tilde{\boldsymbol{\xi}} = w^{-1/2}\mathbf{u}$ if $\mathbf{u}$ is a normalized eigenvector of $\mathbf{V}$. Finally, to obtain an approximate eigenfunction ξ from the discrete values $\tilde{\boldsymbol{\xi}}$, we can use any convenient interpolation method. If the discretization values s_j are closely spaced, the choice of interpolation method will not usually have a great effect.

The discretization approach is the earliest approach to functional principal components analysis, used by Rao (1958, 1987) and Tucker (1958), who applied multivariate principal components analysis without modification to observed function values. We discuss the idea of discretizing the integral in more detail in Section 8.4.3, but first we consider an alternative approach that makes use of basis expansions.

8.4.2 Basis function expansion of the functions

One way of reducing the eigenequation (8.9) to discrete or matrix form is to express each function x_i as a linear combination of known basis functions

ϕ_k. The number K of basis functions used depends on many considerations: how many discrete sampling points n were in the original data, whether some level of smoothing was to be imposed by using $K < n$, how efficient or powerful the basis functions are in reproducing the behavior of the original functions, and so forth. For the monthly temperature data, for example, it would be logical to use a Fourier series basis orthonormal over the interval [0,12], with $K = 12$ the maximum possible dimension of the basis for the monthly temperature data, because only 12 sampling points are available per curve. Actually, for these data, a value of K as small as 7 would capture most of the interesting variation in the original data, but there is little point in reducing K below the value of 12.

Now suppose that each function has basis expansion

$$x_i(t) = \sum_{k=1}^{K} c_{ik}\phi_k(t). \tag{8.16}$$

We may write this more compactly by defining the vector-valued function $\mathbf{x}$ to have components $x_1, \ldots, x_N$, and the vector-valued function $\boldsymbol{\phi}$ to have components $\phi_1, \ldots, \phi_K$. We may then express the simultaneous expansion of all N curves as

$$\mathbf{x} = \mathbf{C}\boldsymbol{\phi},$$

where the coefficient matrix $\mathbf{C}$ is $N \times K$. In matrix terms the variance-covariance function is

$$v(s,t) = N^{-1}\boldsymbol{\phi}(s)'\mathbf{C}'\mathbf{C}\boldsymbol{\phi}(t),$$

remembering that $\boldsymbol{\phi}(s)'$ denotes the transpose of the vector $\boldsymbol{\phi}(s)$ and has nothing to do with differentiation.

Define the order K symmetric matrix $\mathbf{W}$ to have entries

$$w_{k_1,k_2} = \int \phi_{k_1}\phi_{k_2}$$

or $\mathbf{W} = \int \boldsymbol{\phi}\boldsymbol{\phi}'$. For some choices of bases, $\mathbf{W}$ will be readily available. For example, for the orthonormal Fourier series that we might use for the temperature data, $\mathbf{W} = \mathbf{I}$, the order K identity matrix. In other cases, we may have to resort to numerical integration to evaluate $\mathbf{W}$.

Now suppose that an eigenfunction ξ for the eigenequation (8.9) has an expansion

$$\xi(s) = \sum_{k=1}^{K} b_k\phi_k(s)$$

or, in matrix notation, $\xi(s) = \boldsymbol{\phi}(s)'\mathbf{b}$. This yields

$$\begin{aligned}\int v(s,t)\xi(t)\,dt &= \int N^{-1}\boldsymbol{\phi}(s)'\mathbf{C}'\mathbf{C}\boldsymbol{\phi}(t)\boldsymbol{\phi}(t)'\mathbf{b}\,dt \\ &= \boldsymbol{\phi}(s)'N^{-1}\mathbf{C}'\mathbf{C}\mathbf{W}\mathbf{b}.\end{aligned}$$

Therefore the eigenequation (8.9) can be expressed as

$$\boldsymbol{\phi}(s)'N^{-1}\mathbf{C}'\mathbf{C}\mathbf{W}\mathbf{b} = \rho\boldsymbol{\phi}(s)'\mathbf{b}.$$

Since this equation must hold for all s, this implies the purely matrix equation

$$N^{-1}\mathbf{C}'\mathbf{C}\mathbf{W}\mathbf{b} = \rho\mathbf{b}.$$

But note that $\|\xi\| = 1$ implies that $\mathbf{b}'\mathbf{W}\mathbf{b} = 1$ and, similarly, two functions ξ_1 and ξ_2 will be orthogonal if and only if the corresponding vectors of coefficients satisfy $\mathbf{b}_1'\mathbf{W}\mathbf{b}_2 = 0$. To get the required principal components, we define $\mathbf{u} = \mathbf{W}^{1/2}\mathbf{b}$, solve the equivalent symmetric eigenvalue problem

$$N^{-1}\mathbf{W}^{1/2}\mathbf{C}'\mathbf{C}\mathbf{W}^{1/2}\mathbf{u} = \rho\mathbf{u}$$

and compute $\mathbf{b} = \mathbf{W}^{-1/2}\mathbf{u}$ for each eigenvector.

Two special cases deserve particular attention. As already mentioned, if the basis is orthonormal, meaning that $\mathbf{W} = \mathbf{I}$, the functional PCA problem finally reduces to the standard multivariate PCA of the coefficient array $\mathbf{C}$, and we need only carry out the eigenanalysis of the order K symmetric array $N^{-1}\mathbf{C}'\mathbf{C}$.

As a rather different special case, particularly appropriate if the number of observed functions is not enormous, we may also view the observed functions x_i as their *own* basis expansions. In other words, there are N basis functions, and they happen to be the observed functions. This implies, of course, that $\mathbf{C} = \mathbf{I}$, and now the problem becomes one of the eigenanalysis of the symmetric matrix $N^{-1}\mathbf{W}$, which has entries

$$w_{ij} = \int x_i x_j.$$

As a rule, these entries will have to be computed by some quadrature technique.

In every case, the maximum number of eigenfunctions that can in principle be computed by the basis function approach is K, the dimension of the basis. However, if the basis expansions have involved any approximation of the observed functions, then it is not advisable to use a basis expansion to K terms to calculate more than a fairly small proportion of K eigenfunctions.

The results of both strategies that we have discussed are illustrated in Figure 8.1, which shows the first four estimated eigenfunctions ξ_m of the centered temperature functions

$$x_i = \texttt{Temp}_i - \frac{1}{35}\sum_j \texttt{Temp}_j.$$

The smooth curves give the estimated eigenfunctions using the complete 12-term Fourier series expansion. For comparison purposes, the results of applying the discretization approach to the data are also displayed as points

indicating the values of the eigenvectors. There is little discrepancy between the two sets of results. The proportions of variances for the basis function analysis turn out to be identical to those computed for the discretization approach. No attempt has been made to interpolate the discretized values to give continuous eigenfunctions, but if the Fourier series interpolation method were used, the results would be identical to the results obtained by the basis method; this is a consequence of special properties of Fourier series.

8.4.3 More general numerical quadrature

The eigenequation (8.9) involves the integral $\int x_i(s)\xi(s)\,ds$, and the discretization strategy is to approximate this integral by a sum of discrete values. Most schemes for numerical integration or quadrature (Stoer and Bulirsch, 2002, is a good reference) involve an approximation of the form

$$\int f(s)\,ds \approx \sum_{j=1}^{n} w_j f(s_j) \tag{8.17}$$

and the method set out in Section 8.4.1 is a fairly crude special case. We restrict our attention to linear quadrature schemes of the form (8.17). There are three aspects of the approximation that can be manipulated to meet various objectives:

- n, the number of discrete argument values s_j
- s_j, the argument values, called *quadrature points*
- w_j, the weights, called *quadrature weights*, attached to each function value in the sum.

A simple example is the *trapezoidal rule*, in which the interval of integration is divided into $n-1$ equal intervals, each of width h. The s_j are the boundaries of the interval with s_1 and s_n the lower and upper limits of integration, respectively, and the approximation is

$$\int f(s)\,ds \approx h[f(s_1)/2 + \sum_{j=2}^{n-1} f(s_j) + f(s_n)/2]. \tag{8.18}$$

Note that the weights w_j are $h/2, h, \ldots, h, h/2$ and that accuracy is controlled simply by the choice of n. The trapezoidal rule has some important advantages: the original raw data are often collected for equally spaced argument values, the weights are trivial, and although the accuracy of the method is modest relative to other more sophisticated schemes, it is often entirely sufficient for the objectives at hand. The method we set out in Section 8.4.1 is similar to the trapezoidal rule, and indeed if we use periodic boundary conditions, the methods are the same, since the values $f(s_n)$ and $f(s_1)$ are identical.

Other techniques, Gaussian quadrature schemes for example, define quadrature weights and points that yield much higher accuracy for fixed n under suitable additional conditions on the integrand. Another class of procedures chooses the quadrature points adaptively to provide more resolution in regions of high integrand curvature; for these to be relevant to the present discussion, we must choose the quadrature points once for all the functions considered in the analysis.

Applying quadrature schemes of the type (8.17) to the operator V in (8.10), yields the discrete approximation

$$V\xi \approx \mathbf{V}\mathbf{W}\tilde{\boldsymbol{\xi}}, \tag{8.19}$$

where, as in Section 8.4.1, the matrix $\mathbf{V}$ contains the values $v(s_j, s_k)$ of the covariance function at the quadrature points, and $\tilde{\boldsymbol{\xi}}$ is an order n vector containing values $\xi(s_j)$. The matrix $\mathbf{W}$ is a diagonal matrix with diagonal values being the quadrature weights w_j.

The approximately equivalent matrix eigenanalysis problem is then

$$\mathbf{V}\mathbf{W}\tilde{\boldsymbol{\xi}} = \rho\tilde{\boldsymbol{\xi}},$$

where the orthonormality requirement is now

$$\tilde{\boldsymbol{\xi}}_m'\mathbf{W}\tilde{\boldsymbol{\xi}}_m = 1 \text{ and } \tilde{\boldsymbol{\xi}}_{m_1}'\mathbf{W}\tilde{\boldsymbol{\xi}}_{m_2} = 0,\ m_1 \neq m_2.$$

Since most quadrature schemes use positive weights, we can put the approximate eigenequation in more standard form, analogous to the calculations carried out in Section 8.4.2:

$$\mathbf{W}^{1/2}\mathbf{V}\mathbf{W}^{1/2}\mathbf{u} = \rho\mathbf{u},$$

where $\mathbf{u} = \mathbf{W}^{1/2}\tilde{\boldsymbol{\xi}}$ and $\mathbf{u}'\mathbf{u} = 1$. Then the whole procedure is as follows:

1. Choose n, the w_j's, and the s_j's.
2. Compute the eigenvalues ρ_m and eigenvectors $\mathbf{u}_m$ of $\mathbf{W}^{1/2}\mathbf{V}\mathbf{W}^{1/2}$.
3. Compute
$$\tilde{\boldsymbol{\xi}}_m = \mathbf{W}^{-1/2}\mathbf{u}_m.$$
4. If needed, use an interpolation technique to convert each vector $\tilde{\boldsymbol{\xi}}_m$ to a function ξ_m.

If the number n of quadrature points is less than the number of curves N, we cannot recover more than n approximate eigenfunctions. However, many applications of PCA require only a small number of the leading eigenfunctions, and any reasonably large n will serve.

To illustrate the application of this discretizing approach, we analyze the acceleration in human growth described in Chapter 1. Each curve consists of 141 equally spaced values of acceleration in height estimated for ages from 14 to 18 years, after spline smoothing and registration by certain

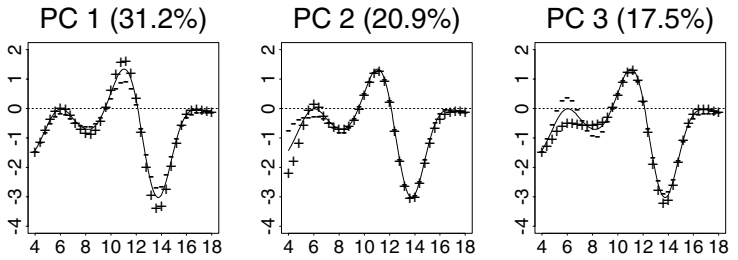

Figure 8.7. The solid curve in each panel is the mean acceleration in height in cm/year^2 for girls in the Zurich growth study. Each principal component is plotted in terms of its effect when added (+) and subtracted (−) from the mean curve.

marker events. Full details of this process can be found in Ramsay, Bock and Gasser (1995). The curves are for 112 girls who took part in the Zurich growth study (Falkner, 1960).

Figure 8.7 shows the first three eigenfunctions or harmonics plotted as perturbations of the mean function. Essentially, the first principal component reflects a general variation in the amplitude of the variation in acceleration that is spread across the entire curve, but is particularly marked during the pubertal growth spurt lasting from 10 to 16 years of age. The second component indicates variation in the size of acceleration only from ages 4 to 6, and the third component, of great interest to growth researchers, shows a variation in intensity of acceleration in the prepubertal period around ages 5 to 9 years.

8.5 Bivariate and multivariate PCA

We often wish to study the simultaneous variation of more than one function. The hip and knee angles described in Chapter 1 are an example; to understand the total system, we want to know how hip and knee angles vary jointly. Similarly, the handwriting data require the study of the simultaneous variation of the X and Y coordinates; there would be little point in studying one coordinate at a time. In both these cases, the two variables being considered are measured relative to the same argument, time in both cases. Furthermore, they are measuring quantities in the same units (degrees in the first case and cm in the second). The discussion in this section is particularly aimed towards problems of this kind.

8.5.1 Defining multivariate functional PCA

For clarity of exposition, we discuss the extension of the PCA idea to deal with bivariate functional data in the specific context of the hip and knee data. Suppose that the observed hip angle curves are $\texttt{Hip}_1, \texttt{Hip}_2, \ldots, \texttt{Hip}_n$ and the observed knee angles are $\texttt{Knee}_1, \texttt{Knee}_2, \ldots, \texttt{Knee}_n$. Let `Hipmn` and `Kneemn` be estimates of the mean functions of the `Hip` and `Knee` processes. Define $v_{\texttt{HH}}$ to be the covariance operator of the $\texttt{Hip}_i$, $v_{\texttt{KK}}$ that of the $\texttt{Knee}_i$, $v_{\texttt{HK}}$ to be the cross-covariance function, and $v_{\texttt{KH}}(t,s) = v_{\texttt{HK}}(s,t)$.

A typical principal component is now defined by a 2-vector $\xi = (\xi^{\texttt{H}}, \xi^{\texttt{K}})'$ of weight functions, with $\xi^{\texttt{H}}$ denoting the variation in the `Hip` curve and $\xi^{\texttt{K}}$ that in the `Knee` curve. To proceed, we need to define an inner product on the space of vector functions of this kind. Once this has been defined, the principal components analysis can be formally set out in exactly the same way as previously.

The most straightforward definition of an inner product between bivariate functions is simply to sum the inner products of the two components. Suppose ξ_1 and ξ_2 are both bivariate functions each with hip and knee components. We then define the inner product of ξ_1 and ξ_2 to be

$$\langle \xi_1, \xi_2 \rangle = \int \xi_1^{\texttt{H}} \xi_2^{\texttt{H}} + \int \xi_1^{\texttt{K}} \xi_2^{\texttt{K}}. \tag{8.20}$$

The corresponding squared norm $\|\xi\|^2$ of a bivariate function ξ is simply the sum of the squared norms of the two component functions $\xi^{\texttt{H}}$ and $\xi^{\texttt{K}}$.

What all this amounts to, in effect, is stringing two (or more) functions together to form a composite function. We do the same thing with the data themselves: define $\texttt{Angles}_i = (\texttt{Hip}_i, \texttt{Knee}_i)$. The weighted linear combination (8.4) becomes

$$f_i = \langle \xi, \texttt{Angles}_i \rangle = \int \xi^{\texttt{H}} \texttt{Hip}_i + \int \xi^{\texttt{K}} \texttt{Knee}_i. \tag{8.21}$$

We now proceed exactly as in the univariate case, extracting solutions of the eigenequation system $V\xi = \rho\xi$, which can be written out in full detail as

$$\begin{aligned} \int v_{\texttt{HH}}(s,t)\xi^{\texttt{H}}(t)\,dt + \int v_{\texttt{HK}}(s,t)\xi^{\texttt{K}}(t)\,dt &= \rho\xi^{\texttt{H}}(s) \\ \int v_{\texttt{KH}}(s,t)\xi^{\texttt{H}}(t)\,dt + \int v_{\texttt{KK}}(s,t)\xi^{\texttt{K}}(t)\,dt &= \rho\xi^{\texttt{K}}(s). \end{aligned} \tag{8.22}$$

In practice, we carry out this calculation by replacing each function $\texttt{Hip}_i$ and $\texttt{Knee}_i$ with a vector of values at a fine grid of points or coefficients in a suitable expansion. For each i these vectors are concatenated into a single long vector Z_i; the covariance matrix of the Z_i is a discretized version of the operator V as defined in (8.7). We carry out a standard principal components analysis on the vectors Z_i, and separate the resulting principal component vectors into the parts corresponding to `Hip` and to `Knee`. The

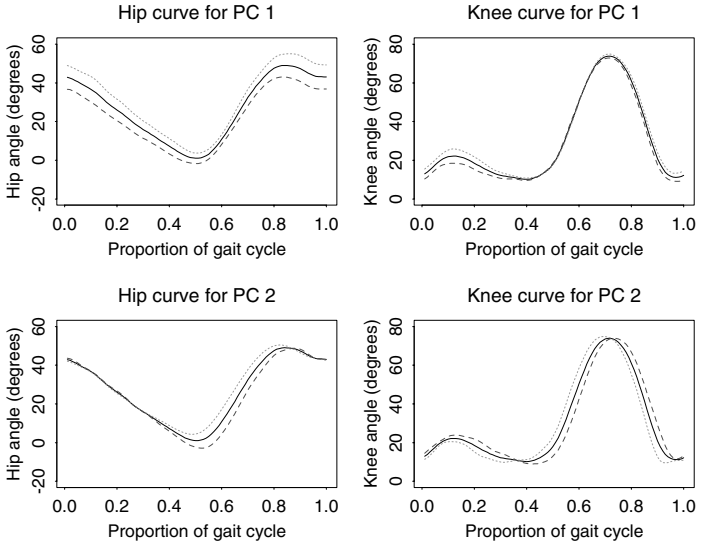

Figure 8.8. The mean hip and knee angle curves and the effects of adding and subtracting a multiple of each of the first two vector principal components.

analysis is completed by applying a suitable inverse transform to each of these parts if necessary.

If the variability in one of the sets of curves is substantially greater than that in the other, then it is advisable to consider down-weighting the corresponding term in the inner product (8.20), and making the consequent changes in the remainder of the procedure. In the case of the hip and knee data, however, both sets of curves have similar amounts of variability and are measured in the same units (degrees) and so there is no need to modify the inner product.

8.5.2 Visualizing the results

In the bivariate case, the best way to display the result depends on the particular context. In some cases it is sufficient to consider the individual parts $\xi_m^{\mathtt{H}}$ and $\xi_m^{\mathtt{K}}$ separately. An example of this is given in Figure 8.8, which displays the first two principal components. Because $\|\xi_m^{\mathtt{H}}\|^2 + \|\xi_m^{\mathtt{K}}\|^2 = 1$ by definition, calculating $\|\xi_m^{\mathtt{H}}\|^2$ gives the proportion of the variability in the mth principal component accounted for by variation in the hip curves.

For the first principal components, this measure indicates that 85% of the variation is due to the hip curves, and this is borne out by the presentation in Figure 8.8. The effect on the hip curves of the first combined principal component of variation is virtually identical to the first principal

component curve extracted from the hip curves considered alone. There is also little associated variation in the knee curves, apart from a small associated increase in the bend of the knee during the part of the cycle where all the weight is on the observed leg. The main effect of the first principal component remains an overall shift in the hip angle. This could be caused by an overall difference in stance; some people stand up more straight than others and therefore hold their trunks at a different angle from the legs through the gait cycle. Alternatively, there may simply be variation in the angle of the marker placed on the trunk.

For the second principal component, the contributions of both hip and knee are important, with somewhat more of the variability (65%) due to the knee than to the hip. We see that this principal component is mainly a distortion in the timing of the cycle, again correlated with the way in which the initial slight bend of the knee takes place. There is some similarity to the second principal component found for the hip alone, but this time there is very substantial interaction between the two joints.

A particularly effective method for displaying principal components in the bivariate case is to construct plots of one variable against the other. Suppose we are interested in displaying the mth principal component function. For equally spaced points t in the time interval on which the observations are taken, we indicate the position of the mean function values $(\texttt{Hipmn}(t), \texttt{Kneemn}(t))$ by a dot in the (x, y) plane, and we join this dot by an arrow to the point $(\texttt{Hipmn}(t) + C\xi_m^{\texttt{H}}(t),\ \texttt{Kneemn}(t) + C\xi_m^{\texttt{K}}(t))$. We choose the constant C to give clarity. Of course, the sign of the principal component functions, and hence the sense of the arrows, is arbitrary, and plots with all the arrows reversed convey the same information.

This technique is displayed in Figure 8.9. The plot of the mean cycle alone demonstrates the overall shape of the gait cycle in the hip-knee plane. The portion of the plot between time points 11 and 19 (roughly the part where the foot is off the ground) is approximately half an ellipse with axes inclined to the coordinate axes. The points on the ellipse are roughly at equal angular coordinates — somewhat closer together near the more highly curved part of the ellipse. This demonstrates that in this part of the cycle, the joints are moving roughly in simple harmonic motion but with different phases. During the other part of the cycle, the hip angle is changing at a approximately constant rate as the body moves forward with the leg approximately straight, and the knee bends slightly in the middle.

Now consider the effect of the first principal component of variation. As we have already seen, this has little effect on the knee angle, and all the arrows are approximately in the x-direction. The increase in the hip angle due to this mode of variation is somewhat larger when the angle itself is larger. This indicates that the effect contains an exaggeration (or diminution) in the amount by which the hip joint is bent during the cycle, and is also related to the overall angle between the trunk and the legs.

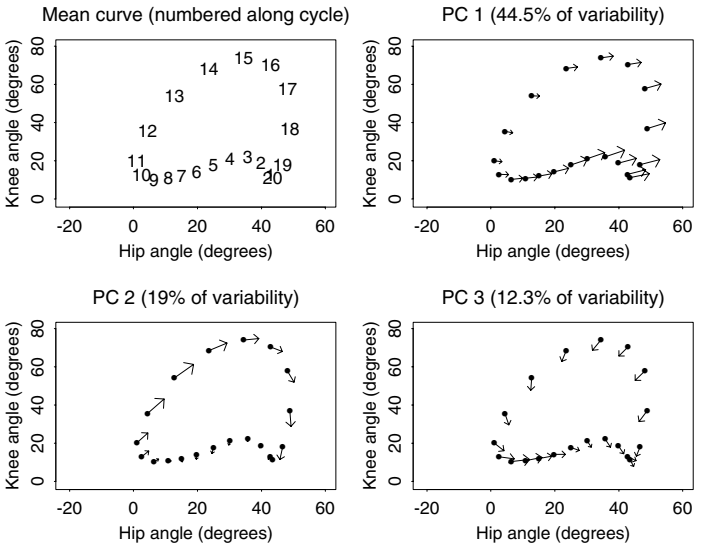

Figure 8.9. A plot of 20 equally spaced points in the average gait cycle, and the effects of adding a multiple of each of the first three principal component cycles in turn.

The second principal component demonstrates an interesting effect. There is little change during the first half of the cycle. However, during the second half, individuals with high values of this principal component would traverse roughly the same cycle but at a roughly constant time ahead. Thus this component represents a uniform time shift during the part of the cycle when the foot is off the ground.

A high score on the third component indicates two effects. There is some time distortion in the first half of the cycle, and then a shrinking of the overall cycle; an individual with a high score would move slowly through the first part of the cycle, and then perform simple harmonic motion of knee and hip joints with somewhat less than average amplitude.

8.5.3 Inner product notation: Concluding remarks

One of the features of the functional data analysis approach to principal components analysis is that, once the inner product has been defined appropriately, principal components analysis looks formally the same, whether the data are the conventional vectors of multivariate analysis, scalar functions as considered in Section 8.2.2, or vector-valued functions as in Section 8.5.1. Indeed, principal component analyses for other possible forms of functional data can be constructed similarly; all that is needed

is a suitable inner product, and in most contexts the definition of such an inner product will be a natural one. For example, if our data are functions defined over a region $\mathcal{S}$ in two-dimensional space, for example temperature profiles over a geographical region, then the natural inner product will be given by

$$\int_{\mathcal{S}} f(\mathbf{s})g(\mathbf{s})d\mathbf{s},$$

and the principal component weight functions will also be functions defined over $\mathbf{s}$ in $\mathcal{S}$.

Much of our subsequent discussion of PCA, and of other functional data analysis methods, will use univariate functions of a single variable as the standard example. This choice simplifies the exposition, but in most or all cases the methods generalize immediately to other forms of functional data, simply by substituting an appropriate definition of inner product.

8.6 Further readings and notes

An especially fascinating and comprehensive application of functional principal components analysis can be found in Locantore, Marron, Simpson, Tripoli, Zhang and Cohen (1999). These authors explore abnormalities in the curvature of the cornea in the human eye, and along the way extend functional principal components methodology in useful ways. Since the variation is over the spherical or elliptical shape of the cornea, they use Zernicke orthogonal basis functions. Their color graphical displays and the importance of the problem make this a showcase paper.

Viviani, Grön and Spitzer (2005) apply PCA to repeated fMRI scans of areas in the human brain, where each curve is associated with a specific voxel. They compare the functional and multivariate versions, and find that the functional approach offers a rather better image of experimental manipulations underlying the data. They also find that the use of the GCV criterion is particularly effective in choosing the smoothing parameter prior to applying functional PCA.

While most of our examples have time as the argument, there are many important problems in the physical and engineering sciences where spectral analysis is involved. An example involving elements of both registration and principal components analysis is reported in Liggett, Cazares and Semmes (2003). Kneip and Utikal (2001) apply functional principal components analysis to the problem of describing a set of density curves where the argument variable is log income.

Besse, Cardot and Ferraty (1997) studied the properties of estimates of curves where these are assumed to lie within a finite-dimensional subspace, and where principal components analysis is used in the estimation process, and Cardot (2004) extended this work.

Valderrama, Aguilera and Ocaña (2000) is a monograph in Spanish that contains many interesting applications of principal components analysis to functional data at the University of Granada, some of which precede the publication of our first edition. Ocaña, Aguilera and Valderrama (1999) discuss the role of the norm used to define functional principal components analysis.

James, Hastie and Sugar (2000) have developed a useful extension of functional principal components analysis that permits the estimation of harmonics from fragments of curves. They analyze measurements of spinal bone mineral density in females children and young adults taken at various ages. Yao, Müller and Wang (2004) is a more recent reference on this important problem.

There is a considerable literature on cluster analysis of samples of curves, a topic not far removed from principal components analysis. Abraham, Cornillion, Matzner-Lober and Molinari (2003) and Tarpey and Kinateder (2003) are recent references. James and Sugar (2003) adapt their functional principal components approach to this problem. Tarpey, Petkova and Ogden (2003) use functional cluster analysis to profile placebo responders.

9

Regularized principal components analysis

9.1 Introduction

In this chapter, we discuss the application of smoothing to functional principal components analysis. In Chapter 5 we have already seen that smoothing methods are useful in functional data analysis in preprocessing the data to obtain functional observations. The emphasis in this chapter is somewhat different, in that we incorporate the smoothing into the principal components analysis itself.

Our discussion provides a further insight into the way the method of regularization, discussed in Chapter 5, can be used rather generally in functional data analysis. The basic idea is to put into practice, in any particular context, the philosophy of combining a measure of goodness-of-fit with a roughness penalty.

Consideration of the third component in Figure 8.1 indicates that some smoothing may be appropriate when estimating functional principal components. A more striking example is provided by the pinch force data discussed in Section 1.5.2. Rather than smoothing the data initially, consider the data in Figure 9.1, which consists of the original records of the force exerted by the thumb and forefinger during each of 20 brief squeezes or pinches. The observed records are not very smooth, and consequently the principal component curves in Figure 9.2 show substantial variability. There is a clear need for smoothing or regularizing of the estimated principal component curves. In this chapter, we develop a method for smoothed principal component analysis, but first of all the application of the method

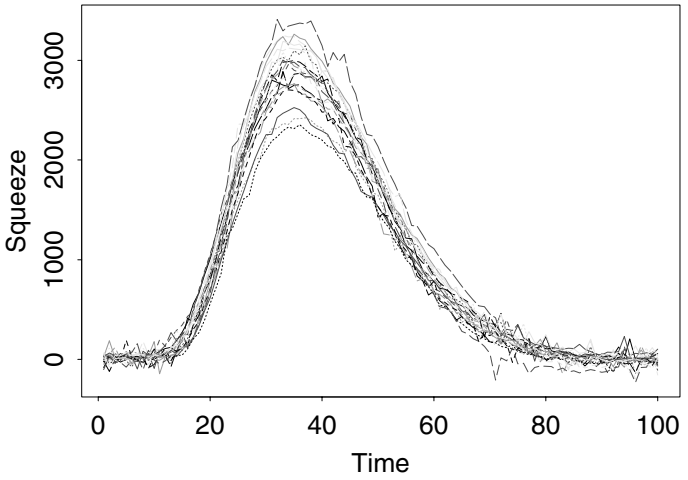

Figure 9.1. The aligned original recordings of the force relative to a baseline value exerted during each of 20 brief pinches.

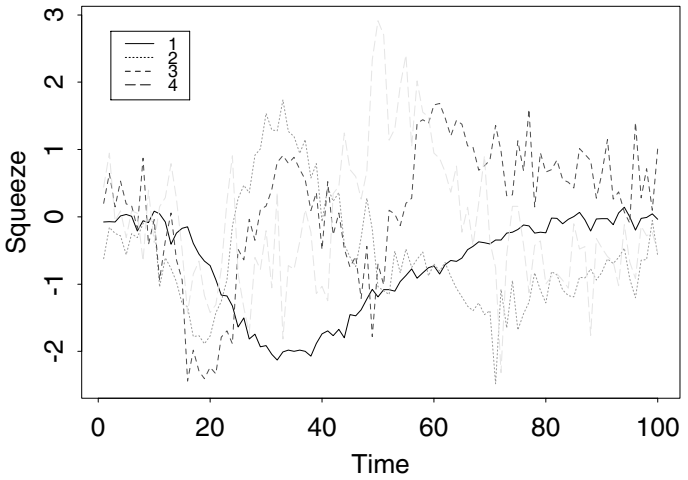

Figure 9.2. The first four principal component curves for the pinch force data without regularization.

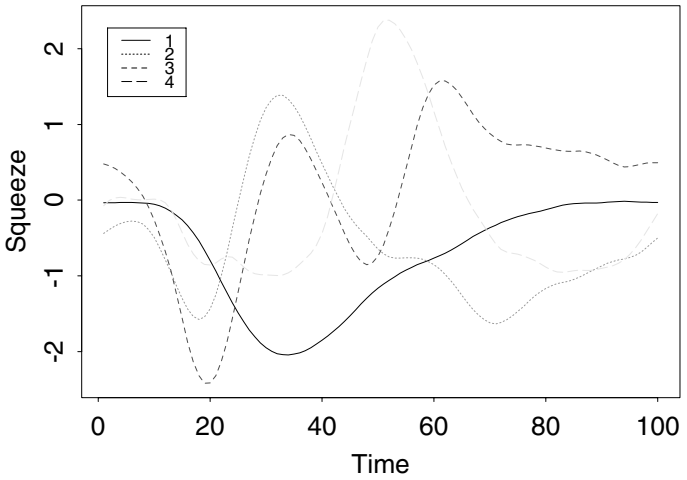

Figure 9.3. The first four smoothed principal components for the pinch force data, smoothed by the method of Section 9.3. The smoothing parameter is chosen by cross-validation.

to the pinch force data is demonstrated. In subsequent sections, the method is defined in detail and various aspects of its implementation are discussed.

9.2 The results of smoothing the PCA

Figure 9.3 shows the effect of applying principal components analysis using the method for smoothed PCA set out subsequently in this chapter. The method incorporates a smoothing parameter λ to control the amount of smoothing applied, and this has been chosen by a cross-validation method set out in Section 9.3.3. The smoothing method achieves the aim of removing the considerable roughness in the raw principal component curves in Figure 9.2.

Figure 9.4 displays the effects on the mean curve of adding and subtracting a multiple of each of the first four smoothed principal components. The first component corresponds to an effect whereby the shape of the impulse is not substantially changed, but its overall scale is increased. The second component (with appropriate sign) corresponds roughly to a compression in the overall time scale during which the squeeze takes place. Both of these effects were removed in the analysis of Ramsay, Wang and Flanagan (1995) before any detailed analysis was carried out. It is, however, interesting to

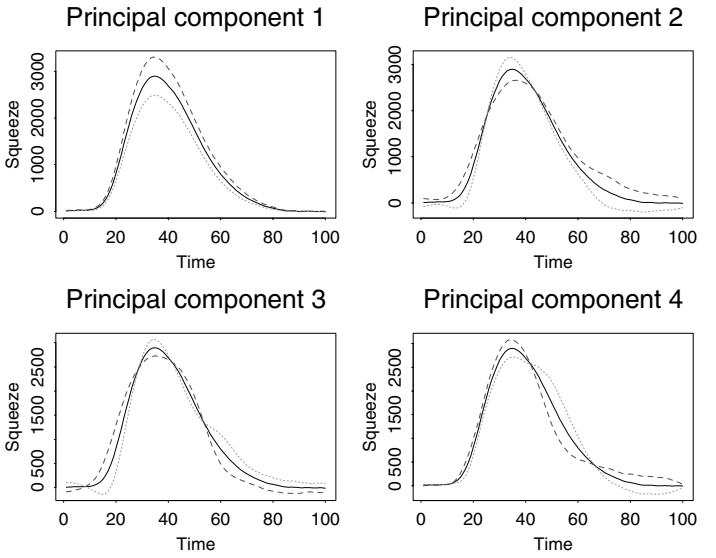

Figure 9.4. The effect on the overall mean curve of adding and subtracting a suitable multiple of each of the first four smoothed principal component curves provided in Figure 9.3.

note that they occur as separate components and therefore are essentially uncorrelated with one another, and with the effects found subsequently. The third component corresponds to an effect whereby the main part takes place more quickly but the tail after the main part is extended to the right. The fourth component corresponds to a higher peak correlated with a tail-off that is faster initially but subsequently slower than the mean. The first and second effects are transparent in their interest, and the third and fourth are of biomechanical interest in indicating how the system compensates for departures from the (remarkably reproducible) overall mean. The smoothing we have described makes the effects very much clearer than they are in the raw principal component plot.

The estimated variances σ^2 indicate that the four components displayed respectively explain 86.2%, 6.7%, 3.5% and 1.7% of the variability in the original data, with 1.9% accounted for by the remaining components. The individual principal component scores indicate that there is one curve with a fairly extreme value of principal component 2 (corresponding to moving more quickly than average through the cycle) but this curve is not unusual in other respects.

9.3 The smoothing approach

9.3.1 Estimating the leading principal component

Our smoothed PCA approach is based on a roughness penalty idea, as discussed in Chapter 7. Suppose ξ is a possible principal component curve. As in standard spline smoothing, we usually penalize the roughness of ξ by its integrated squared second derivative over the interval of interest, $\texttt{PEN}_2(\xi) = \|D^2\xi\|^2$.

Consider, first, the estimation of the leading principal component. In an unsmoothed functional PCA as described in Chapter 8, we work with the sample variance $\texttt{var} \int \xi x_i$ of the principal component scores $\int \xi x_i$ over the observations x_i. The first principal component weight function is chosen to maximize $\texttt{var} \int \xi x_i$ subject to the constraint $\|\xi\|^2 = 1$. As explained in Section 8.2.4, this maximization problem is solved by finding the leading solution of the eigenfunction equation $V\xi = \rho\xi$.

However, maximizing this sample variance is not our only aim. We also want to prevent the roughness $\texttt{PEN}_2(\xi) = \int \xi''(t)^2 dt$ of the estimated principal component ξ from being too large. The key to the roughness penalty approach is to make explicit this possible conflict. As usual in the roughness penalty method, the trade-off is controlled by a smoothing parameter $\lambda \geq 0$ which regulates the importance of the roughness penalty term.

Given any possible principal component function ξ with $\|\xi\|^2 = 1$, one way of penalizing the sample variance $\texttt{var} \int \xi x_i$ is to divide it by $\{1 + \lambda \times \texttt{PEN}_2(\xi)\}$. This gives the *penalized sample variance*

$$\texttt{PCAPSV}(\xi) = \frac{\texttt{var} \int \xi x_i}{\|\xi\|^2 + \lambda \times \texttt{PEN}_2(\xi)}. \tag{9.1}$$

Increasing the roughness of ξ while maintaining λ fixed decreases $\texttt{PCAPSV}(\xi)$, as defined in (9.1), since $\texttt{PEN}_2(\xi)$ increases. Moreover, PCAPSV reverts to the raw sample variance as $\lambda \to 0$. On the other hand, the larger the value of λ, the more the penalized sample variance is affected by the roughness of ξ. In the limit $\lambda \to \infty$, the component ξ is forced to be of the form $\xi = a$ in the periodic case and $\xi = a + bt$ in the nonperiodic case, for some constants a and b.

9.3.2 Estimating subsequent principal components

Of course, it is usually of interest not merely to estimate the leading principal component, but also to estimate the other components. The way our procedure works is to estimate each ξ_j to maximize the penalized variance $\texttt{PCAPSV}(\xi)$ as defined in (9.1), subject to two constraints. The first constraint is the usual requirement that $\|\xi_j\|^2 = 1$. Secondly, we impose a

modified form of orthogonality to the previously estimated components

$$\int \xi_j(s)\xi_k(s)ds + \int D^2\xi_j(s)D^2\xi_k(s)ds = 0 \text{ for } k = 1, \ldots, j-1. \quad (9.2)$$

The use of the modified orthogonality condition (9.2) means that we can find the estimates of all the required principal components by solving a single eigenvalue problem, and this will be explained in Section 9.4, where practical algorithms are discussed. Silverman (1996) provides a detailed investigation of the theoretical advantages of this approach.

9.3.3 Choosing the smoothing parameter by cross-validation

How should the smoothing parameter λ be chosen? It is perfectly adequate for many purposes to choose the smoothing parameter subjectively, but we can also use a cross-validation approach to choose the amount of smoothing automatically. Some general remarks about the use of automatic methods for choosing smoothing parameters are found in Section 3.1 of Green and Silverman (1994).

To consider how a cross-validation score could be calculated, suppose that x is an observation from the population. Then, by the optimal basis property discussed in Section 8.2.3, the principal components have the property that, for each m, an expansion in terms of the functions $\xi_1, \ldots, \xi_m$ can explain more of the variation in x than any other collection of m functions. To quantify the amount of variation in x accounted for by these functions, we define x^* to be the projection of x onto the subspace spanned by $\xi_1, \ldots, \xi_m$ and let ζ_m be the residual component $x - x^*$. Thus, ζ_m is the component of x orthogonal to the functions $\xi_1, \ldots, \xi_m$.

If we wish to consider the efficacy of the first m components, then a measure to consider is $E\|\zeta_m\|^2$; in order not to be tied to a particular m, we can, for example, minimize $\sum_m E\|\zeta_m\|^2$. In both cases, we do not have new observations x to work with, and the usual cross-validation paradigm has to be used, as follows:

1. Subtract the overall mean from the observed data x_i.
2. For a given smoothing parameter λ, let $\xi_j^{[i]}(\lambda)$ be the estimate of ξ_j obtained from all the data except x_i.
3. Define $\zeta_m^{[i]}(\lambda)$ to be the component of x_i orthogonal to the subspace spanned by $\{\xi_j^{[i]}(\lambda) : j = 1, \ldots, m\}$.
4. Combine the $\zeta_m^{[i]}(\lambda)$ to obtain the cross-validation scores

$$\mathtt{CV}_m(\lambda) = \sum_{i=1}^{n} \|\zeta_m^{[i]}(\lambda)\|^2 \quad (9.3)$$

and hence

$$\mathtt{CV}(\lambda) = \sum_{m=1}^{\infty} \mathtt{CV}_m(\lambda). \tag{9.4}$$

In practice, we would of course truncate the sum in (9.4) at some convenient point. Indeed, given n data curves, we can estimate at most $n-1$ principal components, and so the sum must be truncated at $m = n-1$ if not at a smaller value.

5. Minimize $\mathtt{CV}(\lambda)$ to provide the choice of smoothing parameter.

Clearly there are other possible ways of combining the $\mathtt{CV}_m(\lambda)$ to produce a cross-validation score to account for more than one value of m, but we restrict attention to $\mathtt{CV}(\lambda)$ as defined in (9.4).

In the pinch force data example considered Section 9.2, it was found satisfactory to calculate the cross-validation score on a grid (on a logarithmic scale) of values of the smoothing parameter λ and pick out the minimum. The grid can be quite coarse, since small changes in the numerical value of λ do not make very much difference to the smoothed principal components. For this example, we calculated the cross-validation scores for $\lambda = 0$ and $\lambda = 1.5^{i-1}$ for $i = 1, \ldots, 30$, and we attained the minimum of $\mathtt{CV}(\lambda)$ by setting $\lambda = 37$.

9.4 Finding the regularized PCA in practice

In practice, the smoothed principal components are most easily found by working in terms of a suitable basis. First of all, consider the periodic case, for which it is easy to set out an algorithm based on Fourier series.

9.4.1 The periodic case

Suppose, for simplicity, that $\mathcal{T}$ is the interval $[0,1]$ and that periodic boundary conditions are valid for all the functions we are considering. In particular, this means that the data $x_i(s)$ themselves are regarded as being periodic. Let $\{\phi_\nu\}$ be the series of Fourier functions defined in (3.7). For each j, define $\omega_{2j-1} = \omega_{2j} = 2\pi j$. Given any periodic function x, we can expand x as a Fourier series with coefficients $c_\nu = \int x\phi_\nu$, so that

$$x(s) = \sum_\nu c_\nu \phi_\nu(s) = \mathbf{c}'\boldsymbol{\phi}(s).$$

The operator D^2 has the useful property that, for each ν,

$$D^2\phi_\nu = -\omega_\nu^2 \phi_\nu,$$

meaning that we can also expand D^2x as

$$D^2x(s) = -\sum_\nu \omega_\nu^2 c_\nu \phi_\nu(s).$$

By standard orthogonality properties of trigonometric functions, the ϕ_ν are orthonormal, and it follows that the roughness penalty $\|D^2x\|^2$ can be written as a weighted sum of squares of the coefficients c_ν:

$$\|D^2x\|^2 = \int (-\sum_\nu \omega_\nu^2 c_\nu \phi_\nu)(-\sum_\nu \omega_\nu^2 c_\nu \phi_\nu) = \sum_\nu \omega_\nu^4 c_\nu^2.$$

Now proceed by expanding the data functions to sufficient terms in the basis to approximate them closely. We can use a fast Fourier transform on a finely discretized version of the observed data functions to do this efficiently. Denote by $\mathbf{c}_i$ the vector of Fourier coefficients of the observation $x_i(s)$, so that $x_i(s) = \mathbf{c}_i'\boldsymbol{\phi}(s)$ where $\boldsymbol{\phi}$ is the vector of basis functions. Let $\mathbf{V}$ be the covariance matrix of the vectors $\mathbf{c}_i$, and let $\mathbf{S}$ be the diagonal matrix with entries

$$S_{\nu\nu} = (1 + \lambda\omega_\nu^4)^{-1/2}.$$

The matrix $\mathbf{S}$ then corresponds to a smoothing operator S.

Let $\mathbf{y}$ be the vector of coefficients of any potential principal component curve ξ, so that

$$\xi(s) = \sum_\nu y_\nu \phi_\nu(s) = \mathbf{y}'\boldsymbol{\phi}(s). \tag{9.5}$$

In terms of Fourier coefficients, we have

$$\texttt{PCAPSV}(\xi) = \frac{\mathbf{y}'\mathbf{V}\mathbf{y}}{\mathbf{y}'\mathbf{S}^{-2}\mathbf{y}}. \tag{9.6}$$

Furthermore, if $\mathbf{y}_{(j)}$ denotes the vector of Fourier coefficients of the curve ξ_k, then the constraint (9.2) can be written as $\mathbf{y}_{(j)}'\mathbf{S}^{-2}\mathbf{y}_{(k)} = 0$ for $k = 1, \ldots, j-1$.

It follows from standard arguments in linear algebra that the estimates specified in Section 9.3 have Fourier coefficients that satisfy the eigenvector equation

$$\mathbf{V}\mathbf{y} = \rho\mathbf{S}^{-2}\mathbf{y}, \tag{9.7}$$

which can be rewritten

$$(\mathbf{S}\mathbf{V}\mathbf{S})(\mathbf{S}^{-1}\mathbf{y}) = \rho(\mathbf{S}^{-1}\mathbf{y}). \tag{9.8}$$

The matrix $\mathbf{SVS}$ is the covariance matrix of the vectors $\mathbf{S}\mathbf{c}_i$, the Fourier coefficient vectors of the original data smoothed by the application of the smoothing operator S.

To find the solutions of (9.8), suppose that $\mathbf{u}$ is an eigenvector of $\mathbf{SVS}$ with eigenvalue ρ. Finding the eigenvectors and eigenvalues of $\mathbf{SVS}$ corresponds precisely to carrying out an unsmoothed PCA of the *smoothed* data

$\mathbf{Sc}_i$. Then it is apparent that any multiple of $\mathbf{Su}$ is a solution of (9.8) for the same ρ. Because we require $\|\mathbf{y}\|^2 = 1$, renormalize and set $\mathbf{y} = \mathbf{Su}/\|\mathbf{Su}\|$. The functional principal component ξ corresponding to $\mathbf{y}$ is then computed from (9.5).

Putting these steps together gives the following procedure for carrying out the smoothed principal component analysis of the original data:

1. Compute the coefficients $\mathbf{c}_i$ for the expansion of each sample function x_i in terms of basis $\boldsymbol{\phi}$.
2. Operate on these coefficients by the smoothing operator S.
3. Carry out a standard PCA on the resulting smoothed coefficient vectors $\mathbf{Sc}_i$.
4. Apply the smoothing operator S to the resulting eigenvectors $\mathbf{u}$, and renormalize so that the resulting vectors $\mathbf{y}$ have unit norm.
5. Compute the principal component function ξ from (9.5).

9.4.2 The nonperiodic case

Now turn to the nonperiodic case, where Fourier expansions are no longer appropriate because of the boundary conditions. Suppose that $\{\phi_\nu\}$ is a suitable basis for the space of smooth functions $\mathcal{S}$ on $[0, 1]$. Possible bases include B-splines on a fine mesh, or possibly orthogonal polynomials up to some degree. In either case, we choose the dimensionality of the basis to represent the functions $x_i(s)$ well. As in the discussion of the periodic case, let $\mathbf{c}_i$ be the vector of coefficients of the data function $x_i(s)$ in the basis $\{\phi_\nu\}$. Let $\mathbf{V}$ be the covariance matrix of the vectors $\mathbf{c}_i$.

Define $\mathbf{J}$ to be the matrix $\int \boldsymbol{\phi}\boldsymbol{\phi}'$, whose elements are $\int \phi_j \phi_k$ and $\mathbf{K}$ the matrix whose elements are $\int D^2\phi_j D^2\phi_k$. The penalized sample variance can be written as

$$\texttt{PCAPSV} = \frac{\mathbf{y}'\mathbf{JVJy}}{\mathbf{y}'\mathbf{Jy} + \lambda \mathbf{y}'\mathbf{Ky}} \tag{9.9}$$

and the eigenequation corresponding to (9.7) is given by

$$\mathbf{JVJy} = \rho(\mathbf{J} + \lambda\mathbf{K})\mathbf{y}. \tag{9.10}$$

Now perform a factorization $\mathbf{LL}' = \mathbf{J} + \lambda\mathbf{K}$ and define $\mathbf{S} = \mathbf{L}^{-1}$. We can find a suitable matrix $\mathbf{L}$ by an SVD or by Choleski factorization, in which case $\mathbf{L}$ is a lower triangular matrix. The equation (9.10) can now be written as

$$(\mathbf{SJVJS}')(\mathbf{L}'\mathbf{y}) = \rho\mathbf{L}'\mathbf{y}.$$

We can now work through stages corresponding to those for the periodic case. The algorithm obtained is as follows:

1. Expand the observed data x_i with respect to the basis $\boldsymbol{\phi}$ to obtain coefficient vectors $\mathbf{c}_i$.
2. Solve $\mathbf{L}d_i = \mathbf{J}\mathbf{c}_i$ for each i to find the vectors $d_i = \mathbf{S}\mathbf{J}\mathbf{c}_i$.
3. Carry out a standard PCA on the coefficient vectors d_i.
4. Apply the smoothing operator $\mathbf{S}'$ to the resulting eigenvectors $\mathbf{u}$ by solving $\mathbf{L}'\mathbf{y} = u$ in each case, and renormalize so that the resulting vectors $\mathbf{y}$ have $\mathbf{y}'\mathbf{J}\mathbf{y} = 1$.
5. Transform back to find the principal component functions ξ using (9.5).

If we use a B-spline basis and define $\mathbf{L}$ by a Choleski factorization, then the matrices $\mathbf{J}$, $\mathbf{K}$ and $\mathbf{L}$ are all band matrices, and by using appropriate linear algebra routines, we can carry out all the calculations extremely economically. Even in the full matrix case, especially if not too many basis functions are used, the computations are reasonably fast because $\mathbf{S}$ never has to be found explicitly.

9.5 Alternative approaches

In this section, we discuss two alternative approaches to smoothed functional PCA.

9.5.1 Smoothing the data rather than the PCA

In this section, we compare the method of regularized principal components analysis with an approach akin to that discussed earlier in the book. Instead of carrying out our smoothing step within the PCA, we smooth the data first, and then carry out an unsmoothed PCA. This approach to functional PCA was taken by Besse and Ramsay (1986), Ramsay and Dalzell (1991) and Besse, Cardot and Ferraty (1997). Of course, conceivably any smoothing method can be used to smooth the data, but to make a reasonable comparison, we use a roughness penalty smoother based on integrated squared second derivative. For simplicity, let us restrict our attention to the case of periodic boundary conditions.

Suppose that x is a data curve, and that we regard x as the sum of a smooth curve and a noise process. We would obtain the roughness penalty estimate of the smooth curve by minimizing

$$\texttt{PENRSS} = \|x - g\|^2 + \lambda \|D^2 g\|^2$$

over g in $\mathcal{S}$. As usual, λ is a smoothing parameter that controls the trade-off between fidelity to the data and smoothing. This is a generalization of the

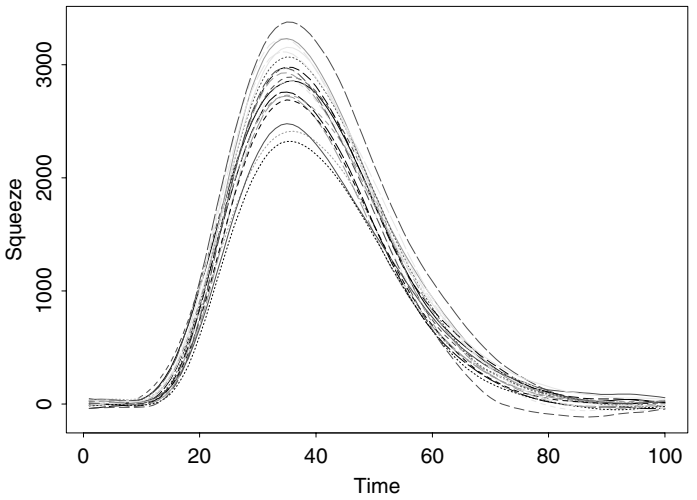

Figure 9.5. The pinch force data curves, smoothed by a roughness penalty method with the same smoothing parameter as used for the smoothed PCA, and with the baseline pressure subtracted.

spline smoothing method discussed in Chapter 5 to the case of functional data.

Consider an expansion of x and g in terms of Fourier series as in Section 9.4.1, and let $\mathbf{c}$ and $\mathbf{d}$ be the resulting vectors of coefficients. Then

$$\texttt{PENRSS} = \|\mathbf{c} - \mathbf{d}\|^2 + \lambda \sum_{\nu} \omega_\nu^4 d_\nu^2,$$

and hence the coefficients of the minimizing g satisfy

$$\mathbf{d} = \mathbf{S}^2\mathbf{c}, \tag{9.11}$$

where $\mathbf{S}$ is as defined in Section 9.4.1. Note that this demonstrates that the smoothing operator $\mathbf{S}$ used twice in the algorithm set out in Section 9.4.1 can be regarded as a half-spline-smooth, since $\mathbf{S}^2$ is the operator corresponding to classical spline smoothing.

Now let us consider the effect of smoothing the data by the operator $\mathbf{S}^2$ using the same smoothing parameter $\lambda = 37$ as in the construction of Figures 9.3 and 9.4. The effect of this smoothing on the data is illustrated in Figure 9.5. Figure 9.6 shows the first four principal component curves of the smoothed data. Although the two methods do not give identical results, the differences between them are too small to affect any interpretation.

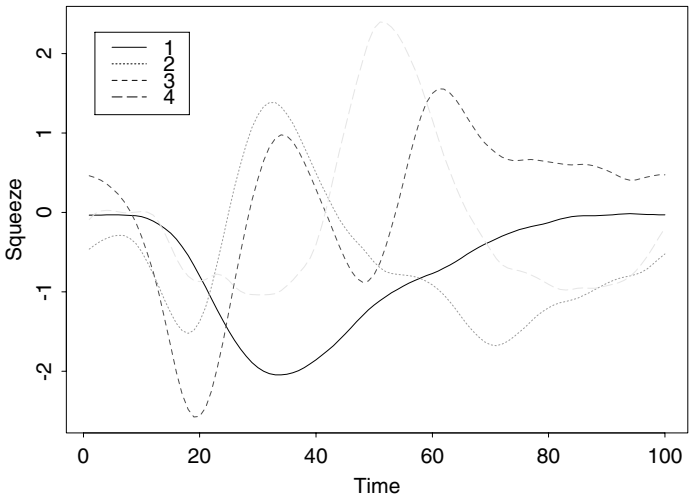

Figure 9.6. The first four principal component curves of the smoothed data as shown in Figure 9.5.

However, this favorable comparison depends rather crucially on the way in which the data curves are smoothed, and in particular on the match between the smoothing level implied in (9.11) and the smoothing level used for the PCA itself. For example, we tried smoothing the force functions curves individually, selecting the smoothing parameters by the generalized cross-validation approach used in the S-PLUS function `smooth.spline`. The result was much less successful, in the sense that the components were far less smooth. The reason appears to be that this smoothing technique tended to choose much smaller values of the smoothing parameter λ.

Kneip (1994) considers several aspects of an approach that first smooths the data and then extracts principal components. Under a model where the data are corrupted by a white noise error process, he investigates the dependence of the quality of estimation of the principal components on both sample size and sampling rate. In an application based on economics data, he shows that smoothing is clearly beneficial in a practical sense.

9.5.2 A stepwise roughness penalty procedure

Another approach to the smoothing of functional PCA was set out by Rice and Silverman (1991). They considered a stepwise procedure incorporating the roughness penalty in a different way. Their proposal requires a separate smoothing parameter λ_j for each principal component. The prin-

cipal components are estimated successively, the estimate $\xi_j^\dagger$ of ξ_j being found by maximizing $\mathtt{var} \int \xi x_i - \lambda_j \|D^2 \xi\|^2$ subject to the conventional orthonormality conditions $\|\xi\|^2 = 1$ and $\int \xi \xi_k^\dagger = 0$ for $k = 1, \ldots, j-1$.

This approach is computationally more complicated because a separate eigenproblem has to be posed and solved for each principal component; for more details, see the original paper. Theoretical results in Pezzulli and Silverman (1993) and Silverman (1996) also suggest that the procedure described in Section 9.3 is likely to be advantageous under conditions somewhat milder than those for the Rice-Silverman procedure.

9.5.3 A further approach

Yao, Müller, Clifford, Dueker, Follet, Lin, Bucholz and Vogel (2003) regularize the principal component scores f_{im} by shrinking them towards zero.

10

Principal components analysis of mixed data

10.1 Introduction

It is a characteristic of statistical methodology that problems do not always fall into neat categories. In the context of the methods discussed in this book, we often have *both* a vector of data *and* an observed function on each individual of interest. In this chapter, we consider some ways of approaching such mixed data, extending the ideas of PCA that we have already developed.

In Chapter 7 we have discussed one way in which mixed data can arise. Consider the Canadian temperature data as a specific example. The registration process finds, for each weather station, a suitable phase shift to apply to the raw observed curve; the phase shifts are chosen to make the shifted records fit together as well as possible. The *vector part* of the record is in this case just the single number giving the size of the shift. The *functional part* of the record is the shifted curve.

The method we will develop in this chapter produces principal component weights that have the same structure as the mixed data themselves. So the variability accounted for by each principal component can itself be split into two parts, the part corresponding to variability in the phase shifts and the part corresponding to variability in the registered functions. The first four principal components for the Canadian temperature data are shown in Figure 10.1. Let $\hat{\mu}(s)$ be the mean of all the *registered* curves, in other words the mean of the functional parts of all the observations. We assume that the mean of all the phase shifts is zero.

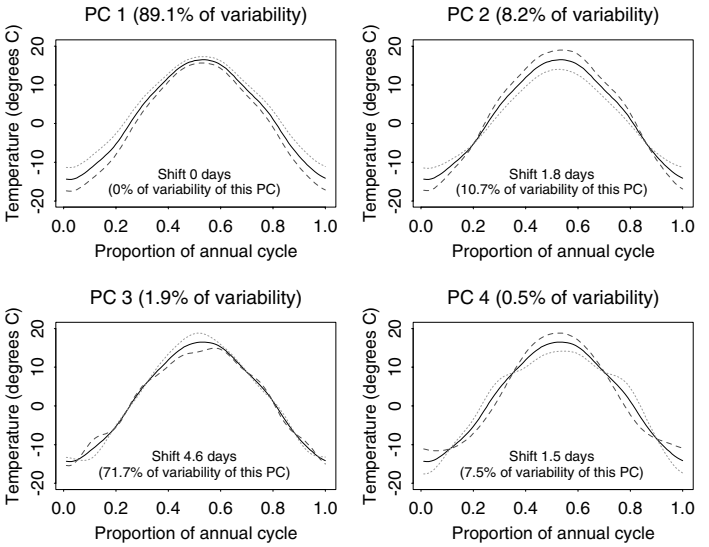

Figure 10.1. The mean Canadian temperature curve and the effects of adding and subtracting a suitable multiple of each PC curve, with the shift considered as a separate parameter.

The effect of each principal component is specified by a pair $(\xi_i(s), v_i)$, where $\xi(s)$ is the effect of variation in that component on the functional part, and v_i is the effect on the shift. Suppose, just for example, that the score at a particular weather station is 2.5 on the ith principal component and zero on all others. Then the functional part of the observation would be $\hat{\mu}(s) + 2.5\xi_i(s)$ and the phase shift would be $2.5v_i$. Note that the two effects go together, and the multiple of $\xi_i(s)$ is the same as that of v_i. In each case, the sign of the principal component has been taken to make the shift positive; this is by no means essential, but it leads to some simplicity of interpretation.

In the figure, the functional part of each principal component is illustrated by showing the effect on the overall mean $\hat{\mu}$ of adding and subtracting a suitable multiple of the relevant ξ_j. The fine dotted curve corresponds to adding the ξ_j and the dashed curve to subtracting. The shift part v_i is given numerically, for example 1.8 days for the second component. The figure also states what percentage of variability of each PC is accounted for by shift variability as opposed to the variability of the functional part.

Now consider the figure in detail. The first principal component—accounting for 89.1% of the variability in the original observations—entirely concerns the functional part, with 0% of the variability being in the shift component. A high score on this component goes along with a weather

station that is warmer than average all the year round, but with a larger variation in the winter months.

The second component has 10.7% of its variability accounted for by a shift component, of size 1.8 days. The functional part of this component corresponds to a change in amplitude of the annual temperature variation. High positive scores on this component would indicate lower-than-average temperature variation over the year (cool summers and relatively warm winters) *together with* a positive shift value. The third component is very largely shift variation (71.7% of the variability). Associated with a positive shift is an increase in temperature at the high point of the summer, with very little effect elsewhere.

A comparison between Figure 10.1 and Figure 8.2 is instructive. Because the shift component has been explicitly separated out, less skill is needed to interpret the principal components in Figure 10.1. The percentage of variation explained by each of the first four principal components is very similar, but not quite identical, in the two analyses, for a reason discussed further in Section 10.4.2.

Of course, there are many other situations where we have numerical observations as well as functional observations on the individuals of interest, and the PCA methodology we set out can be easily generalized to deal with them.

10.2 General approaches to mixed data

We now consider mixed data in a more general context, bearing in mind the Canadian temperature data as a specific example. To be precise about notation, suppose that our observations consist of pairs $(x_i, \mathbf{y}_i)$, where x_i is a function on the interval $\mathcal{T}$ and $\mathbf{y}_i$ is a vector of length M. How might we use PCA to analyze such data?

There are three different ways of viewing the $\mathbf{y}_i$. First, it may be that the $\mathbf{y}_i$ are simply nuisance parameters, of no real interest to us in the analysis, for example corresponding to the time at which a recording instrument is activated. In this case we would quite simply ignore them. The $\mathbf{y}_i$ can be thought of as one of the features of almost all real data sets that we choose not to include in the analysis.

On the other hand, as in the temperature data example, both the functions x_i and the observations $\mathbf{y}_i$ may be of primary importance. The PCA of such *hybrid* data $(x_i, \mathbf{y}_i)$ is the case to which we give the most attention, from Section 10.3 onwards. There is some connection with the methodology described in Section 8.5 for bivariate curve data with values $(x_i(t), y_i(t))$, though in our case the second component is a scalar or vector rather than a function.

As a third and somewhat intermediate possibility, the $\mathbf{y}_i$ may be of marginal importance, our central interest being in the functions x_i. In this case, we could ignore the $\mathbf{y}_i$ initially and carry out a PCA of the curves $x_i(t)$ alone. Having done this, we could investigate the connection between the scores on the principal component scores and the variable(s) $\mathbf{y}_i$. We could calculate the sample correlations between the principal component scores and the components of the $\mathbf{y}_i$. Alternatively or additionally, we could plot the $\mathbf{y}_i$ against the principal component scores or use other methods for investigating dependence. In this general approach, the $\mathbf{y}_i$ would not have been used in the first part of the analysis itself; however, they would have played a key part in interpreting the analysis. It would be interesting, for example, to notice that a particular principal component of the x_i was highly correlated with $\mathbf{y}_i$. We develop this approach further in Section 10.5.2.

10.3 The PCA of hybrid data

10.3.1 Combining function and vector spaces

A typical principal component weight function would consist of a *pair* $(\xi, \mathbf{v})$, where $\mathbf{v}$ is an M-vector, and the principal component score of a particular observation would then be the sum

$$\eta_i = \int x_i(s)\xi(s)\,ds + \mathbf{y}_i'\mathbf{v}. \tag{10.1}$$

Another way of saying this is that the principal component would be made up of a functional part ξ and a vector part $\mathbf{v}$, corresponding to the functional and vector (or numerical) parts of the original data. A typical observation from the distribution of the data would be modelled as

$$\begin{pmatrix} x_i \\ \mathbf{y}_i \end{pmatrix} = \sum_j \eta_{ij} \begin{pmatrix} \xi_j \\ \mathbf{v}_j \end{pmatrix}, \tag{10.2}$$

where $(\xi_j, \mathbf{v}_j)$ is the jth principal component weight and, as j varies, the vectors of principal component scores $\eta_{ij} = \int x_i\xi_j + \mathbf{y}_i'\mathbf{v}_j$ are uncorrelated variables with mean zero.

This kind of hybrid data PCA can very easily be dealt with in our general functional framework. Define $\mathcal{Z}$ to the space of pairs $z = (x, \mathbf{y})$, where x is a smooth function and $\mathbf{y}$ is a vector of length M. Given any two elements $z_{(1)} = (x_{(1)}, \mathbf{y}_{(1)})$ and $z_{(2)} = (x_{(2)}, \mathbf{y}_{(2)})$ of $\mathcal{Z}$, define the inner product

$$\langle z_{(1)}, z_{(2)} \rangle = \int x_{(1)}x_{(2)} + \mathbf{y}_{(1)}'\mathbf{y}_{(2)}. \tag{10.3}$$

From (10.3) we can define the norm $\|z\|^2 = \langle z, z \rangle$ of any z in $\mathcal{Z}$.

Now that we have defined an inner product and norm on $\mathcal{Z}$, write z_i for the data pair $(x_i, \mathbf{y}_i)$. To find the leading principal component, we wish to

find $\zeta = (\xi, \mathbf{v})$ in $\mathcal{Z}$ to maximize the sample variance of the $\langle \zeta, z_i \rangle$ subject to $\|\zeta\|^2 = 1$. The $\langle \zeta, z_i \rangle$ are of course exactly the same as the quantities $\eta_i = \int x_i(s)\xi(s)\,ds + \mathbf{y}_i'\mathbf{v}$ specified in equation (10.1).

Subsequent principal components maximize the same sample variance subject to the additional condition of orthogonality to the principal components already found, orthogonality being defined by the hybrid inner product (10.3). Principal components found in this way yield principal component scores that are uncorrelated, just as for conventional multivariate PCA.

The PCA of hybrid data is thus very easily specified in principle. However, there are several important issues raised by this idea, and we discuss these in the following sections.

10.3.2 Finding the principal components in practice

How do we carry out the constrained maximization of the sample variance of the $\langle \zeta, z_i \rangle$ in practice? Suppose that ϕ_k is a basis of K functions in which the functional parts x_i of the hybrid data z_i can be well approximated. Given any element $z = (x, \mathbf{y})$ of $\mathcal{Z}$, define the K-vector $\mathbf{c}$ to be the coefficients of x relative to the basis $\boldsymbol{\phi}$. Now let $p = K + M$, and let $\mathbf{w}$ be the p-vector

$$\mathbf{w} = \begin{bmatrix} \mathbf{c} \\ \mathbf{y} \end{bmatrix}.$$

Suppose that the basis $\boldsymbol{\phi}$ is an orthonormal basis, the Fourier functions, for example. Then the inner product (10.3) of any two elements $z_{(1)}$ and $z_{(2)}$ of $\mathcal{Z}$ is precisely equal to the ordinary vector inner product $\mathbf{w}_{(1)}'\mathbf{w}_{(2)}$ of the corresponding p-vectors of coefficients. Thus, if we use this method of representing members of $\mathcal{Z}$ by vectors, we have a representation in which the vectors behave exactly as if they were p-dimensional multivariate observations, with the usual Euclidean inner product and norm. It follows that we can use standard multivariate methods to find the PCA.

In summary, we can proceed as follows to carry out a PCA:

1. For each i, let $\mathbf{c}_i$ be the vector of the first K Fourier coefficients of x_i.

2. Augment each $\mathbf{c}_i$ by $\mathbf{y}_i$ to form the p-vector $\mathbf{w}_i$.

3. Carry out a standard PCA of the $\mathbf{w}_i$, by finding the eigenvalues and eigenvectors of the matrix $N^{-1}\sum_i \mathbf{w}_i\mathbf{w}_i'$.

4. If $\mathbf{u}$ is any resulting eigenvector, the first K elements of $\mathbf{u}$ are the Fourier coefficients of the functional part of the principal component, and the remaining elements are the vector part.

Since the procedure we have set out is a generalization of ordinary functional PCA, we may wish to incorporate some smoothing, and this is discussed in the next section.

10.3.3 Incorporating smoothing

To incorporate smoothing into our procedure, we can easily generalize the smoothing methods discussed in Chapter 9. The key step in the method is to define the roughness of an element $z = (x, \mathbf{y})$ of $\mathcal{Z}$. Let us take the roughness of z to be that of the functional part x of z, without any reference to the vector part $\mathbf{y}$. To do this, define $D^2 z$ to be equal to the element $(D^2 x, 0)$ of $\mathcal{Z}$ so that the roughness of z can then be written $\|D^2 z\|^2$, just as in the ordinary functional case. The norm is taken in $\mathcal{Z}$, but since the vector part of $D^2 z$ is defined to be zero, $\|D^2 z\|^2 = \|D^2 x\|^2$ as required.

Once we have defined the roughness of z, we can proceed to carry out a smoothed PCA using exactly the same ideas as in Chapter 9. As far as algorithms are concerned, the Fourier transform algorithm for the periodic case requires slight modification. Let z^* be the vector representation of an element z, of length $K + p$. The first K elements of z^* are the Fourier coefficients of the functional part x and the last p elements simply the vector part $\mathbf{y}$. The roughness of z is $\sum_{k=0}^{K-1} \omega_k^4 z_k^{*2}$ so the matrix $\mathbf{S}$ used in the algorithm described in Section 9.4.1 must be modified to have diagonal elements $(1 + \lambda\omega_k^4)^{-1/2}$ for $k < K$, and 1 for $K \leq k < p$.

Apart from this modification, and of course the modified procedures for mapping between the function/vector and basis representations of elements of $\mathcal{Z}$, the algorithm is exactly the same as in Section 9.4.1. Furthermore, the way in which we can apply cross-validation to choose the smoothing parameter is the same as in Section 9.3.3.

To deal with the nonperiodic case, we modify the algorithm of Section 9.4.2 in the same way. The matrix $\mathbf{J}$ is a block diagonal matrix where the first K rows and columns have elements $\int \phi_j \phi_k$ and the last M rows and columns are the identity matrix of order M. The matrix $\mathbf{K}$ has elements $\int (D^2\phi_j)(D^2\phi_k)$ in its first K rows and columns, and zeroes elsewhere.

10.3.4 Balance between functional and vector variation

Readers who are familiar with PCA may have noted one potential difficulty with the methodology set out above. The variations in the functional and vector parts of a hybrid observation z are really like chalk and cheese: they are measured in units which are almost inevitably not comparable, and therefore it may well not be appropriate to weight them as we have. In the registration example, the functional part consists of the difference between the pattern of temperature on the transformed time scale and its population mean; the vector part is made up of the parameters of the

time transformation. Clearly, these are not measured in directly compatible units!

One way of noticing the effect of noncomparability is to consider the construction of the inner product (10.3) on $\mathcal{Z}$, which we defined by adding the inner product of the two functional parts and that of the two vector parts. In many problems, there is no intrinsic reason to give these two inner products equal weight in the sum, and a more general inner product we could consider is

$$\langle z_{(1)}, z_{(2)} \rangle = \int x_{(1)} x_{(2)} + C^2 y'_{(1)} y_{(2)} \tag{10.4}$$

for some suitably chosen constant C. Often, the choice of C (for example $C = 1$) is somewhat arbitrary, but we can make some remarks that may guide its choice.

First, if the interval $\mathcal{T}$ is of length $|\mathcal{T}|$, then setting $C^2 = |\mathcal{T}|$ gives the same weight to overall differences between $x_{(1)}$ and $x_{(2)}$ as to differences of similar size in a single component of the vector part y. If the measurements are of cognate or comparable quantities, this may well be a good method of choosing C. On the other hand, setting $C^2 = |\mathcal{T}|/M$ tends to weight differences in functional parts the same as differences in all vector components.

Another approach, corresponding to the standard method of PCA relative to correlation matrices, is to ensure that the overall variability in the functional parts is given weight equal to that in the vector part. To do this, we would set

$$C^2 = \frac{\sum_i \|x_i - \bar{x}\|^2}{\sum_i \|\mathbf{y}_i - \bar{y}\|^2},$$

taking the norm in the functional sense in the numerator, and in the usual vector sense in the denominator.

Finally, in specific problems, there may be a particular rationale for some other choice of constant C^2, an example of which is discussed in Section 10.4.

Whatever the choice of C^2, the most straightforward algorithmic approach is to construct the vector representation z of any element $z = (x, y)$ of $\mathcal{Z}$ to have last M elements Cy, rather than just y. The first K elements are the coefficients of the representation of x in an appropriate basis, as before. With this modification, we can use the algorithms set out above. Some care must be taken in interpreting the results, however, because any particular principal component weight function has to be combined with the data values using the inner product (10.4) to get the corresponding principal component scores.

10.4 Combining registration and PCA

10.4.1 Expressing the observations as mixed data

We now return to the special case of mixed data obtained by registering a set of observed curves. For the moment, concentrate on data that may be assumed to be periodic on $[0, 1]$. We suppose that an observation can be modelled as

$$x(t + \tau) = \mu(t) + \sum_j \eta_j \xi_j(t) \tag{10.5}$$

for a suitable sequence of orthonormal functions ξ_j, and where η_j are uncorrelated random variables with mean zero and variances σ_j^2. The model (10.5) differs from the usual PCA model in allowing for a shift in time τ as well as for the addition of multiples of the principal component functions. Because of the periodicity, the shifted function $x(t + \tau)$ may still be considered as a function on $[0, 1]$.

Given a data set $x_1, \ldots, x_n$, we can use the Procrustes approach set out in Chapter 7 to obtain an estimate $\hat{\mu}$ of μ and to give values of the shifts $\tau_1, \ldots, \tau_n$ appropriate to each observation. Then we can regard the data as pairs $z_i = (\tilde{x}_i, \tau_i)$, where the τ_i are the estimated values of the shift parameter and the $\tilde{x}_i$ are the shifted mean-corrected temperature curves with values $x_i(t + \tau_i) - \hat{\mu}(t)$. Recall that a consequence of the Procrustes fitting is that the $\tilde{x}_i$ satisfy the orthogonality property

$$\int \tilde{x}_i D\hat{\mu} = 0. \tag{10.6}$$

10.4.2 Balancing temperature and time shift effects

We can now consider the effect of the methodology of Section 10.3 to the mixed data z_i obtained in the registration context. We seek principal components (ξ, v) that have two effects within the model (10.5): the addition of the function ξ to the overall mean $\hat{\mu}$, together with a contribution of v to the time shift τ.

In the special case of the registration data, there is a natural way of choosing the constant C^2 that controls the balance between the functional and shift components in the inner product (10.4). Suppose that x is a function in the original data function space, and that $z = (\tilde{x}, \tau)$ is the corresponding pair in $\mathcal{Z}$, so that

$$x(t) = \hat{\mu}(t - \tau) + \tilde{x}(t - \tau).$$

Because of the orthogonality property (10.6), we can confine attention to $\tilde{x}$ that are orthogonal to $\hat{\mu}$.

To define a norm on $\mathcal{Z}$, a requirement is that, at least to first order,

$$\|z\|^2 \approx \|x - \hat{\mu}\|^2 = \int [x(s) - \hat{\mu}(s)]^2 \, ds, \tag{10.7}$$

the standard squared function norm for $x - \hat{\mu}$. This means that the norm of any small perturbation of the mean function $\hat{\mu}$ must the same, whether it is specified in the usual function space setting as $x - \hat{\mu}$, or expressed as a pair z in $\mathcal{Z}$, consisting of a perturbation $\tilde{x}$ orthogonal to $\hat{\mu}$ and a time shift.

Suppose $\|\tilde{x}\|$ and τ are small. If we let

$$C^2 = \|D\hat{\mu}\|^2, \tag{10.8}$$

then, to first order in $\|\tilde{x}\|$ and τ,

$$x(t) - \hat{\mu}(t) \approx -\tau D\hat{\mu}(t - \tau) + \tilde{x}(t - \tau).$$

By the orthogonality of $\tilde{x}$ and $D\hat{\mu}$,

$$\|z - \hat{\mu}\|^2 \approx \int \tilde{x}^2(s) + C^2\tau^2(s) \, ds = \|\tilde{x}\|^2 + C^2\|\tau\|^2, \tag{10.9}$$

as required.

With this calculation in mind, we perform our PCA of the pairs $(\tilde{x}_i, \tau_i)$ relative to the inner product (10.4) with $C^2 = \|D\hat{\mu}\|^2$, and this was the way that C was chosen in Section 10.1. The percentage of variability of each principal component due to the shift was then worked out as $100C^2v_j^2$.

The use of this value of C provides approximate compatibility between the quantification of variation caused simply by the addition of a curve to the overall mean, and variation that also involves a time shift. It therefore accounts for the similarity of the percentages of variation explained by the various components in Figures 8.2 and 10.1.

10.5 The temperature data reconsidered

10.5.1 Taking account of effects beyond phase shift

In the temperature example, the shift effect is not necessarily the only effect that can be extracted explicitly and dealt with separately in the functional principal components analysis. We can also take account of the overall annual average temperature for each weather station, and we do this by extending the model (10.5) to a model of the form

$$x(t + \tau) - \theta = \alpha + \mu(t) + \sum_j \eta_j \xi_j(t), \tag{10.10}$$

where θ is an annual temperature effect with zero population mean. The η_j are assumed to be uncorrelated random variables with mean zero. The

parameter α is the overall average temperature (averaged both over time and over the population). For identifiability we assume that $\int \mu(s)\, ds = 0$.

The data we would use to fit such a model consist of triples $(\breve{x}_i, \tau_i, \theta_i)$, where $\breve{x}_i$ are the observed temperature curves registered to one another by shifts τ_i, and with each curve modified by subtracting its overall annual average $\hat{\alpha}+\theta_i$. Here the number $\hat{\alpha}$ is the time average of all the temperatures observed at all weather stations, and the individual θ_i therefore sum to zero. Because the annual average $\hat{\alpha}+\theta_i$ has been subtracted from each curve $\breve{x}_i$, the curves $\breve{x}_i$ each integrate to zero as well as satisfying the orthogonality condition (10.6). The mean curve $\hat{\mu}$ is then an estimate of the mean of the registered curves $\breve{x}_i$, most straightforwardly the sample mean. In the hybrid data terms we have set up, the functional part of each data point is the curve $\breve{x}$, whereas the vector part is the 2-vector $(\tau_i, \theta_i)'$.

To complete the specification of (10.10) as a hybrid data principal components model, we regard τ and θ as random variables which can be expanded for the same η_j, as

$$\tau = \sum_j \eta_j v_j \text{ and } \theta = \sum_j \eta_j u_j,$$

where the v_j and u_j are fixed quantities. Thus, the jth principal component is characterized by a triple (ξ_j, v_j, u_j), constituting a distortion of the mean curve by the addition of a multiple of ξ_j, together with shifts in time and in overall temperature by the same multiples of v_j and u_j, respectively.

Just as before, we carry out a PCA of the hybrid data $\{(\breve{x}_i, \tau_i, \theta_i)\}$ with respect to a suitably chosen norm. To define the norm of a triple $(\breve{x}, \tau, \theta)$, consider the corresponding unregistered and uncorrected curve x, defined by

$$x(t+\tau) = \hat{\alpha} + \theta + \hat{\mu}(t) + \breve{x}(t).$$

Define $C_1 = \|D\hat{\mu}\|^2$ and $C_2 = |\mathcal{T}|$. Assume that $\breve{x}$ integrates to zero and satisfies (10.6).

By arguments similar to those used previously, using the standard square integral norm for $\breve{x}$,

$$\|x - \hat{\mu}\|^2 \approx \|\breve{x}\|^2 + C_1^2\tau^2 + C_2^2\theta^2.$$

Thus an appropriate definition of the norm of the triple is given by

$$\|(\breve{x}, \tau, \theta)\|^2 = \|\breve{x}\|^2 + C_1^2\tau^2 + C_2^2\theta^2.$$

In practice, a PCA with respect to this norm is carried out by the same general approach as before. For each i, the function $\breve{x}_i$ is represented by a vector $\breve{c}_i$ of its first K Fourier coefficients. The vector is augmented by the two values $C_1\tau_i$ and $C_2\theta_i$ to form the vector z_i. We then carry out a standard PCA on the augmented vectors z_i. The resulting principal component weight vectors are then unpacked into the parts corresponding to ξ_j, v_j and u_j, and the appropriate inverse transforms applied—just dividing by

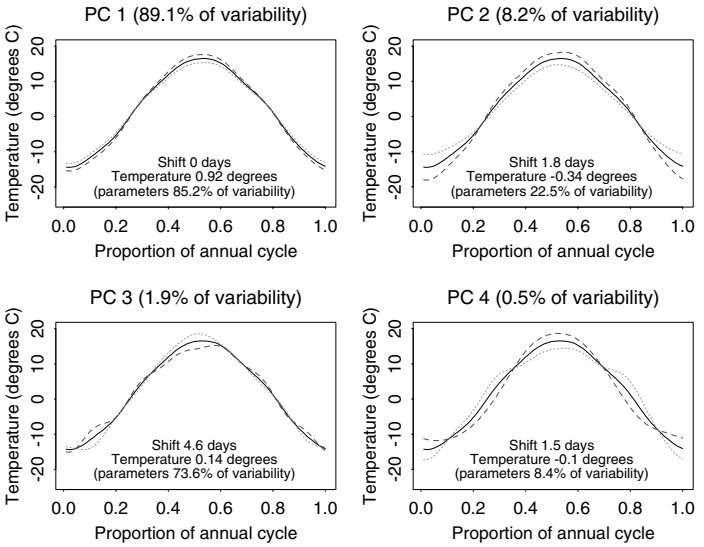

Figure 10.2. The mean Canadian temperature curve and the effect of adding and subtracting a suitable multiple of each PC curve, with the shift and annual average temperature considered as separate parameters.

C_1 and C_2 respectively in the case of the shift and overall temperature effects, and applying an inverse Fourier transform to the first K components of the vector to find ξ_j.

Figure 10.2 shows the effect of this approach applied to the Canadian temperature data. Notice that a component that was entirely variation in overall temperature would have a temperature effect of ± 1 degree, because time is scaled to make the cycle of unit length (with time measured in years) so that $C_2 = 1$. Because each principal component is scaled to have unit norm, the maximum possible value of $(C_2 u_i)^2$ is 1, with equality if and only if the other components are zero. Similarly, since $C_1 = 365/5.4$, a component that was entirely a time shift would have $v_i = \pm 5.4/365$ years, i.e., ± 5.4 days.

In each case in the figure, the proportions of variability due to the two parametric effects, shift and overall average temperature, are combined to give the percentage of variability due to the vector parameters. Principal component 1 is almost entirely due to the variation in overall temperature, with a small effect corresponding to a decrease in range between summer and winter. (Recall that the dotted curve corresponds to a positive multiple of the principal component curve ξ_i, and the dashed curve to a negative multiple.) Principal component 2 has some shift component, a moderate negative temperature effect, and mainly comprises the effect

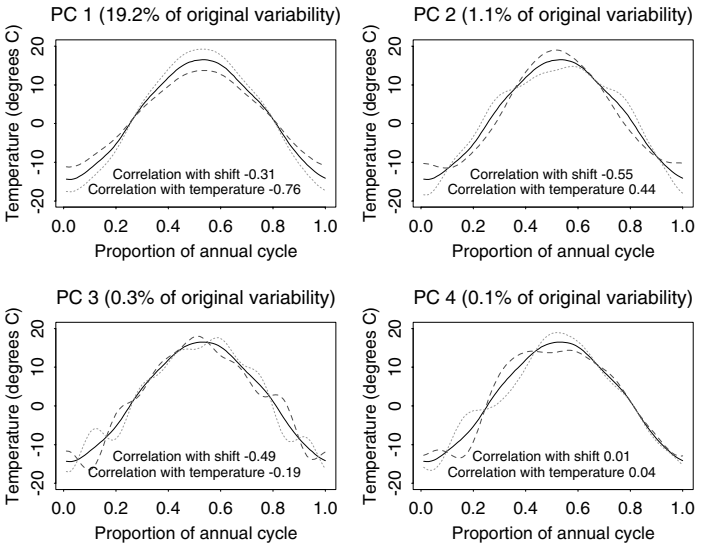

Figure 10.3. Principal component analysis carried out on the Canadian temperature curves adjusted for time shift and for annual average temperature.

of a decreased annual temperature range. Within this component, overall average temperature is positively associated with increased range, whereas in component 1 the association was negative. Principal component 1 accounts for a much larger proportion of the variability in the original data, and a slightly different approach in Section 10.5.2 shows that within the data as a whole, increased overall temperature is negatively correlated with higher range between summer and winter—colder places have more extreme temperatures.

Neither principal component 3 nor 4 contains much of an effect due to overall temperature. As before, component 3 is very largely shift, whereas component 4 corresponds to an effect unconnected to shift or overall temperature.

10.5.2 Separating out the vector component

This section demonstrates the other procedure suggested in Section 10.2. We carry out a principal components analysis on the *registered* curves $\breve{x}_i$ and then investigate the relationship between the resulting principal component scores and the parameters τ_i and θ_i arising in the registration process. Thus we analyze only the functional part of the mixed data, and the vector part is only considered later.

The effect of doing this is demonstrated in Figure 10.3. Removing the temperature and shift effects accounts for 79.2% of the variability in the original data, and the percentages of variability explained by the various principal components have been multiplied by 0.208, to make them express parts of the variability of the original data, rather than the adjusted data. For each weather station, we have a shift and annual average temperature as well as the principal component scores. Figure 10.3 shows the correlations between the score on the relevant principal component and the two parameters estimated in the registration.

We see that the components 3 and 4 in this analysis account for very little of the original variability and have no clear interpretation. Component 1 corresponds to an increase in range between winter and summer—the effect highlighted by component 2 in the previous analysis. We see that this effect is strongly negatively correlated with annual average temperature, and mildly negatively correlated with shift. Component 2 corresponds approximately to component 4 in the previous analysis, and is the effect whereby the length of summer is lengthened relative to that of winter. This effect is positively correlated with average temperature and negatively correlated with shift.

11

Canonical correlation and discriminant analysis

11.1 Introduction

11.1.1 The basic problem

In this chapter, we continue our consideration of exploratory approaches to functional data, specifically the case where we have observed *pairs* of functions $(X_i, Y_i), \quad i = 1, \ldots, N$, such as the hip and knee angles for the gait cycles of a number of children as discussed in Chapters 1 and 8. Suppose we wanted to know how variability in the knee angle cycle is related to that in the hip angle. In Section 8.5 we saw how principal components analysis can examine the variability in the two sets of curves taken together, but we did not explicitly address the issue of interaction between the two curves. In this chapter, we pursue a somewhat different emphasis by considering *canonical correlation analysis* (CCA), which seeks to investigate which modes of variability in the two sets of curves are most associated with one another.

In the functional context, canonical correlation analysis provides a pair of functions $(\xi(s), \eta(s))$ such that $\int \xi X_i$ and $\int \eta Y_i$ are well correlated with one another. We can think of $\xi(s)$ and $\eta(s)$ as the components of variation in the two curves that most account for the interaction between the hip and knee angles. Our method gives the curves shown in Figure 11.1. The values $\int \xi X_i$ and $\int \eta Y_i$ are called *canonical variates*, and the sample correlation between these variates is about 0.81 in this case.

In the figure, the curves ξ and η are rather similar, and the broad interpretation is that there is correlation between the two measurements $X_i(s)$

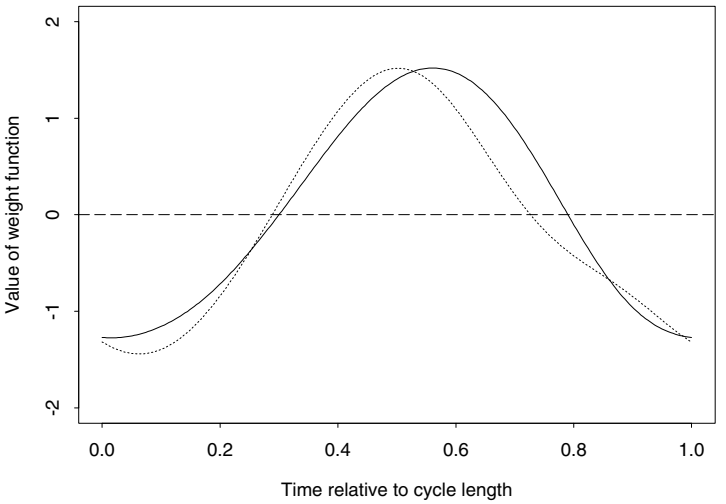

Figure 11.1. Estimated canonical variate weight functions for the gait data. Solid curve: weight function for hip observations; dotted curve: weight function for knee observations.

and $Y_i(s)$ at any particular time. But it is interesting that the extreme in the hip curve in the middle of the cycle occurs a little later than that in the knee curve, whereas the order of the extremes near the beginning of the cycle is reversed. This suggests that, in the middle of the cycle, high variability from the norm in the hip follows that in the knee; near the ends of the cycle, the effects occur in the opposite order. This may indicate a physical propagation of errors caused by the relevant strike of the heel at the beginning and in the middle of the cycle.

Having found these components of variability, we can go on to find further components of variation. Call the (ξ, η) we have already found (ξ_1, η_1). We can now look for another pair of functions (ξ_2, η_2) such that

- There is a high correlation between the variation in the hip angles described by a multiple of ξ_2 and that in the knee angles accounted for by η_2, but ...
- these effects are uncorrelated with the previously found contributions to variability corresponding to ξ_1 and η_1.

The functions ξ_2 and η_2 are shown in Figure 11.2. In this case the correlation between $\int \xi_2 X_i$ and $\int \eta_2 Y_i$ is about 0.72, only slightly lower than that for the first pair of canonical variates. The points at which the functions ξ_2 and η_2 cross the axis indicate conclusions similar to those outlined with respect to the leading variates. In the middle of the cycle the hip curve

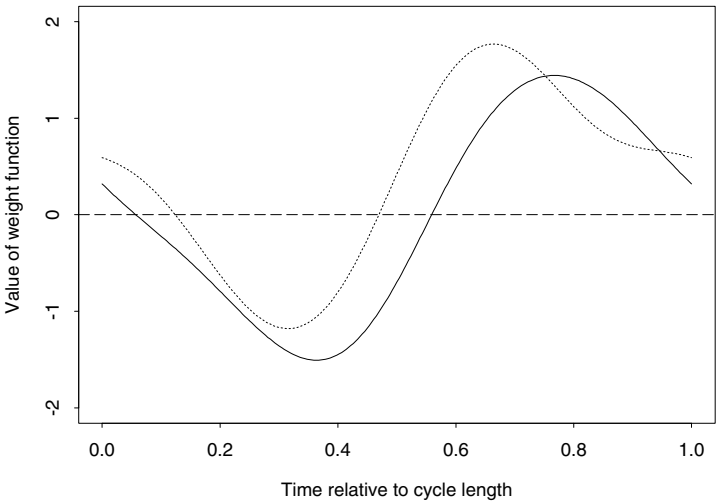

Figure 11.2. Second pair of smoothed canonical variate weight functions for the gait data. Solid curve: weight function for hip observations; dashed curve: weight function for knee observations.

crosses zero considerably later than the knee curve, whereas near the beginning of the cycle the hip curve crosses first. Put another way, we could roughly transform both the first and the second canonical variates to be identical for the hip and the knee by speeding up the hip cycle relative to the knee cycle in the first half of the cycle, and slowing it down in the second.

We shall see that the estimation of the weight functions as shown in Figures 11.1 and 11.2 is not quite straightforward and that an appropriate form of smoothing is essential. But first we review classical multivariate CCA; a fuller discussion can be found in most multivariate analysis textbooks, such as Anderson (1984). We then go on to develop our approach to functional CCA, largely based on the paper of Leurgans, Moyeed and Silverman (1993), and using the gait data as a running example. Another application is considered in Section 11.4. We shall see that some regularization is essential to obtain meaningful results, for reasons discussed briefly in Section 11.5. In Section 11.6, various algorithmic approaches and connections with other FDA topics are explored.

Finally, in Section 11.7, we present some extensions of the ideas of functional CCA to deal with problems of optimal scoring and discriminant analysis. This is based on work of Hastie, Buja and Tibshirani (1995).

11.2 Principles of classical canonical correlation analysis

Suppose we have n pairs of observed vectors (x_i, y_i), each x_i being a p-vector and each y_i being a q-vector. The object of canonical correlation analysis is to reduce the dimensionality of the data by finding the vectors $\mathbf{a}_1$ and $\mathbf{b}_1$ (p- and q-vectors respectively) for which the linear combinations $\mathbf{a}_1' x_i$ and $\mathbf{b}_1' y_i$ are as highly correlated as possible. The *canonical variates* $\mathbf{a}_1' x_i$ and $\mathbf{b}_1' y_i$ are the linear compounds of the original observations whose variability is most closely related in terms of correlation. The vectors $\mathbf{a}_1$ and $\mathbf{b}_1$ are called the *leading canonical variate weight vectors*.

Note that multiplying $\mathbf{a}_1$ and/or $\mathbf{b}_1$ by nonzero constants of the same sign does not alter the correlation. If the constants are opposite in sign, the correlation itself is reversed in sign but has the same magnitude. By convention, we choose $\mathbf{a}_1$ and $\mathbf{b}_1$ so that $\{\mathbf{a}_1' x_i\}$ and $\{\mathbf{b}_1' y_i\}$ both have sample variance equal to 1, and the correlation ρ_1 between the $\mathbf{a}_1' x_i$ and $\mathbf{b}_1' y_i$ is positive.

We can now go on to find subsidiary canonical variates. The jth pair of canonical variates is defined by a p-vector a_j and a q-vector b_j, chosen to maximize the sample correlation $\rho_j = \mathtt{corr}(\mathbf{a}_j' x_i, \mathbf{b}_j' y_i)$ subject to the constraints that

(a) $\mathtt{corr}(\mathbf{a}_j' x_i, \mathbf{a}_k' x_i) = 0$

(b) $\mathtt{corr}(\mathbf{b}_j' y_i, \mathbf{b}_k' y_i) = 0$

(c) $\mathtt{corr}(\mathbf{a}_j' x_i, \mathbf{b}_k' y_i) = 0,$

where in each case the correlations are the sample correlations as i takes the values $1, \ldots, n$.

11.3 Functional canonical correlation analysis

11.3.1 Notation and assumptions

We now return to the functional case, which is our main concern. As usual, assume that the N observed pairs of data curves (X_i, Y_i) are available for argument t in some finite interval $\mathcal{T}$, and that all integrals are taken over $\mathcal{T}$. Given functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$, we define $\mathtt{ccorsq}(\boldsymbol{\xi}, \boldsymbol{\eta})$ to be the sample squared correlation of $\int \boldsymbol{\xi} X_i$ and $\int \boldsymbol{\eta} Y_i$, and therefore

$$\mathtt{ccorsq}(\boldsymbol{\xi}, \boldsymbol{\eta}) = \frac{\{\mathtt{cov}(\int \xi X_i, \int \eta Y_i)\}^2}{(\mathtt{var} \int \xi X_i)(\mathtt{var} \int \eta Y_i)}.$$

The use of a roughness penalty is central to our methodology. As usual we quantify the roughness of a function f by its integrated squared curvature $\|D^2 f\|^2 = \int (D^2 f)^2$.

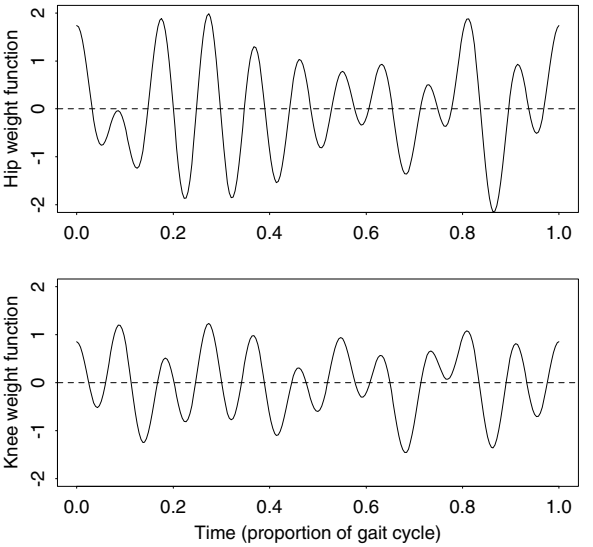

Figure 11.3. Unsmoothed canonical variate weight functions for the gait data that attain perfect correlation. Top panel: weight function for hip observations; bottom panel: weight function for knee observations.

11.3.2 The naive approach does not give meaningful results

For the moment concentrate on the leading canonical variates. We might imagine that the obvious way to proceed is simply to find functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$ that maximize $\texttt{ccorsq}(\boldsymbol{\xi}, \boldsymbol{\eta})$. This would be equivalent to maximizing $\texttt{cov}(\int \xi X_i, \int \eta Y_i)$ subject to the constraints

$$\texttt{var}(\textstyle\int \xi X_i) = \texttt{var}(\int \eta Y_i) = 1. \tag{11.1}$$

However, simply carrying out this maximization does not produce a meaningful result. Figure 11.3 shows functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$ that maximize the sample correlation `ccorsq` for the gait data example. The sample correlation achieved by these functions is 1. The functions displayed in Figure 11.3 do not give any meaningful information about the data and clearly demonstrate the need for a technique involving smoothing. In Section 11.5, we explain why this behavior is not specific to this particular data set but is an intrinsic property of CCA applied in the functional context.

A straightforward way of introducing smoothing is to modify the constraints (11.1) by adding roughness penalty terms to give

$$\texttt{var}(\textstyle\int \xi X_i) + \lambda \|D^2 \boldsymbol{\xi}\|^2 = \texttt{var}(\int \eta Y_i) + \lambda \|D^2 \boldsymbol{\eta}\|^2 = 1, \tag{11.2}$$

where λ is a positive smoothing parameter.

The effect of introducing the roughness penalty terms into the constraints is that, in evaluating particular candidates to be canonical variates, we

consider not only their variances, but also their roughness, and compare a weighted sum of these two quantities with the covariance term. The problem of maximizing the covariance $\mathtt{cov}(\int \xi X_i, \int \eta Y_i)$ subject to the constraints (11.2) is equivalent to maximizing the penalized squared sample correlation defined by

$$\mathtt{ccorsq}_\lambda(\boldsymbol{\xi}, \boldsymbol{\eta}) = \frac{\{\mathtt{cov}(\int \xi X_i, \int \eta Y_i)\}^2}{\{\mathtt{var}(\int \xi X_i) + \lambda \|D^2 \boldsymbol{\xi}\|^2\}\{\mathtt{var}(\int \eta Y_i) + \lambda \|D^2 \boldsymbol{\eta}\|^2\}}. \tag{11.3}$$

We refer to this procedure as *smoothed canonical correlation analysis*.

Our method of introducing smoothing or regularization is similar to the technique of ridge regression, which is often used in image processing and ill-posed problems to improve the conditioning of the variance matrices considered. The technique of ridge regression was applied to CCA by Vinod (1976). Multiplying the curves ξ and η by constants does not affect the value of the criterion $\mathtt{ccorsq}_\lambda(\boldsymbol{\xi}, \boldsymbol{\eta})$, and in the figures they are normalized to set $\int \xi^2 = \int \eta^2 = 1$.

11.3.3 Choice of the smoothing parameter

The larger the value of λ, the more emphasis is placed on the roughness penalty and the smaller will be the true correlation of the variates found by smoothed CCA. A good choice of the smoothing parameter is essential to give a pair of canonical variates with fairly smooth weight functions and a correlation that is not unreasonably low. The smoothing parameter can be chosen subjectively, but if we require an automatic procedure, a reasonable form of cross-validation is as follows:

Let $\mathtt{ccorsq}_\lambda^{-i}(\boldsymbol{\xi}, \boldsymbol{\eta})$ be the sample penalized squared correlation calculated as in (11.3) but with the observation (X_i, Y_i) omitted. Let $(\boldsymbol{\xi}_\lambda^{(-i)}, \boldsymbol{\eta}_\lambda^{(-i)})$ be the functions that maximize $\mathtt{ccorsq}_\lambda^{-i}(\boldsymbol{\xi}, \boldsymbol{\eta})$. The cross-validation score for λ is defined to be the squared correlation of the N pairs of numbers

$$(\int \boldsymbol{\xi}_\lambda^{(-i)} X_i, \int \boldsymbol{\eta}_\lambda^{(-i)} Y_i)$$

for $i = 1, \ldots, n$. We then choose λ to maximize this correlation. It is this choice of λ that was used for the gait data in Figures 11.1 and 11.2. The degree of smoothing chosen by cross-validation appears to be quite heavy, and to test the sensitivity of these conclusions, Leurgans, Moyeed and Silverman (1993) examined the first two pairs of canonical variates estimated with a value of λ reduced by a factor of 10. Though there was a little more variability in the canonical variate curves, the broad features remained the same.

Throughout this section, we have concentrated on the choice of smoothing parameter for the leading canonical variates. If we were particularly interested in the ideal smoothing parameter for a subsidiary canonical cor-

Table 11.1. Smoothed and unsmoothed sample correlations for the first three pairs of smoothed canonical variates for the gait data.

Canonical variates	Sample squared correlations	
	$\texttt{ccorsq}_\lambda(\boldsymbol{\xi}_\lambda, \boldsymbol{\eta}_\lambda)$	$\texttt{ccorsq}(\boldsymbol{\xi}_\lambda, \boldsymbol{\eta}_\lambda)$
First	0.755	0.810
Second	0.618	0.717
Third	0.141	0.198

relation, we could formulate a relevant cross-validation score. However, our practical experience has shown us that, although cross-validation works well for the leading canonical variate, its behavior is much more disappointing for subsequent canonical variates. We have found it to be more satisfactory simply to use the same value of λ for any subsidiary canonical variates considered.

We have used a single smoothing parameter λ for both ξ and η. It is possible to use separate smoothing parameters λ_1 and λ_2; the conceptual and algorithmic extensions are straightforward, but we have found a single smoothing parameter to be adequate in the examples we have considered.

11.3.4 The values of the correlations

Once the canonical variates have been found, we can consider the values of the correlations themselves. We can consider either the smoothed squared correlation $\texttt{ccorsq}_\lambda$ or the unsmoothed value $\texttt{ccorsq}$; there is no firm theoretical footing for the choice between them and in any case it would be a matter of some concern if the effect of smoothing was to make the values dramatically different.

For the gait data, Table 11.1 shows the values of the smoothed and unsmoothed squared correlations, and also includes corresponding values for the second and third pairs of smoothed canonical variates, estimated with the same λ. Table 11.1 shows that the second pair of canonical variates is almost as important as the first. On the other hand, the third pair of canonical variates have low estimated correlation, and we do not consider them further.

Before we leave the gait example, we note that scatterplots of the canonical variate scores $(\int \boldsymbol{\xi} X_i, \int \boldsymbol{\eta} Y_i)$ show that no particular curves have outlying scores for either of the first two canonical variates. In Section 8.5, we saw that the first principal component of variation in the hip curves alone corresponded to an overall vertical shift in the curves. If this shift were in any way correlated with a variation in the knee curves, the hip canonical variate curves would be more like constants than sine waves. Since this is not the case, we can see that this vertical shift is a property of the hip curves alone, independent of any variation in the knee angles.

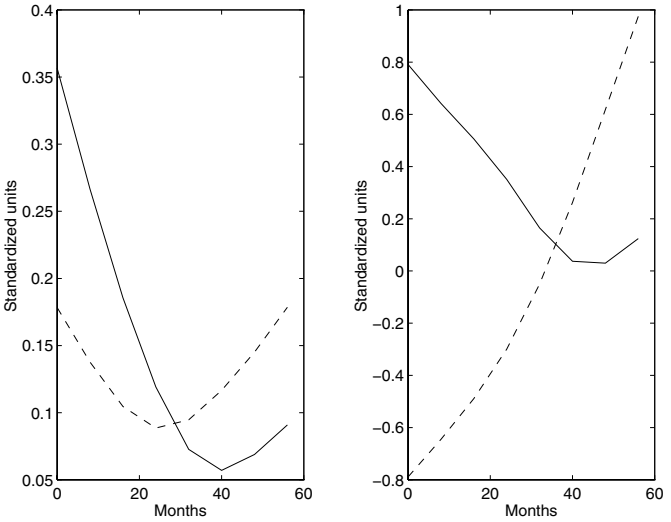

Figure 11.4. Smoothed canonical variate weight functions for the lupus data, from Buckheit et al. (1997). Left panel: results of CCA applied to GFR and KUC with solid curve corresponding to GFR and dashed curve to KUC. Right panel: results of CCA applied to GFR and GOP, with solid curve corresponding to GFR and dashed curve to GOP.

11.4 Application to the study of lupus nephritis

Buckheit, Olshen, Blouch and Myers (1997) applied functional CCA to renal physiology, in the study of diffuse proliferative lupus nephritis, and we present their results here as an illustration. The original paper should be consulted for further details; we are extremely grateful to Richard Olshen for his generosity in sharing and discussing this work with us prior to its publication.

They had available various measurements on a number of patients over a 60-month period. These include the glomerular filtration rate (GFR), the glomerular oncotic pressure (GOP) and the two-kidney ultrafiltration coefficient (KUC). They focused on nine patients labelled *progressors*, those whose kidney function, as measured by GFR was clearly declining over the period of study. The GFR measure is currently favored by clinicians as an overall indicator of progressive glomerular disease, a particular form of kidney degeneration, and therefore the progressors are the group suffering long-term kidney damage, likely to require eventual dialysis or transplantation. It is important to understand the kidney filtration dynamics in this disease, and this is facilitated by investigating the covariation between measured variables.

Within the progressor group, GFR and KUC tend to decrease considerably over the 60 month period, whereas the GOP measure increases somewhat. This contrasts with well-functioning kidneys, where an increase in GOP would be counteracted by an increase in KUC, resulting in steady GFR. Functional smoothed CCA was applied to explore variability and interaction effects in the progressor group. The correlations between GFR and each of KUC and GOP were investigated. Figure 11.4 shows the leading pairs of canonical variate weight functions. It is interesting that the linear functional of GFR most highly correlated with the other two variables is virtually the same in both cases.

To interpret the figure, remember that all patients concerned show an overall declining value of GFR. The U-shaped solid curves in the figure therefore correspond to a canonical variate where a positive value indicates a GFR record that starts at a value higher than average, but then declines more rapidly than average in the first 40 months, finally switching to a relatively less rapid decline in the last 20 months.

The left-hand panel shows that this variate is correlated with a similar effect for KUC, but the switch in rate happens earlier. This indicates not only that strong decline of GFR is associated with strong decline of KUC but also suggests that the pattern of GFR in some sense follows that of KUC, raising the hope that KUC could be used to predict future GFR behavior. On the other hand, the right-hand panel shows that this aspect of GFR behavior is correlated with an increase of GOP stronger than average over the entire time period. Thus, patients with rapidly increasing GOP are likely to be those whose GFR declines rapidly at first, though there may be some reduction in the rate of decline after about 36 months.

In broad terms, the CCA gives insights broadly consistent with those for the average behavior of the sample as a whole. It is interesting that the relationships between the variables are borne out on an individual level, not merely on an average level. Furthermore the detailed conclusions yielded by the CCA give important avenues for future thought and investigation concerning the way in which the variables interrelate. Of course, given the small sample size, any conclusions must be relatively tentative unless supported by other evidence.

11.5 Why is regularization necessary?

Apart from its importance as a practical method, canonical correlation analysis of functional data has an interesting philosophical aspect. In the principal components analysis context we have already seen that appropriately applied smoothing may improve the estimation accuracy. However, in most circumstances, we obtain reasonable estimates of the population principal components even if no smoothing is applied. By contrast, as we saw

in the gait example, in the context of functional CCA some regularization is absolutely essential to obtain meaningful results. This is the same conclusion that we will draw for the functional regression context discussed in Chapter 16. But in the canonical correlation case, the impact of smoothing is even more dramatic.

To understand the need for regularization, compare functional CCA with standard multivariate CCA. A standard condition of classical CCA is that $n > p + q + 1$ which ensures (with probability 1, under reasonable conditions) that the sample covariance matrix $\mathbf{V}_{12}$ of the n vectors (x_i, y_i) is nonsingular (see Eaton and Perlman,1973). In the functional case, p and q are essentially infinite, and so this condition cannot be fulfilled.

Furthermore, consider a sample $X_1, \ldots, X_N$ of functional data, and assume for the moment that the N curves are linearly independent. Now suppose that $z_1, \ldots, z_N$ is any real vector. By results that will be discussed in Chapter 16, it is possible to find a curve $\boldsymbol{\xi}$ such that, for some constant α_X, $z_i = \alpha_X + \int \boldsymbol{\xi} X_i$ for all i. Essentially, the reason for this is that we only have N constraints on ξ, but infinitely many degrees of freedom in the choice of ξ, because ξ is a function. Now suppose we have a second sample of curves Y_i, which may be correlated with the X_i in some way, and again are linearly independent. We can find a function $\boldsymbol{\eta}$ such that, for some constant α_Y, $z_i = \alpha_Y + \int \boldsymbol{\eta} Y_i$ for all i. This means that the given values z_i can be predicted perfectly either from the X_i or from the Y_i.

It follows that not only have we found functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$ such that $\texttt{ccorsq}(\boldsymbol{\xi}, \boldsymbol{\eta}) = 1$, because the variates $\int \boldsymbol{\xi} X_i$ and $\int \boldsymbol{\eta} Y_i$ are perfectly correlated, but that we can prescribe the values z_i taken by the canonical variates to be whatever we please, up to a constant. In particular, we could start with *any* function $\boldsymbol{\xi}$, construct $z_i = \int \boldsymbol{\xi} X_i$, and then find a function $\boldsymbol{\eta}$ such that $\texttt{ccorsq}(\boldsymbol{\xi}, \boldsymbol{\eta}) = 1$. In this sense, every possible function can arise as a canonical variate weight function with perfect correlation!

Leurgans, Moyeed and Silverman (1993) discuss this result in greater detail. They demonstrate that the assumption of linear independence among the curves is a very mild one, and, by proving an appropriate consistency result, they show that regularization indeed makes meaningful estimates possible.

11.6 Algorithmic considerations

11.6.1 Discretization and basis approaches

There are several ways of carrying out our method of smoothed functional CCA numerically. For completeness, we present the methodology for the general case of different parameters λ_1 and λ_2. A direct approach is to set up a discrete version of the covariance `ccorsq` and of the constraints (11.2). Discretize the functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$ and the covariance operators $v_{jk}(s, t)$ using

a fine grid, and replace the operator D^2 by a finite difference approximation. The problem then becomes one of maximizing a quadratic form subject to quadratic constraints, and it can be solved by standard numerical methods.

We can also use a basis for the functions X_i and Y_i, and for the weight functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$. Suppose that $\phi_1, \phi_2, \ldots, \phi_M$ is a suitable basis, which for simplicity we will assume is used for all of these four functions. As usual, define $\mathbf{K}$ to be the matrix with entries $\int (D^2\phi_j)(D^2\phi_k)$ and $\mathbf{J}$ the matrix with entries $\int \phi_j \phi_k$. If we use a Fourier or other orthonormal basis, then $\mathbf{J}$ is the identity matrix.

Define $\mathbf{C}$ and $\mathbf{D}$ to be the matrices of coefficients of the basis expansions of the X_i and Y_i respectively, meaning that

$$X_i = \sum_{\nu=1}^{M} c_{i\nu}\phi_\nu$$

and

$$Y_i = \sum_{\nu=1}^{M} d_{i\nu}\phi_\nu$$

up to the degree of approximation involved in any choice of the number M of basis functions considered. Write $\mathbf{a}$ and $\mathbf{b}$ for the vectors of coefficients of the basis expansions of the functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$.

Define $M \times M$ covariance matrices $\tilde{\mathbf{V}}_{11}$, $\tilde{\mathbf{V}}_{12}$ and $\tilde{\mathbf{V}}_{22}$ to be the matrices with (ν, ρ) entries

$$N^{-1}\sum_i c_{i\nu}c_{i\rho}, \;\; N^{-1}\sum_i c_{i\nu}d_{i\rho}, \;\text{ and }\; N^{-1}\sum_i d_{i\nu}d_{i\rho},$$

respectively, the sample variance and covariance matrices corresponding to the basis expansions of the data. It can be shown that, in the basis expansion domain, we carry out the smoothed CCA of the given data by solving the generalized eigenvalue problem

$$\begin{bmatrix} \mathbf{0} & \mathbf{J}\tilde{\mathbf{V}}_{12}\mathbf{J} \\ \mathbf{J}\tilde{\mathbf{V}}_{21}\mathbf{J} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} = \rho \begin{bmatrix} \mathbf{J}\tilde{\mathbf{V}}_{11}\mathbf{J} + \lambda_1\mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}\tilde{\mathbf{V}}_{22}\mathbf{J} + \lambda_2\mathbf{K} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}.$$

As in Chapter 14, we should choose the number of basis functions M large enough to ensure that the regularization is controlled by the choice of the smoothing parameter(s) λ rather than that of dimensionality M. Values of M of around 20 should give good results without imposing an excessive computational burden.

11.6.2 The roughness of the canonical variates

A third algorithmic possibility is related to the idea of quantifying of the roughness of a variate, as discussed in Chapter 5. Just as in the case of smoothing data, this idea is of both conceptual and algorithmic value,

and can be used to elucidate the regularization method we propose for functional canonical correlation analysis.

Suppose $z_i = \int \boldsymbol{\xi} X_i$ is a possible canonical variate value, and let $\mathbf{z}$ be the N-vector containing these values. Let $\mathbf{R}_X$ be the matrix $\mathbf{R}$ as derived in Section 15.7.3, implying that $\mathbf{z}'\mathbf{R}_X\mathbf{z}$ is the roughness of the smoothest function ξ such that $\int \xi X_i = z_i$ for all i. It may be that $\mathbf{z}'\mathbf{R}_X\mathbf{z}$ is equal to $\|D^2\boldsymbol{\xi}\|^2$, or it may be that z_i can be obtained by integrating a smoother function against the X_i. In any case, we can consider $\mathbf{z}'\mathbf{R}_X\mathbf{z}$ in its own right as a measure of the roughness of z_i as a variate based on the X_i.

Similarly, let $\mathbf{R}_Y$ be a matrix such that the roughness of any vector of canonical variate values $\mathbf{w}$ relative to the observed covariate functions $\{Y_i\}$ is $\mathbf{w}'\mathbf{R}_Y\mathbf{w}$. Our smoothed canonical correlation method can then be recast as the determination of vectors $\mathbf{z}$ and $\mathbf{w}$ to maximize the sample covariance of z_i and w_i subject to

$$\mathtt{var}\{z_i\} + \lambda_1\mathbf{z}'\mathbf{R}_X\mathbf{z} = \mathtt{var}\{w_i\} + \lambda_2\mathbf{w}'\mathbf{R}_Y\mathbf{w} = 1. \tag{11.4}$$

Once we have found in this way a pair of canonical variates, the corresponding weight functions are defined as the smoothest functions $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$ satisfying $z_i = \int \boldsymbol{\xi} X_i$ and $w_i = \int \boldsymbol{\eta} Y_i$ for all i.

We can maximize the sample covariance of $\{z_i\}$ and $\{w_i\}$ subject to the constraints (11.4) by solving an eigenvalue problem. Some care is necessary to deal with a slight complication caused by the presence of the sample mean in the formula for variance and covariance.

Assuming without loss of generality that the canonical variates have sample mean zero, write the constrained maximization problem as that of finding the maximum of $\mathbf{z}'\mathbf{w}$ subject to the constraints

$$\mathbf{z}'\mathbf{z} + \lambda_1\mathbf{z}'\mathbf{R}_X\mathbf{z} = \mathbf{w}'\mathbf{w} + \lambda_2\mathbf{w}'\mathbf{R}_Y\mathbf{w} = 1 \tag{11.5}$$

and the additional constraints

$$1'\mathbf{z} = 1'\mathbf{w} = 0. \tag{11.6}$$

For the moment, neglect the constraint (11.6) and consider the maximization of $\mathbf{z}'\mathbf{w}$ subject only to the constraints (11.5). This corresponds to the eigenvalue problem

$$\begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \end{bmatrix} = \rho \begin{bmatrix} \mathbf{I} + \lambda_1\mathbf{R}_X & \mathbf{0} \\ \mathbf{0} & \mathbf{I} + \lambda_2\mathbf{R}_Y \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \end{bmatrix}. \tag{11.7}$$

By premultiplying (11.7) by $[\mathbf{z}' \;\; \mathbf{w}']$ and taking the product of the two expressions for $\mathbf{z}'\mathbf{w}$ thus obtained, any solution of (11.7) satisfies

$$(\mathbf{z}'\mathbf{w})^2 = \rho^2(\mathbf{z}'\mathbf{z} + \lambda_1\mathbf{z}'\mathbf{R}_X\mathbf{z})(\mathbf{w}'\mathbf{w} + \lambda_2\mathbf{w}'\mathbf{R}_Y\mathbf{w}) \geq \rho^2(\mathbf{z}'\mathbf{z})(\mathbf{w}'\mathbf{w})$$

and so it is necessarily the case that $|\rho| \leq 1$. Since the smoothest functional interpolant of the constant vector has roughness zero, $\mathbf{R}_X 1 = \mathbf{R}_Y 1 = 0$, and so the condition $\mathbf{z} = \mathbf{w} = 1$ yields the leading solution of (11.7), with eigenvalue $\rho = 1$.

The solution of (11.7) with the *second* largest eigenvalue maximizes $\mathbf{z}'\mathbf{w}$ subject to the constraint (11.5) and the additional constraint

$$1'(\mathbf{I} + \lambda_1 \mathbf{R}_X)\mathbf{z} = 1'(\mathbf{I} + \lambda_2 \mathbf{R}_Y)\mathbf{w} = 0. \tag{11.8}$$

But since $\mathbf{R}_X 1 = \mathbf{R}_Y 1 = 0$, the constraint (11.8) is precisely equivalent to the constraint (11.6) that we temporarily neglected. It follows that the second and subsequent eigensolutions of (11.7) are the canonical variates we require, and automatically have sample mean zero; the leading solution is a constant and should be ignored.

11.7 Penalized optimal scoring and discriminant analysis

Hastie, Buja and Tibshirani (1995) consider functional forms of the multivariate techniques of optimal scoring and linear discriminant analysis, making use of ideas closely related to the functional canonical correlation analysis approach discussed in this chapter. We present a brief overview of their work; see the original paper for further details.

11.7.1 The optimal scoring problem

Assume that we have N paired observations (X_i, y_i) where each X_i is a function, and each y_i is a category or class taking values in the set $\{1, 2, \ldots, J\}$. For notational convenience, we code each y_i as a J-vector $\mathbf{y}_i$ with value 1 in position j if $y_i = j$, and 0 elsewhere.

We aim to obtain a function β and a J-vector $\boldsymbol{\theta}$ minimizing the criterion

$$\texttt{OSERR}(\boldsymbol{\theta}, \beta) = N^{-1} \sum_{i=1}^{N} (\int \beta X_i - \boldsymbol{\theta}' \mathbf{y}_i)^2$$

subject to the normalization constraint $N^{-1} \sum_i (\boldsymbol{\theta}' \mathbf{y}_i)^2 = 1$. The idea is to turn the categorical variable coded by the y-vectors into a quantitative variable taking the values θ_j. The θ_j are the scores for the various categories, chosen to give the best available prediction of a linear property $\int \beta X$ of the observed functional data.

For any given $\boldsymbol{\theta}$, the problem of finding the functions β is that of finding a function which satisfies a finite number of linear constraints. Because there are infinitely many degrees of freedom in the choice of a function, it is usually possible to choose β to give perfect prediction of any specified values $\boldsymbol{\theta}' \mathbf{y}_i$. This means that we cannot choose an optimal score vector $\boldsymbol{\theta}$ uniquely on the basis of the observed data. To deal with this difficulty, Hastie et al. (1995) introduced the *penalized* optimal scoring criterion

$$\texttt{OSERR}_\lambda(\boldsymbol{\theta}, \beta) = \texttt{OSERR}(\boldsymbol{\theta}, \beta) + \lambda \times \texttt{PEN}(\beta),$$

where λ is a smoothing parameter and $\texttt{PEN}(\beta)$ a roughness penalty.

11.7.2 The discriminant problem

The discriminant problem is similar to the optimal scoring problem. Again, we have functional observations X_i, each allocated to a category in $\{1, 2, \ldots, J\}$. For any proposed linear discriminant functional $\int \beta X_i$, define θ_j to be the average of the $\int \beta X_i$ for all X_i falling in category j. For each fixed β, this value of $\boldsymbol{\theta}$ minimizes the quantity $\texttt{OSERR}(\boldsymbol{\theta}, \beta)$, which can then be re-interpreted as the *within-class variance* of the $\int \beta X_i$. The *between-class variance* is simply the variance of the discriminant class means $\boldsymbol{\theta}'\mathbf{y}_i$, defining the J-vectors $\mathbf{y}_i$ by the same coding as above. Discriminant analysis aims to maximize the between-class variance subject to a constraint on the within-class variance.

The roles of objective function and constraint are exchanged in passing from optimal scoring to discriminant analysis, and minimization is replaced by maximization. Also, primary attention shifts from the score vector $\boldsymbol{\theta}$ in optimal scoring to the discriminant functional defined by the function β in discriminant analysis. Hastie et al. make the correspondence complete by proposing *penalized discriminant analysis* where we maximize the raw between-class variance subject to a penalized constraint on the within-class variance

$$\texttt{OSERR}(\boldsymbol{\theta}, \beta) + \lambda \times \texttt{PEN}(\beta) = 1.$$

11.7.3 The relationship with CCA

Simple modifications of arguments from multivariate analysis show that the penalized optimal scoring and the penalized discriminant analysis problems are both equivalent to the mixed functional-multivariate canonical correlation analysis problem of maximizing the covariance of $\int \xi X_i$ and $\boldsymbol{\eta}'\mathbf{y}_i$ subject to the constraints

$$\texttt{var}(\int \xi X_i) + \lambda \times \texttt{PEN}(\xi) = \texttt{var}(\boldsymbol{\eta}'\mathbf{y}_i) = 1. \tag{11.9}$$

In the notation we have used for CCA, the weight corresponding to the functional part X_i of the data is itself a function ξ, whereas the vector part $\mathbf{y}_i$ is mapped to its canonical variate by a weight *vector* $\boldsymbol{\eta}$. Only the functional part ξ is penalized for roughness in the constraints (11.9). The numerical approaches we have set out for CCA carry over to this case, with appropriate modifications because only the X_i are functions.

To obtain the solutions $(\beta, \boldsymbol{\theta})$ of the discriminant and optimal scoring problems, it is only necessary to rescale the estimated function ξ and vector $\boldsymbol{\eta}$ appropriately. The subsidiary variates are also interesting for these problems because they yield estimates of vector-valued scores θ_j and discriminants $\int \beta X_i$.

11.7.4 Applications

Hastie et al. present two fascinating applications of these techniques. For speech recognition, the frequency spectra of digitized recordings of various phonemes are used as data. A roughness penalty of the form $\texttt{PEN}(\beta) = \int\{D^2\beta(\omega)\}^2 w(\omega)d\omega$ is used, with the weight function $w(\omega)$ chosen to place different emphasis on different frequencies ω.

Their other application is the recognition of digits in handwritten postal addresses and zip codes. In this case, the observations X_i are functions of a bivariate argument t, defined in practice on a 16×16 pixel grid. The roughness penalty used is a discrete version of the Laplacian penalty $\int\int[\nabla^2\beta(t)]^2\,dt$.

11.8 Further readings and notes

The idea of canonical correlation between two function spaces has a rather substantial history. Lancaster (1969) is considered an early statement of the problem, considered in the context of a treatment of the chi-squared distribution. Caillez, F. and Pagès, J. P. (1976) and Dauxois and Pousse (1976) are two explorations in French of functional canonical correlation, the first being directed to applied statisticians, and the second being a severely abstract treatise that is yet to be published in the conventional sense. A recent contribution on the theoretical side is He, Müller and Wang (2003). Dauxois and Nkiet (2002) discuss some generalizations of canonical correlation analysis within a Hilbert space framework.

12
Functional linear models

12.1 Introduction

We have been exploring the variability of a functional variable without asking how much of its variation is explainable by other variables. It is now time to consider the use of covariates. In classical statistics, the analysis of variance, linear regression and the general linear model serve this purpose, and we now extend the notion of a linear model to the functional context.

Linear models can be functional in one or both of two ways:

1. The dependent or response variable x with argument t is functional.

2. One or more of the independent variables or covariates z is functional.

We will see in Chapter 13 that predicting a functional response with values $x(t)$ by a conventional design matrix (*functional analysis of variance*) or by a set of scalar variables (*functional multiple regression*) involves a fairly straightforward modification of ways of thinking and computational strategies already familiar in ordinary analysis of variance or multiple regression. The main change is that regression coefficients now become regression coefficient functions with values $\beta_j(t)$.

On the other hand, when one or more covariates are themselves functional, a wider range of ways of using a functional covariate to explain a response are available. Let us take a preliminary look at a few of these situations here in this chapter before considering them in detail in later chapters.

Consider the weather at our 35 Canadian sites. The precipitation $\texttt{Prec}(t)$ measured over times t will be the functional dependent variable, and either

- which of four climate zones the station falls in will be a categorical independent variable
- or the temperature $\texttt{Temp}(s)$ measured over times s will be the functional independent variable.

12.2 A functional response and a categorical independent variable

Does the shape of the mean annual precipitation profile depend on which climate zone the station is in? With the zones Atlantic, Arctic, Continental and Pacific, that answer is "almost certainly." Still, we may feel the need to see if the data reject the null hypothesis that there is no difference. And if this happens, then we will want to characterize the differences in functional terms.

In formal terms, we have a number N_g of weather stations in each climate zone $g = 1, \ldots, 4$, and the model for the mth precipitation function in the gth group, indicated by $\texttt{Prec}_{mg}$, is

$$\texttt{Prec}_{mg}(t) = \mu(t) + \alpha_g(t) + \epsilon_{mg}(t). \tag{12.1}$$

In this model, function μ is the grand mean across all 35 weather stations, and the *effect functions* α_g represent departures from the grand mean specific to climate zones. The residual variation left over after we have explained as much as we can using climate zones are captured in the residual functions $\epsilon_{mg}(t)$. Our task is to use the data $\texttt{Prec}_{mg}$ as well as the design matrix coding climate zone membership to estimate the functional parameters μ and α_g.

Moreover, we may want to test more localized hypotheses such as "there are no differences in mid-summer" or "the differences in mid-winter are essentially differences in amount of precipitation rather than in the shape of the precipitation profile." That is, we may have interesting *functional contrasts* specified in advance of looking at the data.

Finally, we can also have the familiar multiple comparisons problem, but this time in functional form. That is, we may simply ask, "Over which time intervals are there significant differences between climate zones?"

More generally, the model may involve a design matrix $\mathbf{Z}$ containing values of p scalar independent variables rather than just 0's and 1's coding category membership, or it may involve both types of predictors. As in the multivariate linear model, these two situations are essentially the same, and this applies here, too.

From an application perspective, a functional response with scalar covariates is a common situation, and our experience indicates that the majority of functional linear model analyses are of this form. We will take up this model in the next chapter, and there we will try to provide as many helpful suggestions for analysis and inference as we can at this point. On the whole, though, tools already familiar to us in working with multivariate data will need only relatively obvious modifications to be adapted to the functional response context.

12.3 A scalar response and a functional independent variable

The converse of the situation considered above may apply. Consider the question, "Does the total amount of precipitation depend on specific features of the temperature profile of a weather station?" Here we can take the response variable as being

$$\texttt{Prectot}_i = \int_0^{365} \texttt{Prec}_i(t)\, dt,$$

where i indexes the 35 weather stations.

Now the issue is how to weight information *within* the single covariate $\texttt{Temp}(s)$ *across* values of s. We do this using the linear model

$$\texttt{Prectot}_i = \alpha + \int_0^{365} \texttt{Temp}_i(s)\beta(s)\, ds + \epsilon_i\,. \tag{12.2}$$

Here the constant α is the usual intercept term that adjusts for the origin of the precipitation variable. The functional parameter of interest is again the regression coefficient function β.

This situation formally resembles conventional multiple regression if we think of each time s as indexing a separate scalar independent variable, namely $\texttt{Temp}(s)$. But then we realize that we now have a potentially unlimited number of independent variables at our disposal to predict 35 scalar values. This seems ridiculous; over-fitting the data now seems inevitable.

The way out of the problem is to force the weighting of information across s to be sufficiently smooth that we can know that a bad fit is in principle possible. This smoothing over s will involve the *regularization* process that we have already seen in action in the spline smoothing chapter 5. Chapter 15 is given over to this situation.

Note that we will always use a different letter s for the argument for a covariate function than we use for the dependent variable. Although in this example context both s and t index time in years over an annual cycle, more generally they could index entirely different continua, such as space for s and time for t.

12.4 A functional response and a functional independent variable

Now we throw open the gates. How does a precipitation profile depend on the associated temperature profile? Now we consider how the functional covariate value $\texttt{Temp}(s)$ influences precipitation $\texttt{Prec}(t)$ specifically at time t. Here are some possibilities.

12.4.1 Concurrent

We might only use the temperature at the same time $s = t$ because we imagine that precipitation now depends only on the temperature now. Our model is

$$\texttt{Prec}_i(t) = \alpha(t) + \texttt{Temp}_i(t)\beta(t) + \epsilon_i(t). \tag{12.3}$$

We might call this model *concurrent* or *point-wise*. Should we use regularization to force β to be smooth in t?

This model has already been discussed in some detail prior to the first edition of this book by Hastie and Tibshirani (1993) under the name of the *varying coefficient model*. It deserves here a chapter of its own, 14, in part because we will show that all functional linear models can be reduced to this form.

12.4.2 Annual or total

We may prefer to allow for temperature influence on $\texttt{Prec}(t)$ to extend over the whole year. The model expands to become

$$\texttt{Prec}_i(t) = \alpha(t) + \int_0^{365} \texttt{Temp}_i(s)\beta(s,t)\,ds + \epsilon_i(t). \tag{12.4}$$

We face the additional complexity of the regression coefficient function β being bivariate; the value $\beta(s,t)$ determines the impact of temperature at time s on precipitation at time t.

We suspect from the discussion of the scalar response and functional covariate that it may be essential to smooth β as a function of s. But what is the difference between s-smoothing and smoothing with respect to t?

12.4.3 Short-term feed-forward

We may choose for reasons of parsimony to use only the temperature now and over an interval back in time in order to allow for some cumulative effects. For example, it may be that what counts is whether the temperature

has been falling rapidly up to time t. The model expands to

$$\texttt{Prec}_i(t) = \alpha(t) + \int_{t-\delta}^{t} \texttt{Temp}_i(s)\beta(s,t)\,ds + \epsilon_i(t). \tag{12.5}$$

Here δ is the time lag over which we use temperature information. In addition to being bivariate, now β is only defined over the somewhat complicated trapezoidal domain: $t \in [0, 365], t - \delta \leq s \leq t$.

Since in this situation the data are periodic, so we won't have particular problems with s being negative at $t = 0$ since we can borrow information from the previous year. But for non-periodic data, we would want to remove the triangle implied by $s < 0$ from the domain.

12.4.4 Local influence

Finally, after some reflection, we may open up the model to allow integration over s within a t-dependent set Ω_t. Why? Well, for example, if the temperature first falls rapidly, and then rises rapidly immediately after, and if the time t in question is in the middle of the summer, this may be a thunderstorm, and will therefore have the potential for a very large amount of rainfall within a short time period. The model may therefore be

$$\texttt{Prec}_i(t) = \alpha(t) + \int_{\Omega_t} \texttt{Temp}_i(s)\beta(s,t)\,ds + \epsilon_i(t). \tag{12.6}$$

Here there is the potential complexity of the domain over which β is defined that will challenge our computational resources.

These examples indicate that the functional linear model has the potential to be rather complex. Indeed, there is no reason why the covariate z might not be a function of both s and t. For example, we may predict rainfall at a station by integrating information over both space and time if we are on the Canadian prairies where precipitation in the summer tends to be *convective*, meaning thunder storms, hail storms and tornadoes that tend to be spatially limited and to follow curvilinear tracks.

12.5 What about predicting derivatives?

We may choose to model the rate of change in precipitation, $D\texttt{Prec}$ instead of precipitation itself. When a model is designed to explain a derivative of some order, we call it a *dynamic model*. In this case, the model is a *differential equation,* meaning simply that a derivative is involved.

When the response is a derivative, then there is the potential for the function itself to be a useful covariate. For example, the concurrent linear model

$$D\texttt{Prec}_i(t) = \texttt{Prec}_i(t)\beta(t) + \epsilon_i(t) \tag{12.7}$$

is a called a *homogeneous first order linear differential equation* in precipitation, and if we also include an influence of temperature,

$$D\texttt{Prec}_i(t) = \texttt{Prec}_i(t)\beta_0(t) + \texttt{Temp}_i(t)\beta_1(t) + \epsilon_i(t), \tag{12.8}$$

the equation is said to be *nonhomogeneous* rather than *homogeneous*. Temperature in the equation is called a *forcing function*.

The final chapters in the book will take up the story of differential equations, and we will see that the power of functional data analysis is remarkably extended in this way.

12.6 Overview

Although we dedicate separate chapters to these situations for the good reason that each involves some specialized techniques and issues, at a broader level the differences between the various models outlined above are more apparent than real. For example, a scalar response can always be expressed as a functional response with a constant basis, and the same is true for a scalar covariate. Of course, specialized computational issues arise as we take advantage at an algorithmic level of the fact that scale variables are involved.

A central theme common to all functional linear models is that of smoothing regression coefficient functions. Functional linear models usually involve more predictive power than we want to use for a finite amount of noisy data. Deciding how much to smooth and how to define smoothness itself will be a central issue in most applications.

Probably the most fundamental issue is the nature of the potentially t-specific domain Ω_t in (12.6). Both the point-wise and total influence models are comparatively easy to deal with computationally, as we shall see. But localized feed-forward influence is often essential, and already well represented in statistics in the form of ARIMA and state-space models in time series analysis.

13

Modelling functional responses with multivariate covariates

13.1 Introduction

We now consider how to use data on a set of scalar predictor variables or covariates $z_j, j = 1, \ldots, p$ to fit the features of a functional response or outcome variable x. Both of the examples in this chapter can be described as *functional analyses of variance* because the values of the covariates are 0's and 1's coding the categories of factor variables, but the techniques that we develop here apply equally well to measured covariates.

13.2 Predicting temperature curves from climate zones

Let's have a look at the Canadian weather data introduced in Chapter 1. Monthly means for temperature and precipitation are available for each of 35 weather stations distributed across the country, and we can use the smoothing techniques of Chapters 4 and 5 to represent each record as a smooth function. Thus, two periodic functions, `Temp` and `Prec`, denoting temperature and precipitation, respectively, are available for each station.

How much of the pattern of annual variation of temperature in a weather station is explainable by its geographical area? Dividing Canada into Atlantic, Continental, Pacific and Arctic meteorological zones, we want to study the characteristic types of temperature patterns in each zone.

This is an *analysis of variance* problem with four treatment groups. *Multivariate analysis of variance* (MANOVA) is the extension of the ideas of analysis of variance to deal with problems where the dependent variable is multivariate. Because our dependent variable is the functional observation `Temp`, the methodology we need is a *functional analysis of variance*, abbreviated FANOVA.

In formal terms, we have a number of stations in each group g, and the model for the mth temperature function in the gth group, indicated by $\texttt{Temp}_{mg}$, is

$$\texttt{Temp}_{mg}(t) = \mu(t) + \alpha_g(t) + \epsilon_{mg}(t). \tag{13.1}$$

The function μ is the grand mean function, and therefore indicates the average temperature profile across all of Canada. The terms α_g are the specific effects on temperature of being in climate zone g. To be able to identify them uniquely, we require that they satisfy the constraint

$$\sum_g \alpha_g(t) = 0 \text{ for all } t. \tag{13.2}$$

The residual function ϵ_{mg} is the unexplained variation specific to the mth weather station within climate group g.

We note in passing that the smoothing problem discussed in Chapters 3, 4, and 5 is contained within this model by using a single covariate whose values are all ones.

We can define a 35×5 design matrix $\mathbf{Z}$ for this model, with one row for each individual weather station, as follows. Use the label (mg) for the row corresponding to station m in group g; this row has a one in the first column, a one in column $g+1$, and zeroes in the rest. Write $z_{(mg)j}$ for the value in this row and in the jth column of $\mathbf{Z}$.

We can then define a corresponding set of five regression functions β_j by setting $\beta_1 = \mu$, $\beta_2 = \alpha_1$, and so on to $\beta_5 = \alpha_4$, so that the functional vector $\boldsymbol{\beta} = (\mu, \alpha_1, \alpha_2, \alpha_3, \alpha_4)'$. In these terms, the model (13.1) has the equivalent formulation

$$\texttt{Temp}_{mg}(t) = \sum_{j=1}^{5} z_{(mg)j}\beta_j(t) + \epsilon_{mg}(t) \tag{13.3}$$

or, more compactly in matrix notation,

$$\texttt{Temp} = \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{13.4}$$

where `Temp` is the functional vector containing the 35 temperature functions, $\boldsymbol{\epsilon}$ is a vector of 35 residual functions, and $\boldsymbol{\beta}$ is the 5-vector of parameter functions. The design matrix $\mathbf{Z}$ has exactly the same structure as for the corresponding univariate or multivariate one-way analysis of variance. The only way in which (13.4) differs from the corresponding equations in standard elementary textbooks on the general linear model is

that the parameter $\boldsymbol{\beta}$, and hence the predicted observations $\mathbf{Z}\boldsymbol{\beta}$, are vectors of functions rather than vectors of numbers.

13.2.1 Fitting the model

If (13.4) were a standard general linear model, the standard least squares criterion would say that $\boldsymbol{\beta}$ should be chosen to minimize the residual sum of squares. To extend the least squares principle to the functional case, we need only reinterpret the residual sum of squares in an appropriate way. The quantity $\texttt{Temp}_i(t) - \mathbf{Z}_i\boldsymbol{\beta}(t)$ is now a function, and so the unweighted least squares fitting criterion becomes

$$\texttt{LMSSE}(\boldsymbol{\beta}) = \sum_{g}^{4} \sum_{m}^{N_g} \int [\texttt{Temp}_{mg}(t) - \sum_{j}^{q} z_{(mg),j}\beta_j(t)]^2 \, dt. \tag{13.5}$$

Minimizing $\texttt{LMSSE}(\boldsymbol{\beta})$ subject to the constraint $\sum_2^5 \beta_j = 0$ (equivalent to $\sum_1^4 \alpha_g = 0$) gives the least squares estimates $\hat{\boldsymbol{\beta}}$ of the functional parameters μ and α_g. Section 13.4 contains some remarks about the way LMSSE is minimized in practice.

Figure 13.1 displays the resulting estimated region effects α_g for the four climatic zones, and Figure 13.2 displays the composite effects $\mu + \alpha_g$. We see that the region effects are more complex than the constant or even sinusoidal effects that one might expect:

- The Atlantic stations appear to have a temperature around 5 degrees C warmer than the Canadian average.
- The Pacific weather stations have a summer temperature close to the Canadian average, but are much warmer in the winter.
- The Continental stations are slightly warmer than average in the summer, but are colder in the winter by about 5 degrees C.
- The Arctic stations are certainly colder than average, but even more so in March than in January.

The cross-hatched areas indicate 95% confidence regions for the location of the curves at fixed points. These will be discussed in Section 13.4.

13.2.2 Assessing the fit

In estimating and plotting the individual regional temperature effects, we have taken our first step towards achieving the goal of characterizing the typical temperature pattern for weather stations in each climate zone. We may wish to move on and not only confirm that the total zone-specific effect α_g is nonzero, but also investigate whether this effect is substantial at

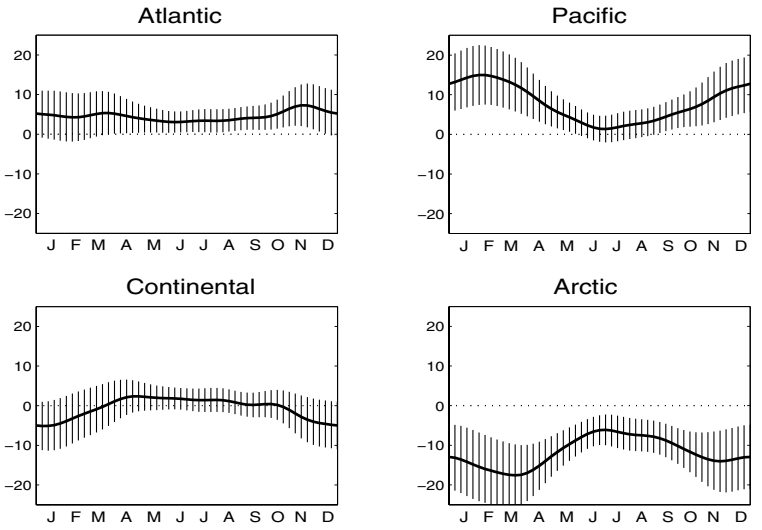

Figure 13.1. The region effects α_g for the temperature functions in the functional analysis of variance model $\texttt{Temp}_{mg}(t) = \mu(t) + \alpha_g(t) + \epsilon_{mg}(t)$. The effects $\alpha_g(t)$ are required to sum to 0 for all t. The cross-hatched areas indicate 95% point-wise confidence intervals for the true effects.

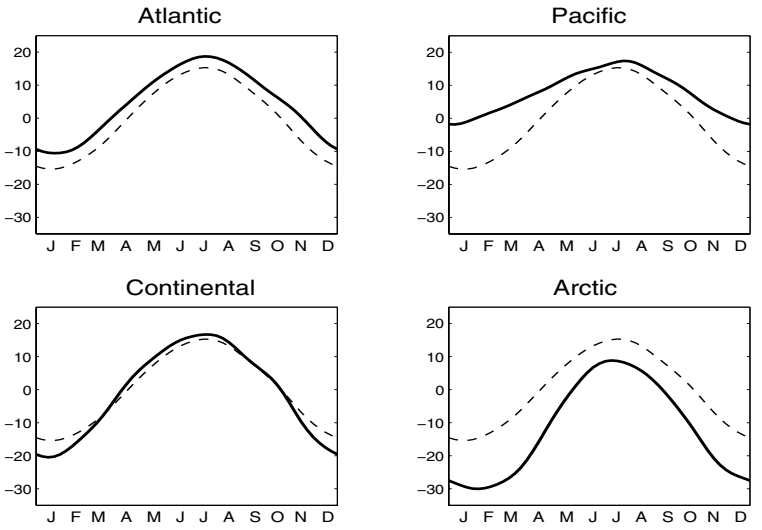

Figure 13.2. The estimated climate zone temperature profiles $\mu + \alpha_g$ for the temperature functions in the functional analysis of variance model (solid curves). The dashed curve is the Canadian mean function μ.

a specific time t. As in ordinary analysis of variance, we look to summarize these issues in terms of error sum of squares functions LMSSE, squared correlation functions RSQ, and F-ratio functions FRATIO. It is the dependence of these quantities on t that makes the procedure different from the standard multivariate case.

As in the multivariate linear model, the primary source of information in investigating the importance of the zone effects α_g is the sum of squares function

$$\mathtt{SSE}(t) = \sum_{mg} [\mathtt{Temp}_{mg}(t) - \mathbf{Z}_{mg}\hat{\boldsymbol{\beta}}(t)]^2. \tag{13.6}$$

This function can be compared to the error sum of squares function based on using only the Canadian average $\hat{\mu}$ as a model,

$$\mathtt{SSY}(t) = \sum_{mg} [\mathtt{Temp}_{mg}(t) - \hat{\mu}(t)]^2$$

and one way to make this comparison is by using the squared multiple correlation function RSQ with values

$$\mathtt{RSQ}(t) = [\mathtt{SSY}(t) - \mathtt{SSE}(t)]/\mathtt{SSY}(t). \tag{13.7}$$

Essentially, this function considers the drop in error sum of squares produced by taking climate zone into effect relative to error sum of squares without using climate zone information.

We can also compute the functional analogues of the quantities entered into the ANOVA table for a univariate analysis. For example, the mean squared for error function MSE has values

$$\mathtt{MSE} = \mathtt{SSE}/\mathtt{df}(\text{error}),$$

where df(error) is the degrees of freedom for error, or the sample size N less the number of mathematically independent functions β_q in the model. In this problem, the zero sum restriction on the climate zone effects α_g implies that there are four degrees of freedom lost to the model, or df(error) = 31.

Similarly, the mean square for regression is the difference between SSY (or, more generally, whatever reference model we employ that is a specialization of the model being assessed) and SSE, divided by the difference between the degrees of freedom for error for the two models. Let the difference in degrees of freedom be denoted by df(regression), which in this case is 3. Thus

$$\mathtt{MSR}(t) = \frac{\mathtt{SSY}(t) - \mathtt{SSE}(t)}{\mathtt{df}(\text{regression})} .$$

Finally, we can compute the F-ratio function,

$$\mathtt{FRATIO} = \frac{\mathtt{MSR}}{\mathtt{MSE}}. \tag{13.8}$$

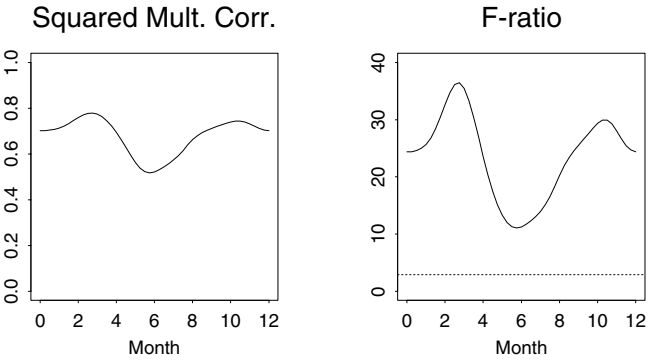

Figure 13.3. The left panel contains the squared multiple correlation function `RSQ` and the right panel the corresponding F-ratio function `FRATIO`. The horizontal dotted line indicates the 5% significance level for the F-distribution with 3 and 31 degrees of freedom.

Figure 13.3 shows the two functions `RSQ` and `FRATIO`. We can see that the squared correlation is relatively high and that the F-ratio is everywhere substantially above the 5% significance level of 2.92. It is interesting to note that the differences between the climate zones are substantially stronger in the spring and autumn, rather than in the summer and winter as we might expect.

Basically, then, most of the statistical machinery available for univariate analysis of variance is readily applicable to this functional problem. We can consider, for example, contrast functions, post-hoc multiple comparison functions, F-ratios associated with constrained estimates of region effects, and so on, essentially because the functional analysis of variance problem is really a univariate ANOVA problem for each specific value of t.

One question not addressed in the discussion of this example is an overall assessment of significance for the difference between the climate zones, rather than an assessment for each individual time t. We remind ourselves that the classical significance level was designed to be used for a single hypothesis test, rather than a continuum of them as in here. Although there is no reasonable doubt here that climate zone has an important effect somewhere in the year, in other applications we will want to protect ourselves more effectively against falsely declaring significance somewhere in the interval. Section 13.3.3 provides an approach to this question using simulation in the context of a different example.

A second question is, "How can we compute confidence intervals for the estimated regression functions?" Because this topic involves substantial mathematical detail, we put this off until Section 13.4.

13.3 Force plate data for walking horses

This section describes some interesting data on equine gait. The data were collected by Dr. Alan Wilson of the Equine Sports Medicine Center, Bristol University, and his collaborators. Their kindness in allowing use of the data is gratefully acknowledged. The data provide an opportunity to discuss various extensions of our functional linear modelling and analysis of variance methodology. For further details of this example, see Wilson et al. (1996).

13.3.1 Structure of the data

The basic structure of the data is as follows. It is of interest to study the effects of various types of shoes, and various walking surfaces, on the gait of a horse. One reason for this is simply biomechanical: the horse is an animal particularly well adapted to walking and running, and the study of its gait is of intrinsic scientific interest. Secondly, it is dangerous to allow horses to race if they are lame or likely to go lame. Careful study of their gait may produce diagnostic tests of incipient lameness which do not involve any invasive investigations and may detect injuries at a very early stage, before they become serious or permanent. Thirdly, it is important to shoe horses to balance their gait, and understanding the effects of different kinds of shoe is necessary to do this. Indeed, once the normal gait of a horse is known, the measurements we describe can be used to test whether a blacksmith has shod a horse correctly, and can therefore be used as an aid in the training of farriers.

In this experiment, horses walk on to a plate about 1 meter square set into the ground and equipped with meters at each corner measuring the force in the vertical and the two horizontal directions. We consider only the vertical force. During the period that the horse's hoof is on the ground (the stance phase) the four measured vertical forces allow the instrument to measure the point of resultant vertical force. The hoof itself does not move during the stance phase, and the position of the hoof is measured by dusting the plate with sawdust or is inferred from the point of force at the end of the stride, when only the front tip of the hoof is in contact with the ground.

The vertical force increases very rapidly at the beginning of the stance phase but reduces more slowly at the end. Operationally, the stance phase is defined as starting at the moment where the total vertical force first reaches 30% of its maximum value and ending where it falls to 8% of its maximum value. For each replication, the point of force is computed for 100 time points equally spaced in this time interval.

A typical functional observation is therefore a two-dimensional function of time $\texttt{Force} = (\texttt{ForceX}, \texttt{ForceY})$ where t varies from 0 to 1 during the stance phase, and $\texttt{ForceX}(t)$ and $\texttt{ForceY}(t)$ are the coordinates of the point

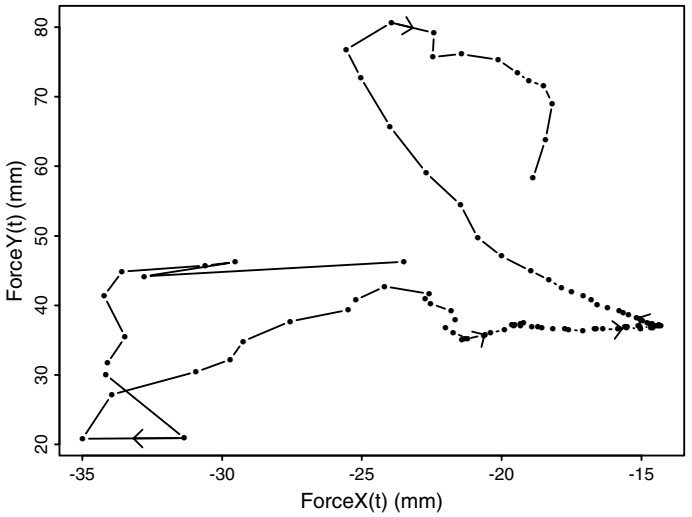

Figure 13.4. A typical trace of the resultant point of force during the stance phase of a horse walking onto a force plate. One hundred points equally spaced in time are indicated on the curve. The arrows indicate the direction of time.

of force at time t. Here Y is the direction of motion of the horse, and X measures distance in a perpendicular direction towards the body of the horse. Thus the coordinates are defined as if looking at the plate from above if a left foot is being measured, but with the X direction reflected if a right foot is being measured.

The data set consists of 592 separate runs and involves 8 horses, each of which has a number of measurements on both its right and left forelimbs. The nine shoeing conditions are as follows: first, the horse is observed unshod; it is then shod and observed again; then its shoe is modified by the addition of various wedges, either building up its toe or heel or building up one side or the other of its hoof. Not every horse has every wedge applied. In the case of the toe and heel wedges, the horse is observed immediately after the wedge is fitted and one day later, after it has become accustomed to the shoe. Finally the wedges are removed and the horse is observed with a normal shoe.

Figure 13.4 shows a typical (`ForceX`, `ForceY`) plot. This realization is among the smoother curves obtained. The 100 points that are equally spaced in time are marked on the curve, and the direction of time is indicated by arrows (also evenly spaced in time). We can see, not surprisingly, that the point of force moves most rapidly near the beginning and end of

the stance phase. The accuracy of a point measurement was about 1 mm in each direction.

13.3.2 A functional linear model for the horse data

The aim of this experiment is to investigate the effects of various shoeing conditions, and particularly to study the effects of the toe and heel wedges, which change as the horse becomes accustomed to the wedge. We fit a model of the form

$$\texttt{Force}_{ijkl} = \mu + \alpha_{ij} + \theta_k + \epsilon_{ijkl}, \tag{13.9}$$

where all the terms are two-dimensional functions of t, $0 \leq t \leq 1$. The suffix $ijkl$ refers to the data collected for the lth observed curve for side j of horse i under condition k.

For any particular curve, use labels x and y where necessary to denote the x and y coordinates of the vector function. The following identifiability constraints are placed on the various effects, each valid for all t:

$$\sum_{i,j} \alpha_{ij}(t) = \sum_{k=1}^{9} \theta_k(t) = 0. \tag{13.10}$$

We estimate the various effects by carrying out a separate general linear model fit for each t and for each of the x and y coordinates. Since the data are observed at 100 discrete times in practice, each $\texttt{Force}_{ijkl}$ corresponds to two vectors, each of length 100, one for the x coordinates and one for the y coordinates. The design matrix relating the expected value of $\texttt{Force}_{ijkl}$ to the various effects is the same for all 200 observed values, so although the procedure involves the fitting of 200 separate models, considerable economy of effort is possible. The model (13.9) can be written as

$$\texttt{Force} = \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{13.11}$$

where Force and $\boldsymbol{\epsilon}$ are both vectors of length 592, each of whose elements is a two-dimensional function on $[0, 1]$. The vector $\boldsymbol{\beta}$ is a vector of the 26 two-dimensional functions μ, α_{ij} and θ_k, and $\mathbf{Z}$ is a 592×26 design matrix relating the observations Force to the effects $\boldsymbol{\beta}$. The identifiability constraints (13.10) are incorporated by augmenting the matrix $\mathbf{Z}$ by additional rows corresponding to the constraints, and by augmenting the data vector Force by zeroes. Standard theory of the general linear model of course then gives as the estimator

$$\hat{\boldsymbol{\beta}} = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\,\texttt{Force}. \tag{13.12}$$

Figure 13.5 plots the estimated overall mean curve $\mu = (\mu_x, \mu_y)$ in the same way as Figure 13.4. Although the individual observations are somewhat irregular, the overall mean is smooth, even though no smoothing is incorporated into the procedure.

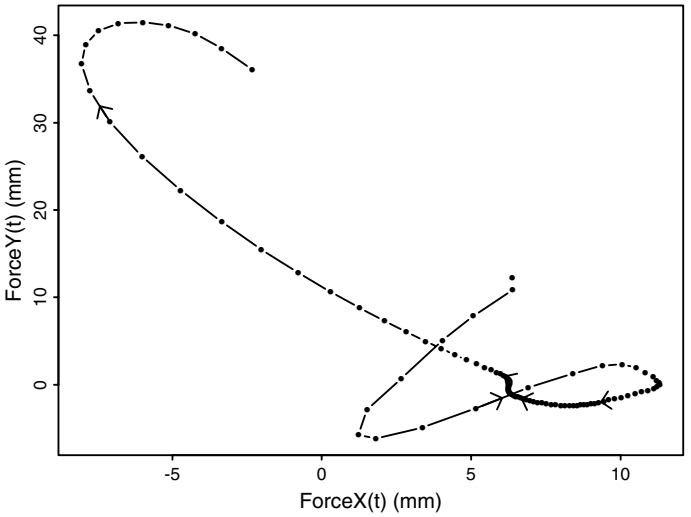

Figure 13.5. Estimate of the overall mean curve (μ_x, μ_y) obtained from the 592 observed point-of-force curves using model (13.9).

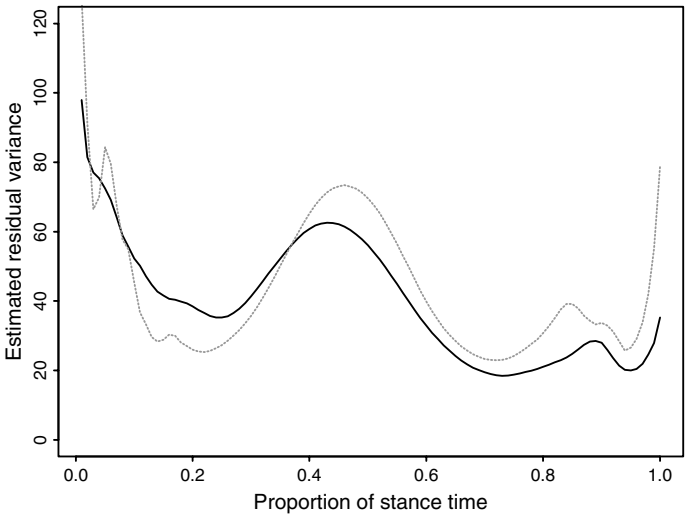

Figure 13.6. The estimated residual variance in the x coordinate (solid curve) and the y coordinate (dotted curve) as the stance phase progresses.

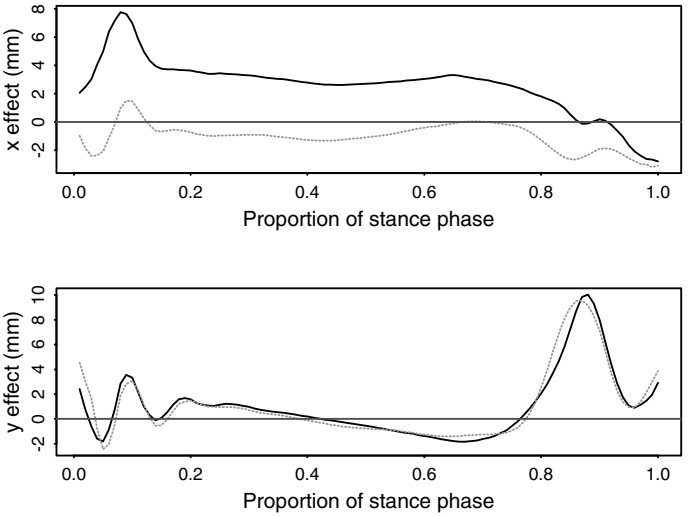

Figure 13.7. The effects of the application of a toe wedge, x coordinate in the upper panel and y coordinate in the lower. Solid curves are the immediate effect, and dashed curves are the effect on the following day.

The general linear model fitted for each coordinate at each time point allows the calculation of a residual sum of squares, and hence an estimated residual variance, at each point. The residual variance curves $\texttt{MSE}_x$ and $\texttt{MSE}_y$ for the x and y coordinates are plotted in Figure 13.6. It is very interesting to note that the residual variances in the two coordinates are approximately the same size, and vary in roughly the same way, as the stance phase progresses.

13.3.3 Effects and contrasts

We can now explain how the linear model can be used to investigate various effects of interest. We concentrate on two specific effects, corresponding to the application of the toe wedges, and illustrate how various inferences can be drawn. In Figure 13.7, we plot the effects of the toe wedge immediately after it has been applied and the following day. The x and y coordinates of the relevant functions θ_k are plotted separately. It is interesting to note that the y effects are virtually the same in both cases: The application of the wedge has an immediate effect on the way in which the point of force moves in the forward-backward direction, and this pattern does not change appreciably as the horse becomes accustomed to the wedge. The effect in the side-to-side direction is rather different. Immediately after the wedge

is applied, the horse tends to put its weight to one side, but the following day the effect becomes much smaller, and the weight is again placed in the same lateral position as in the average stride.

To investigate the significance of this change, we now consider the contrast between the two effects, which shows the expected difference between the point of force function for a horse 24 hours after a toe wedge has been applied and that immediately after applying the wedge. Figure 13.8 shows the x and y coordinates of the difference of these two effects. The standard error of this contrast is easily calculated. Let u be the vector such that the estimated contrast is the vector function $\texttt{Contrast} = u'\hat{\boldsymbol{\beta}}$, so that the component of u corresponding to toe wedge 24 hours after application is $+1$, that corresponding to toe wedge immediately after application is -1, and all the other components are zero. Define a by $a^2 = u'(\mathbf{Z}'\mathbf{Z})^{-1}u$. The squared point-wise standard errors of the x and y coordinates of the estimated contrasts are then $a^2\texttt{MSE}_x$ and $a^2\texttt{MSE}_y$, respectively. Plots of ± 2 times the relevant standard error are included in Figure 13.8. Because the degrees of freedom $(592 - 26 + 2)$ for residual variance are so large, these plots indicate that point-wise t tests at the 5% level would demonstrate that the difference in the y coordinate of the two toe wedge effects is not significant, except possibly just above time 0.8, but that the x coordinate is significantly different from zero for almost the whole stance phase.

How should we account for the correlation in the tests at different times in assessing the significance of any difference between the two conditions? We can consider the summary statistics

$$M_x = \sup_t |\texttt{Contrast}_x(t)/a\sqrt{\texttt{MSE}_x(t)}|$$

and

$$M_y = \sup_t |\texttt{Contrast}_y(t)/a\sqrt{\texttt{MSE}_y(t)}|.$$

The values of these statistics for the data are $M_x = 5.03$ and $M_y = 2.01$. A permutation-based significance value for each of these statistics was obtained by randomly permuting the observed toe wedge data for each leg of each horse between the conditions immediately after fitting of wedge and 24 hours after fitting of wedge, keeping the totals the same within each condition for each leg of each horse. The statistics M_x and M_y were calculated for each random permutation of the data. In 1000 realizations, the smallest value of M_x observed was 3.57, so the observed difference in the x direction of the two conditions is highly significant. A total of 177 of the 1000 simulated M_y values exceeded the observed value of 2.01, and so the estimated p-value of this observation was 0.177, showing no evidence that the y coordinate of point of force alters its time behavior as the horse becomes accustomed to the wedge.

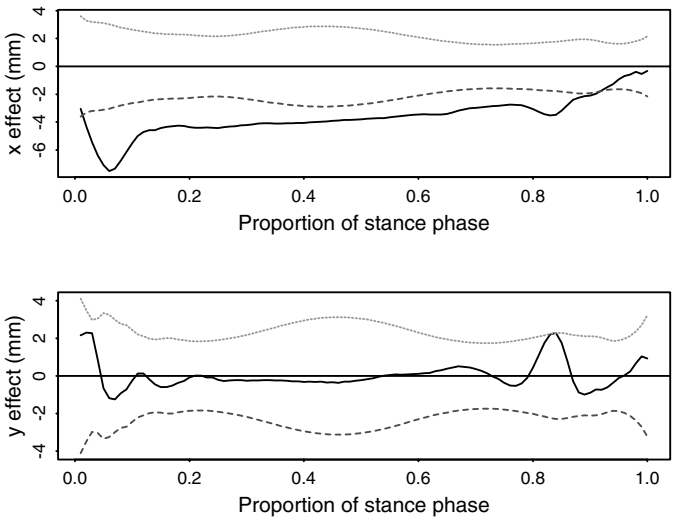

Figure 13.8. Solid curves are the differences between the effect of a toe wedge after 24 hours and its immediate effect. Dotted curves indicate plus and minus two estimated standard errors for the point-wise difference between the effects. The upper panel contains x coordinates and the lower the y coordinates.

13.4 Computational issues

13.4.1 The general model

We explain how to compute the least squares estimates in the functional linear model. Let Y be an N-vector of functional observations, and let the q-vector $\boldsymbol{\beta}$ contain the regression functions. In the force-plate data example, the individual elements of both Y and $\boldsymbol{\beta}$ were themselves two-dimensional functions. We assume that $\mathbf{Z}$ is an $N \times q$ design matrix, and that the expected value of $\mathbf{y}(t)$ for each t is modelled as $\mathbf{Z}\boldsymbol{\beta}(t)$. The functional linear model is then

$$\mathbf{y}(t) = \mathbf{Z}\boldsymbol{\beta}(t) + \boldsymbol{\epsilon}(t) \ . \tag{13.13}$$

Any linear constraints on the parameters $\boldsymbol{\beta}$, such as the requirement in the temperature data example that the individual climate zone effects sum to zero, are expressed as $\mathbf{L}\boldsymbol{\beta} = 0$ for some suitable matrix $\mathbf{L}$ with q columns. By using a technique such as the QR-decomposition, described in Section A.3.3 of the Appendix, we may then say that

$$\boldsymbol{\beta} = \mathbf{C}\alpha \tag{13.14}$$

for some matrix $\mathbf{C}$. In this case we find ourselves back at the basic model (13.13), except that $\mathbf{Z}$ is now replaced by $\mathbf{ZC}$ and $\boldsymbol{\beta}$ by $\boldsymbol{\alpha}$.

Our aim is to minimize the least squares fitting criterion

$$\texttt{LMSSE}(\boldsymbol{\beta}) = \int [\mathbf{y}(t) - \mathbf{Z}\boldsymbol{\beta}(t)]'[\mathbf{y}(t) - \mathbf{Z}\boldsymbol{\beta}(t)]\, dt \ , \tag{13.15}$$

It is possible that an order N weighting matrix $\mathbf{W}$ should be included in this expression between the two factors on the right in order to allow for possible variation of importance across replications, for dependencies. However, most situations assume independence and constant error variation, and so we will not bother with this feature in our discussion.

13.4.2 Pointwise minimization

If there are no particular restrictions on the way in which $\boldsymbol{\beta}(t)$ varies as a function of t, we can minimize $\texttt{LMSSE}(\boldsymbol{\beta})$ by minimizing $\|\mathbf{y}(t) - \mathbf{Z}\boldsymbol{\beta}(t)\|^2$ individually for each t. That is, we calculate $\hat{\boldsymbol{\beta}}(t)$ for a suitable grid of values of t using ordinary regression analysis, and then interpolate between these values. This was the technique used in the force plate example, where the grid of values was chosen to correspond with the discretization of the original data. The fact that the same design matrix is involved for each t makes for considerable economy of numerical effort.

13.4.3 Functional linear modelling with regularized basis expansions

We have already noted, however, that the use of regularized basis function expansions gives us continuous control over smoothness while still permitting as much high frequency detail in the model as the data require. The use of roughness penalties or regularization can play an important role in a functional linear model. In particular, one may adopt the philosophy that the representation of the response functions should be allowed to be of high resolution, and that smoothness is imposed only on the functional parameters to be estimated in $\boldsymbol{\beta}$. In this way, we do not risk smoothing away important information that may impact the estimate of $\boldsymbol{\beta}$ when smoothing the data giving rise to y.

Let us now assume that the observed functions y_i and regression functions β_j are expressed in basis expansion form, as the coefficients of a Fourier series or B-spline or some other basis system. This means that

$$\mathbf{y}(t) = \mathbf{C}\boldsymbol{\phi}(t),$$

where

- the N-vector $\mathbf{y}$ contains the N observed response functions,

- the K_y-vector $\boldsymbol{\phi}$ contains the linearly independent basis functions, and
- the N by K_y matrix matrix $\mathbf{C}$ contains the coefficients of expansion of function y_i in its ith row.

Let us now expand the estimated parameter vector $\hat{\boldsymbol{\beta}}$ in terms of a basis vector $\boldsymbol{\theta}$ of length K_β, expressing $\hat{\boldsymbol{\beta}} = \mathbf{B}\boldsymbol{\theta}$ for a $q \times K_\beta$ matrix $\mathbf{B}$. In some cases, we may choose to use the same basis that was used to expand the response functions, in which case $\boldsymbol{\theta} = \boldsymbol{\phi}$, and consequently some of what follows becomes simpler. However, there are plenty of situations where we need to keep the two basis systems separate.

Note, though, that we have made things somewhat easier on ourselves by assuming that the same basis system $\boldsymbol{\theta}$ is used for all q regression functions β_j. In the next chapter, we will relax this constraint, but for the time being this assumption has the advantage of keeping the notation reasonably simple.

Now suppose that we use a linear differential operator L to define a roughness penalty for $\boldsymbol{\beta}$ as

$$\texttt{PEN}_L(\beta) = \int [L\boldsymbol{\beta}(s)]'[L\boldsymbol{\beta}(s)]\, ds \ . \tag{13.16}$$

In addition, we need to define these four matrices:

$$\mathbf{J}_{\phi\phi} = \int \boldsymbol{\phi}\boldsymbol{\phi}' \ , \quad \mathbf{J}_{\theta\theta} = \int \boldsymbol{\theta}\boldsymbol{\theta}' \ , \quad \mathbf{J}_{\phi\theta} = \int \boldsymbol{\phi}\boldsymbol{\theta}'$$

$$\mathbf{R} = \int (L\boldsymbol{\theta})(L\boldsymbol{\theta})' \ . \tag{13.17}$$

Note that we dropped "(s)" and "ds" from the expressions in (13.17); this makes the expressions more readable, and the context makes it clear that what we really mean is an expression like (13.16) where they were left in. Note, too, that since $\boldsymbol{\phi}$ is a column vector of length K_y of basis functions, $\boldsymbol{\phi}\boldsymbol{\phi}'$ is a square matrix of order K_y containing all possible pairs of these functions, and consequently $\mathbf{J}_{\phi\phi}$ is constant symmetric order K_y matrix of integrated products, and similarly for the other three matrices.

We can then obtain the following expressions for the penalized least squares criterion:

$$\begin{aligned} \texttt{PENSSE}(y|\boldsymbol{\beta}) \quad &= \quad \int (\mathbf{C}\boldsymbol{\phi} - \mathbf{Z}\mathbf{B}\boldsymbol{\theta})'(\mathbf{C}\boldsymbol{\phi} - \mathbf{Z}\mathbf{B}\boldsymbol{\theta}) + \\ &\qquad \lambda \int (L\mathbf{B}\boldsymbol{\theta})'(L\mathbf{B}\boldsymbol{\theta}) \ . \end{aligned} \tag{13.18}$$

The operation of integration and the summations implied by the matrix products in these expressions can be interchanged, and consequently we

can re-expressing this as

$$\begin{aligned}\texttt{PENSSE}(y|\boldsymbol{\beta}) &= \operatorname{trace}(\mathbf{C}'\mathbf{C}\mathbf{J}_{\phi\phi}) + \operatorname{trace}(\mathbf{Z}'\mathbf{Z}\mathbf{B}\mathbf{J}_{\theta\theta}\mathbf{B}') - \\ &\quad 2\operatorname{trace}(\mathbf{B}\mathbf{J}_{\theta\phi}\mathbf{C}'\mathbf{Z}) + \lambda\operatorname{trace}(\mathbf{B}\mathbf{R}\mathbf{B}') \ , \end{aligned} \tag{13.19}$$

where the operation trace is defined as

$$\operatorname{trace}\mathbf{A} = \sum_i a_{ii}$$

for a square matrix $\mathbf{A}$. One of the properties of the trace is that its value remains the same under any cyclic permutation of the matrix factors, so that, for example, $\operatorname{trace}(\mathbf{B}\mathbf{R}\mathbf{B}') = \operatorname{trace}(\mathbf{B}'\mathbf{B}\mathbf{R})$.

13.4.4 Using the Kronecker product to express $\hat{\mathbf{B}}$

We now need to compute the derivative of (13.19) with respect to matrix $\mathbf{B}$ and set the result to zero. Using the fact that the derivative of $\operatorname{trace}(\mathbf{B}'\mathbf{A})$ with respect to matrix $\mathbf{B}$ is $\mathbf{A}$, we find that $\mathbf{B}$ satisfies the matrix system of linear equations

$$(\mathbf{Z}'\mathbf{Z}\mathbf{B}\mathbf{J}_{\theta\theta} + \lambda\mathbf{B}\mathbf{R}) = \mathbf{Z}'\mathbf{C}\mathbf{J}_{\phi\theta} \ . \tag{13.20}$$

The solution $\mathbf{B}$ to this equation can be expressed explicitly in conventional matrix algebra if we use Kronecker products. The Kronecker product $\mathbf{A}\otimes\mathbf{C}$ is the super or composite matrix consisting of sub-matrices $a_{ij}\mathbf{C}$. Its usefulness in this situation derives from the fact that the matrix expression $\mathbf{A}\mathbf{B}\mathbf{C}'$ can be re-expressed as

$$\operatorname{vec}(\mathbf{A}\mathbf{B}\mathbf{C}') = (\mathbf{C}\otimes\mathbf{A})\operatorname{vec}(\mathbf{B}) \ ,$$

where $\operatorname{vec}(\mathbf{B})$ indicates the vector of length qK_θ obtained by writing matrix $\mathbf{B}$ as a vector column-wise. Moreover, the Kronecker product is also *bilinear* in the sense that

$$\operatorname{vec}(\mathbf{A}_1\mathbf{B}\mathbf{C}_1' + \mathbf{A}_2\mathbf{B}\mathbf{C}_2') = (\mathbf{C}_1\otimes\mathbf{A}_1 + \mathbf{C}_2\otimes\mathbf{A}_2)\operatorname{vec}(\mathbf{B}).$$

The Appendix contains a discussion of properties of the Kronecker product that have been used to obtain these expression, and other properties that will be used subsequently.

Consequently, applying these relations to the two terms involving $\mathbf{B}$ on the left side of (13.20), we obtain

$$[\mathbf{J}_{\theta\theta}\otimes(\mathbf{Z}'\mathbf{Z}) + \mathbf{R}\otimes\lambda\mathbf{I}]\operatorname{vec}(\mathbf{B}) = \operatorname{vec}(\mathbf{Z}'\mathbf{C}\mathbf{J}_{\phi\theta}) \ . \tag{13.21}$$

Now we have a system of qK_θ linear equations expressed in the conventional way that must be solved to obtain the elements of $\mathbf{B}$. The solution is

$$\operatorname{vec}(\mathbf{B}) = [\mathbf{J}_{\theta\theta}\otimes(\mathbf{Z}'\mathbf{Z}) + \mathbf{R}\otimes\lambda\mathbf{I}]^{-1}\operatorname{vec}(\mathbf{Z}'\mathbf{C}\mathbf{J}_{\phi\theta}) \ . \tag{13.22}$$

In (13.21) and (13.22) we have assumed a single smoothing parameter λ to impose the same level of smoothness on each component $\beta_j, j = 1, \ldots, q$

of the vector $\boldsymbol{\beta}$, but we may need the additional flexibility of controlling the smoothness of each component independently by using a smoothing parameter λ_j using a separate roughness penalty for each component. This involves the following minor modification of (13.21): Replace $\lambda\mathbf{I}$ in this expression by the diagonal matrix Λ containing the λ_j's in its diagonal.

Constraining $\boldsymbol{\beta}$ to be smooth is not the same thing as constraining the fit $\hat{y}$ to be smooth, and this latter strategy may be more important in some situations. Again a modification of (13.21) will serve: Replace $\mathbf{I}$ by $\mathbf{Z}'\mathbf{Z}$, in which case (13.22) simplifies to

$$\text{vec}\,(\mathbf{B}) = [(\mathbf{J}_{\theta\theta} + \lambda\mathbf{R}) \otimes (\mathbf{Z}'\mathbf{Z})]^{-1}\text{vec}\,(\mathbf{Z}'\mathbf{C}\mathbf{J}_{\phi\theta}) \ . \tag{13.23}$$

As in Chapter 5, smoothing parameters may be chosen by cross-validation, generalized cross-validation and other methods.

13.4.5 Fitting the raw data

We have been assuming that the response variable $\mathbf{y}(t)$ is a result of previously smoothing the discrete data, but in some applications we may prefer to go straight from the raw response data matrix $\mathbf{Y}$ to estimates of the regression coefficient functions. The penalized least squares criterion in this case is

$$\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\boldsymbol{\Theta}'\|^2 + \lambda\|\mathbf{L}\boldsymbol{\beta}(t)\|^2,$$

where $\boldsymbol{\Theta}$ is the N by K_β matrix of values of the basis functions for $\boldsymbol{\beta}$ evaluated at the sampling points for the response functions. The normal equations to be solved are in this case

$$(\mathbf{Z}'\mathbf{Z})\mathbf{B}(\boldsymbol{\Theta}'\boldsymbol{\Theta}) + \lambda\mathbf{B}\mathbf{R} = \mathbf{Z}'\mathbf{Y}\boldsymbol{\Theta} \ , \tag{13.24}$$

or, using Kronecker products,

$$[(\boldsymbol{\Theta}'\boldsymbol{\Theta}) \otimes (\mathbf{Z}'\mathbf{Z}) + \mathbf{R} \otimes \lambda\mathbf{I}]\text{vec}\,(\mathbf{B}) = (\boldsymbol{\Theta}' \otimes \mathbf{Z}')\text{vec}\,(\mathbf{Y}) \ . \tag{13.25}$$

13.5 Confidence intervals for regression functions

13.5.1 How to compute confidence intervals

The technique for computing point-wise confidence limits for regression functions is essentially the same as we used in Section 5.5. Recall that we required there two mappings. The first was the mapping `y2cMap` from the raw data vector y to the coefficient vector c, corresponding to the n by K matrix $\mathbf{S}_{\phi,\lambda}$ in the equation

$$\mathbf{c} = \mathbf{S}_{\phi,\lambda}\mathbf{y}.$$

We still need this mapping here, but we now assume that there are N replications, and consequently that the raw data reside in an N by n

matrix $\mathbf{Y}$ to be mapped to the N by K_y matrix $\mathbf{C}$ of coefficients of the basis function expansions of the functions N functions x_i. Consequently, as in Section 5.5, the `y2cMap` is represented by the matrix equation

$$\mathbf{C} = \mathbf{Y}\mathbf{S}_{\phi,\lambda} \ .$$

This may be re-expressed in Kronecker product notation as

$$\text{vec}\,\mathbf{C} = (\mathbf{S}_{\phi,\lambda} \otimes \mathbf{I})\text{vec}\,\mathbf{Y},$$

and this permits us to express what we can now denote as `Y2CMap` as

$$\texttt{Y2CMap} = \mathbf{S}_{\phi,\lambda} \otimes \mathbf{I}. \tag{13.26}$$

However, we now have a new linear mapping, namely that of the linear model itself, which maps a coefficient matrix $\mathbf{B}$ to an N by n fit to the data $\hat{\mathbf{Y}}$ by

$$\hat{\mathbf{Y}} = \mathbf{Z}\mathbf{B}\mathbf{\Theta}'.$$

We therefore need an expression for the mapping `C2BMap` that maps the coefficient matrix $\mathbf{C}$ for the response functions to the q by K_β coefficient matrix $\mathbf{B}$ for the regression function vector $\boldsymbol{\beta}$.

Finally, we will want to compute confidence intervals for some functional contrast or linear probe $\rho(\boldsymbol{\beta})$, and this will require a mapping that we can indicate by `B2RMap` that maps the coefficient matrix $\mathbf{B}$ to $\rho(\boldsymbol{\beta})$. For example, we may want to estimate the standard error of a regression function at a value t, and this is the value of the evaluation function $\rho_t(\boldsymbol{\beta})$. But we may also be interested in *functional contrasts* that probe for special effects that interest us as well.

The last stage in actually computing confidence limits is the computing of the composite mapping $\texttt{Y2RMap} = \texttt{B2RMap} \circ \texttt{C2BMap} \circ \texttt{Y2CMap}$ and applying it to each side of $\mathbf{\Sigma}_e$ to get an estimate of the sampling variance of the quantity of interest.

Now let us derive each of these matrix mappings, and put them together as required. That is, we compute the matrix mapping the raw data to the coefficients of the basis function expansions for the β_j's, and then we multiply this by the matrix mapping the regression coefficients to whatever quantities or functionals that interest us.

The first step, then, is to compute the matrix mapping $\mathbf{S}_{\phi,\lambda_y}$ from the data to these coefficients. This is, using the results in Section 5.5 for the response functions,

$$\mathbf{S}_{\phi,\lambda_y} = \mathbf{\Phi}(\mathbf{\Phi}'\mathbf{\Phi} + \lambda_y\mathbf{R}_y)^{-1}\mathbf{\Phi}' \ ,$$

where λ_y is the smoothing parameter used to smooth the data and $\mathbf{R}_y$ is the corresponding roughness penalty matrix. Matrix $\mathbf{S}_{\phi,\lambda_y}$ is then substituted in (13.26) to obtain `Y2CMap`.

From (13.25), we have the mapping from the data coefficients to the regression function coefficients expressed as

$$\text{vec}\,(\mathbf{B}) = [\mathbf{J}_{\theta\theta} \otimes \mathbf{Z}'\mathbf{Z} + \mathbf{R}_\beta \otimes \lambda_\beta \mathbf{I}]^{-1}(\mathbf{J}'_{\phi\theta} \otimes \mathbf{Z}')\text{vec}\,(\mathbf{C}'),$$

where λ_β and $\mathbf{R}_\beta$ are the smoothing parameter and roughness matrix associated with the regularization of the functions β_j. The qK_β by NK_y matrix

$$\texttt{C2BMap} = \mathbf{S}_\beta = [\mathbf{J}_{\theta\theta} \otimes \mathbf{Z}'\mathbf{Z} + \mathbf{R}_\beta \otimes \lambda_\beta \mathbf{I}]^{-1}(\mathbf{J}'_{\phi\theta} \otimes \mathbf{Z}') \tag{13.27}$$

is the matrix mapping that we need.

These last two expressions can then be combined into

$$\texttt{Y2BMap} = \text{vec}\,(\mathbf{B}) = \mathbf{S}_\beta(\mathbf{S}_{\phi,\lambda_y} \otimes \mathbf{I})\text{vec}\,(\mathbf{Y}') \ . \tag{13.28}$$

The variance of the raw data arranged as a vector is given by

$$\texttt{Var}[\text{vec}\,(\mathbf{Y}')] = \mathbf{\Sigma}_e \otimes \mathbf{I} \ ,$$

where $\mathbf{\Sigma}_e$ is the variance-covariance matrix of the residual vectors e_i and $\mathbf{I}$ is of order N. Note that these residuals are for the linear model (13.13) and not the residuals involved in smoothing the raw data for the response variable.

We can now put this all together to get what we need in terms of the coefficients of the expansions of the β_j;

$$\texttt{Var}[\text{vec}\,(\mathbf{B})] = \mathbf{S}_\beta(\mathbf{S}_{\phi,\lambda_y} \otimes \mathbf{I})(\mathbf{\Sigma}_e \otimes \mathbf{I})(\mathbf{S}_{\phi,\lambda_y} \otimes \mathbf{I})\mathbf{S}'_\beta \ . \tag{13.29}$$

If our objective is an estimate of $\texttt{Var}[\text{vec}\,(\hat{\boldsymbol{\beta}})]$, then this is

$$\texttt{Var}[\text{vec}\,(\hat{\boldsymbol{\beta}})] = (\mathbf{\Theta}\otimes\mathbf{I})\mathbf{S}_\beta(\mathbf{S}_{\phi,\lambda_y}\otimes\mathbf{I})(\mathbf{\Sigma}_e\otimes\mathbf{I})(\mathbf{S}_{\phi,\lambda_y}\otimes\mathbf{I})\mathbf{S}'_\beta(\mathbf{\Theta}\otimes\mathbf{I})' \ . \tag{13.30}$$

If both $\mathbf{J}_{\theta\theta}$ and $\mathbf{Z}'\mathbf{Z}$ are invertible, then this expression can be simplified to

$$\texttt{Var}[\text{vec}\,(\hat{\boldsymbol{\beta}})] = [\mathbf{J}^{-1}_{\theta\theta}\mathbf{J}'_{\phi\theta}\mathbf{S}_{\phi,\lambda_y}\mathbf{\Sigma}_e\mathbf{S}_{\phi,\lambda_y}\mathbf{J}_{\phi\theta}\mathbf{J}^{-1}_{\theta\theta}] \otimes (\mathbf{Z}'\mathbf{Z})^{-1} \ . \tag{13.31}$$

If the raw data are fit directly, the corresponding expression is

$$\texttt{Var}[\text{vec}\,(\hat{\boldsymbol{\beta}})] = [\mathbf{\Theta}(\mathbf{\Theta}'\mathbf{\Theta})^{-1}\mathbf{\Theta}'\mathbf{\Sigma}_e\mathbf{\Theta}(\mathbf{\Theta}'\mathbf{\Theta})^{-1}\mathbf{\Theta}'] \otimes (\mathbf{Z}'\mathbf{Z})^{-1} \ . \tag{13.32}$$

13.5.2 Confidence intervals for climate zone effects

We now illustrate this method for computing confidence intervals by estimating climate zone effects for the daily mean temperature data for 35 weather stations. We smoothed these data with a Fourier series basis with 65 basis functions without regularization in order to economize on computer time and work with temperature profiles that were somewhat smoother than those that we obtained in Chapter 5. The figures in Section 13.2 were obtained using these functional responses.

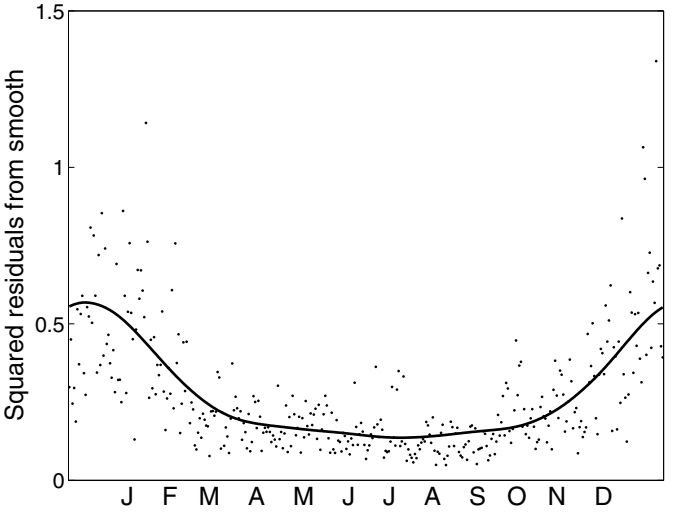

Figure 13.9. The points indicate daily estimates of the standard error of measurement for the mean temperature data computed across 35 weather stations, and the solid line is a positive smooth of these values.

Figure 13.9 shows the raw daily standard error estimates taken across the 35 stations as well as a positive smooth of these estimates of the kind that we discussed in Chapter 6. These vary between 0.4 degrees Celsius in the summer to 0.7 in the winter. This is a fair amount of variation, and so we put the reciprocal of the smoothed variances of measurement in the diagonal of weight matrix $\mathbf{W}$ in (5.3), and then re-smoothed the data.

In the functional analysis of variance step, we defined $\mathbf{Z}$ to be the 35 by 5 matrix containing the value 1 in column 1, and coding membership in the Atlantic, Pacific, Continental and Arctic regions in columns two to five as above. We used 21 Fourier basis functions to represent the 5 regression functions β_j.

The order 365 variance-covariance matrix $\mathbf{\Sigma}_e$ for the residuals from the linear model was estimated in the same way that we described in Chapter 2.

Figure 13.1 displays the 95% point-wise confidence limits on the estimated curve climate zone effect function. We see, for example, that it is only in the winter months that the temperature in Pacific zones can be considered as significantly warmer than those represented by the intercept function. For the record, a direct approximation of the raw average temperatures produced function and confidence limit estimates that were effectively indistinguishable from these results.

13.5.3 Some cautions on interpreting confidence intervals

Many things can go wrong in interpreting confidence interval estimates such as those in Figure 13.1, and it is important here to stress their limitations so as to avoid misleading ourselves and others in applications leading to serious decisions.

First of all, point-wise limits are not the same thing as confidence regions for the entire estimated curve. As mentioned in Section 5.5, although can envisage of pairs of curves between which there is a specified probability that the entire true curve is to be found, this requires the use of computationally intensive resampling methods. Point-wise regions are useful, of course, but they are based on the assumption that conclusions are only to be drawn *at that point.* So when we indicated above that we could conclude that summer temperatures in the Pacific zone are not much different from the national average, this was, strictly speaking, an abuse of the concept of a point-wise region.

Secondly, the confidence region estimates that we have developed are based on strong assumptions that may not be true. We have, as is usual in the analysis of variance, implicitly assumed that the distribution of the residuals is the same within each group. In fact, it is hard to imagine that, given enough weather stations, we would not also see systematic differences in covariances and other aspects of dispersion from one climate zone to another. In any case, the very idea of basing a confidence region on an estimated covariance involves the strong assumption that the joint distribution of two residuals is well summarized by a covariance, as would be the case if they were normally distributed. In fact, if the actual residual distribution is strongly skewed, if it has long tails, if it is multi-modal, or any one of many other violations of what normality implies is operative, these regions will not work as advertised. That is, in this case, we cannot claim them to contain the true curve with the specified probability.

Thirdly, we are estimating something whose potential dimension can outstrip any quantity of data that we can afford to collect. A curve can be made arbitrarily complex given enough information. It seems likely, surely, that someone working with ten times the number of weather stations, fifty years worth of data, and taking spatial dependencies into account will discover features of temperature curves that we could not capture with the number of basis functions that we used. Consequently, any claims that we may make about the precision with which we have estimated these curves must be understood to be conditional on the amount and quality of information that we have at our disposal. There is no asymptotic sample size when it comes to estimating a curve. Period. Although the results we have in this section are what some texts would call "small sample" results, in fact, we had to do here what is almost always done in practice, that is, substitute a sample estimate for a population quantity, namely $\mathbf{\Sigma}_e$. We really don't know what the full implication of this might be.

We can put the problem this way. In classical statistics, we usually work with a fixed number of parameters. But in nonparametric curve estimation, the number of basis functions K has the characteristics of a random variable. Two investigators working with different objectives, different number of sampling points, different residual variance levels, different ranges, and so forth are quite likely to work with different values of K. We really need, therefore, to take into account variation in K in our interval estimates, something that this chapter has not done. The interval estimation problem is, as far as K is concerned, more suitable for a Bayesian approach than for the classical methodology used here.

Elsewhere, moreover, we will have to substitute approximations for so-called "exact" results. Because, for example, the monotone smoother is not a linear function of the data, it was necessary to replace an exact calculation of the mapping `y2cMap` by a first-order approximation. This is inevitably a crude approximation in many situations, and always has the potential to be misleading.

So what to do? In the end, there is probably no safe substitute for computationally intensive methods such as simulation, bootstrapping of various kinds, and cross-validation methods. If these methods give results in essential agreement with these cheaper exact or asymptotically correct estimates, perhaps we can breathe a sigh of relief and carry on. But we should always assume that our decisions will only be reasonable until better data become available.

13.6 Further reading and notes

Brumback and Rice (1998) reported a functional analysis of variance involving daily progesterone metabolite concentrations over the menstrual cycles of 91 women enrolled in an artificial insemination clinic. The main experimental factor was whether conception occurred (21) or not (70). Within and between woman variation was also assessed. This work proceeded independently of Ramsay and Silverman (1997), but ended up using rather similar methods, and identified some serious computational difficulties involved with working with random functional factors. The discussion that followed the paper highlighted a number of issues. We strongly recommend reading this paper as a supplement to this chapter.

Faraway (1997) used functional ANOVA to study three–dimensional movement trajectories in a complex industrial design setting. Muñoz Maldonado, Staniswalis, Irwin and Byers (2002) suggest three ways of testing the equality of curves collected from samples of young and old rats. Another application of functional ANOVA can be found in Ramsay, Munhall, Gracco and Ostry (1996), where variation in lip movement during the production of four syllables is analyzed at the level of both position and acceleration.

Spitzner, Marron and Essick (2003) combine functional ANOVA with a mixed-model approach to study human tactile perception. Fan and Lin (1998) proposed a method for testing for significance when the response variable is functional. Yu and Lambert (1999) fit tree models to functional responses.

Li, Aragon, Shedden and Thomas Agnan (2003) offer an approach that combines elements of the concurrent functional linear model discussed in Chapter 14, principal components analysis and the varying-coefficient model. This paper applies the *sliced inverse regression* or *SIR* method developed by Li (1991) to a functional response predicted by one or scalar independent variables.

Chiou, Müller and Wang (2003) describe an interesting variant of the functional linear model. They propose that principal components scores $f_{im}, m = 1, \ldots, M$, associated with the response functions x_i are related to the covariate values z_{ij} through

$$f_{im} = \alpha_m(\sum_j \beta_{mj} z_{ij}) + \epsilon_{im}. \tag{13.33}$$

This model combines a linear model for the arguments of the regression coefficient functions α_m with a principal components model for the response. Models in which argument values are themselves linear combinations of covariates are often referred to as *single index models*. The med-fly life history data to which the authors apply the model have been analyzed in many fascinating and original ways, and a collection of the papers on these data makes fascinating reading for anyone wishing to see functional data analysis in action.

14

Functional responses, functional covariates and the concurrent model

14.1 Introduction

We now consider a model for a functional response involving one or more functional covariates. In this chapter the influence of a covariate on the response is of a particularly elementary nature: The response y and each covariate z_j are both functions of the same argument t, and the influence is *concurrent, simultaneous* or *point-wise* in the sense that z_j only influences $y(t)$ through its value $z_j(t)$ at time t. This contrasts with the more general situation that we will defer for two chapters in which the influence of z_j can involve a range of argument values $z_j(s)$.

We will see that this functional/functional model involves only minor changes at the computational level of the functional response and multivariate covariate model in the last chapter. Perhaps this is not surprising, since a scalar covariate can be viewed as a functional covariate expanded in terms of a constant basis, where the single coefficient multiplying the basis function value 1 is the value of the covariate. Therefore the functional/multivariate model is really contained within what we take up in this chapter. But of course the fact that the functional covariate is not constant does add new features that now need to be considered. We begin with a concrete problem.

14.2 Predicting precipitation profiles from temperature curves

14.2.1 The model for the daily logarithm of rainfall

Predicting temperature is relatively easy, but predicting rainfall is quite another thing. Certainly there are important precipitation effects due to climate zones, but can we get additional predictability from the behavior of temperature? It seems likely, for example, that on days when the average temperature is high, precipitation tends to be low, at least in the summer. In the winter, on the other hand, most of the snowfall comes when the temperature is only a little below freezing; when it is really cold, it seldom snows since the atmosphere is too dry.

Here is an extension of the functional ANOVA model (13.1) that we could describe as a *functional analysis of covariance* model:

$$\log[\texttt{Prec}_{mg}(t)] = \mu(t) + \alpha_g(t) + \texttt{TempRes}_{mg}(t)\beta(t) + \epsilon_{mg}(t). \qquad (14.1)$$

We consider the log of precipitation as the response since precipitation is a magnitude, and experience indicates that logging magnitudes tends to improve the fitting power of linear models. As in Chapter 13, g indexes climate zones, m indexes weather stations within climate zones, and climate zone effects satisfy the constraint $\sum_g \alpha_g(t) = 0$.

The variable $\texttt{TempRes}_{mg}$ is the residual temperature after removing the temperature effect of climate zone g by using the techniques of Chapter 13. The motivation for removing temperature climate effects from the temperature profiles before using them in this model is that we have already allowed for these effects in the model. We don't want climate zones in the equation twice.

14.2.2 Preliminary steps

The average daily precipitation data for some extremely dry stations such as Resolute contain a number of zeros, and we dealt with this by replacing these with 0.05 mm since the smallest nonzero value was 0.1 mm. This permits us to smooth the logarithm of average precipitation directly. We first used 365 Fourier basis functions, and the same harmonic acceleration roughness penalty that we have been using for the weather data. The generalized cross-validation or GCV criterion was minimized for $\lambda = 10^6$, a level of smoothing that is equivalent to about 9.5 degrees of freedom. In order to speed up computation, we then opted for a simple Fourier basis expansion with eleven basis functions and no roughness penalization. For this analysis, we used an expansion of the daily average temperature residual in terms of 21 Fourier basis functions.

The smooth log precipitation curves for all 35 weather stations are shown in Figure 14.1. The rainiest place in Canada is unlucky Prince Rupert,

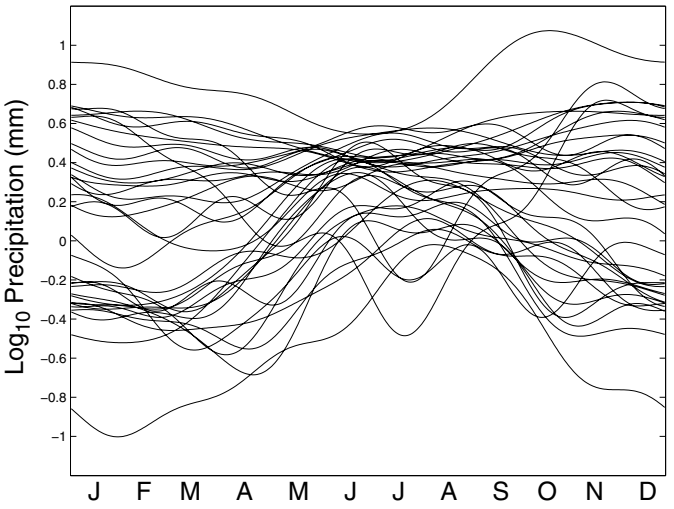

Figure 14.1. The logarithm (base 10) of average daily precipitation after smoothing for 35 Canadian weather stations.

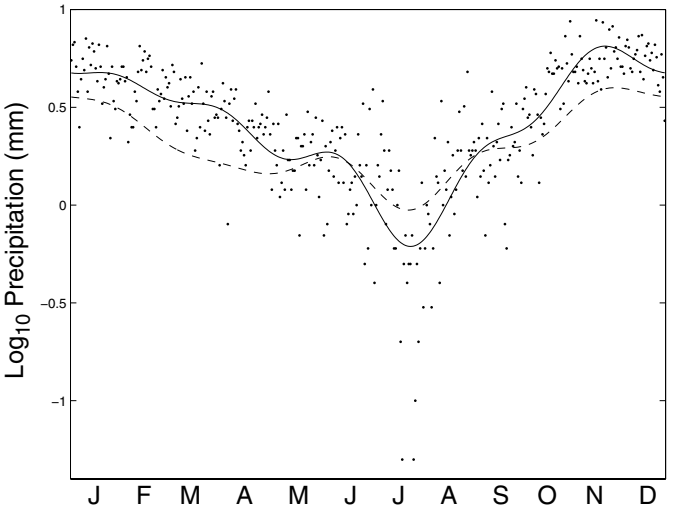

Figure 14.2. The $\log_{10}$ of average precipitation at Vancouver over 34 years is indicated by the dots, the smooth of the data using 11 Fourier basis functions by the solid curve, and the fit to the smooth curves by the point-wise linear model (14.1) by the dashed curve.

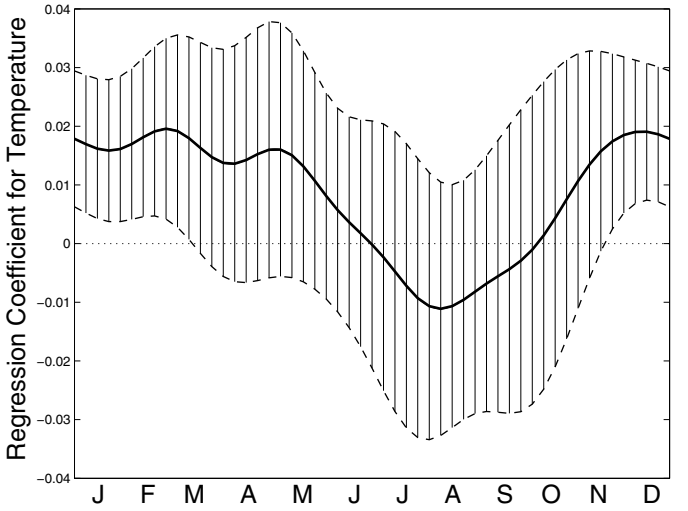

Figure 14.3. The regression coefficient for temperature with climate zone effects removed in a model predicting $\log_{10}$ rainfall. The solid line is the regression function and the cross-hatched area is the point-wise 95% confidence region for the function.

which averages nearly 12 mm of rain a day in October. The driest station is Resolute in the high arctic, where snowfall has a barely measurable rain equivalent of 0.1 mm per day in the winter. Figure 14.2 shows the resulting smooth to the precipitation data for Vancouver, a station that shows a sharp drop in rainfall during the summer months, and even records two days with no precipitation in 34 years.

14.2.3 Fitting the model and assessing fit

The unweighted least squares criterion for assessing fit is

$$\texttt{LMSSE}(\mu, \alpha_g, \beta) = \sum_{g}^{4} \sum_{m}^{N_g} \int \texttt{LogPrecRes}_{mg}^2(t)\, dt, \tag{14.2}$$

where

$$\texttt{LogPrecRes}_{mg}(t) = [\texttt{LogPrec}_{mg}(t) - \mu(t) - \alpha_g(t) - \texttt{TempRes}_{mg}(t)\beta(t)].$$

When we fit the model, using an approach that will be described in detail in the next section, we obtain a standard error of 467.9. If we drop `TempRes` from the model, this increases to 510.8, and these values are equivalent to $R^2 = 0.08$. Overall, the temperature residual functions don't seem to improve the fit by much. Figure 14.3 confirms this by showing point-wise 95%

confidence intervals for the estimated regression function for the residual temperature functions. The only part of the year where temperature seems to make a contribution is December through February.

However, it is potentially misleading to report that the regression coefficient is "significantly different from zero" at the end of January, since we are, in a sense, optimizing significance over a year's worth of results. The right way to proceed is to construct a contrast, a linear weighting of the entire year's information that focusses on the effect of interest. We can reasonably say that focussing on the effect in the winter is a test that we could propose in advance of collecting the data; we knew already that there is much more potential variation in rainfall across weather stations in the winter months and much more variability in temperature available then to predict it. As a contrast function or linear probe, we could propose

$$\xi(t) = \cos[2\pi(t - 64.5)/365],$$

where the shift value of 64.5 is defined by finding the low point in the mean precipitation profile, marking out empirically mid-winter. The inner product of the regression coefficient function with this probe,

$$\int_0^{365} \xi(t)\beta_6(t)\, dt = 2.32,$$

in effect accumulates information across the entire year about the difference between the summer and winter influence in temperature. Using the techniques described in Section 14.4, we can also work out the sampling standard error of this quantity, which in this case works out to 0.77. Taking ratio of the probe value to its standard error, we obtain $z = 3.0$. It is fairly reasonable to interpret this as a standard normal value under the null hypothesis of no difference in influence between summer and winter, and the value that we obtain appears to be inconsistent with this null hypothesis. It seems appropriate to declare that temperature has a small but statistically significant capacity to predict the log precipitation mean in mid-winter. We can conclude that, if the mean temperature residual for a weather station is high in winter, as it would be for marine stations like Prince Rupert, then precipitation will also be high for that station relative to other stations within the same climate zone.

14.3 Long-term and seasonal trends in the nondurable goods index

The nondurable goods manufacturing index, introduced in Chapter 1 and displayed in Figure 14.4 from 1952 to 2000, is a single long time series with a typical multiresolution structure. The global trend across these years is rather linear over large sections after logarithmic scaling. On a shorter

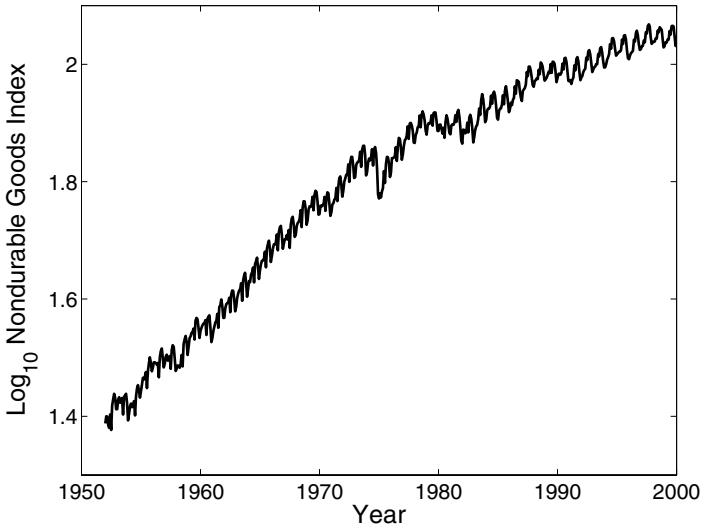

Figure 14.4. The United States nondurable goods manufacturing index plotted in logarithmic coordinates over the years 1952 to 2000.

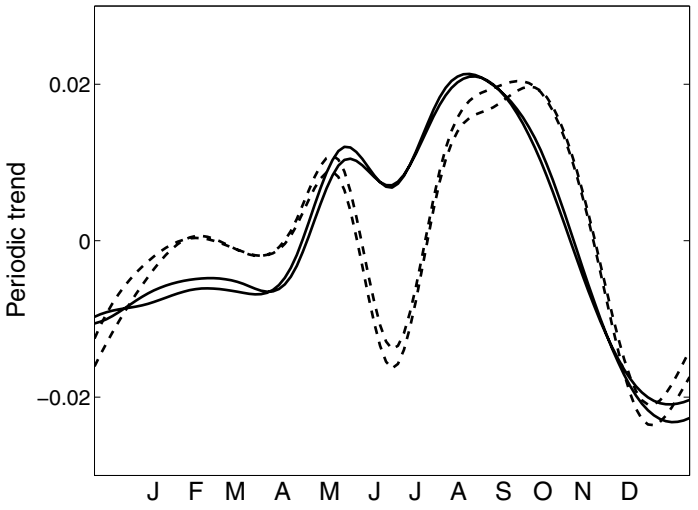

Figure 14.5. Four seasonal cycles for the logged United States nondurable goods manufacturing index are plotted with any overall linear trend removed. Two cycles in the 60's are plotted as dashed lines, and two cycles in the 90's as solid lines.

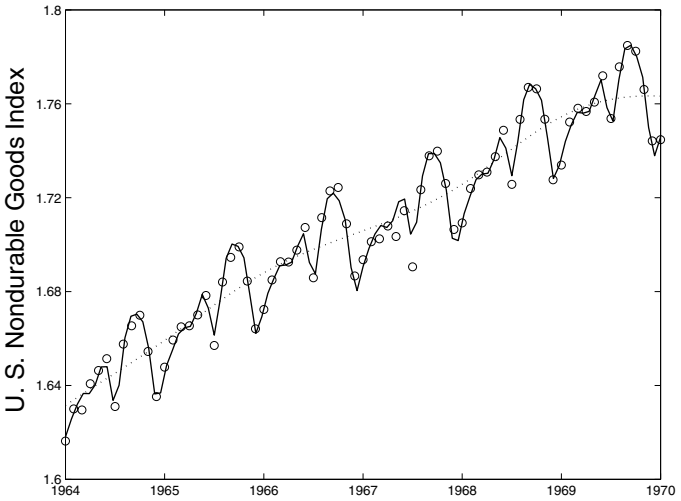

Figure 14.6. The fit to the smoothed logged United States nondurable goods manufacturing index using the point-wise linear model for six typical years. The points indicate the monthly values of the smoothed index, and the solid line is the fit based on the point-wise linear model. The dotted line indicates the estimated smooth nonseasonal trend.

scale, however, we see shocks to the system such as the end of the Vietnam War in 1974, and these seem to result in long-term changes in trend.

Moreover, like most economic indicators, there is a somewhat complex seasonal trend, and this is illustrated for four fairly representative years in Figure 14.5. There are three cycles evident in most years, separated by the Easter/Passover, summer school, and Christmas holidays, respectively.

The seasonal behavior seems to be fairly stable from one year to the next, but exhibits longer-term changes. The large autumn cycle shows a phase shift between the 60's and 90's, but there is little change in amplitude. The small winter cycle is much smaller in the 90's, but the dip due to the summer holidays is much more profound in the 60's.

We can use the point-wise linear model to separate out the smooth long-term trend from the seasonal trend, and at the same time show how the seasonal trend evolves. Our objective here is also to showcase the analysis of a single long time series rather than shorter but replicated series. This analysis used the 577 monthly values in the years 1952 to 2000. The original values were first smoothed by a smoothing spline with curvature penalized with a smoothing parameter value $\lambda = 10^{-6}$, and the smoothed version had a degrees of freedom equivalent of about 521.

The first covariate function z_1 is simply the constant function, and it is multiplied by a regression coefficient function β_1 that was expanded in

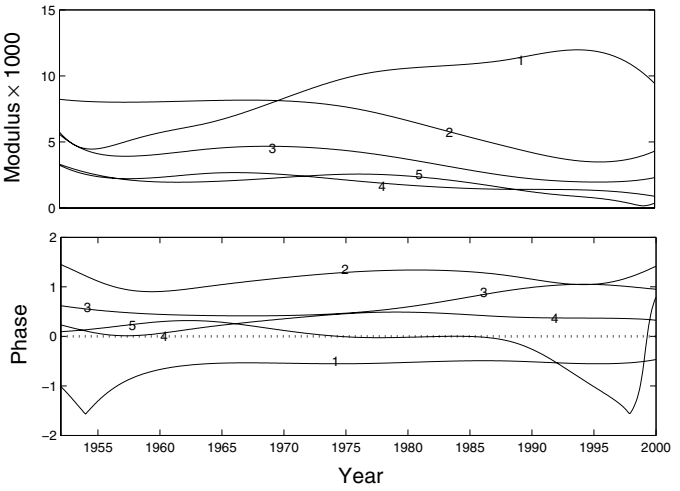

Figure 14.7. The evolution of seasonal trend in the logged United States nondurable goods manufacturing index. The top panel shows the modulus of the five sine/cosine pairs with the frequencies indicated in years. The bottom panel shows the phase for each pair, indicated as an angle in radians between $-\pi$ and π.

terms of cubic B-splines with knots placed at each year. This knot spacing is designed to allow β_1 to show smooth trend, but is too coarse to accommodate any seasonality. To further ensure that β_1 is sufficiently smooth, we penalized curvature with a smoothing parameter $\lambda = 0.01$.

An additional 10 covariate functions $z_2, \ldots, z_{11}$ were set up as a series of sine/cosine pairs with periods 1, 1/2, 1/3, 1/4 and 1/5 years, respectively. These are intended to model periodic seasonal effects. The corresponding β_j's were expanded in terms of seven B-spline basis functions with equal knot spacing, and these coefficient functions permit us to see any smooth changes in the structure of this seasonal tend.

Figure 14.6 shows the fit to the smoothed logged goods index by this model for the years 1964 to 1970 along with the smooth nonseasonal trend estimated by β_1. We see that the fit, based on 121 parameters and some smoothing, is quite good, and certainly captures the seasonal trend in a reasonable way. The turbulent few years in the mid-seventies are not shown, but the fit was not so good there, naturally, since we only allowed for rather smooth seasonal evolution.

How does the seasonal trend evolve? The top panel of Figure 14.7 shows the amplitude or modulus

$$\texttt{Mod}_j(t) = \sqrt{\beta_j^2(t) + \beta_{j+1}^2(t)}$$

of the sine/cosine pairs corresponding to $j = 2, 4, \ldots, 10$ of increasing frequency. The conclusion seems fairly clear: In later years, more of the seasonality is represented by the lowest harmonic with frequency of one year, and the energies in the higher frequency components tend to decline. Seasonal variation is tending to smooth out with time, perhaps due to the effects of automation of production, and the shifting of manufacturing with large seasonality to off-shore locations. The bottom panel shows the phase angle, measured in radians,

$$\texttt{Phase}_j(t) = \arcsin[\beta_j(t)/\texttt{Mod}_j(t)] \ .$$

Here we see little evolution, as we would expect since the timing of the cycles is tied to holidays, in the case of summer and Christmas at least, whose timing is fixed. We no doubt could have done better if we had allowed for the variable timing of the Easter/Passover holiday.

14.4 Computational issues

We have q covariate functions z_{ij}, each multiplied by its regression coefficient function β_j. Our concurrent multiple regression model is

$$y_i(t) = \sum_{j=1}^{q} z_{ij}(t)\beta_j(t) + \epsilon_i(t) \ . \tag{14.3}$$

Let the N by q functional matrix $\mathbf{Z}$ contain these z_{ij}'s, and let the vector coefficient function $\boldsymbol{\beta}$ of length q contain each of the regression functions. The concurrent functional linear model in matrix notation is then

$$\mathbf{y}(t) = \mathbf{Z}(t)\boldsymbol{\beta}(t) + \boldsymbol{\epsilon}(t) \ , \tag{14.4}$$

where $\mathbf{y}$ is a functional vector of length N containing the response functions.

We estimate a basis function expansion for each regression function $\beta_j, j = 1, \ldots, q$ along with roughness penalties to control the smoothness of the estimates for the β_j's. We must allow for both the basis and the roughness penalty to vary from one β_j to another; some regression functions may be assumed to only pick up very smooth effects requiring only a few basis functions, while others may be required to model high-frequency variability in the data. This means that we will have to possibly define a roughness penalty

$$\texttt{PEN}_j(\beta_j) = \lambda_j \int [L_j\beta_j(t)]^2 \, dt$$

separately for each regression coefficient function. Each penalty is defined by choosing a linear differential operator L_j that is appropriate for that functional parameter, such as the curvature operator $L_j = D^2$ or the harmonic acceleration operator $L_j = (2\pi/365)^2 D + D^3$.

The weighted regularized fitting criterion is

$$\texttt{LMSSE}(\boldsymbol{\beta}) = \int \mathbf{r}(t)'\mathbf{r}(t)\,dt + \sum_{j}^{p} \lambda_j \int [L_j\beta_j(t)]^2\,dt, \tag{14.5}$$

where

$$\mathbf{r}(t) = \mathbf{y}(t) - \mathbf{Z}(t)\boldsymbol{\beta}(t).$$

Let regression function β_j have the expansion

$$\beta_j(t) = \sum_{k}^{K_j} b_{kj}\theta_{kj}(t) = \boldsymbol{\theta}_j(t)'\mathbf{b}_j(t)$$

in terms of K_j basis functions θ_{kj}. In order to express (14.4) and (14.5) in matrix notation referring explicitly to these expansions, we need to construct some composite or super matrices.

Defining

$$K_\beta = \sum_{j}^{q} K_j,$$

we first construct vector $\mathbf{b}$ of length K_β by stacking the vectors vertically, that is,

$$\mathbf{b} = (\mathbf{b}_1', \mathbf{b}_2', \ldots, \mathbf{b}_q')' \ .$$

Now assemble q by K_β matrix function $\boldsymbol{\Theta}$ as follows:

$$\boldsymbol{\Theta} = \begin{bmatrix} \boldsymbol{\theta}_1' & 0 & \cdots & 0 \\ 0 & \boldsymbol{\theta}_2' & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{\theta}_q' \end{bmatrix} . \tag{14.6}$$

We can now express model (14.4) as

$$\mathbf{y}(t) = \mathbf{Z}(t)\boldsymbol{\Theta}(t)\mathbf{b} + \boldsymbol{\epsilon}(t) \ . \tag{14.7}$$

Note that the model can be formally transformed to a *constant coefficient linear model* by defining N by K_β functional matrix $\mathbf{Z}^*(t)$ as

$$\mathbf{Z}^*(t) = \mathbf{Z}(t)\boldsymbol{\Theta}(t)$$

so that

$$\mathbf{y}(t) = \mathbf{Z}^*(t)\mathbf{b} + \boldsymbol{\epsilon}(t) \ . \tag{14.8}$$

This doesn't really gain anything computationally since we achieve constant coefficients at the price of going from q covariates to the greatly expanded number of K_β covariates.

But this formalism (14.8) makes clear that the functional linear model has K_β parameters. If each of the Y_i response functions is expanded in

terms of K_y basis functions, then the total number of degrees of freedom for error df_e in the model becomes

$$df_e = NK_y - K_\beta \ .$$

Keeping these numbers in mind helps us to avoid over-fitting the data, an ever-present hazard in the world of functional data analysis. We will show in a couple of chapters that all of the functional linear models considered in this book can be re-expressed in this constant coefficient form (14.8).

In order to take care of the roughness penalties, we also need to arrange the order K_j roughness penalty matrices multiplied by their respective smoothing parameters,

$$\mathbf{R}_j = \lambda_j \int \boldsymbol{\theta}_j(t)\boldsymbol{\theta}_j'(t)\, dt \ ,$$

into the symmetric block diagonal matrix $\mathbf{R}$ of order K_β:

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{R}_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \mathbf{R}_q \end{bmatrix} \ . \tag{14.9}$$

We can now write down the normal equations weighted least squares solution for the composite coefficient vector b:

$$[\int \boldsymbol{\Theta}'(t)\mathbf{Z}'(t)\mathbf{Z}(t)\boldsymbol{\Theta}(t)\, dt + \mathbf{R}]\mathbf{b} = [\int \boldsymbol{\Theta}'(t)\mathbf{Z}'(t)\mathbf{y}(t)\, dt] \ . \tag{14.10}$$

The amount of numerical integration involved in these expressions is really quite manageable. The scalar functions

$$\omega_{j\ell}(t) = \sum_i^N z_{ij}(t)z_{i\ell}(t)$$

play the role of *weighting functions* for the functional inner products

$$\int \boldsymbol{\theta}_j(t)\boldsymbol{\theta}_\ell'(t)\omega_{j\ell}(t)\, dt, j, \ell = 1, \ldots, q \ .$$

Similarly, on the right side, we have a set of inner products of the basis functions $\boldsymbol{\theta}_j$ with the unit function $\mathbf{1}$ weighted by the scalar functions $\sum_i^N z_{ij}(t)y_i(t)$. Computing these inner products by numerical integration is a fairly routine procedure.

14.5 Confidence intervals

In order to compute confidence intervals, we also have to explicate the role of the coefficient matrix $\mathbf{C}$ in the basis function expansions of the response

functions, expressed as $y = \mathbf{C}\boldsymbol{\phi}$, where the basis function vector $\boldsymbol{\phi}$ is of length K_y. This results in

$$\begin{aligned}\hat{b} &= \Big[\int \boldsymbol{\Theta}'\mathbf{Z}'\mathbf{Z}\boldsymbol{\Theta} + \mathbf{R}\Big]^{-1}\Big[\int \boldsymbol{\Theta}'\mathbf{Z}'\mathbf{C}\boldsymbol{\phi}\Big] \\ &= \Big[\int \boldsymbol{\Theta}'\mathbf{Z}'\mathbf{Z}\boldsymbol{\Theta} + \mathbf{R}\Big]^{-1}\Big[\int \boldsymbol{\phi}' \otimes (\boldsymbol{\Theta}'\mathbf{Z}')\Big]\mathrm{vec}(\mathbf{C}) \; . \qquad (14.11)\end{aligned}$$

Here the K_β by $K_y N$ composite matrix $\boldsymbol{\phi}' \otimes (\boldsymbol{\Theta}'\mathbf{Z}')$ has the structure

$$\begin{bmatrix} \phi_1\boldsymbol{\theta}_1\mathbf{Z}_1' & \cdots & \phi_{K_y}\boldsymbol{\theta}_1\mathbf{Z}_1' \\ \vdots & \cdots & \vdots \\ \phi_1\boldsymbol{\theta}_q\mathbf{Z}_q' & \cdots & \phi_{K_y}\boldsymbol{\theta}_q\mathbf{Z}_q' \end{bmatrix} ,$$

where the vector function $\mathbf{Z}_j$ is the jth column of $\mathbf{Z}$. Recall that in this expression ϕ_k is a scalar basis function, whereas $\boldsymbol{\theta}_j$ is a basis function vector of length K_j.

Here again, the numerical integration can be reduced considerably when the jth covariate has the expansion $\mathbf{Z}_j = \mathbf{D}_j\boldsymbol{\psi}_j$. In this event, $\boldsymbol{\phi}' \otimes (\boldsymbol{\Theta}'\mathbf{Z}')$ is

$$\begin{bmatrix} \phi_1\boldsymbol{\theta}_1\boldsymbol{\psi}_1'\mathbf{D}_1' & \cdots & \phi_{K_y}\boldsymbol{\theta}_1\boldsymbol{\psi}_1'\mathbf{D}_1' \\ \vdots & \cdots & \vdots \\ \phi_1\boldsymbol{\theta}_q\boldsymbol{\psi}_q'\mathbf{D}_q' & \cdots & \phi_{K_y}\boldsymbol{\theta}_q\boldsymbol{\psi}_q'\mathbf{D}_q' \end{bmatrix} .$$

We see in this expression that we need inner products $\langle\boldsymbol{\theta}_j, \boldsymbol{\psi}_\ell'\rangle$ with weighting functions ϕ_k.

Finally, the matrix representing the mapping `C2BMap` that we need to put together the mapping `Y2RMap` to construct confidence intervals is

$$\texttt{C2BMap} = \Big[\int \boldsymbol{\Theta}'\mathbf{Z}'\mathbf{Z}\boldsymbol{\Theta} + \mathbf{R}\Big]^{-1}\Big[\int \boldsymbol{\phi}' \otimes (\boldsymbol{\Theta}'\mathbf{Z}')\Big] \; . \qquad (14.12)$$

14.6 Further reading and notes

Models that are closely related to the point-wise linear model have been considered by a number of authors. West, Harrison and Migon (1985) investigated what was essentially model (14.3), but with the restriction that the regression coefficient functions $\beta_j(t)$ have a simple autoregressive time series structure. They referred to this structure as a *dynamic generalized linear model,* and went on to consider various extensions in West and Harrison (1989).

Hastie and Tibshirani (1993) looked at a version of this model within what they called *varying coefficient* models of the form

$$y_i = \sum_j \beta_j(R_{ij})z_{ij} + \epsilon_i. \qquad (14.13)$$

They explored various strategies for obtaining flexible estimates of the functions β_js, including the use of spline basis expansions with roughness penalties. This paper, as well as the work of West, et al (1985, 1989), contain many interesting examples and illustrate the principle that a number of estimation strategies can be developed for models like these. The discussions associated with the two journal articles cited here also contain many useful alternative perspectives.

The varying coefficient model has subsequently received a lot of attention, with much of this devoted to estimation of smooth regression functions by kernel smoothing (Wu, Chiang and Hoover (1998)), local polynomial smoothing (Fan, Yao, and Cai (2003); Neilsen, Nielsen and Joensen, Madsen and Holst (2000); Zhang and Lee (2000); Zhang, Lee and Song (2002)) and local maximum likelihood estimation (Cai, Fan and Li (2000); Cai, Fan and Yao (2000); Dreesman and Tutz (2001)). Gelfand, Kim, Sirmans and Banerjee (2003) used a Bayesian model for spatial variation in regression coefficients.

While the varying coefficient model certainly involves one or more functional parameters, the data involved are more typically multivariate rather than functional. In many applications, the argument variable r_j for β_j is a spatial dimension, and the corresponding covariate $\mathbf{z}_j$ is fixed rather than varying over some argument. From this perspective, the varying coefficient model is closer to the *generalized additive model* (Hastie and Tibshirani, 1990).

It is likely, though, that the techniques associated with varying coefficient problems will prove useful in functional data settings as well. This is especially evident in Eubank, Muñoz Maldonado, Wang and Wang (2004), where the model being investigated is essentially the concurrent functional linear model.

15

Functional linear models for scalar responses

15.1 Introduction

In this chapter, we consider a linear model defined by a set of functions, but where the response variable is scalar or multivariate. This contrasts with Chapter 13, where the responses and the parameters were functional, but, because of the finite and discrete covariate information, the linear transformation from the parameter space to the observation space was still specified by a *design matrix* $\mathbf{Z}$ as in the conventional multivariate general linear model

$$\mathbf{y} = \mathbf{Z}\mathbf{b} + \boldsymbol{\epsilon} \; . \tag{15.1}$$

We now consider a functional extension of linear regression where the prediction of the scalar values y_i is based on functions z_i. This problem is of interest in its own right, and also raises issues about more complicated problems in subsequent chapters.

For illustration, let us predict total annual precipitation for a Canadian weather station from the pattern of temperature variation through the year. To this end, let $y_i = \texttt{LogPrec}_i$ be the logarithm of total annual precipitation at weather station i, and let $z_i = \texttt{Temp}_i$ be its daily temperature function. We now replace the regression vector $\mathbf{b}$ in (15.1) by a function β, so that the model now takes the form

$$\texttt{LogPrec} = \alpha + \int_0^T \texttt{Temp}(s)\beta(s)\, ds + \boldsymbol{\epsilon} \; . \tag{15.2}$$

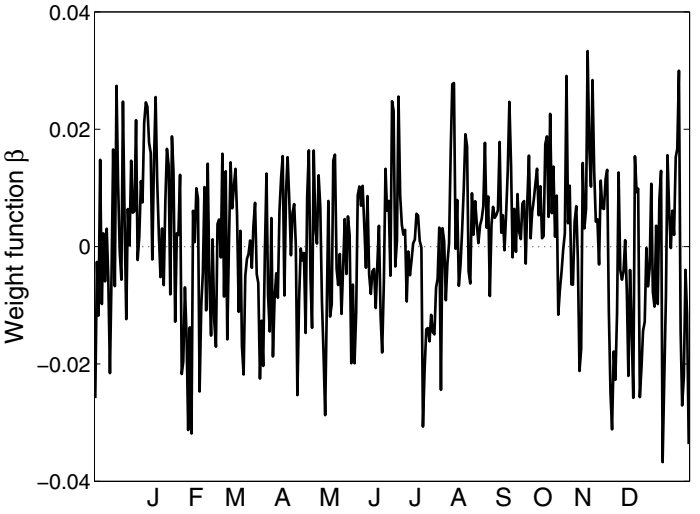

Figure 15.1. The weight function β that allows perfect prediction of log total annual precipitation from observed annual pattern of temperature.

We see that the summation implied in the matrix product **Zb** in (15.1) is now replaced by an integration over a continuous index s in (15.2).

15.2 A naive approach: Discretizing the covariate function

It might occur to us to treat the values of temperature at each observation point as a separate covariate, and then just proceed with ordinary multiple regression. This would certainly get us into trouble! To see why, suppose that $\texttt{Temp}_{ij}$ is the entry for the temperature at station i on day j, and we wish to predict $\texttt{LogPrec}_i$ by

$$\texttt{LogPrec}_i = \alpha + \sum_{j=1}^{365} \texttt{Temp}_{ij}\beta_j + e_i \; i = 1, 2, \ldots, 35. \tag{15.3}$$

We can view this as a finely discretized version of the functional model being considered. This is a system of 35 equations with 366 unknowns. Even if the coefficient matrix is of full rank, there are still infinitely many sets of solutions, all giving a perfect prediction of the observed data. Figure 15.1 plots the b_j's for one such solution, and it is hard to imagine that we can make much practical use out of such a result.

Returning to the functional model (15.2), we now understand that the regression coefficient function β is bound to be under-determined on the

basis of any finite sample (z_i, y_i). This is because, essentially, we have an infinite number of parameters $\beta(s)$ available by discretizing s finely enough, but a finite number of conditions $y_i = \alpha + \int z_i\beta$ to approximate. Usually it is possible to find $\hat{\alpha}$ and $\hat{\beta}$ to reduce the residual sum of squares (15.2) to zero. Furthermore, if β^* is any function satisfying $\int z_i\beta^* = 0$ for $i = 1, \ldots, N$, then adding β^* to $\hat{\beta}$ does not affect the value of the residual sum of squares.

In the weather data example, a possible approach is to reduce the number of unknowns in problem (15.3) by considering the temperatures on a coarser time scale. It is unlikely that overall precipitation is influenced by details of the temperature pattern from day to day, and so, for example, we could investigate how the 12-vectors of monthly average temperatures can be used to predict total annual precipitation. If $\mathbf{Z}$ is the 35×12 matrix containing these values, we can then fit a model of the form $\hat{y} = \hat{\alpha} + \mathbf{Z}\hat{\beta}$, where $\hat{y}$ is the vector of values of log annual precipitation predicted by the model, and $\hat{\beta}$ is a 12-vector of regression parameter estimates. Since the number of parameters to be estimated is now only 13, and thus less than the number of observations $N = 35$, we can use standard multiple regression to fit the model by least squares.

We can summarize the fit in terms of the conventional $R^2 = 1 - \texttt{SSE}/\texttt{SSY}$ measure, and this is 0.84, indicating a rather successful fit, even taking into account the 13 parameters in the model. The corresponding F-ratio is 9.8 with 12 and 22 degrees of freedom, and is significant at the 1% level. The standard error estimate is 0.34, as opposed to the standard deviation of the dependent variable of 0.69.

Figure 15.2 presents the estimated regression function β, obtained by interpolating the individual estimated coefficients $\hat{\beta}_j$ as marked on the figure. It is still not easy to interpret this function directly, although it clearly places considerable emphasis on temperature in the months of April, May, August and September. The lack of any very clear interpretation indicates that this problem raises statistical questions beyond the formal difficulty of fitting an under-determined model. In any case, the model certainly uses up a rather large proportion of the 35 degrees of freedom available in the data.

Since the space of functions satisfying (15.2) is infinite-dimensional, no matter how large our sample size N is, minimizing the residual sum of squares cannot, of itself, produce a meaningful or consistent estimator of the parameters β in the model (15.2). Consequently, to provide an estimate of $\hat{\beta}$ that we can interpret or otherwise use, or even just identify uniquely, we must use some method of regularization, and this is discussed in the following sections.

In short, penalizing roughness when a functional covariate is involved is no longer cosmetic, but an essential aspect of finding a useful solution. We have already seen this issue discussed in Section 11.5 in functional canonical correlation analysis, and we will consider it again in the next chapter.

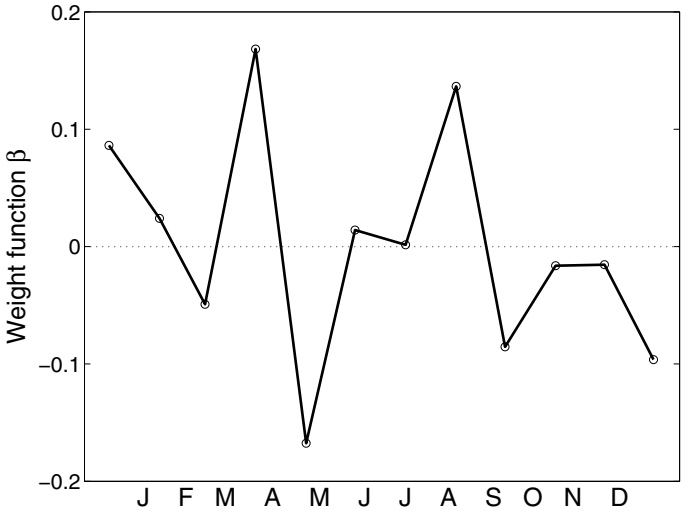

Figure 15.2. The regression function β for the approximation of annual mean log precipitation by the temperature profiles for the Canadian weather stations.

15.3 Regularization using restricted basis functions

To reduce the degrees of freedom in the model still further, we now expand the regression function β in terms of a set of basis functions $\theta_k(s)$, and the Fourier basis is the logical choice here because of the the underlying smoothness and stationarity of the seasonal variation in temperature. Let $\boldsymbol{\theta}$ be a vector of Fourier basis functions of length K_β, so that

$$\beta(s) = \sum_{k}^{K_\beta} b_k \theta_k(s) \quad \text{or} \quad \beta = \boldsymbol{\theta}'\mathbf{b}. \tag{15.4}$$

We choose some suitably large K_β that does not entail any significant loss of information, but hopefully keeps K_β small enough so that we can reasonably interpret β.

At the same time, let us assume that the covariate functions $\texttt{Temp}_i$ are also expanded in terms of Fourier basis vector $\boldsymbol{\psi}$ of length K_z, so that

$$\texttt{Temp}_i(s) = \sum_{k}^{K_z} c_{ik}\psi_k(s) \quad \text{or} \quad \texttt{Temp}(s) = \mathbf{C}\boldsymbol{\psi}(s)\ , \tag{15.5}$$

where coefficient matrix $\mathbf{C}$ is N by K_z. For the monthly and daily temperature data, for example, K_z would be 12 and 365, respectively.

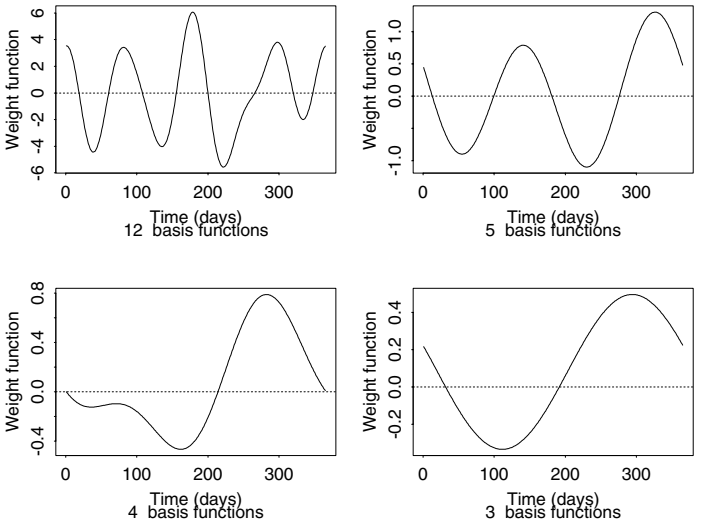

Figure 15.3. Estimated regression weight functions β using $K_\beta = 12$, 5, 4 and 3 basis functions.

Now the model can be expressed as

$$\hat{y}_i = \int_0^T \texttt{Temp}(s)\beta(s)\,ds = \int_0^T \mathbf{C}\psi(s)\theta(s)'\mathbf{b}\,ds = \mathbf{C}\mathbf{J}_{\psi\theta}\mathbf{b}\ , \tag{15.6}$$

where K_z by K_β matrix $\mathbf{J}_{\psi\theta}$ is defined by

$$\mathbf{J}_{\psi\theta} = \int \boldsymbol{\psi}(s)\boldsymbol{\theta}'(s)\,ds\ . \tag{15.7}$$

We can further simplify notation by defining the $(K_\beta + 1)$-vector $\boldsymbol{\zeta} = (\alpha, b_1, \ldots, b_K)'$ and defining the coefficient matrix $\mathbf{Z}$ to be the $N \times (K_\beta + 1)$ matrix $\mathbf{Z} = [\mathbf{1}\ \ \mathbf{C}\mathbf{J}_{\psi\theta}]$. Then the model (15.1) becomes simply

$$\hat{\mathbf{y}} = \mathbf{Z}\hat{\boldsymbol{\zeta}} \tag{15.8}$$

and the least squares estimate of the augmented parameter vector $\boldsymbol{\zeta}$ is the solution of the equation

$$\mathbf{Z}'\mathbf{Z}\hat{\boldsymbol{\zeta}} = \mathbf{Z}'\mathbf{y}\ . \tag{15.9}$$

A convenient method of regularization that we used in Chapter 4 is to truncate the basis by choosing a value $K_\beta < K_z$. We can then fit $\boldsymbol{\zeta}$ by least squares, and the problem is now a standard multiple regression problem.

Figure 15.3 shows the result of carrying out this procedure for the daily weather data with varying numbers K_β of basis functions. The choice $K_\beta =$

12 is intended to correspond to the same amount of discretization as using monthly average data, and we can see that the weight function is similarly uninformative. To obtain results more likely to be meaningful, we have to use a much smaller number of basis functions, and, by considering the graphs for $K_\beta = 4$ and $K_\beta = 3$, it appears that a predictor for high precipitation is a relatively high temperature towards the end of the year.

But the model complexity increases in discrete jumps as K_β varies from three to five, and we might want finer control. Also, to obtain reasonable results, β must be rigidly constrained to lie in a low-dimensional parametric family, and we may worry that we are missing important features in β as a consequence. Section 15.4 develops a more flexible approach making use of a roughness penalty method.

15.4 Regularization with roughness penalties

The estimated function $\hat{\beta}$ in Figure 15.1 illustrates that fidelity to the observed data, as measured by the residual sum of squares, is not the only aim of the estimation. The roughness penalty approach makes explicit the complementary, possibly even conflicting, aim of avoiding excessive local fluctuation in the estimated function.

To this end, we can define the penalized residual sum of squares

$$\texttt{PENSSE}_\lambda(\alpha, \beta) = \sum_{i=1}^{N} [y_i - \alpha - \int z_i(s)\beta(s)\, ds]^2 + \lambda \int [L\beta(s)]^2\, ds \ , \quad (15.10)$$

where L is a linear differential operator that is suitable for the problem. In this situation, it is reasonable to expect that regression function β will be periodic, just like the average temperature function that it multiplies. Consequently, it seems appropriate to choose *harmonic acceleration* as the type of roughness to penalize. That is, we choose

$$L\beta = (\frac{2\pi}{365})^2 D\beta + D^3\beta$$

so that in the limit, as $\lambda \to \infty$, the regression function will approach a shifted sinusoid. Sections 15.5 and 15.7 discuss the algorithmic aspects of minimizing (15.10).

We can choose the smoothing parameter λ either subjectively or by an automatic method such as cross-validation. To apply the cross-validation paradigm in this context, let $\alpha_\lambda^{(-i)}$ and $\beta_\lambda^{(-i)}$ be the estimates of α and β obtained by minimizing the penalized residual sum of squares based on all the data except (z_i, y_i). We can define the cross-validation score as

$$\texttt{CV}(\lambda) = \sum_{i=1}^{N} \left[y_i - \alpha_\lambda^{(-i)} - \int z_i(s)\beta_\lambda^{(-i)}(s)\right]^2 ds \quad (15.11)$$

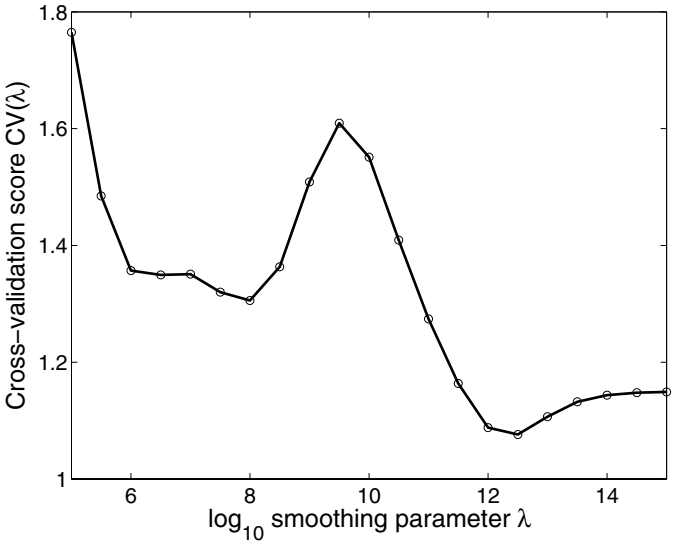

Figure 15.4. The cross-validation score function $\texttt{CV}(\lambda)$ for fitting log annual precipitation by daily temperature variation, with a penalty on the size of harmonic acceleration. The logarithm of the smoothing parameter is taken to base 10.

and minimizing $\texttt{CV}(\lambda)$ over λ gives an automatic choice of λ. In practice, there are efficient algorithms for calculating the cross-validation score, and Section 15.6 discusses these.

We used 65 basis functions to represent the temperature curves and 35 Fourier basis functions to represent β. With this number of basis functions for β, it would be possible to exactly fit the data from the 35 weather stations. However, we wanted to see how well cross-validation would help us in arriving at a reasonable fit by penalizing harmonic acceleration. Figure 15.4 plots the cross-validation score against the logarithms of various values of λ. The plot shows two distinct minima over the range of values plotted. Not shown, however, is the fact that fitting the data exactly or nearly exactly actually gave a smaller cross-validation score than either of these minima. We chose $\lambda = 10^{12.5}$ for the final fit, corresponding to the lower minimum in the plot.

Figure 15.5 shows the estimated regression function along with pointwise 95% confidence limits. The confidence intervals in the earlier summer months contain zero, suggesting that the influence of temperature on precipitation in that period is not important. However, we see a strong peak in the late fall followed by a valley in the early spring. This pattern is, in effect, computing a contrast between fall and early spring temperatures, with more emphasis on the autumn. This pattern favors weather stations that are comparatively warm in October and cool in spring, and where, moreover, spring comes early. This is just what we saw in Chapter 7 for

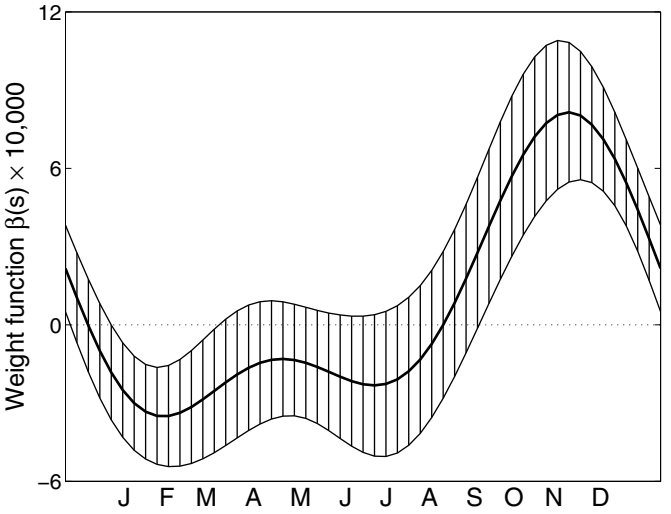

Figure 15.5. The estimated weight function for predicting the log total annual precipitation from the daily temperature pattern. The estimate was constructed by the penalizing the size of harmonic acceleration, with the smoothing parameter $\lambda = 10^{12.5}$ chosen by cross-validation.

the Pacific and Atlantic stations with marine climates, where the seasons are later than average and the fall weather is warm relative to the inland stations.

In Figure 15.6, we have plotted the observed values y_i against the fitted values $\hat{y}_i$ obtained using this functional regression. The squared correlation between the predicted and actual values in the plot is 0.75. This simple regression diagnostic seems to confirm the model assumptions. However, we didn't do so well for Kamloops, whose predicted value of about 2.9 is well above its actual value of a bit under 2.5. But Kamloops is deep in the Thompson River valley, and the rain clouds usually just pass on by. Section 15.6 describes another diagnostic plot.

15.5 Computational issues

A basis function approach has appeal because it is especially simple to apply, and moreover some problems in any case suggest a particular choice of basis. The periodic nature of the temperature and precipitation data, for example, seems naturally to call for the use of a Fourier series basis. Our first strategy is therefore to represent the regularized fitting problem in terms of a basis function expansion, and then to apply the concept of regularization to this representation.

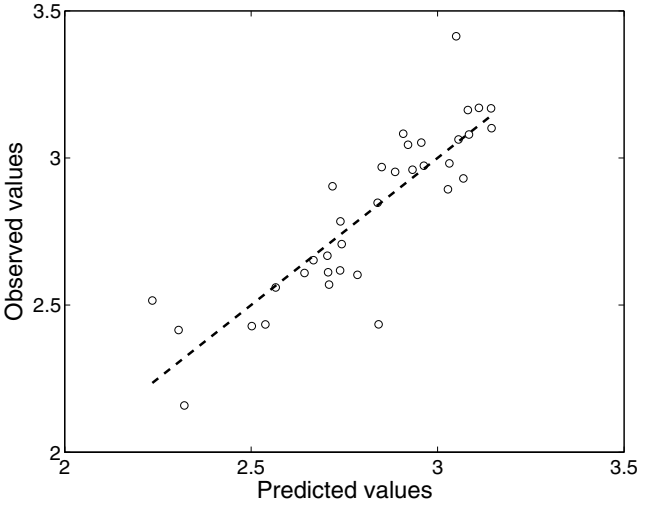

Figure 15.6. Observed values y_i of log annual precipitation plotted against the values $\hat{y}_i$ predicted by the functional regression model with the smoothing parameter chosen by cross-validation. The straight line corresponds to zero residuals.

15.5.1 Computing the regularized solution

Suppose that we expand the covariate functions z_i to K_z terms relative to basis functions ψ_m and the regression function β to K_β terms relative to basis functions θ_k, as in (15.5) and (15.4), respectively. Define a matrix $\mathbf{R}$ as

$$\mathbf{R} = \int [D^2\boldsymbol{\phi}(s)][D^2\boldsymbol{\phi}'(s)]\, ds \ . \tag{15.12}$$

In the Fourier case, note that $\mathbf{R}$ is diagonal, with diagonal elements ω_k^4 as in Section 9.4.1. In general, the penalized residual sum of squares can be written as

$$\texttt{PENSSE}_\lambda(\alpha, \beta) = \|\mathbf{y} - \alpha - \mathbf{C}\mathbf{J}_{\psi\theta}\mathbf{b}\|^2 + \lambda \mathbf{b}'\mathbf{R}\mathbf{b}. \tag{15.13}$$

where $\mathbf{J}_{\psi\theta}$ was defined in (15.7). As before, we deal with the additional parameter α by defining the augmented vector $\boldsymbol{\zeta} = (\alpha, \mathbf{b}')'$, and at the same time use $\mathbf{Z}$ as the $N \times (K_z + 1)$ coefficient matrix $[\mathbf{1} \ \ \mathbf{C}\mathbf{J}_{\psi\theta}]$. Finally, let the penalty matrix $\mathbf{R}$ be augmented by attaching a leading column and row of $K_z + 1$ zeros to yield $\mathbf{R}_0$. In terms of these augmented arrays, the expression (15.13) further simplifies to

$$\texttt{PENSSE}_\lambda(\boldsymbol{\zeta}) = \|\mathbf{y} - \mathbf{Z}\boldsymbol{\zeta}\|^2 + \lambda \boldsymbol{\zeta}'\mathbf{R}_0\boldsymbol{\zeta}. \tag{15.14}$$

It follows that the minimizing value $\hat{\boldsymbol{\zeta}}$ satisfies

$$(\mathbf{Z}'\mathbf{Z} + \lambda\mathbf{R}_0)\hat{\boldsymbol{\zeta}} = \mathbf{Z}'\mathbf{y}. \tag{15.15}$$

15.5.2 Computing confidence limits

We can again follow the procedure that we used in previous chapters to compute sampling standard errors for the coefficients in $\mathbf{b}$ and the intercept α in the composite parameter vector $\boldsymbol{\zeta}$. Things are simpler here in one sense since there is no intermediate step of smoothing the response variable. Consequently, we can drop the mapping `y2cMap`.

The matrix corresponding to `y2bMap` can be simply lifted from (15.15), and is $(\mathbf{Z}'\mathbf{Z}+\lambda\mathbf{R}_0)^{-1}\mathbf{Z}'$. The variance-covariance matrix Σ_e computed from the residuals is now a scalar estimate σ_e^2 of the mean squared residual. The sampling variance of $\hat{\boldsymbol{\zeta}}$ is given by

$$\mathtt{Var}[\hat{\boldsymbol{\zeta}}] = \sigma_e^2(\mathbf{Z}'\mathbf{Z} + \lambda\mathbf{R}_0)^{-1}\mathbf{Z}'\mathbf{Z}(\mathbf{Z}'\mathbf{Z} + \lambda\mathbf{R}_0)^{-1} \ . \tag{15.16}$$

15.6 Cross-validation and regression diagnostics

We have already noted the possibility of choosing the smoothing parameter λ by cross-validation. Various economies are possible in calculating the cross-validation score $\mathtt{CV}(\lambda)$ as defined in (15.11).

Let $\mathbf{S}$ be the so-called *hat matrix* of the smoothing procedure which maps the data values y to their fitted values $\hat{y}$ for any particular value of λ. A calculation described, for example, in Section 3.2 of Green and Silverman (1994), shows that the cross-validation score satisfies

$$\mathtt{CV}(\lambda) = \sum_{i=1}^{N}\left(\frac{y - \hat{y}_i}{1 - S_{ii}}\right)^2 .$$

If N is large and we are considering an expansion in a moderate number K of basis functions, then we can find the diagonal elements of $\mathbf{S}$ directly from

$$\mathbf{S} = \mathbf{Z}(\mathbf{Z}'\mathbf{Z} + \lambda\mathbf{R})^{-1}\mathbf{Z}'.$$

From $\mathbf{S}$, we can also compute an indicator of the effective degrees of freedom used up in the approximation. Either trace $\mathbf{S}$ or trace $\mathbf{S}^2$ were recommended for this purpose by Buja, Hastie, and Tibshirani (1989). For the fit in Figure 15.5, defined by minimizing the cross-validation criterion, the effective degrees of freedom are estimated to be trace $\mathbf{S} = 4.7$.

Another important use of the hat matrix $\mathbf{S}$ is in constructing various regression diagnostics. The diagonal elements of the hat matrix are often called *leverage values*; they determine the amount by which the fitted value $\hat{y}_i$ is influenced by the particular observation y_i. If the leverage value

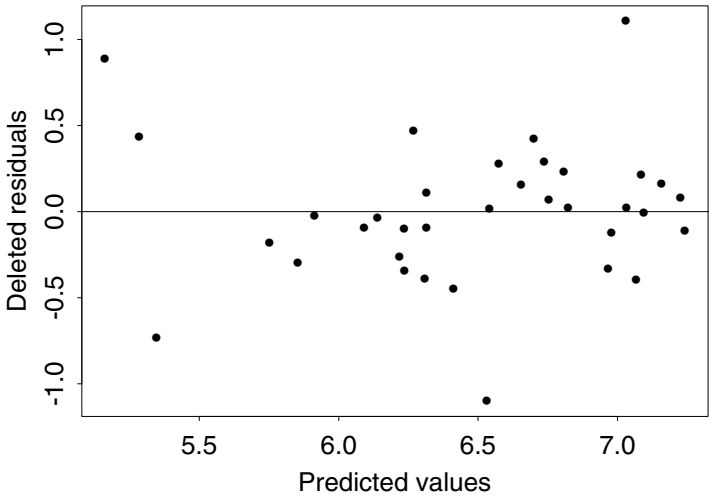

Figure 15.7. Deleted residuals from the fitted prediction of log annual precipitation from overall temperature pattern.

is particularly high, the fitted value needs to be treated with some care. Two standard ways of assessing the regression fit are to examine the raw residuals $y_i - \hat{y}_i$ and the *deleted residuals* $(y_i - \hat{y}_i)/(1 - S_{ii})$; the latter give the residual between y_i and the value predicted from the data set with case i deleted. We refer readers to works on regression diagnostics such as Cook and Weisberg (1982).

Figure 15.7 shows a plot of deleted residuals against fitted values for the log precipitation and temperature example, with the smoothing parameter chosen by cross-validation. The three observations with small predicted values have somewhat larger leverage values (around 0.4) than the others (generally in the range 0.1 to 0.2). This is not surprising, given that they are somewhat isolated from the main part of the data.

15.7 The direct penalty method for computing β

We now turn to a more direct way of using the roughness penalty approach that computes $\hat{\beta}$ direction without using basis functions. Our first task is to show how we can set up this approach as a two-stage process involving: (1) minimizing a simple quadratic expression to obtain the vector of values $\hat{y}$ approximating the data vector y, and (2) computing the smoothest linear functional interpolant of these values.

15.7.1 Functional interpolation

We have already seen that the observed data can in general be fitted exactly by an infinite number of possible parameter choices (α, β). In some contexts, it may be of interest to define a functional interpolant $(\tilde{\alpha}, \tilde{\beta})$ to the given data by the smoothest parameter function choice that fits the data exactly. In any case, we need to consider this problem in defining the technique used to compute the estimate for β in Figure 15.5. Therefore, we require that estimate $(\tilde{\alpha}, \tilde{\beta})$ minimizes $\|D^2\beta\|^2$ subject to the N constraints

$$y_i = \tilde{\alpha} + \langle x_i, \tilde{\beta} \rangle. \tag{15.17}$$

The functional interpolant is the limiting case of the regularized estimator as $\lambda \to 0$. In fact, the curve $\tilde{\beta}$ resulting from interpolating the weather data is identical to that shown in Figure 15.1.

We can consider this minimization problem (15.17) as a way of quantifying the roughness or irregularity of the response vector y relative to the observed functional covariates x_i. More generally, if $z_1, \ldots, z_N$ is any sequence of values, then we can define the roughness of z relative to the functional covariates x_i as being the roughness of the smoothest function β_z such that

$$z_i = \alpha_z + \langle x_i, \beta_z \rangle$$

for all i, for some constant α_z. This method of defining the roughness of a variate z_i will be of considerable conceptual and practical use later.

15.7.2 The two-stage minimization process

Section 15.7.3 shows that we can define an order N matrix $\mathbf{R}$ in such a way that the roughness of a variate z can be expressed as the quadratic form

$$\int [D^2\beta(s)]^2 \, ds = \mathbf{b}'\mathbf{R}\mathbf{b}.$$

Assuming this to be true for the moment, we can conceptualize the smoothing problem as being solved by dividing the minimization of the penalized residual sum of squares into two stages:

Stage 1: Find predicted values $\hat{y}$ that minimize $\texttt{PENSSE}_\lambda(\hat{y}) = \sum_i (y_i - \hat{y}_i)^2 + \lambda \hat{\mathbf{y}}'\mathbf{R}\hat{\mathbf{y}}$, the solution to which is

$$\hat{\mathbf{y}} = (\mathbf{I} + \lambda\mathbf{R})^{-1}\mathbf{y}.$$

Stage 2: Find the smoothest linear functional interpolant (α, β) satisfying

$$\hat{y}_i = \alpha + \int x_i(s)\beta(s) \, ds. \tag{15.18}$$

This two-stage procedure does indeed minimize $\texttt{PENSSE}_\lambda(\alpha, \beta)$ by the following argument. Write the minimization problem as one of first minimizing $\texttt{PENSSE}_\lambda(\alpha, \beta)$ as a function of (α, β) but with $\hat{y}$ fixed, and then

minimizing the result with respect to $\hat{y}$. Formally, this is

$$\begin{aligned} \min_{\hat{y}} \ & [\min_{\alpha,\beta}\{\texttt{PENSSE}_\lambda(\alpha,\beta)\}\ ] \\ = \ & \min_{\hat{y}}\{\sum(y_i-\hat{y}_i)^2+\lambda\min_{\beta}\int[D^2\beta(s)]^2\,ds\}, \end{aligned} \tag{15.19}$$

where the inner minimizations over α and β are carried out keeping the values of the linear functionals $\hat{y}_i$ as defined in (15.18) fixed.

But according to our assumption, these inner minimizations yield (α,β) as the smoothest functional interpolant to the variate $\hat{y}$, so we may now write the equation as

$$\texttt{PENSSE}_\lambda(\alpha,\beta)=\min_{\hat{y}}\{\sum(y_i-\hat{y}_i)^2+\lambda\hat{\mathbf{y}}'\mathbf{R}\hat{\mathbf{y}}\}. \tag{15.20}$$

Setting aside the question of how $\mathbf{R}$ is defined for a moment, one of the advantages of the roughness penalty approach to regularization is that it allows this conceptual division to be made, in a sense uncoupling the two aspects of the smoothing procedure. However, it should not be forgotten that the roughness penalty is used in the construction of the matrix $\mathbf{R}$, and so the functional nature of the covariates x_i, and the use of $\int(D^2\beta)^2$ to measure the variability of the regression coefficient function β, are implicit in both stages set out above.

We can think of the two-stage procedure in two ways: First as a practical algorithm in its own right, and second as an aid to understanding and intuition. We also see in subsequent chapters that it has wider implications than those discussed here.

In order to use the algorithm in practice, it is necessary to derive the matrix $\mathbf{R}$, and we now show how to do this.

15.7.3 Functional interpolation revisited

In this section, we present an algorithmic solution to the linear functional interpolation problem presented in Stage 2 in the two-stage procedure set out in Section 15.7.2. That is, it is of interest to find the smoothest functional interpolant $(\tilde{\alpha},\tilde{\beta})$ to a specified N-vector $\hat{y}$ relative to the given covariates $z_i, i=1,\ldots,N$. For practical purposes, our algorithm is suitable for the case where the sample size N is moderate, where matrix manipulations of $N\times N$ matrices do not present an unacceptable computational burden.

Let matrix $\mathbf{Z}$ be defined in terms of the functional covariates z_i as described in Section 15.3. In terms of basis expansions, we wish to solve the problem

$$\min\{\boldsymbol{\zeta}'\mathbf{R}\boldsymbol{\zeta}\} \ \text{ subject to } \ \mathbf{Z}\boldsymbol{\zeta}=\hat{\mathbf{y}}. \tag{15.21}$$

We first define some more notation. By rotating the basis if necessary, assume that the first M_0 basis functions ϕ_ν span the space of all functions f that have roughness $\int (D^2 f)^2 = 0$. In the Fourier case, this is true without any rotation: The only periodic functions with zero roughness are constants, so $M_0 = 1$, and the basis ϕ_ν consists of just the constant function.

Let $\mathbf{K}_2$ be the matrix obtained by removing the first M_0 rows and columns of $\mathbf{K}$. Then $\mathbf{K}_2$ is strictly positive-definite, and the rows and columns removed are all zeroes. In the Fourier case, $\mathbf{K}_2$ is diagonal.

Corresponding to the above partitioning, let $\mathbf{Z}_1$ be the matrix of the first $M_0 + 1$ columns of $\mathbf{Z}$, and let $\mathbf{Z}_2$ be the remaining columns. Defining $\mathbf{P}$ to be the $N \times N$ projection matrix $\mathbf{P} = \mathbf{I} - \mathbf{Z}_1(\mathbf{Z}_1'\mathbf{Z}_1)^{-1}\mathbf{Z}_1'$ permits us to define $\mathbf{Z}^* = \mathbf{P}\mathbf{Z}_2$. In the periodic case, $\mathbf{Z}_1$ has columns $(1, \ldots, 1)$ and $(\bar{x}_1, \ldots, \bar{x}_N)$, where $\bar{x}_i = \int z_i(s)\, ds$ for each i. Thus $\mathbf{P}$ is the $N \times N$ matrix that projects any N-vector z to its residuals from its linear regression on $\bar{x}_i$.

Continuing with this partitioning process, let $\boldsymbol{\zeta}_1$ be the vector of the first $M_0 + 1$ components of $\boldsymbol{\zeta}$, and let $\boldsymbol{\zeta}_2$ be the remaining components of $\boldsymbol{\zeta}$. Then the constraint

$$\mathbf{Z}\boldsymbol{\zeta} = \mathbf{Z}_1\boldsymbol{\zeta}_1 + \mathbf{Z}_2\boldsymbol{\zeta}_2 = \hat{\mathbf{y}}$$

implies, by multiplying both sides by $\mathbf{Z}'$, that

$$\mathbf{Z}_1'\mathbf{Z}_1\boldsymbol{\zeta}_1 + \mathbf{Z}_1'\mathbf{Z}_2\boldsymbol{\zeta}_2 = \mathbf{Z}_1'\hat{\mathbf{y}}. \tag{15.22}$$

Solving for ζ_1 alone gives

$$\boldsymbol{\zeta}_1 = (\mathbf{Z}_1'\mathbf{Z}_1)^{-1}\mathbf{Z}_1'(\hat{\mathbf{y}} - \mathbf{Z}_2\boldsymbol{\zeta}_2) \text{ and } \mathbf{Z}_1\boldsymbol{\zeta}_1 = (\mathbf{I} - \mathbf{P})(\hat{\mathbf{y}} - \mathbf{Z}_2\boldsymbol{\zeta}_2). \tag{15.23}$$

In the periodic case, equation (15.23) indicates that $\boldsymbol{\zeta}_1$ is obtained by linear regression of the values $\hat{\mathbf{y}} - \mathbf{Z}_2\boldsymbol{\zeta}_2$ on the vector with components $\bar{x}_i$. Thus, once $\boldsymbol{\zeta}_2$ has been determined, we can find $\boldsymbol{\zeta}_1$.

Now substitute solution (15.23) for $\boldsymbol{\zeta}_1$ back into the constraint (15.22) and rearrange to show that we can find $\boldsymbol{\zeta}_2$ by solving the minimization problem

$$\min_{\boldsymbol{\zeta}_2}\{\boldsymbol{\zeta}_2'\mathbf{K}_2\boldsymbol{\zeta}_2\} \text{ subject to } \mathbf{Z}^*\boldsymbol{\zeta} = \mathbf{P}\hat{\mathbf{y}} \tag{15.24}$$

using the fact that $\boldsymbol{\zeta}'\mathbf{K}\boldsymbol{\zeta} = \boldsymbol{\zeta}_2'\mathbf{K}_2\boldsymbol{\zeta}_2$.

Let $\mathbf{R}$ be defined as the Moore-Penrose g-inverse

$$\mathbf{R} = (\mathbf{Z}^*\mathbf{K}_2^{-1}\mathbf{Z}^{*\prime})^+. \tag{15.25}$$

The solution of the minimization (15.24) is then given by

$$\boldsymbol{\zeta}_2 = \mathbf{K}_2^{-1}\mathbf{Z}^{*\prime}\mathbf{R}\hat{\mathbf{y}} \tag{15.26}$$

and the minimum value of the objective function $\boldsymbol{\zeta}'\mathbf{R}\boldsymbol{\zeta}$ is therefore

$$\boldsymbol{\zeta}'\mathbf{R}\boldsymbol{\zeta} \quad = \quad \boldsymbol{\zeta}_2'\mathbf{K}_2\boldsymbol{\zeta}_2$$

$$\begin{aligned} &= \hat{\mathbf{y}}'\mathbf{R}\mathbf{Z}^{*\prime}\mathbf{K}_2^{-1}\mathbf{K}_2\mathbf{K}_2^{-1}\mathbf{Z}^*\mathbf{R}\hat{\mathbf{y}} \\ &= \hat{\mathbf{y}}'\mathbf{R}\mathbf{R}^+\mathbf{R}\hat{\mathbf{y}} \\ &= \hat{\mathbf{y}}'\mathbf{R}\hat{\mathbf{y}}. \end{aligned} \tag{15.27}$$

This is the assumption we made above in defining the two-step procedure, and moreover we have now defined the matrix $\mathbf{R}$.

We can now sum up this discussion by setting out an algorithm for functional interpolation as follows:

Step 1: Calculate matrices $\mathbf{P} = \mathbf{I} - \mathbf{Z}_1(\mathbf{Z}_1'\mathbf{Z}_1)^{-1}\mathbf{Z}_1'$ and $\mathbf{Z}^* = \mathbf{P}\mathbf{Z}_2$. In effect, the columns of $\mathbf{Z}^*$ are the residuals from a standard regression of the corresponding columns of $\mathbf{Z}_2$ on the design matrix $\mathbf{Z}_1$.

Step 2: Compute $\mathbf{R}$ as defined in (15.25) above.

Step 3: Compute $\boldsymbol{\zeta}_2$ from (15.26) and use (15.23) to find $\boldsymbol{\zeta}_1$.

Of course, if all we require is the roughness of $\boldsymbol{\zeta}$, then we can find $\hat{\mathbf{y}}'\mathbf{R}\hat{\mathbf{y}}$ from (15.25) without actually calculating $\boldsymbol{\zeta}$.

Finally, returning now to our two-stage technique for smoothing, we can now carry out the first step by solving the equation

$$(\mathbf{I} + \lambda\mathbf{R})\hat{\mathbf{y}} = \mathbf{y}.$$

Note, by the way, that if $\mathbf{R}$ is either diagonal (as for the Fourier basis) or band-structured (as for the B-spline basis), that this solution is rapidly computable, and hence trying out various values for λ is quite feasible.

If we are dealing with a large data set by truncating or restricting the basis expansion to a reasonable dimensionality K as described in Section 15.3, then we only wish in general to assess the roughness of variates of the form $\mathbf{Z}\zeta$ for known $\boldsymbol{\zeta}$ with $\boldsymbol{\zeta}_j = 0$ for $j > m$. It is usually more appropriate to calculate $\boldsymbol{\zeta}'\mathbf{R}\boldsymbol{\zeta}$ for such variates directly if it is needed.

15.8 Functional regression and integral equations

Functional interpolation and regression can be viewed as a different formalization of a problem already considered in detail in Chapter 6, that of reconstructing a curve given certain indirect observations. Suppose that g is a curve of interest, and that we have noisy observations of a number of linear functionals $l_i(g)$. Such a problem was explored by Engle, Granger, Rice and Weiss (1986); see also Section 4.7 of Green and Silverman (1994). The problem involved in reconstructing the effect of temperature t on electricity consumption, so that $g(t)$ is the expected use of electricity per consumer on a day with average temperature t. Various covariates were also considered, but these need not concern us here.

Electricity bills are issued on various days and always cover the previous 28 days. For bills issued on day i, the average consumption (after correcting

for covariates) would be modelled to satisfy

$$\frac{1}{28}\mathrm{E}Y_i = \langle \theta_i, g \rangle,$$

where θ_i is the probability density function of temperature over the previous 28 day period. By setting $z_i = 28\theta_i$ and $\beta = g$, we see that this problem falls precisely into the functional regression context, and indeed the method used by the original authors to solve it corresponds precisely to the regularization method we have set out.

More generally, regularization is a very well-known tool for the solution of integral equations; see, for example, Section 12.3 of Delves and Mohamed (1985).

15.9 Further reading and notes

The subject of this chapter is probably the area in functional data analysis that has undergone the most development since the publication of the first version of this volume. The STAPH group that meets regularly at Paul Sabatier University in Toulouse has been especially active in terms of both applications and theory. To learn more about their work, consult the website `http://www.lsp.ups-tlse.fr/Fp/Ferraty/staph.html`.

Cardot, Faivre and Goulard (2003) predicted type of land use based on the evolution of the reflectance of a parcel of land in a specified wavelength over time as measured by satellite imagery. They also used functional principal components analysis to reduce the dimensionality of the reflectance curves prior to estimating the functional linear model, an approach first developed in Cardot, Ferraty and Sarda (1999) and discussed further in Cardot, Ferraty and Sarda (2003). Cardot, Goia and Sarda (2004) developed a test of the hypothesis that there is no effect on the outcome variable by the predictor variable, and Cardot, Ferraty, Mas and Sarda (2004) report further developments. Cardot, Faivre and Maisongrande (2004) use a mixed effects formulation of this model. Ferraty, Goia and Vieu (2002) forecast United States monthly electricity consumption, and Ferraty and Vieu (2002) predict the fat content of meat samples from spectrometric curves. Cardot (2002) used a roughness penalty that is similar to that used by Eilers and Marx (1996).

Escabias, Aguilera and Valderrama (2004), James (2002) and Cardot and Sarda (2004) look at the larger problem of how to adapt the generalized linear model to the presence of a functional predictor variable, and offer a number of examples, including the situation considered here of a continuous dependent variable. Escabias et al. (2004) combine the functional linear model with principal components analysis to reduce the dimensionality of the covariate space. James (2002) also describes an interesting method for estimating the between-curve variation as well as the

within-curve structure. Müller and Stadtmüller (2004) also investigate that they call the *generalized functional linear model*. James and Hastie (2001) consider linear discriminant analysis where at one of the of independent variables used for prediction is a function, and where the curves are irregularly sampled. Ratcliffe, Leader and Heller (2002) and Ratcliffe, Heller and Leader (2002) use the functional covariate foetal heart rate to model continuous and binary outcome variables.

16

Functional linear models for functional responses

16.1 Introduction: Predicting log precipitation from temperature

The aim of Chapter 15 was to predict a scalar response y from a functional covariate z. We now consider a fully functional linear model in which both the response y and the covariate z are functions. For instance, in the Canadian weather example, we might wish to investigate to what extent we can predict the complete log daily precipitation profile `LogPrec` of a weather station from information in its complete daily temperature profile `Temp`.

Because all the functions in this example are intrinsically periodic, we can expand both the log precipitations and the temperatures in Fourier series. We preprocessed the data by fitting a Fourier series with 65 terms, applying a roughness penalty smoother by tapering the series to eliminate very local variation.

We are now interested in the functional linear model

$$\texttt{LogPrec}_i(t) = \alpha(t) + \int_0^{365} \texttt{Temp}_i(s)\beta(s,t)\,ds + \epsilon_i(t)\ . \tag{16.1}$$

In contrast to the concurrent model discussed in Chapter 14, the regression function β is now a function of both s and t. We can interpret the regression function $\beta(s,t)$ for a fixed value of t as the relative weight placed on the temperature at day s that is required to predict log precipitation on day t.

Temperatures on day s quite far from day t may be important for predicting $\texttt{LogPrec}_i(t)$. For example, continental weather stations, where the winters are cold and the summers hot, have most of their precipitation in the summer months, in contrast to Atlantic stations, which tend to have a fairly even distribution of precipitation over the whole year. Also, the influence of temperature at a time $s > t$ is allowed here since we assume that the weather patterns are periodic, and therefore that the information "wraps around" so that we can predict rainfall in January by using temperature information in December.

The function α plays the part of the constant term in the standard regression setup, and allows for a different functional origin for the log precipitation curves than the origin for the temperature curves. In effect, the second term involving $\beta(s,t)$ indicates the advantage of using temperature information over what could be achieved by using mean log precipitation as a predictor, which is what $\alpha(t)$ would be without any covariate.

The unweighted fitting criterion is the integrated residual sum of squares that we already used in Chapter 14:

$$\texttt{LMSSE}(\alpha,\beta) = \int \sum_{i=1}^{N} [\texttt{LogPrec}_i(t) - \alpha(t) - \int \texttt{Temp}_i(s)\beta(s,t)\,ds]^2\,dt. \quad (16.2)$$

16.1.1 Fitting the model without regularization

We consider the expression of β as a double expansion in terms of K_1 basis functions η_k and K_1 basis functions θ_ℓ to give

$$\beta(s,t) = \sum_{k=1}^{K_1}\sum_{\ell=1}^{K_2} b_{k\ell}\,\eta_k(s)\theta_\ell(t) = \boldsymbol{\eta}(s)'\mathbf{B}\boldsymbol{\theta}(t), \quad (16.3)$$

where $\mathbf{B}$ is a $K_1 \times K_2$ matrix of coefficients $b_{\ell k}$, or, more compactly, as

$$\beta = \boldsymbol{\eta}'\mathbf{B}\boldsymbol{\theta}\ .$$

We will also use the basis vector $\boldsymbol{\theta}$ to expand the intercept function α as

$$\alpha(t) = \sum_{\ell=1}^{K_2} a_\ell\theta_\ell(t) = \boldsymbol{\theta}'(t)\mathbf{a}. \quad (16.4)$$

The unweighted fitting criterion (16.2) now becomes

$$\texttt{LMSSE}(\mathbf{a},\mathbf{B}) = \int \sum_{i=1}^{N} [\texttt{LogPrec}_i(t) - \boldsymbol{\theta}'(t)\mathbf{a} - \int \texttt{Temp}_i(s)\boldsymbol{\eta}(s)'\mathbf{B}\boldsymbol{\theta}(t)\,ds]^2\,dt. \quad (16.5)$$

We defer any further discussion of how to estimate the coefficient vector a for α and the coefficient matrix $\mathbf{B}$ for β to Section 16.4.

As our first attempt to fit the model, we used the same 65 basis functions used to expand the log precipitation and temperature functions for both

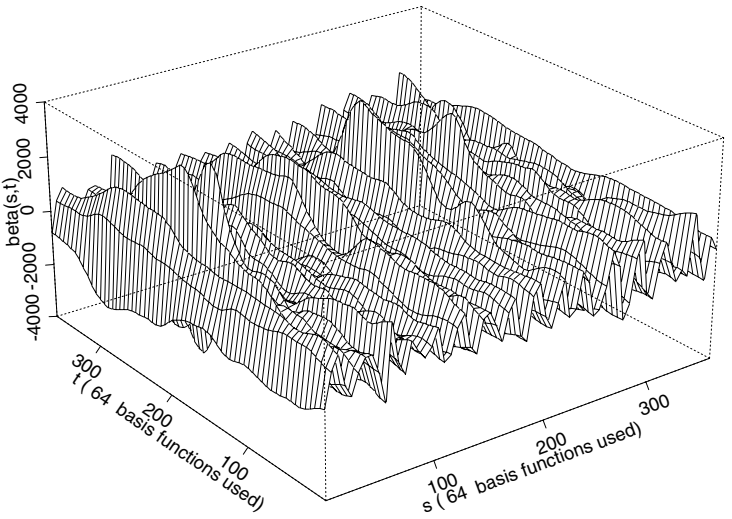

Figure 16.1. The functional parameter function β for the prediction of log precipitation from temperature, estimated direct from the data. The value $\beta(s,t)$ shows the influence of temperature at time s on log precipitation at time t.

basis systems $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$. We minimized the fit criterion (16.5) to obtain the estimated function β plotted in Figure 16.1.

We see that the function β estimated by this method is extremely variable. It also turns out that this β gives perfect prediction of the given data in `LogPrec`. This does not make physical sense; whatever influence temperature patterns may have on precipitation patterns, it is naive to imagine that the precipitation pattern at a place can be entirely accounted for by its temperature pattern.

The reason for this over-fitting is an extension of the discussion in Chapter 14 on the concurrent linear model. Consider any fixed t: as in Section 15.2, we can find a number α_t and a function β_t such that, for all i,

$$\texttt{LogPrec}_i(t) = \alpha_t + \langle \texttt{Temp}_i, \beta_t \rangle$$

without any error. Just as in Chapter 15, we must somehow regularize the functional predictor variable. Regularization by limiting the number of basis functions is discussed in the next section, and the use of roughness penalties is taken up in Section 16.4.

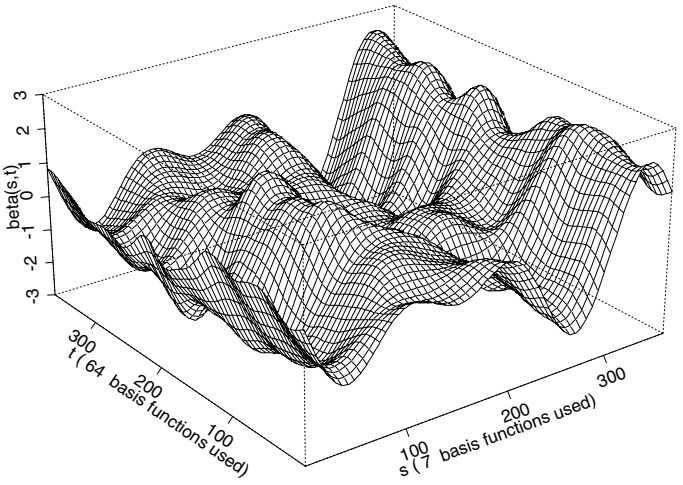

Figure 16.2. Perspective plot of estimated β function truncating the basis for the temperature covariates to 7 terms.

16.2 Regularizing the fit by restricting the bases

We now consider the effect of restricting each of the bases $\boldsymbol{\eta}$ and $\boldsymbol{\theta}$ in turn for expanding $\beta(s,t)$.

16.2.1 Restricting the basis $\boldsymbol{\eta}(s)$

We regularize β by setting the number of basis functions for its variation over argument s at $K_1 = 7$. Figure 16.2 shows the resulting estimated β function. The resulting prediction of the annual *pattern* of log precipitation at four selected stations is demonstrated in Figure 16.3. In this figure, both the original data and the predictions for log precipitation have their annual mean subtracted, to highlight the pattern of precipitation rather than its overall level. The precipitation pattern is quite well predicted except for Edmonton, which has a precipitation pattern different from other weather stations with similar temperature profiles.

Although the plot of the estimated β function demonstrates a more plausible influence of temperature pattern on precipitation pattern, it is not easy to interpret. As a function of t for any fixed s it is irregular, and this irregularity is easily explained. Because every Fourier coefficient of log precipitation is allowed to be predicted by the temperature covariate, the prediction contains frequency elements at all levels. By the arguments

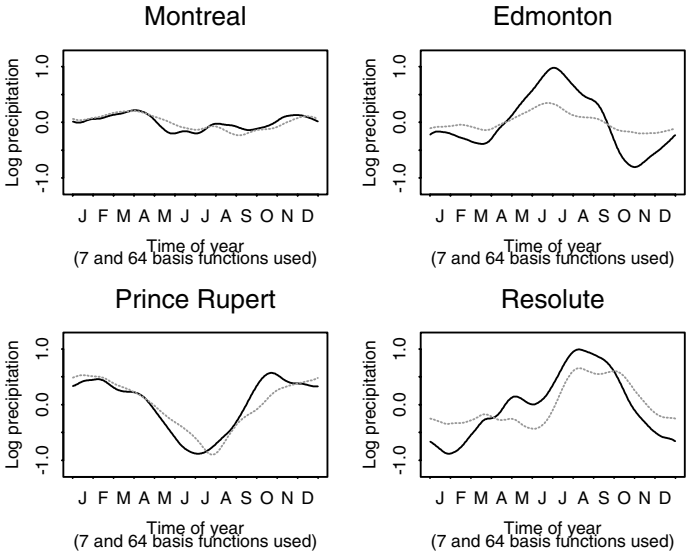

Figure 16.3. Original data (solid) and predictions (dashed) of log precipitation relative to annual mean for each of 4 weather stations. The prediction is carried out using an estimated β function with the temperature covariate truncated to 7 terms.

given in Chapter 15, we expect that each individual Fourier coefficient will be predicted sensibly as a scalar response. However, in putting these together to give a functional prediction, the high-frequency terms are given inappropriate weight. From a common-sense point of view, we cannot expect overall temperature patterns to affect a very high frequency aspect of log precipitation at all. To address this difficulty, we consider the idea of restricting or truncating the $\boldsymbol{\theta}$ basis in terms of which the functional response variable is expanded.

16.2.2 Restricting the basis $\boldsymbol{\theta}(t)$

In this section, we consider the approach of truncating the $\boldsymbol{\theta}$ basis, allowing the prediction of only low-frequency aspects of the response variable. In our example, this would correspond to the idea that the very fine detail of log precipitation could not be predicted from temperature. For the moment, suppose that we do not truncate the $\boldsymbol{\eta}$ basis, but that we allow only $K_0 = 7$ terms in the expansion of the y_i, with corresponding adjustments to the matrices $\mathbf{C}$ and $\mathbf{B}$. Figures 16.4 and 16.5 show the resulting β functions and sample predictions. The predictions are smooth, but otherwise very close to the original data. The function β is similar in overall character to

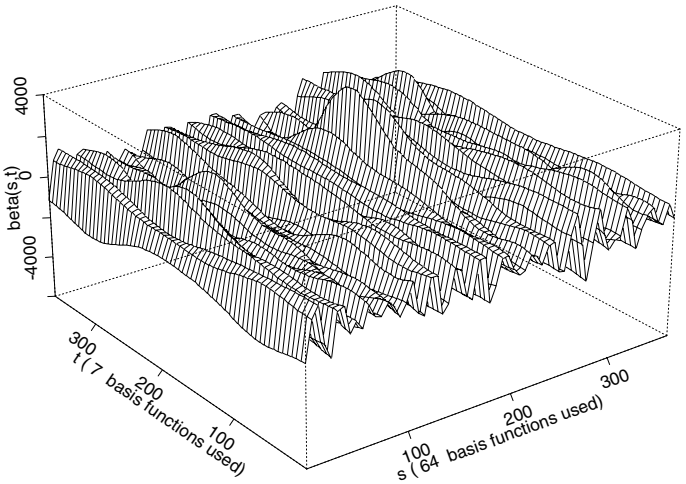

Figure 16.4. Perspective plot of estimated β function truncating the basis for the log precipitations to 7 terms.

the unsmoothed function shown in Figure 16.1, except that it is smoother as a function of t. However, it is excessively rough as a function of s. Thus, although the predictions are aesthetically attractive as smooth functions, they provide an optimistic assessment of the quality of the prediction, and an implausible mechanism by which the prediction takes place.

16.2.3 Restricting both bases

Sections 16.2.1 and 16.2.2 illustrated advantages in truncating both the $\boldsymbol{\eta}$ basis of the predictors and the $\boldsymbol{\theta}$ basis of the responses to obtain useful and sensible estimates. It should be stressed that the reason for doing this is not the same in both cases. Truncating the $\boldsymbol{\eta}$ basis for the covariates is essential to avoid over-fitting, while the $\boldsymbol{\theta}$ basis is truncated to ensure that the predictions are smooth.

Let us combine these different reasons for truncating the bases, and truncate both the predictor basis $\boldsymbol{\eta}$ and the response basis $\boldsymbol{\theta}$. Figures 16.6, 16.7 and 16.8 show the effects of truncating both bases to seven terms. We can discern several aspects of the effect of temperature on log precipitation. Temperature in February is negatively associated with precipitation throughout the year. Temperature around May is positively associated with precipitation in the summer months. Temperature in September has a strong negative association with precipitation in the autumn and

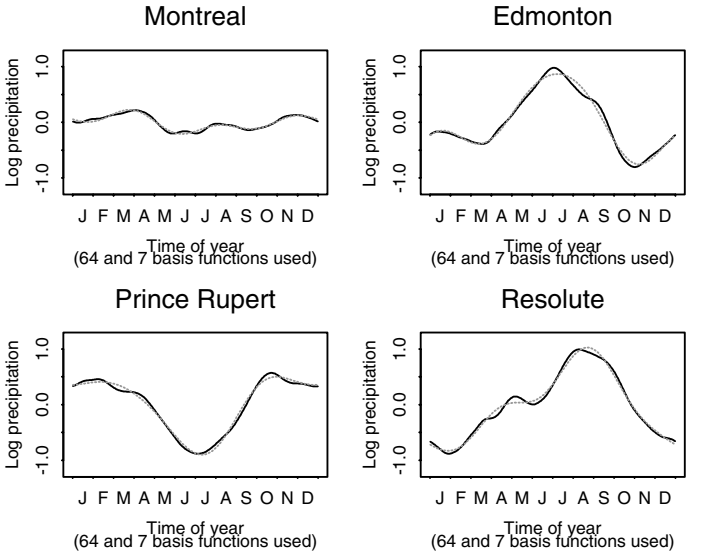

Figure 16.5. Original data (solid) and predictions (dashed) of log precipitation relative to annual mean for each of four weather stations. The prediction is carried out using an estimated β function with the basis for the log precipitations truncated to 7 terms.

winter, and finally, temperature in December associates positively with precipitation throughout the year, particularly with winter precipitation.

16.3 Assessing goodness of fit

There are various ways of assessing the fit of a functional linear model as estimated in Section 16.2. An approach borrowed from the conventional linear model is to consider the squared correlation function

$$R^2(t) = 1 - \sum_i \{\hat{y}_i(t) - y_i(t)\}^2 \Big/ \sum_i \{y_i(t) - \bar{y}(t)\}^2.$$

If we require a single numerical measure of fit, then the average of R^2 over t is useful, but using the entire function R^2 offers more detailed information about the fit. Figure 16.9 plots the R^2 function for the fit to the log precipitation data in Figure 16.6. The fit is generally reasonable, and is particularly good in the first five months of the year.

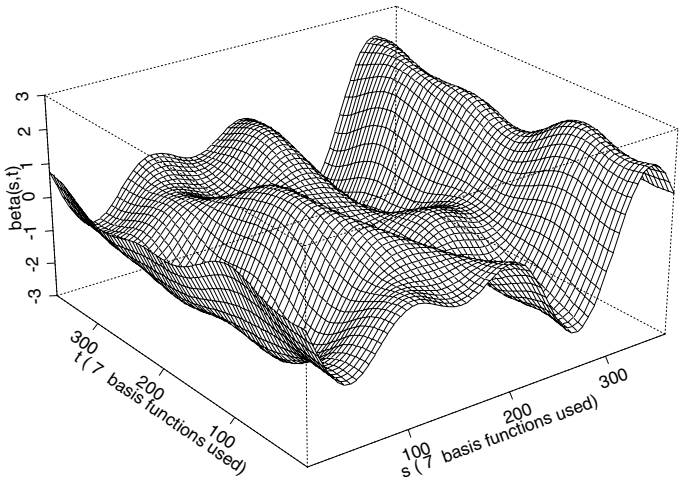

Figure 16.6. Perspective plot of estimated β function truncating both bases to seven terms.

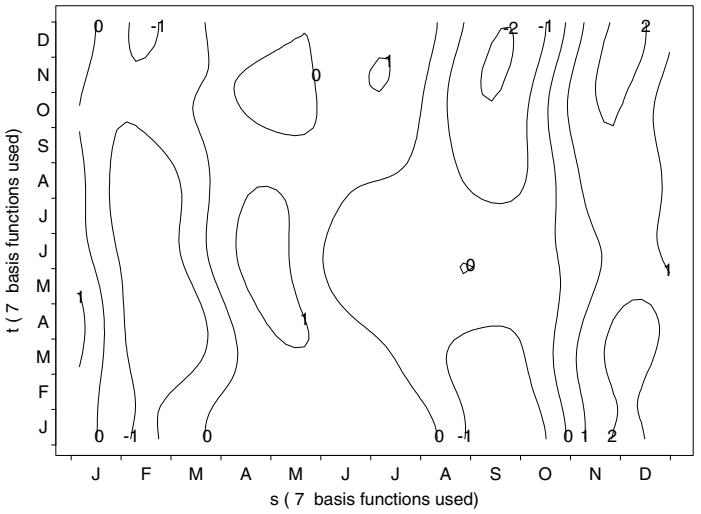

Figure 16.7. Contour plot of estimated β function truncating both bases to seven terms.

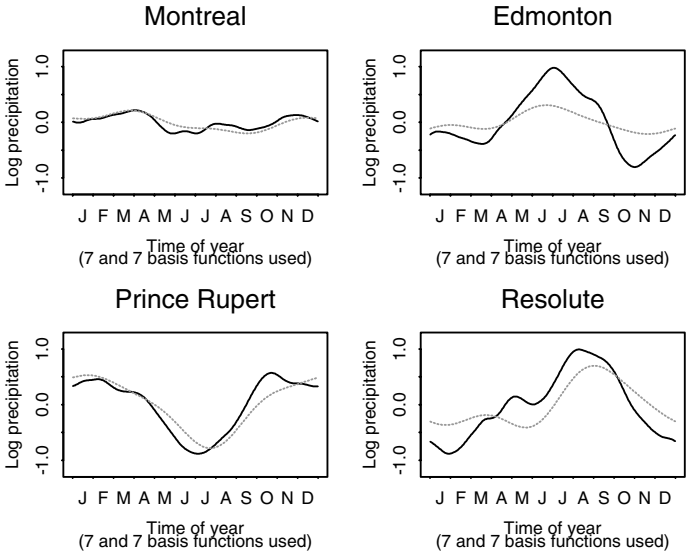

Figure 16.8. Original data (solid) and predictions (dashed) of log precipitation relative to annual mean for each of four weather stations. The prediction is carried out using an estimated β function with the both bases truncated to seven terms.

A complementary approach to goodness of fit is to consider an overall R^2 measure for each individual functional datum, defined by

$$R_i^2 = 1 - \int \{\hat{y}_i(t) - y_i(t)\}^2 \, dt \Big/ \int \{y_i(t) - \bar{y}(t)\}^2 \, dt.$$

For the four particular stations plotted in Figure 16.8, for instance, the values of R_i^2 are 0.96, 0.67, 0.63 and 0.81 respectively, illustrating that Montreal and Resolute are places whose precipitations fit closely to those predicted by the model on the basis of their observed temperature profiles; for Edmonton and Prince Rupert the fit is of course still quite good in that the temperature pattern accounts for over 60% of the variation of the log precipitation from the overall population mean. However, Figure 16.8 demonstrates that the pattern of precipitation, judged by comparing the predictions with the original data after subtracting the annual mean for the individual places, is predicted only moderately well for Resolute and is not well predicted for Edmonton. Figure 16.10 displays a histogram of all 35 R_i^2 values. At most of the stations, the R_i^2 value indicates reasonable or excellent prediction, but for a small proportion the precipitation pattern is not at all well predicted. Indeed, four stations (Dawson, Schefferville, Toronto and Prince George) have negative R_i^2 values, indicating that for

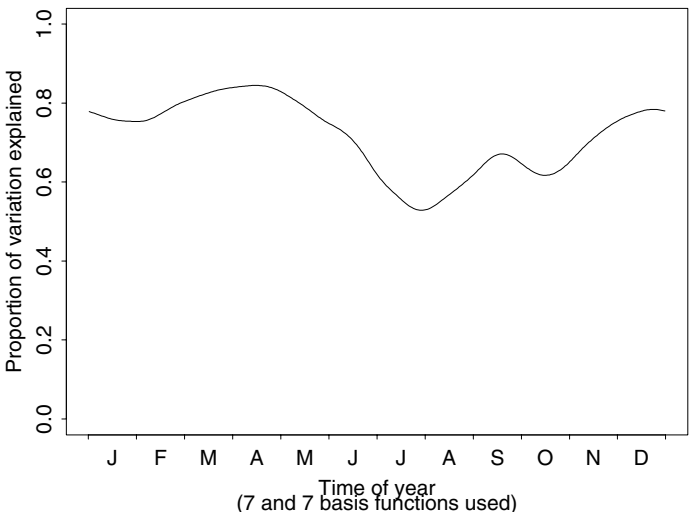

Figure 16.9. Proportion of variance of log precipitation explained by a linear model based on daily temperature records. The prediction is carried out using an estimated β function with both bases truncated to seven terms.

these places the population mean $\bar{y}$ actually gives a better fit to y_i than does the predictor $\hat{y}_i$.

To investigate this effect further, we use Figure 16.11 to show a plot of the residual mean square prediction error $\int(\hat{y}_i - y_i)^2$ against the mean square variation from the overall mean, $\int(y_i - \bar{y})^2$. The four places with negative values of R_i^2 are indicated by 0's on the plot. Each of the four places plotted in Figure 16.8 is indicated by the initial letter of its name. For most places the predictor has about one quarter the mean squared error of the overall population mean, and for many places the predictor is even better. The four places that yielded a negative value of R_i^2 did so because they were close (in three cases very close) to the overall population mean, not because the predictor did not work well for them. To judge accuracy of prediction for an individual place, it is clear that one needs to look a little further than just at the statistic R_i^2.

It is possible to conceive of an F-ratio function for the fit. We have

$$\hat{y}_i(t) - \bar{y}(t) = \sum_{j=1}^{J_0} C_{ij} \left(\sum_{k=1}^{K_0} B_{jk}\theta_k(t) \right) = \sum_{j=1}^{J_0} C_{ij}\theta_j(t).$$

By analogy with the standard linear model, we can ascribe $K_0 - 1$ degrees of freedom to the point-wise sum of squares $\sum_i \{\hat{y}_i(t) - \bar{y}(t)\}^2$ and $n - K_0$

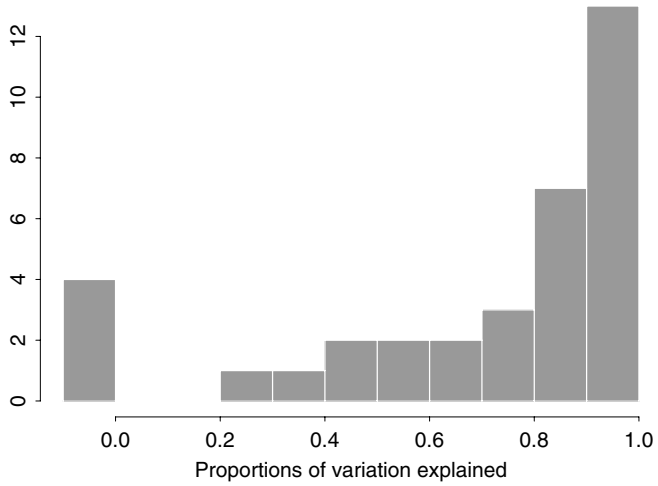

Figure 16.10. Histogram of individual proportions of variance R_i^2 in log precipitation explained by a linear model based on daily temperature records. The prediction is carried out using an estimated β function with both bases truncated to seven terms. The left-hand cell of the histogram includes all cases with negative R_i^2 values.

degrees of freedom to the residual sum of squares $\sum_i \{y_i(t) - \hat{y}_i(t)\}^2$. An F-ratio plot would be constructed by plotting

$$\texttt{FRATIO}(t) = \frac{\sum_i \{\hat{y}_i(t) - \bar{y}(t)\}^2/(K_0 - 1)}{\sum_i \{y_i(t) - \hat{y}_i(t)\}^2/(n - K_0)}.$$

However, the parameters $\theta_j(t)$ are not directly chosen to give the best fit of $\hat{y}_i(t)$ to the observed $y_i(t)$, and so the classical distribution theory of the F-ratio could be used only as an approximation to the distribution of $\texttt{FRATIO}(t)$ for each t.

Figure 16.12 plots the F-ratio for the fit to the log precipitation data. The upper 5% and 1% points of the $F_{6:28}$ distribution are given; within this model, this indicates that the effect of daily temperature on precipitation is highly significant overall.

We have not given much attention to the method by which the truncation parameters J_0 and K_0 could be chosen in practice. For many smoothing and regularization problems, the appropriate method of choice is probably subjective. The different roles of J_0 and K_0 lead to different ways of considering their automatic choice, if one is desired. The variable J_0 corresponds to a number of terms in a regression model, and so we could use a vari-

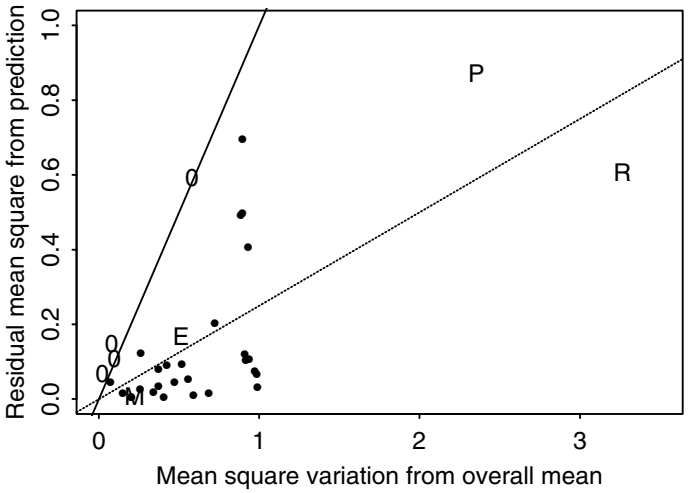

Figure 16.11. Comparison, for log precipitation, between mean square prediction errors and mean square variation from overall mean of log precipitation. The prediction is carried out using an estimated β function with both bases truncated to seven terms. The points for Montreal, Edmonton, Prince Rupert and Resolute are marked as M, E, P and R respectively. The points marked 0 yield negative R_i^2 values. The lines $y = x$ and $y = 0.25x$ are drawn on the plot as solid and dotted, respectively.

able selection technique from conventional regression, possibly adapted to give a functional rather than a numerical criterion, to indicate a possible value. On the other hand, K_0 is more akin to a smoothing parameter in a smoothing method, and so a method such as cross-validation might be a more appropriate choice. These questions are interesting topics for future investigation and research.

16.4 Computational details

Here we indicate how the fits discussed in Section 16.1 are computed. First, we have a look at the simpler case, used in that section, where the only regularization principle used is restricting the number of basis functions.

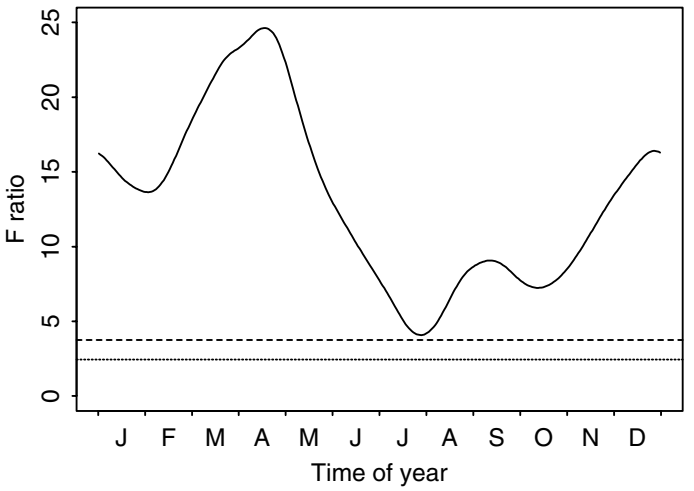

Figure 16.12. A plot of the F-ratio function for the prediction of log precipitation from daily temperature data. The prediction is carried out using an estimated β function with both bases truncated to seven terms. The horizontal lines show the upper 5% and 1% points of the $F_{6:28}$ distribution.

16.4.1 Fitting the model without regularization

Using more general matrix notation, our model is

$$\mathbf{y}^*(t) = \int \mathbf{z}^*(s)\beta(s,t)\,ds + \boldsymbol{\epsilon}(t)\ . \tag{16.6}$$

Recall that the bivariate regression function β has the expansion

$$\beta(s,t) = \boldsymbol{\theta}'(s)\mathbf{B}\boldsymbol{\eta}(t)\ ,$$

where basis system $\boldsymbol{\theta}$ has K_1 functions and $\boldsymbol{\eta}$ has K_2 functions. By substituting this expansion, the model becomes

$$\begin{aligned}\mathbf{y}^*(t) &= \int \mathbf{z}^*(s)\boldsymbol{\theta}'(s)\mathbf{B}\boldsymbol{\eta}(t)\,ds + \boldsymbol{\epsilon}(t)\\ &= \mathbf{Z}^*\mathbf{B}\boldsymbol{\eta}(t) + \epsilon(t),\end{aligned} \tag{16.7}$$

where the N by K_1 matrix $\mathbf{Z}^*$ is

$$\mathbf{Z}^* = \int \mathbf{z}^*(s)\boldsymbol{\theta}'(s)\,ds\ . \tag{16.8}$$

The second equation in (16.7), in effect, brings us back to the situation in Chapter 13, and in addition has the further simplification that the basis $\boldsymbol{\eta}$

is used for all the regression functions. Thus, we can use the computational details in that chapter to derive the following set of normal equations that must be solved for the regression function coefficient matrix $\mathbf{B}$:

$$\mathbf{Z}^{*'}\mathbf{Z}^{*}\mathbf{B}\int \boldsymbol{\eta}(t)\boldsymbol{\eta}'(t)\,dt = \mathbf{Z}^{*'}\int y(t)\boldsymbol{\eta}'(t)\,dt\ . \tag{16.9}$$

As before, we can re-express this equation in Kronecker product notation as

$$[\mathbf{J}_{\eta\eta}\otimes(\mathbf{Z}^{*'}\mathbf{Z}^{*})]\mathrm{vec}(\mathbf{B}) = \mathrm{vec}(\mathbf{Z}^{*'}\int \mathbf{y}(t)\boldsymbol{\eta}'(t)\,dt)\ , \tag{16.10}$$

where

$$\mathbf{J}_{\eta\eta} = \int \boldsymbol{\eta}(t)\boldsymbol{\eta}'(t)\,dt\ .$$

We note again that the numerical integration involved in the right side $\mathbf{Z}^{*'}\int \mathbf{y}\boldsymbol{\eta}'$ is, in effect, the set of inner products of the basis function vector $\boldsymbol{\eta}$ with the unit function $\mathbf{1}$ using the K_1 weighting functions $\mathbf{Z}^{*'}y$.

16.4.2 Fitting the model with regularization

Now let us consider the alternative strategy of endowing β with a generous number of basis functions for both $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$. We need to define two roughness penalties: one for β's variation with respect to s, and another for its variation with respect to t.

Consider the s situation first. Let linear differential operator L_s be an appropriate operator for "curvature" in the larger sense to be applied to β as a function of s only. Our penalty is

$$\begin{aligned}
\mathtt{PEN}_s(\beta) &= \int\int [L_s\beta(s,t)]^2\,ds\,dt \\
&= \int\int [L_s\boldsymbol{\theta}'(s)\mathbf{B}\boldsymbol{\eta}(t)][L_s\boldsymbol{\theta}'(s)\mathbf{B}\boldsymbol{\eta}(t)]'\,ds\,dt \\
&= \int\int [L_s\boldsymbol{\theta}'(s)]\mathbf{B}\boldsymbol{\eta}(t)\boldsymbol{\eta}'(t)\mathbf{B}'[L_s\boldsymbol{\theta}(s)]\,ds\,dt \\
&= \int \mathrm{trace}[\mathbf{B}\boldsymbol{\eta}(t)\boldsymbol{\eta}'(t)\mathbf{B}'\mathbf{R}]\,dt \\
&= \mathrm{trace}[\mathbf{B}'\mathbf{R}\mathbf{B}\mathbf{J}_{\eta\eta}]\ ,
\end{aligned} \tag{16.11}$$

where order K_1 symmetric matrix $\mathbf{R}$ is

$$\mathbf{R} = \int [L_s\boldsymbol{\theta}(s)][L_s\boldsymbol{\theta}'(s)]\,ds\ .$$

Penalization of β with respect to t requires an analogous linear differential operator L_t to be applied to β as a function of t only. Following through

the derivation above, we arrive at the following for the roughness penalty for t:

$$\begin{aligned} \texttt{PEN}_t(\beta) &= \int\int [L_t\beta(s,t)]^2\, ds\, dt \\ &= \text{trace}[\mathbf{B}'\mathbf{J}_{\theta\theta}\mathbf{S}\mathbf{B}] \ , \end{aligned} \tag{16.12}$$

where order K_2 symmetric matrix $\mathbf{S}$ is

$$\mathbf{S} = \int [L_t\boldsymbol{\eta}(t)][L_t\boldsymbol{\eta}'(t)]\, dt \ ,$$

where

$$\mathbf{J}_{\theta\theta} = \int \boldsymbol{\theta}(t)\boldsymbol{\theta}'(t)\, dt \ .$$

When we add these two penalties, each multiplied by their respective smoothing parameters, the equations for $\mathbf{B}$ become

$$\mathbf{Z}^{*'}\mathbf{Z}^*\mathbf{B}\mathbf{J}_{\eta\eta} + \lambda_s\mathbf{R}\mathbf{B}\mathbf{J}_{\eta\eta} + \lambda_t\mathbf{J}_{\theta\theta}\mathbf{B}\mathbf{S} = \mathbf{Z}^{*'}\int \mathbf{y}\boldsymbol{\eta}' \ , \tag{16.13}$$

with the Kronecker product equivalent

$$[\mathbf{J}_{\eta\eta}\otimes(\mathbf{Z}^{*'}\mathbf{Z}^*) + \lambda_s\mathbf{J}_{\eta\eta}\otimes\mathbf{R} + \lambda_t\mathbf{S}\otimes\mathbf{J}_{\theta\theta}]\text{vec}(\mathbf{B}) = \text{vec}(\mathbf{Z}^{*'}\int \mathbf{y}\boldsymbol{\eta}') \ . \tag{16.14}$$

Finally, in order to compute standard errors, we will need to specify that $\mathbf{y} = \mathbf{C}\boldsymbol{\phi}$ where $\boldsymbol{\phi}$ is a system of K_y basis functions and $\mathbf{C}$ is the associated N by K_y coefficient matrix. In this case, we can express the estimate for $\mathbf{B}$ as

$$\text{vec}(\hat{\mathbf{B}}) = [\mathbf{J}_{\eta\eta}\otimes(\mathbf{Z}^{*'}\mathbf{Z}^*) + \lambda_s\mathbf{J}_{\eta\eta}\otimes\mathbf{R} + \lambda_t\mathbf{S}\otimes\mathbf{J}_{\theta\theta}]^{-1}(\mathbf{J}_{\theta\eta}\otimes\mathbf{Z}^{*'})\text{vec}(\mathbf{C}) \ , \tag{16.15}$$

where

$$\mathbf{J}_{\theta\eta} = \int \boldsymbol{\theta}(t)\boldsymbol{\eta}'(t)\, dt \ .$$

16.5 The general case

The functional linear model (16.6) has been useful for exploring the implications of regularizing $\beta(s,t)$ with respect to each of its arguments. But it is, nevertheless, a model that is restricted in three important ways:

- The covariate function z that we used was a function of s alone. In fact, there is no reason why it might not also vary as a function of t, that is, take values $z(s,t)$. In fact, we assumed variation over t for the point-wise linear model in Chapter 14, and we may do so here, too.

- That we were able to integrate the influence of z over the entire year was made possible by the covariate being periodic, as we already observed. In many situations, and especially when both s and t index time, the case $s > t$ is inadmissible since causality does not operate backwards in time. In this case, we would have the model

$$y_i(t) = \int_0^t z_i(s)\beta(s,t)\,ds + \epsilon_i(t)$$

  and the matrix $\mathbf{Z}^*$ defined in (16.8) would now be a function of t rather than being constant.

- The expansion of β in (16.3), called a *tensor product* expansion, is highly specialized. In fact, it tends only be suitable when argument s can be integrated over its entire range, that is, including when the covariate is periodic.

Consequently, we need a formulation that can encompass not only the models that we have used up to now, but a range of others as well that may be important in other applications.

To get away from tensor-product expansions, let us now propose the general expansion

$$\beta(s,t) = \sum_k^{K_\beta} b_k\theta_k(s,t) \;\; = \boldsymbol{\theta}'(s,t)\mathbf{b}\;. \tag{16.16}$$

Moreover, let us assume that the covariate z takes values $z(s,t)$ varying over both arguments. Finally, let the interval of integration for argument s be allowed to vary over t, and we can use the notation Ω_t to indicate the interval associated with t. Our model now becomes

$$\begin{aligned} y_i(t) &= \int_{\Omega_t} z_i(s,t)\beta(s,t)\,ds + \epsilon_i(t) \\ &= \int_{\Omega_t} z_i(s,t)\boldsymbol{\theta}'(s,t)\mathbf{b}\,ds + \epsilon_i(t). \end{aligned} \tag{16.17}$$

Each of the previous models is contained within this one. For example, the multivariate covariate z_i in Chapter 13 does not vary with either argument, and no integration is involved, so that we can simply drop argument s from the model. The point-wise model in Chapter 14 is similar in that, since $\Omega_t = t$, argument s is again irrelevant in the sense that it can be folded into t itself. But in this case z_i does vary over t.

In this general case, we can still make an important simplification. As we did for the earlier specialized model, we can integrate out s by defining

$$z^*_{ik}(t) = \int_{\Omega_t} z_i(s,t)\theta'_k(s,t)\,ds \tag{16.18}$$

and then re-express this general model in a matrix version that no longer has an explicit role for s:

$$\mathbf{y}(t) = \mathbf{Z}^*(t)\mathbf{b} + \boldsymbol{\epsilon}(t), \tag{16.19}$$

where $\mathbf{Z}^*$ is an N by K_β matrix function. Of course, the removal of s is actually an illusion, since any evaluation of $\mathbf{Z}^*$ will automatically involve an integration over s.

Nevertheless, formally we now have a concurrent or point-wise functional linear model, but where the regression coefficient functions are all constant. In short, the most general case finally reduces to the point-wise linear model, but of course at the expense of replacing a single covariate $z_i(s,t)$ by a vector of K computed covariates $z^*_{ik}(t)$.

The estimated parameter vector $\hat{\mathbf{b}}$ satisfies

$$[\int \mathbf{Z}^{*'}\mathbf{Z}^* + \lambda_s\mathbf{R} + \lambda_t\mathbf{S}]\hat{\mathbf{b}} = \int \mathbf{Z}^{*'}\mathbf{y}, \tag{16.20}$$

where

$$\mathbf{R} = \int\int_{\Omega_t} [L_s\boldsymbol{\theta}(s,t)][L_s\boldsymbol{\theta}(s,t)]'\, ds\, dt$$

and

$$\mathbf{S} = \int\int_{\Omega_t} [L_t\boldsymbol{\theta}(s,t)][L_t\boldsymbol{\theta}(s,t)]'\, ds\, dt\ .$$

What kind of bivariate basis functions might we propose? The topic of smoothing data over higher numbers of dimensions has generated several examples. In *thin-plate spline smoothing*, for example, *radial basis functions* are used of the form

$$\theta_k(s,t) = \zeta_k(s^2 + t^2),$$

where the functions ζ_k are a *univariate* basis system.

An especially convenient and powerful class of bivariate basis functions are associated with *finite element* methods developed for numerical methods for solving partial differential equations over complex regions. These typically involve the approximate coverage of the two-dimensional domain $\Omega_t, t \in \mathcal{T}$ by a system of triangles, and the basis functions are piecewise linear "hat" functions having value one at a vertex shared by six triangles, and decreasing to zero on each of the distal edges. See Ramsay and Silverman (2002, Chapter 10) for an example.

16.6 Further reading and notes

Many important issues need further attention, but they would carry us beyond the objectives of this volume and would require technical resources rather out of line with what we are assuming.

There has been a considerable amount of work on reformulating time series methods for functional data, and especially in France and Spain. Bosq (2000) has a rather technical but broad overview. Besse, Cardot and Stephenson (2000) and Aguilera, Ocaña and Valderrama, M. J. (1999) developed autoregressive forecasting models for climatic variations.

From the perspective of classical frequentist statistics, just what is being estimated here and in Chapter 15 where the covariate structure is potentially of infinite dimension? As sample size increases, what conditions are required to assure the convergence of estimates $\hat{\boldsymbol{\beta}}$ to their population values? A good coverage of this issue along with some interesting consistency results can be found in Cuevas, Febrero and Fraiman (2002), as well as in earlier papers by Cardot, Ferraty and Sarda (1999) and Ferraty and Vieu (2001). These latter researchers are part of a group called STAPH that meets regularly to exchange findings on both foundational and application issues. Frank and Friedman (1993) also raised some of these issues in their survey on the use of regression in chemometrics.

If we adopt a Bayesian perspective, how can we propose prior distributions for the functional parameters like β that fulfill certain regularity conditions essential for coherent estimation? We have tended to adopt the perspective that the basis function expansion that we use is chosen essentially as a matter of both capturing certain known features of the problem and of computational convenience. This would imply that we wouldn't want to assume that the coefficients in the matrix $\mathbf{B}$ were the parameters, since we would freely admit that other investigators might use different basis expansions for perfectly good reasons.

17

Derivatives and functional linear models

17.1 Introduction

This chapter is an introduction to the idea of a *differential equation*, and aims to provide for readers unfamiliar with differential equations some of the basic ideas that will carry them forward into the next chapters. We begin with an example where we see the advantages of modelling the *rate of change* of a function as the dependent variable. Of course, by term "rate of change" we mean a derivative of a function, and in this case the first derivative. Models for derivatives are often termed models for the *dynamics* of a system, or *dynamic models.*

We will see how these dynamic models, expressed as differential equations, permit us to model both the function itself and one or more of its derivatives *at the same time.* How does this differ from what we have already been doing, say, with the growth data? There, by contrast with a truly dynamic model, we begin with a model for the observed data, the height measurements. To be sure, we selected this model with an eye to looking at derivatives, but fundamentally we modelled the data and then let the derivatives emerge as by-products. Now, however, and in the next chapters, we look at linking derivatives and function values together so as to take away the privileged place of the function as the object to be estimated.

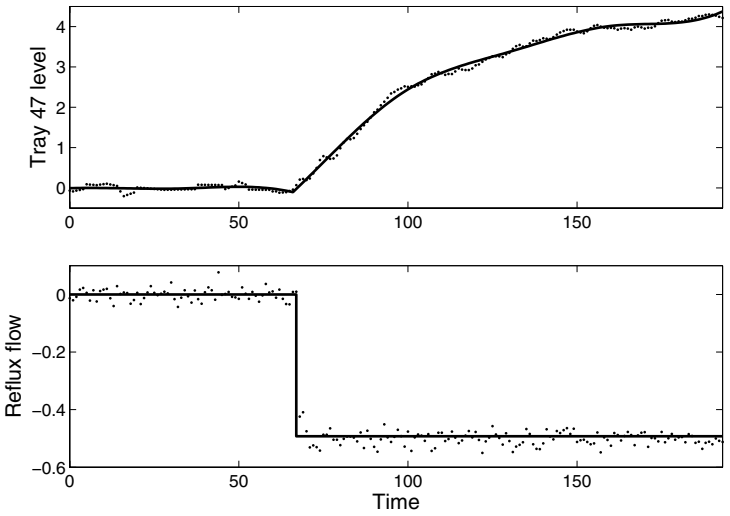

Figure 17.1. The upper panel shows the level of material in a tray of a distillation column in an oil refinery, and the lower level shows the flow of material being distilled into the tray. The points are measured values, and the solid lines are smooths of the data using regression splines. Time is in minutes.

17.2 The oil refinery data

A distillation column or cracking tower in an oil refinery converts crude oil to refined petroleum products like gasoline by boiling the crude and passing the vapor through a series of trays where, at each level, the condensate becomes more refined. Figure 17.1 shows the output from tray number 47 in the upper panel in response to the input shown in the lower panel. Both functions have been centered on their values at time 0, the time and flow units are unknown, and input flow has been measured in the downward direction.

We see that the output changes slowly in response to an abrupt change in input, although it is clearly headed toward some stable upper level between four and five units. It seems to have a fair amount of inertia, and the results are analogous to those of a person pushing a car on level ground. Otherwise there does not seem to be much to understand here; we increase the flow into a tank with an outlet, and the level rises.

The refinery data show variation on two time scales: The long-term scale involves the overall change in level from zero to near five that takes place over several hundred time units, and the the shorter time scale covers period from time 67 to where the new level is achieved, covering about one hundred units. We would like to find a way to model both the long-term change in

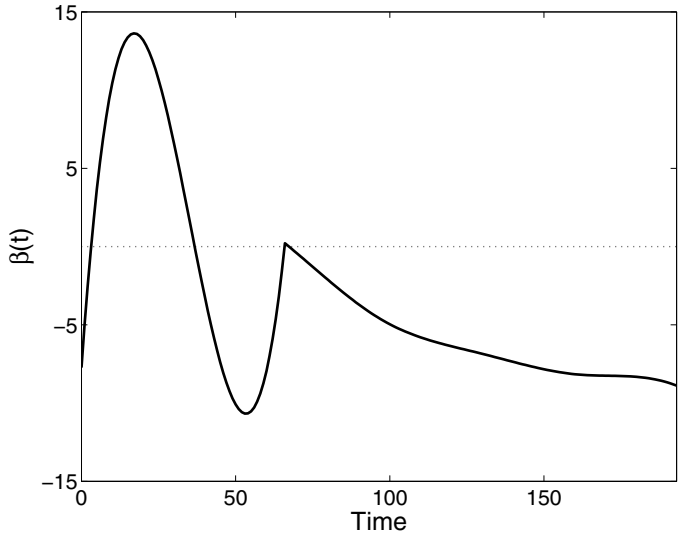

Figure 17.2. The regression coefficient function β for model (17.1) for the oil refinery data.

the output, and the rate of change over the shorter period that produces this change.

Here are a few technical asides on how we smoothed these data. Both functions show a sharp break at time number 67; the upper curve has a discontinuous derivative, and the lower curve is itself discontinuous. In order to have the smooth curve for the output to have a derivative discontinuity at 67, we used order four splines and placed three coincident knot values at that time. There was also one knot positioned midway between 0 and 67, and three equally spaced knots between 67 and 193. These knot choices imply a total of eleven basis functions. The lower curve was fit with order one splines with a single interior knot placed at 67.

Suppose that we model these data using the concurrent functional linear model described in Chapter 14, so that

$$\texttt{Tray}(t) = \texttt{Reflux}(t)\beta(t) + \epsilon(t). \tag{17.1}$$

We used nearly the same basis system for the single regression coefficient $\beta(t)$ except that we dropped the interior knot in the first interval, thus using ten splines. Figure 17.2 displays the estimated regression function. After time 67, β simply mirrors the behavior of the output, and we have little interest in its behavior before time 67, where it captures some of the data's wanderings around zero. The fit to the data, not shown, is virtually the same as that shown in Figure 17.1.

This seems disappointing. We haven't learned much from the shape of the regression function that we couldn't see in the original data. In fact, a

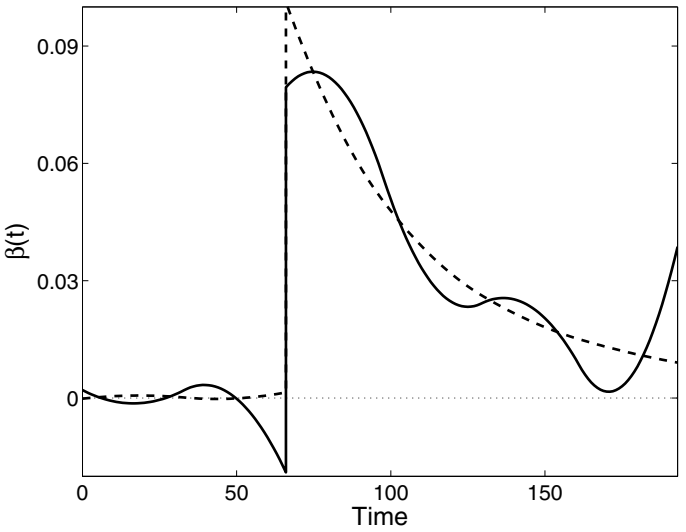

Figure 17.3. The first estimated derivative of Tray 47 level is shown as a solid line, and the fit to this derivative from model (17.2) is shown as a dashed line.

little thought convinces us that simply making β negatively proportional to `Tray` is going to work just fine.

Now let's take a different approach, involving explicit use of tray 47's first derivative as computed from the smooth in Figure 17.1, and shown in Figure 17.3. There is a fair amount of variability in this derivative estimate, but we do see something like exponential decay in the derivative after time 67, which seems consistent with what we see in Figure 17.1.

We propose to make this derivative the dependent variable, and to use two independent variables, namely `Tray` level itself and the input, Reflex flow. The model is therefore

$$\texttt{DTray}(t) = -\beta_1(t)\texttt{Tray}(t) + \beta_2(t)\texttt{Reflux}(t) + \epsilon(t). \qquad (17.2)$$

It is the usual practice in formulating a linear differential equation model to place a minus sign in front of coefficient functions such as $\beta_1(t)$.

The motivation here is to model the behavior of the rate of change of the output as a function of both the output level and the input. This time we will impose extreme simplicity on both the regression functions by using a constant basis for each. The results that we obtain are $\beta_1(t) = 0.02$ and $\beta_2(t) = -0.20$. Figure 17.3 shows the fit to the first derivative offered by this model, and we have captured nicely the idea of zero derivative up to time 67 and exponential decay afterwards.

Model (17.2) is an example of a first order linear differential equation with constant coefficients. This is to say that the equation links the first derivative to the function value and the input function, and that the linking

equation is linear with coefficients that are constant. In order to see how well the result fits the data, we need to solve equation (17.2) for `Tray`. Fortunately, any basic text on differential equations will give us the solution, which is, using $y(t)$ to stand for `Tray`(t) and $u(t)$ to stand for `Reflux`(t),

$$y(t) = e^{-\beta_1 t}[y(0) - (\beta_2/\beta_1)\int_0^t e^{\beta_1 s}u(s)\,ds]. \tag{17.3}$$

We can simplify this further by specifying that $y(0) = 0$, $u(t) = 0, t \leq 67$, and $u(t) = -0.4924, t > 67$ to get

$$y(t) = 0.4924\frac{\beta_2}{\beta_1}[1 - e^{-\beta_1(t-67)}],\ t \geq 67,\ \text{and 0 otherwise.} \tag{17.4}$$

The fit to the data offered by this equation is shown in Figure 17.4. The two parameter values define a model that fits the data beautifully, and predicts that the new level that `Tray` is approaching is 4.7.

Here's a summary of what we learn from the model by studying equations (17.3) and (17.4):

- When there is no input, `Tray` level will decay exponentially with a rate constant of -0.02 from whatever its level is at time 0.
- When `Reflux` increases by one unit, the level of Tray 47 will increase at an exponentially declining rate (rate constant again -0.02) to a new level $0.2/0.02 = 10$ units higher. This is the long-term change in the output.
- The time from increase in `Reflux` to the time `Tray` achieves its new level is about $4/0.02 = 200$ time units, and this is the shorter term period in which the actual change takes place.
- β_1 is the rate constant, and therefore controls the rate of change of `Tray` level. It models the dynamic behavior of `Tray`.
- β_2, along with β_1, controls the ultimate change; the long-term *gain* per unit increase in Reflux flow is β_2/β_1.

17.3 The melanoma data

Figure 17.5 presents age-adjusted melanoma incidences for 37 years from the Connecticut Tumor Registry (Houghton et al. 1980). The solid line is a smoothing spline fit by penalizing the size of the fourth derivative D^4x and choosing the penalty parameter by minimizing generalized cross-validation or GCV. Two types of trends are obvious: a steady linear increase and a periodic component. The latter is related to sunspot activity and the accompanying fluctuations in solar radiation. If we look closely, though, we can also see that there are some changes in the periodic trend; the peaks

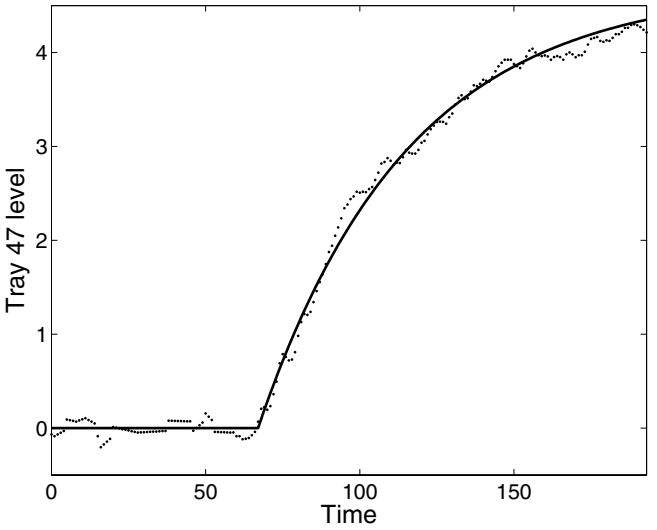

Figure 17.4. The fit to the data defined by model (17.2) is shown as a solid line, and the data as points.

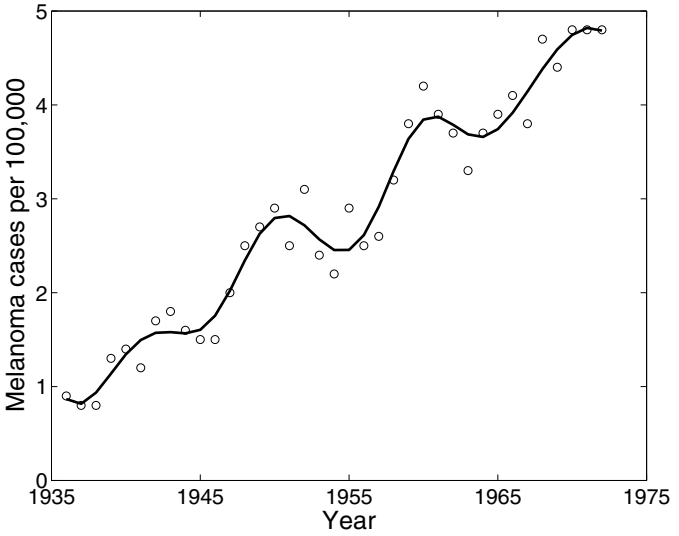

Figure 17.5. Age-adjusted incidences of melanoma for the years 1936 to 1972. The solid curve is the polynomial smoothing spline fit to the data penalizing the norm of the fourth derivative, with the smoothing parameter chosen by minimizing the `GCV` criterion.

around 1950 and 1960 seem stronger than those near 1940 and 1970, and perhaps the length of each cycle changes a little, too.

In short, there are three time scales here: the unlimited time over which linear trend is maintained, the short-term sunspot cycle of about ten years, and the medium term covering the range of years in the data in which the cycles themselves change.

We again want to find a simple model that will capture changes on these three time spans, and that will also tell us something about the dynamics of the cyclical variation. We already know that a straight line solves the differential equation $D^2x = 0$ and that $\sin(\omega t)$ and $\cos(\omega t)$ solve the equation $D^2x = -\omega^2 x$ for some period $2\pi/\omega$. We can put these two ideas together working with the fourth order equation $D^4x = -\omega^2 D^2 x$. Let's add one more parameter to define the differential equation

$$D^4x = -\beta_1 D^2x - \beta_2 D^3x, \tag{17.5}$$

where $\beta_1 = -\omega^2$ and β_2, called the *damping coefficient*, allows for an exponential decay in the oscillations by multiplying $\sin(\omega t)$ and $\cos(\omega t)$ by the factor $\exp(-\beta_2 t/2)$ where $t = \text{year} - 1935$.

Here's an algorithm for estimating the unknown coefficients β_1 and β_2:

1. Start by smoothing the data, as we have already done, using smoothing splines penalized by using D^4 with the smoothing parameter λ that minimizes GCV.
2. Compute the derivatives of the smooth up to order four.
3. Carry out a regression of the fourth derivative values, taken at each year, on the corresponding values for the second and third derivatives. The regression coefficients are estimates of β_1 and β_2.
4. Define the linear differential operator L as

   $$Lx = \beta_1 D^2x + \beta_2 D^3x + D^4x. \tag{17.6}$$

   Operator L is just a re-arrangement of differential equation (17.5); x satisfies the equation if and only if $Lx = 0$.
5. Now smooth the data using the roughness penalty defined by this linear differential operator, and again choose λ to minimize GCV. Hopefully, because this operator will annihilate more of the variation in the data than D^4 would, the smooth will be better and the estimates of the derivatives will also improve.
6. Check for convergence in the regression coefficients, or in the value of GCV. If convergence occurs, continue on to the last step; otherwise, return to step 2.
7. As we did for the refinery data, see how well the smooth fits the data, and also how well the data are fit by a solution to the differential equation (17.5).

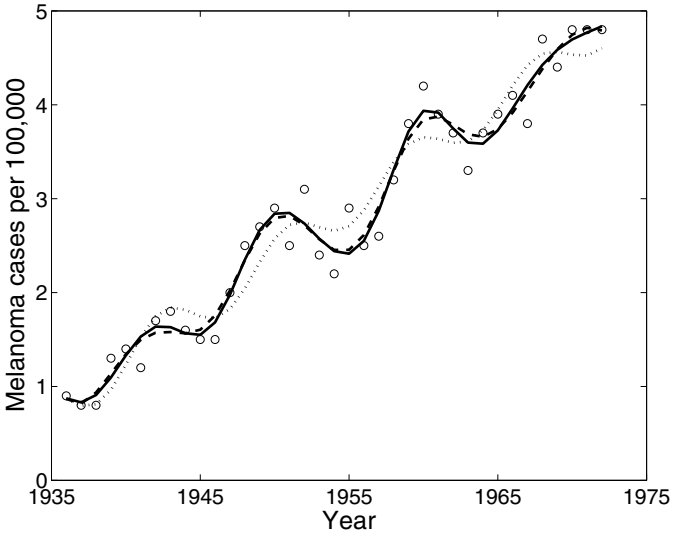

Figure 17.6. The light dashed line is the minimum–GCV fit to the data using a roughness penalty on D^4. The heavy solid line is the fit using a roughness penalty on Lx, where L is defined by (17.6). The dotted line is the best fit for functions satisfying differential equation (17.5).

This process effectively converged in five iterations, at which point β_1 and β_2 are 0.56 and 0.018, respectively. We can work out that $\omega^2 = 3\beta_2^2/4 + \beta_1$, and this corresponds to a period of 8.39 years. The period was estimated in the first iteration as 11.22 years.

Now we're in a position to compare the various fits to the data:

- The same fit as in Figure 17.5 using the D^4 operator.
- The smoothing fit using the converged value of operator L.
- The fit Lx satisfying $Lx = 0$, that is, satisfying the differential equation (17.5).

Each of these fits are shown in Figure 17.6. The final smooth tracks the data a bit better, especially between 1960 and 1965. But now we have a good estimate of the trend that can be fit with an exponentially decaying sinusoid plus linear trend, and we see that there are indeed phase differences between the smooth and the strictly periodic fit. Actually, the exponential decay is small, and scarcely visible in the plot.

The changes in the cycles resulting from iteratively updating the smoothing function and its derivatives are more visible in the phase-plane plot in Figure 17.7. In the right panel, showing the results for the estimated roughness penalty, the amplitudes of the cycles are stronger and the behavior of

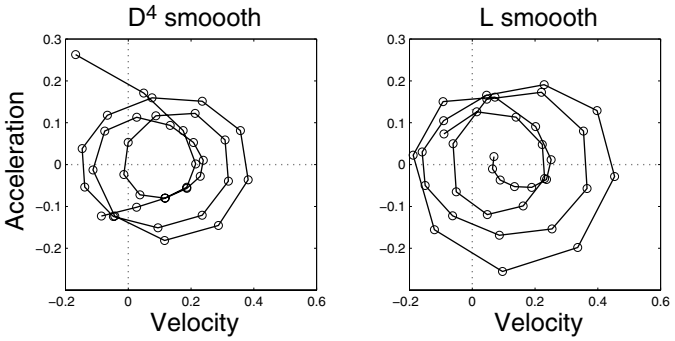

Figure 17.7. Two phase–plane plots for the fit to the melanoma data. The left panel is for the initial roughness penalty defined by the differential operator D^4, and the right panel is for the estimated operator L defined in (17.6).

the fit at the beginning and end of the curve is consistent with its behavior elsewhere.

17.4 Some comparisons of the refinery and melanoma analyses

Why was the differential equation (17.2) for the refinery data of order one and (17.5) for the melanoma model of order four? The reason is that we could express the shape of `Tray` in terms of only a single function, whereas we required four component or basis functions to express the essential structure of the melanoma data. Of course, the melanoma model required only two constants to be estimated, but that was because we could assume that the multipliers of x and Dx were zero. They are there, after all, but are just not estimated from the data.

On the other hand, the refinery data involved both an input and an output. Hence, we needed a parameter to model the impact of a change in the input, as well as a parameter to model the internal dynamics of the output. In the melanoma data, there was no input (although we could well have used sunspot activity records as an input), but the internal dynamics were, in effect, four dimensional.

In both problems we used two levels of fitting. The low-dimensional fit was defined by the solution of a differential equation, and the higher-dimensional fit was achieved by keeping smoothing parameter λ low enough that the roughness penalty did not overwhelm the data fit. This means that we *partitioned the functional variance* into two parts: the low-dimensional part captured by the differential equation, and the balance which is the difference between the low- and high-dimensional fits. The differential equations played key roles in this process.

For both models we estimated some parameters defining the differential equation from the data. In effect, the process that we used for the refinery data was simply a one-step version of the more sophisticated algorithm that we used for the melanoma data.

Perhaps this is the most important conclusion to take away from this chapter: We can use noisy data to estimate a differential equation that expresses at least a substantial part of the variation in the data. This problem is taken up in Chapter 19. First, though, you may want to read the next chapter, which offers a review of a number of results about linear differential equations and linear differential operators.

18
Differential equations and operators

18.1 Introduction

The derivatives of functional observations have played a strong role from the beginning of this book. For example, we chose to work with acceleration directly rather than height for growth curves and handwriting coordinate functions, and to inspect functions $(\pi/6)^2 D\,\texttt{Temp} + D^3\texttt{Temp}$ for temperature profiles. We used $D^2\beta$ as a measure of curvature in an estimated regression function β so as to regularize or smooth the estimate, and applied this idea in functional principal components analysis, canonical correlation, and various types of linear models. When the objective was a smooth estimate of a derivative $D^m x$, we used $D^{m+2}x$ to define the roughness penalty. Thus, derivatives can be used both as the object of inquiry and as tools for stabilizing solutions.

In Chapter 17, we introduced the idea of incorporating derivatives into linear models for functional data. We saw that this permitted a model for the simultaneous variation in a function and one or more of its derivatives, and in the oil refinery example in Section 17.2, the approach came up with an elegant little model with only two parameters that fit the data beautifully.

It is time to look more systematically at how derivatives might be employed in modelling functional data. Are there other ways of using derivatives, for example? Can we use mixtures of derivatives instead of simple derivatives? Can we extend models so that derivatives can be used on either the covariate or response side? Can our smoothing and regulariza-

tion techniques be extended in useful ways? Are new methods of analysis making explicit use of derivative information possible?

This chapter provides some background on differential equations and their use in applications. Readers either considering differential equations for the first time or whose memories of their first contact has dimmed may appreciate this material. We begin with the simplest of input/output systems commonly described by a differential equation. After considering possible extensions, we review how linear differential operators may be used in various ways and some basic theory. The last three sections, on constraint functionals, Green's functions and reproducing kernels, are more advanced. They may therefore be profitable to those already having a working knowledge of this field. We nevertheless consider these topics to be of potential importance for statistical applications, and they play a role in subsequent chapters.

18.2 Exploring a simple linear differential equation

An input/output system has an input function u that in some way modifies an output function x. Perhaps you might like to return to the refinery data in Figure 1.4 for an example.

Here is the simplest prototype for such equations:

$$Dx(t) = -\beta x(t) + \alpha u(t) + \epsilon(t). \tag{18.1}$$

This is a functional linear model in which the dependent variable is the derivative of output x, and the two independent variables are x itself and input function u. To keep things as simple as possible, we have specified that the regression coefficient functions are constant. Function ϵ allows for noise and other forms of ignorable variation in the functional data. It is a useful convention to place a minus before terms on the right side involving output function x; most real-life systems modelled by differential equations have positive values of β if we do this, reflecting their natural tendency to return to their resting state.

We could, however, make things even simpler by dropping u from the equation. Situations do arise where the goal is to model the behavior of a function x and its derivatives without considering any external influences. The no-input version of the equation,

$$Dx(t) = -\beta x(t) + \epsilon(t), \tag{18.2}$$

is said to be *homogeneous*, while (18.1) is called *nonhomogeneous.* Input function u, when it is present, is often called a *forcing function*, and the homogeneous version of the equation is said to be *forced* by αu.

Let x_0 be a solution to the homogeneous equation. Given parameter β and assuming that the noise function ϵ is zero, a moment of reflection

reveals that the solution to $Dx_0 = -\beta x_0$, is

$$x_0(t) = Ce^{-\beta t}$$

for some nonzero constant C. If we knew the value of x_0 at time $t = 0$, then $C = x_0(0)$ and the solution is completely determined.

It is a bit harder to work out the solution of (18.1), or $Dx = -\beta x + \alpha u$, but here it is:

$$x(t) = Ce^{-\beta t} + \alpha \int_0^t e^{-\beta(t-s)} u(s)\, ds \ . \tag{18.3}$$

As with the homogeneous equation, constant C is simply $x(0)$.

A graph helps us to see the role played by the two parameters α and β. Engineers often study how an industrial process reacts to changes in its inputs by stepping these inputs up or down abruptly. Accordingly, let $u(t) = 0$ for $0 \le t \le 1$, and $u(t) = 1$ for $t > 1$. Also, let's set $x(0) = C = 1$. Then solution (18.3) becomes

$$\begin{aligned} x(t) &= e^{-\beta t}, 0 \le t \le 1, \\ &= e^{-\beta t} + (\alpha/\beta)[1 - e^{-\beta(t-1)}], t > 1. \end{aligned}$$

Figure 18.1 shows the solution x for $\beta = 2$, and $\beta = 4$, while fixing $\alpha/\beta = 2$. Over the first half of the interval, x behaves like x_0, and we see that the solution decays to zero in about $4/\beta$ time units. Over the second half of the interval, the solution grows at an exponentially decreasing rate towards an upper asymptote of α/β, often called the *gain* of the system. Again, the gain level is achieved in about $4/\beta$ time units. The role of β is now clear; it determines the rate of change in x in response to a step change in u.

We can summarize the roles of these two parameters by comparing α to the volume control on a radio playing a song carried by radio signal u; the bigger α, the louder the sound. The bass/treble control, on the other hand, corresponds to β; the larger β, the higher the frequency of what you hear.

We may rearrange differential equation (18.1) to put it in the form

$$Lx(t) = \beta x(t) + Dx(t) - \alpha u(t) - \epsilon(t) \ . \tag{18.4}$$

Function x is a solution of the original equation when $\epsilon = 0$ if and only if $Lx = 0$. We call $L = \beta I + D$, where I is the identity operator, or $Ix = x$, a *linear differential operator*, in this case with constant coefficients. This alternative expression of the differential equation is handy, as we now know, for defining roughness penalties, and using the roughness penalty

$$\texttt{PEN}(x) = \int [Lx(t)]^2\, dt$$

is equivalent to penalizing the failure of x to satisfy the differential equation $Dx = -\beta x$ corresponding to operator L.

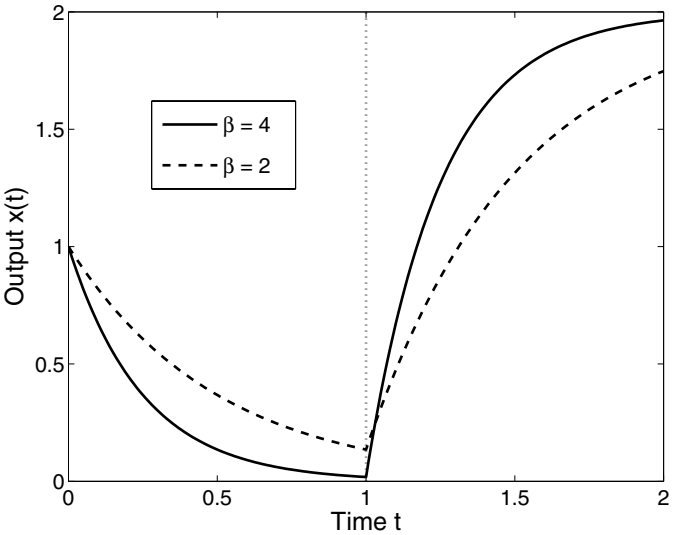

Figure 18.1. The solid and dashed lines are two solutions (18.3) to a first-order constant-coefficient differential equation for two different values of the rate constant β.

18.3 Beyond the constant coefficient first-order linear equation

18.3.1 Nonconstant coefficients

Returning to Figure 18.1, we might be struck by an anti-symmetry: The rate of decay over the first interval, $Dx = -\beta e^{-\beta t}$ is the negative of the rate of increase over the second, $Dx = \beta e^{-\beta t}$. Many systems, however, increase more rapidly than they decrease, or vice versa. We acquire common cold symptoms within hours and take days to recover from them, for example. This suggests that allowing β to vary over time might be useful, and similar arguments could be made for α. Then (18.1) becomes

$$Dx(t) = -\beta(t)x(t) + \alpha(t)u(t) + \epsilon(t). \tag{18.5}$$

The solution to (18.5) is

$$x(t) = Cx_0(t) + \int_0^t \alpha(s)u(s)x_0(t)/x_0(s)\,ds, \tag{18.6}$$

where

$$x_0(t) = \exp[-\int_0^t \beta(s)\,ds].$$

The functional ratio $x_0(t)/x_0(s)$ that occurs in the second term of (18.6) defines the Green's function for the differential equation; see Section 20.2 for details.

18.3.2 Higher order equations

More generally, the order of the derivative on the left side of (18.5) may be m, with lower order derivatives appearing in the right side:

$$\begin{aligned} D^m x(t) &= -\beta_0(t)x(t) - \beta_1(t)Dx(t) - \ldots - \beta_{m-1}(t)D^{m-1}x(t) \\ &\quad +\alpha(t)u(t) + \epsilon(t) \\ &= -\sum_{j=0}^{m-1} \beta_j(t)D^j x(t) + \alpha(t)u(t) + \epsilon(t). \end{aligned} \tag{18.7}$$

These higher order systems are needed when there are more than two time scales for events. This means that, in the case of a second order system, there is a long-term trend, medium-term changes, and sharper shorter-term events.

Figure 18.2 shows the forced second order equation

$$D^2x(t) = -4.04x(t) - 0.4Dx(t) + 2u(t), \tag{18.8}$$

where forcing function $u(t)$ steps from 0 to 1 at time $t = 2\pi$. The corresponding homogeneous solution is

$$x_0(t) = e^{-0.2t}[\sin(2t) + \cos(2t)].$$

There are three time scales involved here. The longest scale is the overall oscillation level, first about 0 and then later about 0.5. The medium scale trend is the exponential decay in the amplitude of the oscillation, and of course the shortest scale is the oscillation with period π.

Consider handwriting; Ramsay (2000) observed that there were features in script at four time scales:

1. The overall spatial position of the script, that is, the line on which it is written, requiring some considerable seconds per line.
2. The movement of the script from left to right within a line, taking place over several seconds.
3. The strokes and loops within the script, produced about eight times a second.
4. Sharper transient effects due to the pen striking or leaving the paper, lasting of the order of 10 milliseconds.

The differential equation developed in this study consequently was of the third order.

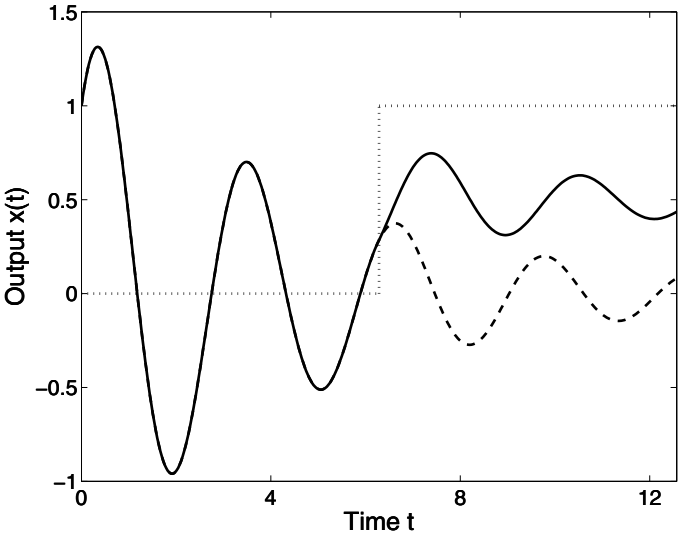

Figure 18.2. The solid line is the solution to the second order equation (18.8). The dashed line is the corresponding homogeneous equation solution, and the dotted line indicates the step function u forcing the equation.

18.3.3 Systems of equations

Often the processes that we study produce more than one output, and so we need several output functions x_i. As an example, suppose that $\beta(t)$ in (18.7) is also affected by $u(t)$, and that we can develop a differential equation that defines its behavior. We now have two differential equations, one for x and one for β.

Or, as another example, suppose that an engineer develops a feedback loop for the process permitting the output x to have an effect on the input u. For example, a doctor adjusts the medication u of the patient according to changes in the symptoms x. Then u and x can each be expressed as a differential equation, and in each equation the other variable now plays the role of an input. That is,

$$\begin{aligned} Dx(t) &= -\beta_x(t)x(t) + \alpha_x(t)u(t) \\ Du(t) &= -\beta_u(t)u(t) + \alpha_u(t)x(t). \end{aligned} \tag{18.9}$$

In fact, any differential equation of order m can be expressed as a system of m first-order equations. For a second order system,

$$D^2x(t) = -\beta_0(t)x(t) - \beta_1(t)Dx(t), \tag{18.10}$$

for example, define $y(t) = Dx(t)$. Then we have the equivalent system of two linear differential equations

$$\begin{aligned} Dx(t) &= y(t) \\ Dy(t) &= -\beta_1(t)y(t) - \beta_0(t)x(t). \end{aligned} \tag{18.11}$$

18.3.4 Beyond linearity

Equation (18.7) is a linear differential equation in the sense that each derivative or input function is multiplied by a coefficient function, and the products added to yield the output. That is, it is linear in the same sense that the models in Chapters 12 to 16 are linear.

The general form of a nonlinear differential equation of the first order is

$$Dx(t) = f[t, x(t), u(t)]$$

for some function f.

Linear differential equations are easier to work with. They have solutions for all values of t, and their properties are much better understood by mathematicians than nonlinear equations. However, simple nonlinear systems can define remarkable and often complex behavior in a solution x. The world of *nonlinear dynamics* is vast and fascinating, but unfortunately beyond the scope of this book.

The term "linear" is often used in engineering and elsewhere to refer only to linear constant coefficient systems. In this restricted case, the use of Laplace transformations leads to expressing the behavior of solutions in terms of *transfer functions.*

18.4 Some applications of linear differential equations and operators

In this section, we review a number of ways in which linear differential equations and operators are useful in functional data analysis. Many of these we have already encountered, but a few new ones are also suggested. We will assume that the linear differential operator is in the form

$$Lx = \sum_{j=0}^{m-1} \beta_j D^j x + D^m x. \tag{18.12}$$

18.4.1 Differential operators to produce new functional observations

Derivatives of various orders and mixtures of them are of immediate interest in many applications. We have already noted that there is much

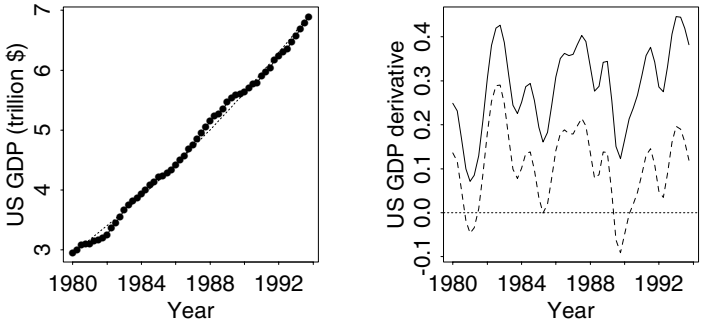

Figure 18.3. The left panel shows the gross domestic product of the United States in trillion US dollars. The solid curve mostly obscured by the dots is a polynomial smoothing spline constructed with a penalty on the integrated squared fourth derivative, and the dotted curve is a purely exponential trend fit by least squares. The solid curve in the right panel is the estimated first derivative of GDP. The dashed curve in this panel is the value of the differential operator $L = \beta\,\texttt{GDP} + D\,\texttt{GDP}$.

to be learned about human growth by examining acceleration profiles. There is an analogy with mechanical systems; a version of Newton's third law, $a(t) = F(t)/M$, asserts that the application of some force $F(t)$ at time t on an object with mass M has an immediate impact on acceleration $a(t)$. However, force has only an indirect impact on velocity, through $v(t) = v_0 + M^{-1}\int_0^t F(u)du$, and an even less direct impact on what we directly observe, namely position, $s(t) = s_0 + v_0 t + M^{-1}\int_0^t\int_0^u F(z)\,dz\,du$. From the standpoint of mechanics, the world that we experience is two integrals away from reality! The release of adrenal hormones during puberty tends to play the role of the force function F, and so does a muscle contraction with respect to position of a limb or other part of the body.

18.4.2 The gross domestic product data

The gross domestic product (GDP) of a country is the financial value of all goods and services produced in that country, whether by the private sector of the economy or by government. Like most economic measures, GDP tends to exhibit a percentage change each year in times of domestic and international stability. Although this change can fluctuate considerably from year to year, over long periods the fluctuations tend to even out for most countries and the long-range trend in GDP tends to be roughly exponential.

We obtained quarterly GDP values for 15 countries in the Organization for Economic Cooperation and Development (OECD) for the years 1980

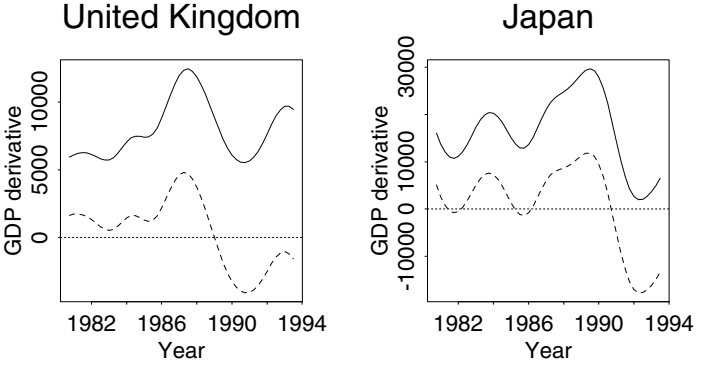

Figure 18.4. The solid curves are the derivatives of GDP of the United Kingdom and Japan estimated by order 4 smoothing splines. The dashed curves are the corresponding values of the differential operator $L = \beta x + Dx$.

through 1994 (OECD, 1995). The values for any country are expressed in its own currency, and thus scales are not comparable across countries. Also, there are strong seasonal effects in GDP values reported by some countries, whereas others smooth them out before reporting.

The left panel of Figure 18.3 displays the GDP of the United States. The seasonal trend, if any, is hardly visible, and the solid line indicates a smooth of the data using a penalty on D^4 `GDP`. It also shows a best fitting exponential trend, $C\exp(\gamma t)$, with rate constant $\gamma = 0.038$. Thus, over this period the U.S. economy tended to grow at about 4% per year. The right panel displays the first derivative of GDP as a solid line. The economy advanced especially rapidly in 1983, 1987 and 1993, but there were slowdowns in 1981, 1985 and 1990.

If we define Lx to be $\beta x + Dx$, then we may say even more compactly that $Lx = 0$ implies exponential growth. When studying processes that exhibit exponential growth or decay to some extent, it can be helpful to look at Lx defined in this way; the extent to which the result is a nonzero function with substantial variation is a measure of departure from exponential growth, just as the appearance of a nonzero phase in D^2x for a mechanical system indicates the application of a force.

The right panel of Figure 18.3 shows the result of applying this differential operator to the U.S. GDP data. The result is clearly not zero; there seem to be three cycles of shorter term growth in GDP that depart from the longer-term exponential trend. Figure 18.4 shows the comparable curves for the United Kingdom and Japan, and we note that the U.K. had only one boom period with an uncertain recovery after the recession, while Japan experienced a deep and late recession.

18.4.3 Differential operators to regularize or smooth models

Although we have covered this topic elsewhere, we should still point out that we may substitute Lx for D^2x in any of the regularization schemes covered so far. Why? The answer lies in the homogeneous equation $Lx = 0$; functions satisfying this equation are deemed to be *ultrasmooth* in the sense that we choose to ignore any component of variation of this form in calculating roughness or irregularity. In the case of the operator D^2, linear trend is considered to be so smooth that any function may have an arbitrary amount of it, since the penalty term $\lambda \int (D^2x)^2$ is unaffected. Suppose, on the other hand, that we are working with a process that is predominantly exponential growth with rate parameter β. We may choose in this case to do nonparametric regression with the fitting criterion

$$\texttt{PENSSE}_\lambda(x) = n^{-1} \sum_{j=1}^{n} [y_j - x(t_j)]^2 + \lambda \int [\beta x(t) + Dx(t)]^2 \, dt$$

in order to leave untouched any component of variation of this form.

More generally, suppose we observe a set of discrete data values generated by the process

$$y_j = x(t_j) + \epsilon_j,$$

where, as in previous chapters, x is some unobserved smooth function that we wish to estimate by means of nonparametric regression, and ϵ_j is a disturbance or error assumed to be independently distributed over j and to have mean zero and finite variance. Suppose, moreover, that we employ the general smoothing criterion

$$\texttt{PENSSE}_\lambda(\hat{x}) = n^{-1} \sum_{j} [y_j - \hat{x}(t_j)]^2 + \lambda \int (L\hat{x})^2(t) \, dt \tag{18.13}$$

for some differential operator L.

It is not difficult to show (see Wahba, 1990) that, if we choose $\hat{x}$ to minimize $\texttt{PENSSE}_\lambda$, then the integrated squared bias

$$\texttt{Bias}^2(\hat{x}) = \{ \int \mathrm{E}[\hat{x}(t) - x(t)] \, dt \}^2$$

cannot exceed $\int (Lx)^2(t) \, dt$. This is useful, because if we choose L so as to approximate $Lx = 0$, then the bias is likely to be small. It then follows that we can use a relatively large value of the smoothing parameter λ, leading to lower variance, without introducing excessive bias. Also, we can achieve a small value of the integrated mean squared error

$$\texttt{IMSE}(\hat{x}) = \int \mathrm{E}[\hat{x}(t) - x(t)]^2 \, dt$$

since

$$\texttt{IMSE}(\hat{x}) = \texttt{Bias}^2(\hat{x}) + \texttt{Var}(\hat{x}),$$

where

$$\mathtt{Var}(\hat{x}) = \int \mathrm{E}\{\hat{x}(t) - \mathrm{E}[\hat{x}(t)]\}^2 \, dt.$$

The conclusion, therefore, is that if we have any prior knowledge at all about the predominant shape of x, it is worth choosing a linear differential operator L so as to annihilate functions having that shape. We show how to construct customized spline smoothers of this type in the next two chapters.

This insight about the role of L in the regularization process also leads to the following interesting question: Can we use the information in N replications x_i of functional observations such as growth or temperature curves to *estimate* an operator L that comes close in some sense to satisfying $Lx_i = 0$? If so, then we should certainly use this information to improve on our smoothing techniques. This matter is taken up in Chapter 21.

18.4.4 Differential operators to partition variation

Linear differential operators L of the form (18.12) of degree m have m linearly independent solutions ξ_j of the homogeneous equation $L\xi_j = 0$. There is no unique way of choosing these m functions ξ_j, but any choice is related by a linear transformation to any other choice. The set of all functions z for which $Lz = 0$ is called the *null space* of L, and the functions ξ_j form a basis for this space. The notation $\ker L$ is often used to indicate this null space.

Consider, for example, the derivative operator $L = D^m$: The monomials $\{1, t, \ldots t^{m-1}\}$ are a basis for the null space, as is the set of m polynomials formed by any nonsingular linear transformation of these. Likewise the functions $\{1, e^{-\beta t}\}$ are a solution set for $\beta Dx + D^2x = 0$. And $\{1, \sin \omega t, \cos \omega t\}$ were cited as the solution set or null space functions for $Lx = \omega^2 Dx + D^3x = 0$ in Chapter 1.

This means, then, that we can use linear differential operators L to partition functional variation in the sense that Lx splits x into two parts, the first consisting of what is in x that can be expressed in terms of a linear combination of the null space functions ξ_j, and the second being whatever is orthogonal to these functions.

This partitioning of variation is just what happens, as we already know from Section 4.4, with basis functions ϕ_k and the projection operator P that expands x as a linear combination of these basis functions. That is,

$$Px = \hat{x} = \sum_{k=1}^{m} c_k \phi_k$$

also splits any function x into the component $\hat{x}$ that is an optimal combination of the basis functions in a least squares sense, and an orthogonal residual component $x - \hat{x} = (I - P)x$. The complementary projection operator $Q = I - P$ therefore satisfies the linear homogeneous equation $Q\hat{x} = 0$,

as well as the m equations $Q\phi_k = 0$. Thus the projection operator Q and the differential operator L have analogous properties.

But there are some important differences, too. First, the projection operator P does not pay any attention to derivative information, whereas L does. Second, we have the closely related fact that Q is chosen to make Qx small, while L is chosen to make Lx small. Since Lx involves derivatives up to order m, making Lx small inevitably means paying attention to the size of $D^m x$. If we think there is important information in derivatives, then it seems right to exploit this in splitting variation.

It is particularly easy to compare the two operators, differential and projection, in situations where there is an orthonormal basis expansion for the function space in question. Consider, for example, the space of infinitely differentiable periodic functions defined on the interval $[0, 1]$ that would be natural to model our temperature and precipitation records. A function x has the Fourier expansion

$$x(t) = c_0 + \sum_{k=1}^{\infty} [c_{2k-1} \sin(2\pi kt) + c_{2k} \cos(2\pi kt)].$$

Suppose our two operators L and Q are of order 3 and designed to eliminate the first three terms of the expansion, that is, vertically shifted sinusoidal variation of period 1. Then

$$Qx(t) = \sum_{k=2}^{\infty} [c_{2k-1} \sin(2\pi kt) + c_{2k} \cos(2\pi kt)]$$

while

$$\begin{aligned} Lx &= 4\pi^2 Dx + D^3 x \\ &= \sum_{k=2}^{\infty} 8\pi^3 k(k^2 - 1)[-c_{2k-1} \cos(2\pi kt) + c_{2k} \sin(2\pi kt)]. \end{aligned}$$

Note that applying Q does not change the expansion beyond the third term, while L multiplies each successive pair of sines and cosines by an ever-increasing factor proportional to $k(k^2 - 1)$. Thus, L actually accentuates high-frequency variation while Q leaves it untouched; functions that are passed through L are going to come out rougher than those passing through Q.

The consequences for smoothing are especially important: If we penalize the size of $\|Lx\|^2$ in spline smoothing by minimizing the criterion (18.13), the roughening action of L means that high-frequency components are forced to be smaller than they would be in the original function, or than they would be if we penalized using Q by using the criterion

$$\texttt{PENSSE}_\lambda^Q(\hat{x}) = n^{-1} \sum_j [y_j - \hat{x}(t_j)]^2 + \lambda \int (Q\hat{x})^2(t)\, dt. \tag{18.14}$$

But customizing a regularization process is only one reason for splitting functional variation, and in Chapter 19 we look at a differential operator analogue of principal components analysis, called *principal differential analysis*, that can prove to be a valuable exploratory tool.

18.4.5 Operators to define solutions to problems

We have already considered a number of situations in Chapter 6 requiring smoothing functions x that had constraints such as positivity, monotonicity, values in (0,1), and so forth. We saw there that functions having these constraints can often be expressed as solutions to linear or nonlinear differential equations. This insight helped us to modify conventional linear least squares smoothing methods to accommodate these constraints.

18.5 Some linear differential equation facts

So far in this chapter, we have set the scene for the use of linear differential operators and equations in FDA. We now move on to a more detailed discussion of techniques and ideas that we use in this and the following chapters. Readers with some familiarity with the theory of linear ordinary differential equations may wish to skip on to the next two chapters, and refer back to this material only where necessary.

18.5.1 Derivatives are rougher

First, it is useful to point out a few things of general importance. For example, taking a derivative is generally a roughening operation, as we have observed in the context of periodic functions. This means that Dx has in general rather more curvature and variability than x. It is perhaps unfortunate that our intuitions about functions are shaped by our early exposure to polynomials, where derivatives are smoother than the original functions, and transcendental functions such as e^t and $\sin t$, where taking derivatives produces essentially no change in shape. In fact, the general situation is more like the growth curve accelerations in Figure 1.2, which are much more variable than the height curves in Figure 1.1, or the roughening effect of applying the third order linear differential operator to temperature functions displayed in Figure 1.7.

By contrast, the operation of partial integration essentially reverses the process of differentiation (except for the constant of integration), and therefore is a smoothing operation. It is convenient to use the notation $D^{-1}x$ for

$$D^{-1}x(t) = \int_{t_0}^{t} x(s)\, ds,$$

relying on context to specify the lower limit of integration t_0. This means, of course, that $D^{-1}Dx = x$.

18.5.2 Finding a linear differential operator that annihilates known functions

We have already cited a number of examples where we had a set of known functions $\{\xi_1, \ldots, \xi_m\}$ and where at the same time we were aware of the operator L that solved the homogeneous linear differential equations $L\xi_j = 0, j = 1, \ldots, m$. Suppose, however, that we have the ξ_j's in mind but that the L that annihilates them is not obvious, and we want to find it.

The process of identifying the linear differential operator that sets m linearly independent functions to 0, as well as other aspects of working with linear differential operators, can be exhibited through the following example: Suppose we are considering an amplitude-modulated sinusoidal signal with fixed period ω. Such a signal would be of the form

$$x(t) = A(t)[c_1 \sin(\omega t) + c_2 \cos(\omega t)]. \tag{18.15}$$

The function A determines the amplitude pattern. If A is regarded as a known time-varying function, the constants c_1 and c_2 determine the overall size of the amplitude of the signal and also the phase of the signal.

Our aim, for given ω and $A(t)$, is to find a differential operator L such that the null space of L consists of all functions of the form 18.15. Because these functions form a linear space of dimension 2, we seek an annihilating operator of order 2, of the form

$$Lx = \beta_0 x + \beta_1 Dx + D^2 x.$$

The task is to calculate the two weight functions β_0 and β_1.

First, let's do a few things to streamline the notation. Define the vector functions $\boldsymbol{\xi}(t)$ and $\boldsymbol{\beta}(t)$ as

$$\boldsymbol{\xi}(t) = \begin{bmatrix} A(t)\sin(\omega t) \\ A(t)\cos(\omega t) \end{bmatrix} \text{ and } \boldsymbol{\beta}(t) = \begin{bmatrix} \beta_0(t) \\ \beta_1(t) \end{bmatrix}. \tag{18.16}$$

Also, let us use $\mathtt{S}(t)$ to stand for $\sin(\omega t)$ and $\mathtt{C}(t)$ for $\cos(\omega t)$. Then

$$\boldsymbol{\xi} = \begin{bmatrix} A\mathtt{S} \\ A\mathtt{C} \end{bmatrix}. \tag{18.17}$$

The required differential operator L satisfies the vector equation $L\boldsymbol{\xi} = 0$.

Recall that the first and second derivatives of $\mathtt{S}$ are $\omega\mathtt{C}$ and $-\omega^2\mathtt{S}$, respectively, and that those of $\mathtt{C}$ are $-\omega\mathtt{S}$ and $-\omega^2\mathtt{C}$, respectively. Then the first two derivatives of ζ are, after a bit of simplification,

$$D\boldsymbol{\xi} = \begin{bmatrix} (DA)\mathtt{S} + \omega A\mathtt{C} \\ (DA)\mathtt{C} - \omega A\mathtt{S} \end{bmatrix}$$

and

$$D^2\boldsymbol{\xi} = \begin{bmatrix} (D^2A)\mathtt{S} + 2\omega(DA)\mathtt{C} - \omega^2 A\mathtt{S} \\ (D^2A)\mathtt{C} - 2\omega(DA)\mathtt{S} - \omega^2 A\mathtt{C} \end{bmatrix}. \tag{18.18}$$

The relation $L\boldsymbol{\xi} = 0$ can be expressed as follows, by taking the second derivatives over to the other side of the equation:

$$\beta_0\boldsymbol{\xi} + \beta_1 D\boldsymbol{\xi} = -D^2\boldsymbol{\xi} \tag{18.19}$$

or, in matrix notation

$$\begin{bmatrix} \boldsymbol{\xi} & D\boldsymbol{\xi} \end{bmatrix} \boldsymbol{\beta} = -D^2\boldsymbol{\xi}. \tag{18.20}$$

This is a linear matrix equation for the unknown weight functions β_0 and β_1, and its solution is simple provided that the matrix

$$\mathbf{W}(t) = \begin{bmatrix} \boldsymbol{\xi}(t) & D\boldsymbol{\xi}(t) \end{bmatrix} \tag{18.21}$$

is nowhere singular, or in other words that its determinant $|\mathbf{W}(t)|$ does not vanish for any value of the argument t. This coefficient matrix, which plays an important role in linear differential operator theory, is called the *Wronskian matrix*, and its determinant is called the *Wronskian* for the system.

Substituting the specific functions $A\mathtt{S}$ and $A\mathtt{C}$ for this example for ξ_1 and ξ_2 gives

$$\mathbf{W} = \begin{bmatrix} A\mathtt{S} & (DA)\mathtt{S} + \omega A\mathtt{C} \\ A\mathtt{C} & (DA)\mathtt{C} - \omega A\mathtt{S} \end{bmatrix}. \tag{18.22}$$

Thus the Wronskian is

$$|\mathbf{W}| = A\mathtt{S}[(DA)\mathtt{C} - \omega A\mathtt{S}] - A\mathtt{C}[(DA)\mathtt{S} + \omega A\mathtt{C}] = -\omega A^2 \tag{18.23}$$

after some simplification. We have no worries about the singularity of $\mathbf{W}(t)$, then, so long as the amplitude function $A(t)$ does not vanish.

The solutions for the weight functions are then given by

$$\boldsymbol{\beta} = -\mathbf{W}^{-1} D^2\boldsymbol{\xi}.$$

This takes a couple of sheets of paper to work out, or may be solved using symbolic computation software such as Maple (Char et al. 1991) or Mathematica (Wolfram, 1991). Considerable simplification is possible because of the identity $\mathtt{S}^2 + \mathtt{C}^2 = 1$, and the final result is that

$$\boldsymbol{\beta} = \begin{bmatrix} \omega^2 + 2(DA/A)^2 - D^2A/A \\ -2DA/A \end{bmatrix},$$

so that, for any function x,

$$Lx = [\omega^2 + 2(DA/A)^2 - D^2A/A]x - 2[(DA)/A](Dx) + D^2x. \tag{18.24}$$

Note that the weight coefficients in (18.24) are, as we should expect, scale free in the sense that multiplying $A(t)$ by any constant does not change them.

Consider two simple possibilities for amplitude modulation functions. When $A(t)$ is a constant, both derivatives vanish, the operator reduces to $L = \omega^2 I + D^2$ and $Lx = 0$ is the equation for simple harmonic motion. On the other hand, if $A(t) = e^{-\lambda t}$ so that the signal damps out exponentially with rate λ, then things simplify to

$$\boldsymbol{\beta} = \begin{bmatrix} \omega^2 + \lambda^2 \\ 2\lambda \end{bmatrix} \text{ or } Lx = (\omega^2 + \lambda^2)x + 2\lambda Dx + D^2 x. \tag{18.25}$$

This is the equation for damped harmonic motion with a damping coefficient 2λ.

The example illustrates the following general principles: First, the order m Wronskian matrix

$$\mathbf{W}(t) = \begin{bmatrix} \boldsymbol{\xi}(t) & D\boldsymbol{\xi}(t) & \dots & D^{m-1}\boldsymbol{\xi}(t) \end{bmatrix} \tag{18.26}$$

must be invertible, implying that its determinant should not vanish over the range of t being considered. There are ways of dealing with isolated singularities, however. Second, finding the vector of weight functions $\boldsymbol{\beta} = (\beta_0(t), \dots, \beta_{m-1}(t))'$ is then is a matter of solving the system of m linear equations

$$\mathbf{W}(t)\boldsymbol{\beta}(t) = -D^m \boldsymbol{\xi}(t),$$

again with the possible aid of symbolic computation software.

18.5.3 Finding the functions ξ_j satisfying $L\xi_j = 0$

Let us now consider the problem converse to that considered in Section 18.5.2. Given a linear differential operator L of order m, we might wish to identify m linearly independent solutions ξ_j to the homogeneous equation $Lx = 0$. We can do this directly by elementary calculus in simple cases, but more generally there is a variety of analytic and numerical approaches to this problem. For full details, see a standard reference on numerical methods, such as Stoer and Bulirsch (2002).

Specifically, given (18.7), a common procedure is to use a numerical differential equation solving algorithm, such as one of the Runge-Kutta methods, to solve the equation for *initial value constraints*, described below, of the form $B_0 x = \mathbf{I}_i$, where $\mathbf{I}_i$ is the ith column of the identity matrix of order m. This will yield m linearly independent solutions ξ_i that can be used as a basis for obtaining all possible solutions.

18.6 Initial conditions, boundary conditions and other constraints

18.6.1 Why additional constraints are needed to define a solution

We have already noted that the space of solutions of the linear differential equation $Lx = 0$ is, in general, a function space of dimension m, called the null space of L, and denoted by $\ker L$. We now assume that the linearly independent functions $\xi_1, \ldots, \xi_m$ form a basis of the null space.

Any specific solution of $Lx = 0$ requires m additional pieces of information about x. For example, we can solve the equation $\beta Dx + D^2x = 0$, defining a shifted exponential, uniquely provided that we are able to specify that

$$x(0) = 0 \text{ and } Dx(0) = 1,$$

in which case

$$x(t) = \frac{1}{\beta}(1 - e^{-\beta t}).$$

Alternatively, $x(0) = 1$ and $Dx(0) = 0$ implies that $x_0 = 1$ and $\alpha = 0$, or simply that $x = 1$.

We introduce the notion of a *constraint operator* B to specify the m pieces of information about x that we require to identify a specific function x as the unique solution to $Lx = 0$. This operator simply *evaluates* x or its derivatives in m different ways. The most important example is the *initial value* operator used in the theory of ordinary differential equations defined over an interval $\mathcal{T} = [0, T]$,

$$\textbf{Initial Operator: } B_0x = \begin{bmatrix} x(0) \\ Dx(0) \\ \vdots \\ D^{m-1}x(0) \end{bmatrix}. \tag{18.27}$$

When B_0x is set to an m-vector, initial value constraints are defined. In the example above, we considered the two cases $B_0x = (0, 1)'$ and $B_0x = (1, 0)'$, implying the two solutions given there.

The following *boundary value* operator is also of great importance in applications involving linear differential operators of even degree:

$$\textbf{Boundary Operator: } B_Bx = \begin{bmatrix} x(0) \\ x(T) \\ \vdots \\ D^{(m-2)/2}x(0) \\ D^{(m-2)/2}x(T) \end{bmatrix}. \tag{18.28}$$

Specifying $B_B x = c$ gives the values of x and its first $(m-2)/2$ derivatives at both ends of the interval of interest.

The *periodic constraint* operator is

$$\textbf{Periodic Operator:}\quad B_P x = \begin{bmatrix} x(T) - x(0) \\ Dx(T) - Dx(0) \\ \vdots \\ D^{m-1}x(T) - D^{m-1}x(0) \end{bmatrix}. \tag{18.29}$$

Functions satisfying $B_P x = 0$ are periodic up to the derivative D^{m-1} over $\mathcal{T}$, and are said to obey periodic boundary conditions.

The *integral constraint* operator is

$$\textbf{Integral Operator:}\quad B_I x = \begin{bmatrix} \int \xi_1(t)x(t)\,dx \\ \int \xi_2(t)x(t)\,dx \\ \vdots \\ \int \xi_m(t)x(t)\,dx \end{bmatrix}, \tag{18.30}$$

where $\xi_1, \ldots, \xi_m$ are m linearly independent weight functions.

18.6.2 How L and B partition functions

Whatever constraint operator we use, consider the problem of expressing any particular function x as a sum of two components z and e, such that $Lz = 0$ and $Be = 0$. When can we carry out this partitioning in a unique way? This happens if and only if $x = 0$ is the only function satisfying both $Bx = 0$ and $Lx = 0$. Or, in algebraic notation,

$$\ker B \cap \ker L = 0. \tag{18.31}$$

Thus, the two operators B and L complement each other; the equation $Lx = 0$ defines a space of functions $\ker L$ that is of dimension m, and within this space B is a non-singular transformation. Or, looking at it the other way, the equation $Bx = 0$ defines a space of functions $\ker B$ of *codimension* m, within which L is a one-to-one transformation.

Note that the condition (18.31) can break down. Consider, for example, the operator $L = \omega^2 I + D^2$ on the interval $[0, T]$. The space $\ker L$ contains all linear combinations of $\sin\omega t$ and $\cos\omega t$. If $\omega = 2\pi k/T$ for some integer k and we use boundary constraints, all multiples of $\sin\omega t$ satisfy $B_B x = 0$, and so the condition (18.31) is violated. Some functions, namely those that satisfy $x(0) = x(T)$ and $Dx(0) = Dx(T)$, have infinitely many decompositions as $z + e$ with $Lz = Be = 0$, and are called the eigenfunctions of the differential operator.

18.6.3 The inner product defined by operators L and B

All the functional data analysis techniques and tools in this book depend on the notion of an inner product between two functions x and y. We have seen numerous examples of how a careful choice of inner product can produce more useful results, especially in controlling the roughness of estimated functions, such as functional principal components or regression functions. In these and other examples, it is important to use derivative information in defining an inner product.

Let us assume that the constraint operator is such that the orthogonality condition (18.31) is satisfied. We can define a large family of inner products as follows:

$$\langle x, y \rangle_{B,L} = (Bx)'(By) + \int (Lx)(t)(Ly)(t)\, dt \tag{18.32}$$

with the corresponding norm

$$\|x\|^2_{B,L} = (Bx)'(Bx) + \int (Lx)^2(t)\, dt. \tag{18.33}$$

The condition (18.31) ensures that this is a norm; the only function x for which $\|x\|_{B,L} = 0$ is zero itself, since this is the only function simultaneously satisfying $Bx = 0$ and $Lx = 0$.

In fact, this inner product works by splitting the function x into two parts:

$$x = z + e \text{ where } z \in \ker L \text{ and } e \in \ker B.$$

The first term in (18.33) simply measures the size of the component z, since $Be = 0$ and therefore $Bx = Bz$, while the second term depends only on the size of the component e since $Lx = Le$. The first term in (18.32) is essentially an inner product for the m-dimensional subspace in which z lives and which is defined by $Lz = 0$. The second term is an inner product for the function space of *codimension* m defined by $Be = 0$. Thus, we can write

$$\|x\|^2_{B,L} = \|z\|^2_B + \|e\|^2_L.$$

With this composite inner product in hand, that is, with a particular operator L and constraint operator B in mind, we can go back and revisit each of our functional data analytic techniques to see how they perform with this inner product. This is the central point explored by Ramsay and Dalzell (1991), to which we refer the reader for further discussion.

18.7 Further reading and notes

It is beyond the scope of this book to offer more than a cursory treatment of a topic as rich as the theory of differential equations, and there would be

little point, since there are many fine texts on the topic. Texts on differential equations that are designed for engineering students tend to have two advantages. The amount mathematical detail is kept minimal and one gets to see differential equations applied to real world problems and is thereby helped to see them as conceptual as opposed to technical tools.

Some of our favorites references that are also classics are Coddington (1989), Coddington and Levinson (1955), Ince (1956) and Tenenbaum and Pollard (1963). For advice on a wide range of computational and otherwise practical matters we recommend Press et al. (1992).

For more general results for arbitrary constraint operators B, including the integral operator conditions that we need in the following section, see Dalzell and Ramsay (1993) and Heckman and Ramsay (2000).

19

Fitting differential equations to functional data: Principal differential analysis

19.1 Introduction

Now that we have fastened a belt of tools around our waists for tinkering with differential equations, we return to the problems introduced in Chapter 17 ready to get down to some serious work.

Using a differential equation as a modelling object involves concepts drawn from both the functional linear model and from principal components analysis. A differential equation can certainly capture the shape features in both the curve and its derivative for a single functional datum such as the oil refinery observation shown in Figure 1.4. But because the set of solutions to a differential equation is an entire function space, it can also model variation across observations when $N > 1$. In this sense, it also has the flavor of principal components analysis where we find a subspace of functions able to capture the dominant modes of variation in the data.

We have, then, a question of emphasis or perspective. On one hand, the data analyst may want to capture important features of the dynamics of a single observation, and thus look *within* the m-dimensional space of solutions of an estimated equation to find that which gives the best account of the data. On the other hand, the goal may be to see how much functional variation can be explained *across* multiple realizations of a process. Thus, linear modelling and variance decomposition merge into one analysis in this environment.

We introduce a new term here: *principal differential analysis* means the fitting of a differential equation to noisy data so as to capture either the

features of a single curve or the variation in features across curves. This term was first used in Ramsay (1996a), and will be motivated in some detail in Section 19.6. The abbreviation PDA will be handy, and will also serve to remind us of the close connection with PCA.

19.2 Defining the problem

Our challenge is the identification of a linear differential operator

$$L = \beta_0 I + \ldots + \beta_{m-1} D^{m-1} + D^m \tag{19.1}$$

and its associated homogeneous differential equation

$$D^m x = -\beta_0 x - \ldots - \beta_{m-1} D^{m-1} x \tag{19.2}$$

using a set of N functional observations x_i along with, possibly, a set of associated functional covariates $f_{i\ell}, \ell = 1, \ldots, L$. We now call these covariates *forcing functions* so as to keep the nomenclature already current in fields such as engineering and physics. Although, in the examples used in this chapter, the x_i's are univariate functions, and only one forcing function, if at all, is used, we certainly have in mind that systems of differential equations and multiple forcing functions may be involved, and the differential equations may be nonlinear.

First, consider the homogeneous case, where no forcing function is present. We want to find the operator L that comes as close as possible to satisfying the homogeneous linear differential equation

$$Lx_i = 0, \ i = 1, \ldots, N. \tag{19.3}$$

In order to achieve this, we have to estimate up to m coefficient functional parameters $\beta_j, j = 0, \ldots, m-1$. Of course, some of these parameters may be fixed, often to zero as we have already seen, and the constant coefficient case is included within this framework by using a constant basis where required.

Since we wish the operator L to annihilate as nearly as possible the given data functions x_i, we regard the function Lx_i as being the residual from the fit provided by the corresponding linear differential equation (19.2). The least squares approach defines as the fitting criterion the sum of squared norms of the residual functions Lx_i:

$$\texttt{SSE}_{\text{PDA}}(L|\mathbf{x}) = \sum_{i=1}^{N} \int [Lx_i(t)]^2 \, dt = \sum_{i=1}^{N} \|Lx_i\|^2. \tag{19.4}$$

If an input forcing function f_i has also been observed along with the output x_i for a system, then we aim to solve as closely as possible the nonhomogeneous equation

$$Lx_i = f_i, i = 1, \ldots, N.$$

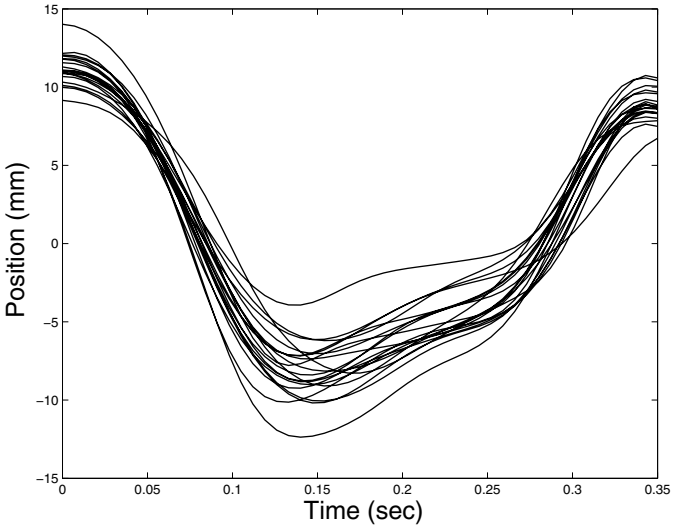

Figure 19.1. Twenty records of position of the center of the lower lip during the uttering of the syllable "bob."

The least squares fitting criterion now becomes

$$\mathtt{SSE}_{\mathrm{PDA}}(L|\mathbf{x},\mathbf{f}) = \sum_{i=1}^{N} \int [Lx_i(t) - f_i(t)]^2 \, dt = \sum_{i=1}^{N} \|Lx_i - f_i\|^2 \qquad (19.5)$$

It will be evident, when we compare these criteria with those for the concurrent functional linear model (14.5), that we may use the same methods here. Indeed, that is what we did in Chapter 17 for the oil refinery and melanoma data. However, there are other estimation techniques available that may be better. But before we consider these, we offer two examples to illustrate some of the issues involved in PDA.

19.3 A principal differential analysis of lip movement

There are several reasons why a PDA can provide important information about the data and the phenomenon under study. Certainly, in many applications the differential equation $Lx = 0$ offers an interesting and useful way of understanding the processes that generated the data.

Consider as an example to be used throughout this chapter the curves presented in Figure 19.1. These indicate the movement of the center of the lower lip as a single speaker said "bob." The displayed curves are the result of considerable preprocessing, including smoothing and the use of

functional PCA to identify the direction in which most of the motion was found. Details can be found in Ramsay, Munhall, Gracco and Ostry (1996). We see in broad terms that lower lip motion shows three phases: an initial rapid opening, a sharp transition to a relatively slow and nearly linear motion, and a final rapid closure.

19.3.1 The biomechanics of lip movement

Because the lower lip is part of a mechanical system, inevitably having certain natural resonating frequencies and a stiffness or resistance to movement, it seems appropriate to explore to what extent this motion can be expressed in terms of a second order linear differential equation of the type useful in the analysis of such systems,

$$Lx_i = \beta_0 x_i + \beta_1 Dx_i + D^2 x_i = 0. \tag{19.6}$$

Discussions of second order mechanical systems can be found in most applied texts on ordinary differential equations, such as Tenenbaum and Pollard (1963).

The first coefficient, β_0, essentially reflects the position-dependent force applied to the system at position x. Coefficient values $\beta_0 > 0$ and $\beta_1 = 0$ correspond to a system with sinusoidal or harmonic motion, with $\beta_0^{1/2}/(2\pi)$ cycles per unit time and wavelength or period $2\pi\beta_0^{-1/2}$; β_0 is often called the *spring constant*. The second coefficient, β_1, indicates influences on the system that are proportional to velocity rather than position, and are often internal or external frictional forces or viscosity in mechanical systems.

The *discriminant* of the second order operator and the mechanical system that it represents is defined as $d = (\beta_1/2)^2 - \beta_0$, and is critical in terms of its sign. When β_1 is small, meaning that d is negative, the system is underdamped, and tends to exhibit some oscillation that gradually disappears. When d is positive because β_1 is relatively large, the system is called overdamped, and either becomes stable so quickly that no oscillation is observed ($\beta_1 > 0$), or oscillates out of control ($\beta_1 < 0$). A critically damped system is one for which $d = 0$, and it exhibits non-oscillatory motion that decays rapidly to zero.

These mechanical interpretations of the roles of coefficient functions β_0 and β_1 are, strictly speaking, only appropriate if these functions are constants, but higher-order effects can be ignored if they do not vary too rapidly with t, in which case β_0, β_1, and d can be viewed as describing the instantaneous state of the system. When $\beta_0 = \beta_1 = 0$ the system is in linear motion, for which $D^2 x = 0$.

The techniques we develop were used to obtain the weight functions displayed in Figure 19.2. These are of rather limited help in interpreting the system, but one does note that β_0 is positive except for a brief episode near the beginning, and near zero in the central portion corresponding to the

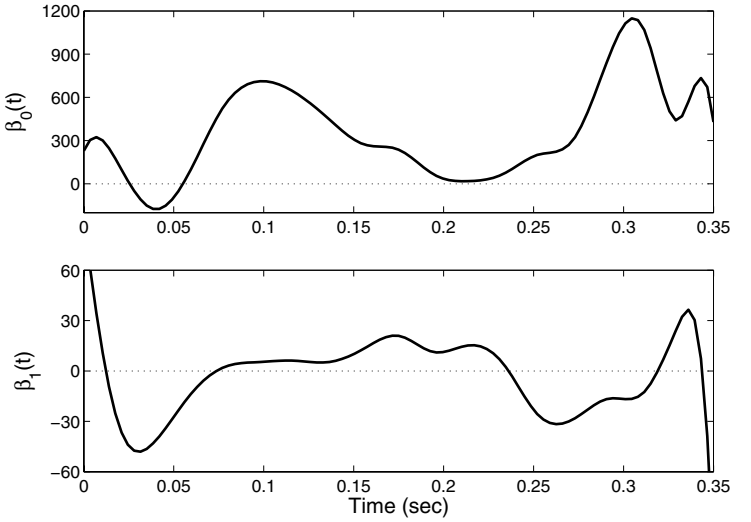

Figure 19.2. The two weight functions β_0 and β_1 for the second order linear differential equation estimated from the lip movement data.

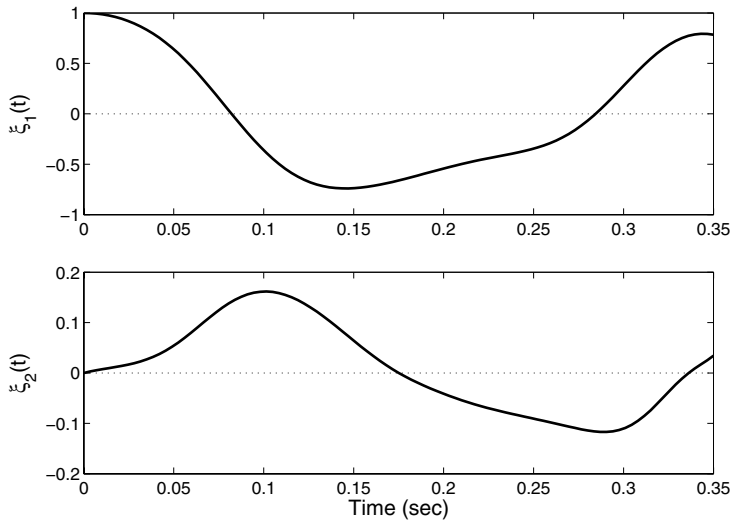

Figure 19.3. Two solutions of the second order linear differential equation estimated for the lip movement data corresponding to initial values conditions $(x(0) = 1, Dx(0) = 0)$ and $(x(0) = 0, Dx(0) = 1)$.

near linear phase of lip movement. The two solutions to the homogeneous differential equation $Lu = 0$ defined by the initial value conditions $(x(0) = 1, Dx(0) = 0)$ and $(x(0) = 0, Dx(0) = 1)$ are shown in Figure 19.3.

19.3.2 Visualizing the PDA results

How effective is the differential operator L at annihilating variation in the x_i? We can see this by plotting the *empirical forcing functions* Lx_i. If these are small and mainly noise-like, we can have some confidence that the equation is doing a good job of representing the data. It is easier to see how successful we have been if we have a null or benchmark hypothesis. A reasonable choice is the model defined by $\beta_0 = \ldots = \beta_{m-1} = 0$. The $D^m x_i$'s are the empirical forcing functions corresponding to this null hypothesis, and we can therefore compare the size of the Lx_i's to these derivatives.

Figure 19.4 shows the acceleration curves for the lip data in the left panel, and the empirical forcing functions in the right. We see that the forcing functions corresponding to L are indeed much smaller in magnitude, and more or less noise-like except for two bursts of signal near the beginning and end of the time interval.

The value of the criterion $\texttt{SSE}_{\text{PDA}}$ defined above is 7.7×10^6, while the same measure of the size of $D^2 x_i$'s is 90.4×10^6. If we call the latter measure $\texttt{SSY}_{\text{PDA}}$, then we can also summarize these results in the squared correlation measure

$$\texttt{RSQ}_{\text{PDA}} = (\texttt{SSY}_{\text{PDA}} - \texttt{SSE}_{\text{PDA}})/\texttt{SSY}_{\text{PDA}}, \tag{19.7}$$

the value of which is 0.92 for this problem.

While it is strictly speaking not the task of PDA to approximate the original curves (this would be a job for PCA), we can nevertheless wonder how well the two solution curves would serve this purpose. Figure 19.5 shows the least squares approximation of the first two curves in terms of the two solution functions in Figure 19.3, and we see that the fit is fairly satisfactory.

Finally, we return to the discriminant function $d = (\beta_1/2)^2 - \beta_0$, presented in Figure 19.6, and its interpretation. This system is more or less critically damped over the interval $0.18 \leq t \leq 0.26$, suggesting that its behavior may be under external control. But in the vicinities of $t = 0.08$ and $t = 0.30$, the system is substantially under-damped, and thus behaving locally like a spring. The period of the spring would be around 30 to 40 msec, and this is in the range of values estimated in studies of the mechanical properties of flaccid soft tissue. These results suggest that the external input to lip motions tends to be concentrated in the brief period near $t = 0.20$, when the natural tendency for the lip to close is retarded in order to allow for the articulation of the vowel.

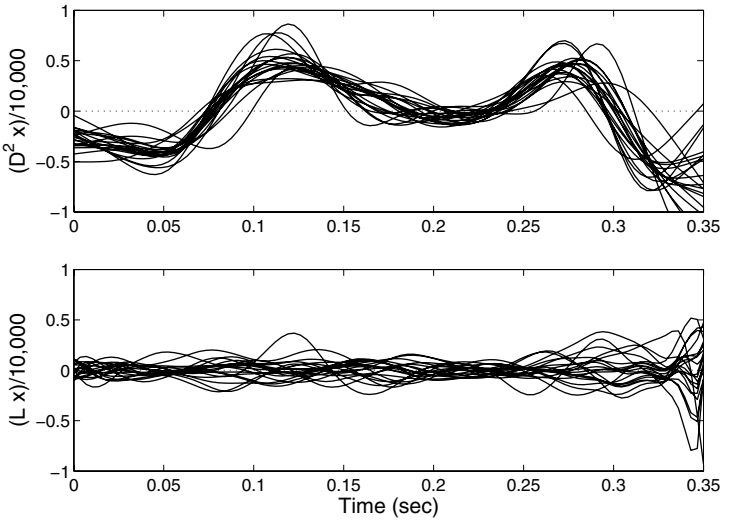

Figure 19.4. The left panel displays the acceleration curves D^2x_i for the lip position data, and the right panel the forcing functions Lx_i.

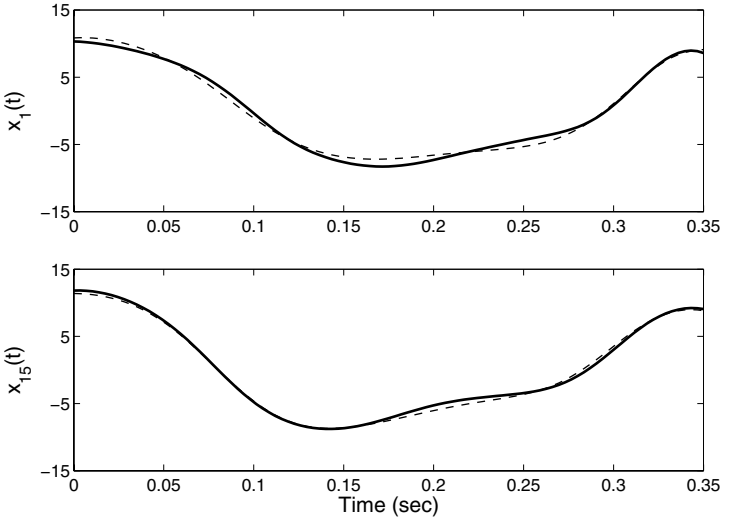

Figure 19.5. The solid curves are the first two observed lip position functions, and the dashed lines are their approximations on the basis of the two solution functions in Figure 19.3.

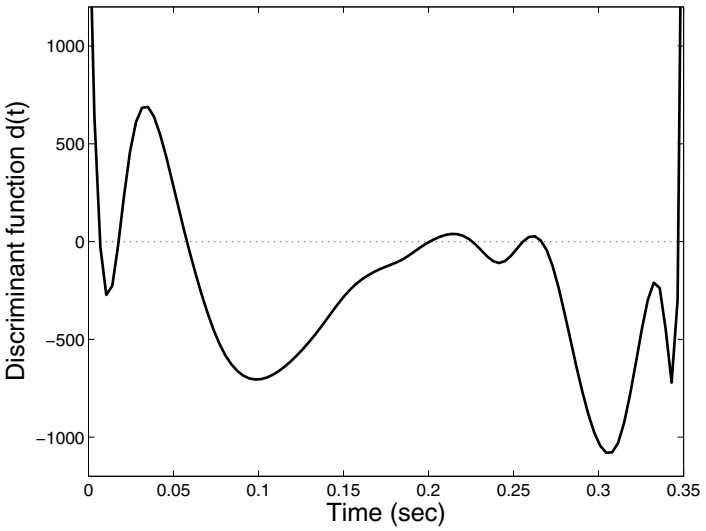

Figure 19.6. The discriminant function $d = (\beta_1/2)^2 - \beta_0$ for the second order differential equation describing lip position.

19.4 PDA of the pinch force data

In this section we take up an example in which the estimated linear differential operator is compared with a theoretically defined operator. The data in this example consisted of the 20 records of brief force impulses exerted by the thumb and forefinger in the experiment in motor physiology described in Section 1.5.2. For the purposes of this discussion, the force impulses were preprocessed to transform time linearly to a common metric, and to remove some simple shape variation. The resulting curves are displayed in Figure 19.7. Details concerning the preprocessing stages can be found in Ramsay, Wang and Flanagan (1995).

There are some theoretical considerations which suggest that the model

$$y_i(t) = C_i \exp[-\log^2 t/(2\sigma^2)] \tag{19.8}$$

offers a good account of any specific force function. In this application, the data were preprocessed to conform to a fixed shape parameter σ^2 of 0.05. Functions of the form (19.8) are annihilated by the differential operator $L_0 = [(t\sigma)^{-1} \log t]I + D$. A goal of this analysis is to compare this theoretical operator with the first order differential operator $L = \beta_0 I + D$ estimated from the data, or to compare the theoretical weight function $\omega_0(t) = (t\sigma)^{-1} \log t$ with its empirical counterpart β_0.

We smoothed the records using splines, with the size of the third derivative being penalized in order to get a smooth first derivative estimate. It is clear from Figure 19.7 that the size of error variation is not constant over

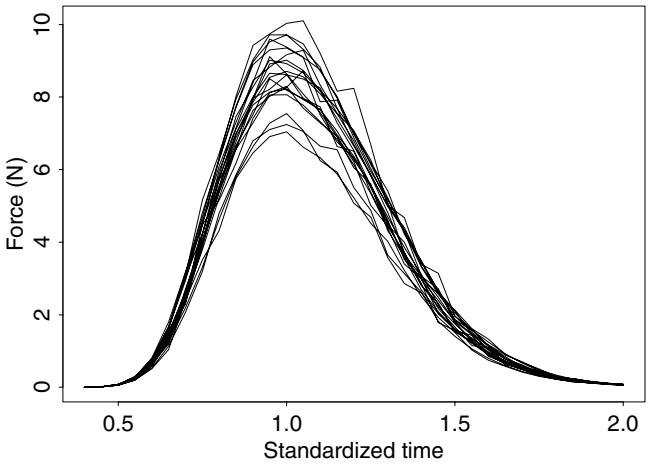

Figure 19.7. Twenty recordings of the force exerted by the thumb and forefinger during a brief squeeze of a force meter. The data have been preprocessed to register the functions and remove some shape variability, and the values displayed are for the 33 values $t = 0.4(.05)2.0$.

time. Accordingly, we estimated the residuals in a first smoothing step, and smoothed the logs of their standard deviations to estimate the variation of the typical residual size over time. We then took the weights σ_j^2 in the weighted spline smoothing criterion

$$\texttt{PENSSE}_\lambda(\mathbf{x}|\mathbf{y}) = \sum_j [y_j - x(t_j)]^2/\sigma_j^2 + \lambda \|D^3 x\|^2 \tag{19.9}$$

to be the squares of the exponential-transformed smooth values. Finally, we re-smoothed the data to get the spline smoothing curves and their derivatives. Figure 19.8 displays the discrete data points, the smoothing function, and also the theoretical function (19.8) fit by least squares for a single record. The theoretical function fits very well, but in the right panel we see that the discrepancy between the theoretical model and the smoothing spline fit is nevertheless smooth and of the order of the largest deviations of the points from this flexible spline fit. While this discordance between the model and the spline is less than 2% of the size of the force itself, we are nevertheless entitled to wonder if this theoretical model can be improved.

We applied both the point-wise and basis expansion procedures for estimating β_0 to the smooth functions and their derivatives, as described in Section 19.5. The basis used for the basis expansion procedure was

$$\phi(t) = (t^{-1}\log t, 1, t-1, (t-1)^2)',$$

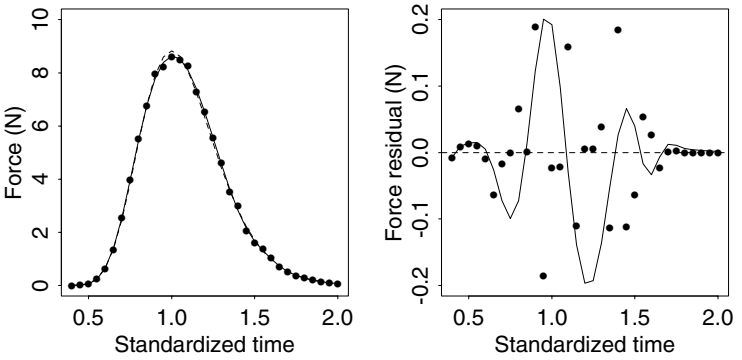

Figure 19.8. The left figure contains the data values for the first record (the points), the smoothing spline (solid curve), and the least squares fit by the model (19.8) (dotted curve). The right display shows the residuals arising from fitting the points by a spline function (the points) and the difference between the theoretical model and the spline (solid curve).

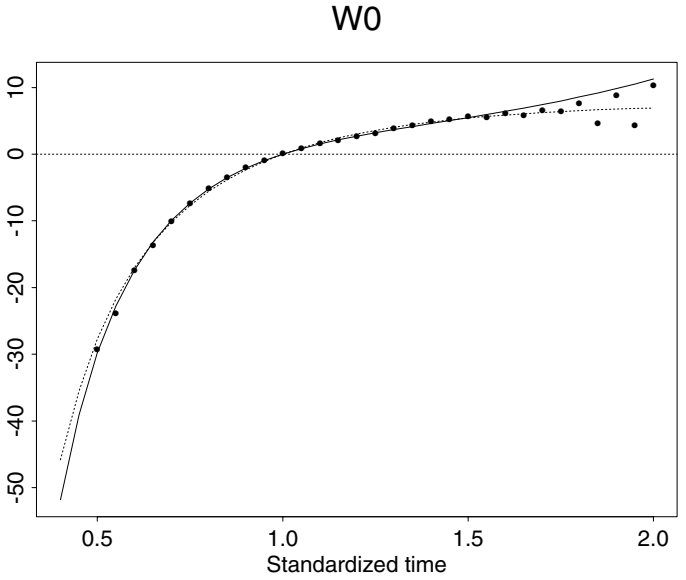

Figure 19.9. The weight function estimated by the basis expansion method for the pinch force data is indicated by the solid line, the theoretical function by the dotted line, and the point-wise estimates by the dots.

chosen after some experimentation; the first basis function was suggested by the theoretical model, and the remaining polynomial terms served to extend this model as required. Figure 19.9 shows the theoretical, the point-wise and the global estimates of the weight functions. These are admittedly close

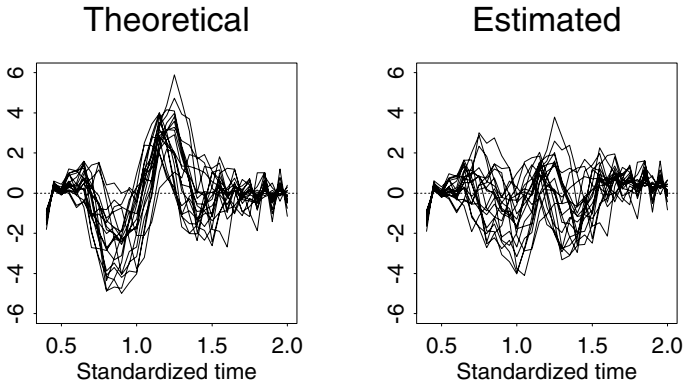

Figure 19.10. The left panel displays the forcing or impulse functions Ly_i produced by the theoretical operator, and the right panel shows the corresponding empirical operator functions.

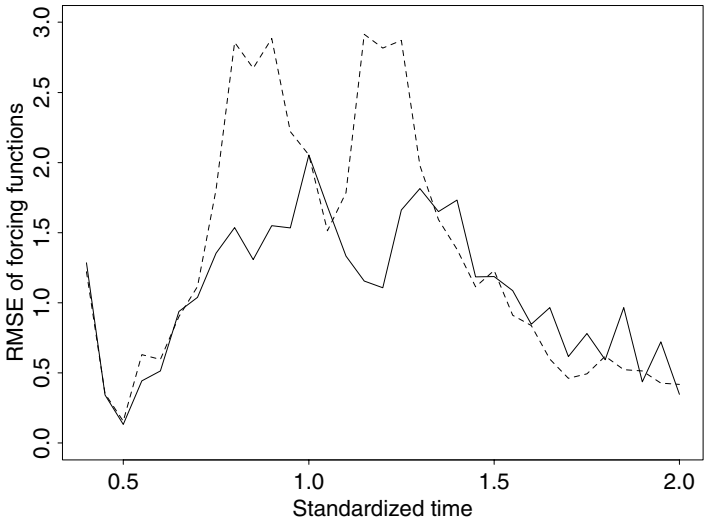

Figure 19.11. The solid line indicates the square root of the mean squared forcing function for the estimated operator, and the dotted line the same quantity for the theoretical operator.

to one another, at least in the central ranges of adjusted time, but again we observe some slight but consistent differences between the theoretical and empirical weight functions.

However, the forcing functions Ly_i, displayed in Figure 19.10, show a systematic trend for the theoretical operator, while the empirical forcing functions exhibit much less pattern. Figure 19.11 displays the root-mean-squares of the two sets of forcing functions, and this permits us to see more clearly that the estimated operator is superior in the epochs just before and after the peak force, where it produces a forcing function about half the size of its theoretical counterpart. It seems appropriate to conclude that the estimated operator has produced an important improvement in fit on either side of the time of maximum force. Ramsay, Wang and Flanagan (1995) conjecture that the discrepancy between the two forcing functions is due to drag or viscosity in the thumb-forefinger joint.

19.5 Techniques for principal differential analysis

We turn now to some methods for estimating the weight functions β_j defining the linear differential operator that comes closest to annihilating the observed functions in the sense of criterion (19.4). All but the final method assume that we have already estimated the function and its derivatives up to order m by smoothing the raw discrete data.

19.5.1 PDA by point-wise minimization

The first approach yields a point-wise estimate of the weight functions β_j computable by standard least squares estimation. Define the point-wise fitting criterion

$$\texttt{PSSE}_L(t) = \sum_i [Lx_i(t) - f_i(t)]^2 = \sum_i [\sum_{j=0}^{m} \beta_j(t) D^j x_i(t) - f_i(t)]^2, \quad (19.10)$$

where, as above, $\beta_m(t) = 1$ for all t. If t is regarded as fixed, this following argument shows that this is simply a least squares fitting criterion.

First define the m-dimensional coefficient vector

$$\boldsymbol{\beta}(t) = (\beta_0(t), \ldots, \beta_{m-1}(t))',$$

the $N \times (m+1)$ point-wise design matrix $\mathbf{Z}$ with rows

$$\mathbf{z}_i(t) = \{-x_i(t), \ldots, -D^{m-1}x_i(t), f_i(t)\}$$

and the N-dimensional dependent variable vector $\mathbf{y}$ with elements

$$y_i(t) = D^m x_i(t).$$

We can express the fitting criterion (19.10) in matrix terms as

$$\texttt{PSSE}_L(t) = [\mathbf{y}(t) - \mathbf{Z}(t)\boldsymbol{\beta}(t)]'[\mathbf{y}(t) - \mathbf{Z}(t)\boldsymbol{\beta}(t)].$$

Then, holding t fixed, the least squares solution minimizing $\mathtt{PSSE}_L(t)$ with respect to values $\beta_j(t)$ is

$$\boldsymbol{\beta}(t) = [\mathbf{Z}(t)'\mathbf{Z}(t)]^{-1}\mathbf{Z}(t)'\mathbf{y}(t). \tag{19.11}$$

The existence of these point-wise values $\boldsymbol{\beta}(t)$ depends on the determinant of $\mathbf{Z}(t)'\mathbf{Z}(t)$ being bounded away from zero for all values of t, and it is wise to compute and display this determinant as a routine part of the computation. Assuming that the determinant is nonzero is equivalent to assuming that $\mathbf{Z}(t)$ is of full column rank for all t.

Of course, if m is not large, then we can express the solution in closed form. For example, for $m = 1$ we have

$$\beta_0(t) = -\sum_i x_i(t)(Dx_i)(t) / \sum_i x_i^2(t) \tag{19.12}$$

and the full-rank condition requires that for each value of t some $x_i(t)$ be nonzero.

Some brief comments about the connections with Section 18.5.2 are in order. There, we were concerned with finding a linear operator of order m that annihilated a set of exactly m functions u_i. In order for this to be possible, an important condition was the nonsingularity of the Wronskian matrix values $\mathbf{W}(t)$ whose elements were $D^j u_i(t)$. We obtain the matrix $\mathbf{Z}(t)$ from the functions x_i in the same way, but it is no longer a square matrix, since in general we will have $N > m$. However, the condition that $\mathbf{Z}(t)$ is of full column rank is entirely analogous.

19.5.2 PDA using the concurrent functional linear model

The point-wise approach can pose problems in some applications. First, solving the equation $Lu = 0$ requires that the β_j's be available at a fine level of detail, with the required resolution depending on their smoothness. Whether or not these functions are smooth depends in turn on the smoothness of the derivatives $D^j x_i$. Since we often estimate these derivatives by smoothing procedures that may not always yield smooth estimates for higher order derivatives, the resolution we require may be very fine indeed. Moreover, for larger orders m, computing the functions β_j point-wise at a fine resolution level can be computationally intensive, since we must solve a linear equation for every value of t for which w is required. We need an approximate solution which can be quickly computed and which is reasonably regular or smooth.

It may also be desirable to circumvent the restriction that the rank of $\mathbf{Z}$ be full, especially if the failure is highly localized within the interval of integration. As a rule, an isolated singularity for $\mathbf{Z}(t)'\mathbf{Z}(t)$ corresponds to an isolated singularity in one or more weight functions β_j, and it may be desirable to bypass these by using weight functions sure to be sufficiently

smooth. More generally, we may seek weight functions more smooth or regular than those resulting from the point-wise solution.

Finally, the point-wise procedure only works if the number of functional observations N exceeds the number of columns of the point-wise design matrix $\mathbf{Z}$. But we often need to fit a differential equation to a single functional observation, and we want a method that will accommodate this case.

A strategy for identifying smooth weight functions β_j is to approximate them by using a fixed set of basis functions. This takes us directly back to Chapter 14 on the concurrent linear model, where the computational procedure is exactly what we need here. The only differences between PDA and the analyses in Chapter 14 is that here the dependent variable is $D^m x$, and the lower order derivatives can appear on the independent variable side of the equation.

19.5.3 PDA by iterating the concurrent linear model

The application of the concurrent functional linear model to this problem presupposes that the estimated derivatives $D^j x_i$ are reasonable. The melanoma analysis in Chapter 17 suggests, however, that it may be worth re-estimating the derivatives once an initial differential equation has been estimated. We can do this by using the corresponding linear differential operator to define the roughness penalty. This cycle can be repeated as many times as are required in order to achieve stable derivative estimates.

A simulated data experiment illustrates the consequences of this iterative refinement of the roughness penalty using PDA. A sample of 1000 sets of functional data were generated using the tilted sinusoid model

$$x_i(t_j) = c_{i1} + c_{i2}t_j + c_{i3}\sin(6\pi t_j) + c_{i4}\cos(6\pi t_j) + \epsilon_{ij}$$

for the 101 values $t_j = (0, 0.01, \ldots, 1)$. The coefficients $c_{ik}, k = 1, \ldots, 4$ were independently generated from a normal distribution with mean zero. The standard deviations were 1 for $k = 1$, 3 and 4, and 4 for c_{i_2}. The errors ϵ_{ij} were independent standard normal deviates. Figure 19.12 shows the first set of samples.

The errorless curves are annihilated by the operator

$$Lx = (6\pi)^2 D^2 x + D^4 x,$$

where $(6\pi)^2 = 355.3$. How well can we estimate this operator from these data? Does estimating this operator buy us anything in terms of the qualities of the estimates of the curves and their derivatives? For example, how well is the second derivative estimated when we use an estimated operator L rather than the default choice $L = D^4$?

The initial operator was consequently $L = D^4$, and was used to define the initial penalty matrix $\mathbf{R}$. The basis system that we used to estimate the true curves and their derivatives consisted of 105 order 6 B-spline basis

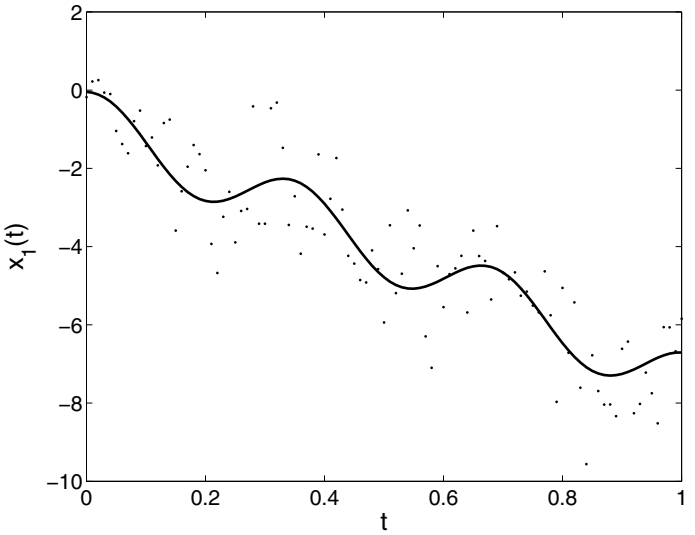

Figure 19.12. The dots indicate the data generated by adding independent standard normal deviates to the tilted sinusoid shown as the solid line.

functions with knots at the sampling points t_j. We cycled through the following process five times:

1. The value of the smoothing parameter λ minimizing the GCV criterion was found using a numerical optimization method. In order to avoid rounding error problems, an upper limit on the allowable estimate was set to $10 - \log_{10} \text{trace}\mathbf{R}$.

2. The data were smoothed using this value of λ.

3. A principal differential analysis was performed based on the concurrent linear model method described above. All four coefficient functions $\beta_j, j = 0, 1, 2, 3$ were estimated using the constant basis for each.

4. The linear differential operator estimated by PDA was then used to redefine the penalty matrix $\mathbf{R}$.

The estimated λ after the first cycle was $10^{-9.9}$, and after the second cycle it came up against the upper limit that we imposed, which for these data was $10^{-8.8}$. Subsequent iterations hardly changed the results at all.

After the first iteration, defined by $L = D^4$, the PDA estimated the operator as

$$Lx = 2360.3x - 123.8Dx + 376.1D^2x - 0.3D^3x + D^4x$$

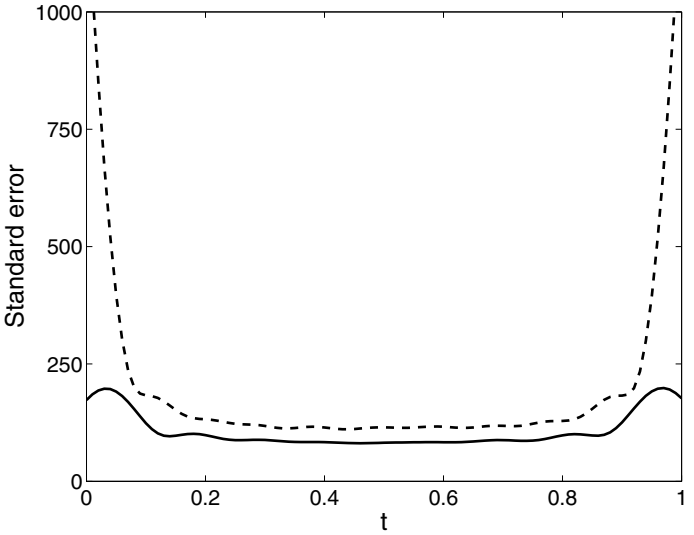

Figure 19.13. The solid curve is the standard error of estimate for the second derivative of the tilted sine data after five iterations, and the dashed line is after the first iteration.

and after all five iterations, the estimate was

$$Lx = 310.4x - 125.4Dx + 357.0D^2x - 0.4D^3x + D^4x.$$

The estimate after one iteration is quite good, judging by the coefficients for the key derivatives of order 2 and 3 that determine the period and phase of the sinusoid, and only slightly improved by going through all five iterations. For the record, we also tried fixing the first two coefficients to zero, but the estimates of the second two coefficients were not appreciably better.

The most striking benefit is in terms of the precision of the function and derivative estimates. The ratios of the first iteration integrated mean squared error to that on the fifth iteration are 1.2, 2.0, 3.8, 5.3 and 8.1 for derivatives of order 0, 1, 2, 3 and 4, respectively. The function values are modestly improved, but the improvement brought about by iterative refinement of L increases with the order of the derivative. To see better both the improvement and how it is achieved, we turn to Figure 19.13 which shows the point-wise standard errors of the second derivatives after the first and fifth iterations. The big impact is at the endpoints, where estimating a linkage between the function value and the second derivative greatly diminishes the standard error. Because function value estimates are less affected by having half the number of neighbors at the endpoints than are derivatives, estimating this linkage passes along the function value stability to the derivatives.

These results are probably better than we would encounter in practice, mainly because the true curves could all be annihilated in principle by a linear differential operator within the family of those that we could estimate. A similar study of growth curves generated by the Jolicoeur model used in Chapters 5 and 6 came up with a much more modest improvement in the second derivative because the variation from curve to curve is more complex than can be modelled with a single order four operator. Nevertheless, the improvements there were also more pronounced at the endpoints.

19.5.4 Assessing fit in PDA

Since the objective of PDA is to minimize the norm $\|L\mathbf{y}\|$ of the forcing function associated with an estimated differential operator, and since the quality of fit can vary over the domain T, it seems appropriate to assess fit in terms of the point-wise error sum of squares $\texttt{PSSE}_L(t)$ as defined in (19.10). As in linear modelling, the logical baseline against which we should compare $\texttt{PSSE}_L$ is the error sum of squares defined by a theoretical model and its associated weight functions ω_j:

$$\texttt{PSSE}_0(t) = \sum_i [\sum_{j=0}^{m-1} \omega_j(t)(D^j y_i)(t) + (D^m y_i)(t)]^2. \tag{19.13}$$

In the event that there is no theoretical model at hand, we may use $\omega_j = 0$, so that the comparison is simply with the sum of squares of the $D^m y_i$. From these loss functions, we may examine the point-wise squared multiple correlation function

$$\texttt{RSQ}(t) = \frac{\texttt{PSSE}_0(t) - \texttt{PSSE}_L(t)}{\texttt{PSSE}_0(t)} \tag{19.14}$$

and the point-wise F-ratio

$$\texttt{FRATIO}(t) = \frac{(\texttt{PSSE}_0(t) - \texttt{PSSE}_L(t))/m}{\texttt{PSSE}_0(t)/(N-m)}. \tag{19.15}$$

19.6 Comparing PDA and PCA

19.6.1 PDA and PCA both minimize sums of squared errors

Once we have found the operator L, we can in general define m linearly independent functions $\xi_1, \ldots, \xi_m$ that span the null space of L, so that any function x that satisfies $Lx = 0$ can be expressed precisely as a linear combination of the ξ_j. This means that the functions ξ_j form a basis for this space of solutions. Just how we compute such a basis is taken up in Chapter 18.

Let us assume that we have a sample of N observed functions x_i, where $N = 1$ is allowed. If these functions are not necessarily solutions to (19.3), then we can quantify the extent to which they approach being solutions by the size of the forcing functions ϵ_i defined by

$$Lx_i = \epsilon_i.$$

An example of this idea was given in Figure 1.7 where we applied the harmonic acceleration operator to four temperature profiles and discovered that these forcing functions were substantially nonzero.

The algorithm that we used in Section 17.3 aimed, at each iteration, to find the operator L that minimized the integrated square of the residual function ϵ. Why? Because it used the concurrent functional linear model developed in Chapter 14 to minimize a measure of discrepancy between the derivative that acted as the dependent variable and the fit based on two lower order derivatives that acted as independent variables. In effect, this minimizes a sum of squares measure for the corresponding L operator.

Consequently, we have a technique for choosing L so as to make the Lx_i as small as possible. If the technique is successful, then the residual functions will be small relative to the highest order of derivative. We should then expect to obtain a good approximation of the x_i by expanding them in terms of the ξ_j that span the subspace defined by the corresponding differential equation.

This is closely reminiscent of PCA, where the first m principal component functions ξ_j also define an m-dimensional subspace for approximating the given data.

19.6.2 PDA and PCA both involve finding linear operators

We can pursue the comparison between PCA and PDA further by noting that PCA can also be considered to involve the identification of a linear operator, which we can denote by Q, such that the equation $Qx_i = 0$ is solved as nearly as possible. To see this, recall from Chapter 8 that the goal of functional PCA is to find a set of m basis functions ξ_j such that the least squares criterion

$$\texttt{SSE}_{\text{PCA}} = \sum_{i=1}^{N} \int [x_i(t) - \sum_{j=1}^{m} f_{ij}\xi_j(t)]^2 \, dt \tag{19.16}$$

is minimized with respect both to the basis functions ξ_j and with respect to the coefficients of the expansions of each observed principal component score f_{ij}.

Because the fitting criterion (19.16) is least squares, we can think of PCA as a two-stage process: First identify a set of m orthonormal basis functions ξ_j, and then approximate any specific curve x_i by $\hat{x}_i = \sum_{j=1}^{m} f_{ij}\xi_j$. This second basis expansion step is the projection of each of the observed

functions onto the m-dimensional space spanned by the basis functions ξ_j, and takes place after having first identified the optimal basis for these expansions. Thus $\hat{x}_i$ is the image of x_i resulting from applying a least squares fit.

Suppose we indicate this projection as P_ξ, with the subscript indicating that the nature of the projection depends on the basis functions ξ_j. That is, $P_\xi x_i = \hat{x}_i$.

Associated with the projection P_ξ is the complementary projection

$$Q_\xi = I - P_\xi,$$

which produces as its result the residuals

$$Q_\xi x_i = x_i - P_\xi x_i = x_i - \hat{x}_i.$$

Using these projection operators, we can alternatively and equivalently define the PCA problem in a way that is much more analogous to the problem of identifying the linear differential operator L: In PCA, one seeks a projection operator Q_ξ such that the residual sum of squares

$$\mathtt{SSE}_{\mathrm{PCA}} = \sum_{i=1}^{N} \int [Q_\xi x_i(t)]^2 \, dt \tag{19.17}$$

is minimized. Indeed, one might think of the first m eigenfunctions as the functional *parameters* defining the projection operator Q_ξ, just as the weight functions β are the functional parameters defining L in PDA. These eigenfunctions, and any linear combinations of them, exactly satisfy the equation $Q_\xi \xi_j = 0$, just as the m functions ξ_j referred to above exactly satisfy the equation $L_\beta \xi_j = 0$, where we now add the subscript β to L to remind ourselves that L is defined by the vector $\boldsymbol{\beta}$ containing the m weight functions β_j.

Principal differential analysis is defined, therefore, as the identification of the differential operator L_β that minimizes least squares criterion $\mathtt{SSE}_{\mathrm{PDA}}$; principal components analysis is defined as the identification of the projection operator Q_ξ that minimizes the least squares criterion $\mathtt{SSE}_{\mathrm{PCA}}$. Both operators are linear.

19.6.3 Differences between differential operators (PDA) and projection operators (PCA)

Since the basic structures of the least squares criteria (19.17) and (19.4) are the same, clearly the only difference between the two criteria is in terms of the actions represented by the two operators L_β and Q_ξ. Since $Q_\xi x$ is in the same vector space as x, the definition of the operator identification problem as the minimization of $\|Q_\xi x\|^2$ is also in the same space, in the sense that we measure the performance of Q_ξ in the same space as the functions x to which it is applied.

On the other hand, L_β is a roughening transform in the sense that $L_\beta x$ has m fewer derivatives than x and is usually more variable. We may want to either penalize or otherwise manipulate x at this rough level.

Put another way, it may be plausible to conjecture that the noise or unwanted variational component in x is found only at the rough level $L_\beta x$. Thus, a second motivating factor for the use of L_β rather than Q_ξ is that PDA process explicitly takes account of the smoothness of the data by first roughening the data before minimizing error, while PCA does not.

Once we have found the operator L, we can in general define m linearly independent functions $u_1, \ldots, u_m$ that span the null space of L, and so any function x that satisfies $Lx = 0$ can be expressed precisely as a linear combination of the u_j. Hence, since L has been chosen to make the Lx_i as small as possible, we would expect to obtain a good approximation of the x_i by expanding them in terms of the u_j. This is closely reminiscent of PCA, where the first m principal component functions ξ_j form a good m-dimensional set for approximating the given data. The spirit of the approximation is rather different, however.

We can pursue the comparison between PCA and PDA by noting that PCA can also be considered to involve the identification of a linear operator, which we can denote by Q, such that the equation $Qx_i = 0$ is solved as nearly as possible. To see this, recall from Chapter 8 that one method of defining functional PCA is to propose to find a set of m basis functions ξ_j such that the least squares criterion

$$\mathtt{SSE}_{\mathrm{PCA}} = \sum_{i=1}^{N} \int [x_i(t) - \sum_{j=1}^{m} f_{ij}\xi_j(t)]^2 \, dt \tag{19.18}$$

with respect both to the basis functions ξ_j and with respect to the coefficients of the expansions of each observed function, f_{ij}. Because the fitting criterion (19.18) is least squares, we can think of PCA as a two-stage process: first identify a set of m orthonormal basis functions ξ_j, and then approximate any specific curve x_i by $\hat{x}_i = \sum_{j=1}^{m} f_{ij}\xi_j$. This second basis expansion step is the projection of each of the observed functions onto the m-dimensional space spanned by the basis functions ξ, and takes place after having first identified the optimal basis for these expansions. Thus $\hat{x}_i$ is the image of x_i resulting from applying a least squares fit.

Suppose we indicate this projection as P_ξ, with the subscript indicating that the nature of the projection depends on the basis functions ξ_j. That is, $P_\xi x_i = \hat{x}_i$. Associated with the projection P_ξ is the complementary projection

$$Q_\xi = I - P_\xi,$$

which produces as its result the residuals

$$Q_\xi x_i = x_i - P_\xi x_i = x_i - \hat{x}_i.$$

Using this concept, we can alternatively and equivalently define the PCA problem in a way that is much more analogous to the problem of identifying the linear differential operator L: In PCA, one seeks a projection operator Q_ξ such that the residual sum of squares

$$\mathtt{SSE}_{\mathrm{PCA}} = \sum_{i=1}^{N} \int [Q_\xi x_i(t)]^2 \, dt \tag{19.19}$$

is minimized. Indeed, one might think of the first m eigenfunctions as the functional *parameters* defining the projection operator Q_ξ, just as the weight functions w are the functional parameters of the LDO L in PDA. These eigenfunctions, and any linear combinations of them, exactly satisfy the equation $Q_\xi \xi_j = 0$, just as the m functions u_j referred to above exactly satisfy the equation $L_w u_j = 0$, where we now add the subscript w to L to remind ourselves that L is defined by the vector w of m weight functions β_j.

Principal differential analysis is defined, therefore, as the identification of the operator L_w that minimizes least squares criterion $\mathtt{SSE}_{\mathrm{PDA}}$, just as we can define PCA as the identification the projection operator Q_ξ that minimizes the least squares criterion $\mathtt{SSE}_{\mathrm{PCA}}$.

Since the basic structures of the least squares criteria (19.19) and (19.4) are the same, clearly the only difference between the two criteria is in terms of the actions represented by the two operators L_w and Q_ξ. Since $Q_\xi x$ is in the same vector space as x, the definition of the operator identification problem as the minimization of $\|Q_\xi x\|^2$ is also in the same space, in the sense that we measure the performance of Q_ξ in the same space as the functions x to which it is applied.

On the other hand, L_w is a roughening transform in the sense that $L_w x$ has m fewer derivatives than x and is usually more variable. We may want to either penalize or otherwise manipulate x at this rough level. Put another way, it may be plausible to conjecture that the noise or unwanted variational component in x is found only at the rough level $L_w x$. Thus, a second motivating factor for the use of L_w rather than Q_ξ is that PDA process explicitly takes account of the smoothness of the data by first roughening the data before minimizing error, while PCA does not.

As an example, imagine that we are analyzing the trajectories x_i of several rockets of the same type launched successively from some site. We observe that not all trajectories are identical, and we conjecture that some random process is at work that contributes variability to our observations. Naively, we might look for that variability in the trajectories themselves, but our friends in physics will be quick to point out that, first, the major source of variability is probably in the propulsion system, and second since the force that it applies is proportional to acceleration, we ought to study the acceleration $D^2 x_i$ instead. That is, if the function x_i is the trajectory along a specific coordinate axis (straight up, for example), the systematic

or errorless trajectory should obey the law

$$f_i(t) = M(t)D^2x_i(t),$$

where $M(t)$ is the mass of the rocket at time t. Alternatively,

$$-f_i/M + D^2x_i = 0.$$

Taking a more empirical approach, however, we agree on the compromise of looking for a second order linear differential equation

$$Lx = \beta_0 x + \beta_1 Dx + D^2x$$

and, if our friends in physics are right, the systematic or errorless component in the data should yield

$$\beta_0 x_i = -f_i/M \text{ and } \beta_1 = 0.$$

What we do understand, in any case, is that the sources of variability are likely to be at the rough level D^2x_i, rather than at the raw trajectory level x_i.

Returning to the lip position curves, we might reason that variation in lip position from curve to curve is due to variation in the forces resulting from muscle contraction, and that these forces have a direct or proportional impact on the acceleration of the lip tissue, and thus only indirectly on position itself. Position is two derivatives away from the action, in short.

More generally, an important motivation for finding the operator L_w is substantive: Applications in the physical sciences, engineering, biology and elsewhere often make extensive use of differential equation models of the form

$$Lx_i = f_i.$$

The result f_i is often called a *forcing or impulse function*, and in physical science and engineering applications is often taken to indicate the influence of exogenous agents on the system defined by $Lx = 0$.

Section 19.5 presents techniques for principal differential analysis, along with some measures of fit to the data. We also take up the possibility of regularizing or smoothing the estimated weight functions β_j.

19.7 Further readings and notes

Viele (2001) also analyzed the pinch force data with an alternative strategy for testing whether the model (19.8) adequately fits the data.

20

Green's functions and reproducing kernels

20.1 Introduction

We now introduce two concepts that are useful for both computation and theory. Green's functions are important because they permit the solution for a nonhomogeneous linear differential equation $Lx = u$ to be explicitly represented and calculated, no matter what the forcing function u. Well, this is a slight overstatement, since what we mean is that the explicit solution is available provided that we know the solution to the corresponding homogeneous equation $Lx = 0$. But it is often the case that we do, and even if we only have available an approximation to the homogeneous solution, it can still be the case that we want to compute solutions for a wide range of forcing functions. Green's functions can, therefore, make a real difference in applications.

A reproducing kernel is a somewhat more theoretical concept, but many texts, such as Gu (2002) and Wahba (1990) use the notion freely, and one often encounters the term *reproducing kernel Hilbert space* in the literature using spline functions. In fact, the term has a standard abbreviation, namely *RKHS*. So it can be useful to know what it means. We try in this chapter to demystify reproducing kernels by showing their relationship to Green's functions.

In Chapter 21 we will use both Green's functions and reproducing kernels to develop new designer bases associated with any specific choice of linear differential operator L. These bases will, like B-splines, be nonzero only over

a small number of adjacent intervals. That is, they have band-structured coefficient matrices, and permit smoothing in order n operations.

20.2 The Green's function for solving a linear differential equation

It often happens in engineering and science that the researcher has a homogeneous linear differential equation corresponding to a linear differential operator L that has either been worked out using fundamental principles or determined empirically. This equation describes the internal or endogenous dynamics of some system. What he or she wants to know, however, is that the consequences will be of adding an external or exogenous influence u to the system, and there are a wide variety of these potentially available. For example, a rocket may be a well-understood system as long as it is on the launching pad, but what will happen when it is in flight under the influence of atmospheric turbulence?

That is, what we want to know is the solution of the nonhomogeneous equation

$$Lx = u \tag{20.1}$$

for known L but arbitrary u? In effect, we want to reverse the effect of applying operator L because we have a *forcing function* u and we want to find x.

Of course, we recognize that the solution is not unique; if we add to any solution x some linear combination of the functions $\xi_j \in \ker L$ that span the null space, $\ker L$, of L, then this function also satisfies the equation.

But let us assume that the investigator has a set of known conditions defining a constraint operator B, and that these satisfy the complementarity condition (18.31). Typically, these will be initial value conditions describing, for example, the status of the rocket on the launching pad. Let m be the order of the equation. Then these constraints will be in the form

$$Bx = \mathbf{b} \tag{20.2}$$

for some known fixed m-vector $\mathbf{b}$.

Define the matrix $\mathbf{A}$ as the result of applying constraint operator B to each of the ξ_j's in turn:

$$\mathbf{A} = B\boldsymbol{\xi}', \tag{20.3}$$

so that the element in row i and column j of $\mathbf{A}$ is the ith element of vector $B\xi_j$. Since every ξ in $\ker L$ can be written as

$$\xi(t) = \sum_j c_j \xi_j(t) = \boldsymbol{\xi}'\mathbf{c}$$

for an m-vector of coefficients $\mathbf{c}$, then by the definition of $\mathbf{A}$ we have that

$$B\xi = \mathbf{b} = \mathbf{A}\mathbf{c}.$$

The conditions we have specified ensure that $\mathbf{A}$ is invertible, and consequently we have that

$$\mathbf{c} = \mathbf{A}^{-1}\mathbf{b}.$$

Now suppose that ν satisfies $L\nu = u$ and also $B\nu = 0$. That is, $\nu \in \ker B$, and in this sense is the complement of $\xi \in \ker L$. Then

$$x(t) = \xi(t) + \nu(t)$$

satisfies

$$Lx = u \ \text{ subject to } \ Bx = \mathbf{b}.$$

Consequently, if we can solve the problem

$$L\nu = u \ \text{ subject to } \ \nu \in \ker B, \tag{20.4}$$

we can find a solution subject to the more general constraint $Bx = \mathbf{b}$.

20.2.1 The definition of the Green's function

It can be shown that there exists a bivariate function $G(t;s)$ called the *Green's function*, associated with the pair of operators (B, L) that satisfies

$$\nu(t) = \int G(t;s)Lx(s)\,ds \ \text{ for } \ \nu \in \ker B. \tag{20.5}$$

Thus, for $L\nu = u$, the Green's function defines an integral transform

$$\mathcal{G}u = \int G(t;s)u(s)\,ds \tag{20.6}$$

that inverts the linear differential operator L. That is, $\mathcal{G}L\nu = \nu$, given that $B\nu = 0$.

Before giving a general recipe for computing the Green's function G, let's look at a few specific examples. The first is nearly trivial: If our interval is $[0, T]$ and our constraint operator is the initial value constraint $B_0 x = x(0)$, then for $L = D$,

$$G(t;s) = 1, s \le t, \text{ and 0 otherwise.}$$

That is, for ν such that $\nu(0) = 0$,

$$\nu(t) = \int_0^t D\nu(s)\,ds = \int_0^t u(s)\,ds.$$

Now consider the first order constant coefficient equation (18.1). Looking at the solution (18.3) for $\alpha(t) = 1$, we see by inspection that

$$G(t;s) = e^{-\beta(t-s)}, s \le t, \text{ and 0 otherwise.}$$

Progressing from this situation to the variable coefficient version (18.5) is now easy:

$$G(t:s) = \xi(t)/\xi(s), s \leq t, \text{ and 0 otherwise.}$$

20.2.2 A matrix analogue of the Green's function

Readers of this book may be familiar enough with matrix algebra to welcome a closely related concept in that domain. Suppose that we have, for $n > m$ an $n - m$ by n matrix $\mathbf{L}$ of rank $n - m$. If n is very large, then we approach the functional situation where $n \to \infty$.

Then there exists a subspace of n-vectors $\boldsymbol{\xi} \in \ker \mathbf{L}$ such that

$$\mathbf{L}\boldsymbol{\xi} = 0,$$

and that space is of dimension m. This means that we can construct a n by m matrix $\mathbf{Z}$ whose columns span this subspace such that $\mathbf{LZ} = 0$.

Also, we can always find an m by n matrix $\mathbf{B}$ of rank m such that there exists a space of dimension m of n vectors $\boldsymbol{\nu}$ such that

$$\mathbf{B}\boldsymbol{\nu} = 0 \ ;$$

and, moreover, such that the only vector $\mathbf{x}$ satisfying simultaneously $\mathbf{Lx} = 0$ and $\mathbf{Bx} = 0$ is $\mathbf{x} = 0$. For example, one way to compute such a matrix $\mathbf{B}$ is through the singular value decomposition of $\mathbf{L}$, but there are many other ways in which to define $\mathbf{B}$, which is not uniquely defined, just as the defining conditions and operator B for differential equations are not unique.

Corresponding to a particular choice of $\mathbf{B}$, we can find an n by $n - m$ matrix $\mathbf{N}$ such that $\mathbf{BN} = 0$.

Now suppose that we have an arbitrary n-vector $\mathbf{u}$. Then it follows that

$$\boldsymbol{\nu} = \mathbf{N}(\mathbf{LN})^{-1}\mathbf{u} \tag{20.7}$$

solves the equation

$$\mathbf{L}\boldsymbol{\nu} = \mathbf{u}$$

and, moreover, $\boldsymbol{\nu} \in \ker \mathbf{B}$ since $\mathbf{BN} = 0$. Matrix

$$\mathbf{G} = \mathbf{N}(\mathbf{LN})^{-1} \tag{20.8}$$

is the analogue of the Green's function $G(s;t)$.

A special choice of $\mathbf{B}$ leads to an interesting result. Let $\mathbf{B}$ be chosen so that $\mathbf{N} = \mathbf{L}'$. In that case, $\mathbf{G} = \mathbf{L}'(\mathbf{LL}')^{-1}$ and $\mathbf{G}'$ is the pseudo-inverse of $\mathbf{L}$.

20.2.3 A recipe for the Green's function

We can now offer a recipe for constructing the Green's function for any linear differential operator L of the form (18.12) and the initial value constraint B_I of the corresponding order. First, compute the Wronskian matrix

$\mathbf{W}(t)$ defined in (18.26). Secondly, define the functions

$$\mathbf{v}(t) = (v_1(t), \ldots, v_m(t))'$$

to be the vector containing the elements of the last row of $\mathbf{W}^{-1}$. Then, it turns out that initial value constraint Green's function $G_0(t; s)$ is

$$G_0(t; s) = \sum_{j=1}^{m} \xi_j(t) v_j(s) = \boldsymbol{\xi}(t)' \mathbf{v}(s), s \leq t, \text{ and 0 otherwise.} \tag{20.9}$$

Let's see how this works for

$$L = \beta D + D^2.$$

The space $\ker L$ is spanned by the two functions $\xi_1(t) = 1$ and $\xi_2(t) = \exp(-\beta t)$. The Wronskian matrix is

$$\mathbf{W}(t) = \begin{bmatrix} \xi_1(t) & D\xi_1(t) \\ \xi_2(t) & D\xi_2(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \exp(-\beta t) & -\beta \exp(-\beta t) \end{bmatrix}$$

and consequently

$$\mathbf{W}^{-1}(t) = \begin{bmatrix} 1 & 0 \\ \beta^{-1} & -\beta^{-1} \exp(\beta t) \end{bmatrix},$$

from which we have

$$\mathbf{v}(s) = -\beta^{-1}[-1, \exp(\beta s)]'$$

and finally

$$G_0(t; s) = -\beta^{-1}[e^{-\beta(t-s)} - 1], s \leq t, \text{ and 0 otherwise.} \tag{20.10}$$

We can verify that this is the required Green's function by integration by parts.

We do not discuss in any detail the case of any constraint functions B other than initial value constraints. Under boundary or periodic constraints, it may be that additional conditions are required on the function f or on the constraint values c, but nevertheless we can extend the basic ideas of Green's functions.

20.3 Reproducing kernels and Green's functions

A bivariate function called the *reproducing kernel* plays a central role in the theory of spline functions, and we will use reproducing kernels in Chapter 21 to define a basis function system $\boldsymbol{\phi}$ specific to any linear differential operator L used to define a roughness penalty.

20.3.1 What is a reproducing kernel?

We remarked in Section 18.6.3 that the concept of an *inner product* underlies perhaps about 95% of all applied mathematics and statistics. There is no possibility here of doing more than recalling the most basic elements of inner product spaces, and perhaps no particular need, either.

A *Hilbert space* is a collection of objects x for which there exists:

- linear combinations $ax_1 + bx_2$,
- an inner product $\langle x_1, x_2 \rangle$ for any pair x_1 and x_2
- a property called *completeness*, namely that convergent sequences of elements converge to elements within the space.

Both vectors and functions as used in applied work are typically elements of Hilbert spaces, and Section 18.6.3 gave some functional examples of useful inner products.

There is a sense, however, in which the Hilbert space is too loose a concept. This revolves around the linear mapping

$$\rho_t(x) = x(t),$$

which we called the *evaluation mapping* in Section 5.5. If a function x is smooth, we imagine that knowing $x(t)$ tells us a great deal about $x(t + \delta)$ when perturbation δ is sufficiently small. Unfortunately, such need not be the case for Hilbert spaces in general.

Consequently, we need to focus on the more specialized Hilbert space for which the evaluation map is *continuous*. It would be nice to imagine that these would be called something like smooth Hilbert spaces, or continuous Hilbert spaces, but alas, mathematics does not tend to generate its nomenclature in such a kindly way! Instead, spaces of this nature are called *reproducing kernel Hilbert spaces*, not surprisingly often abbreviated to *RKHS*s.

It is a basic theorem in functional analysis, called the *Riesz representation theorem*, that if a linear mapping $\rho(x)$ in a Hilbert space is continuous, then there exists a function k in the space such that

$$\rho(x) = \langle x, k \rangle \ .$$

Consequently, applying this idea to the evaluation map $\rho_t(x)$, there must exist a *bivariate* function $k(s, t)$ such that $k(\cdot, t)$ is in the space for any t, and that

$$\rho_t(x) = \langle x, k(\cdot, t) \rangle \ .$$

The term *reproducing kernel* comes from the consequence that

$$k(s, t) = \langle k(\cdot, s), k(\cdot, t) \rangle \ .$$

The existence of $k(s,t)$ has many wide-ranging consequences, and plays an especially important role in the history of the development of spline smoothing.

So, given that we have a Hilbert space with a continuous evaluation map, how do we find the reproducing kernel? The surprising result is: If you know the Green's function for the linear differential operator that defines inner products of the type described in Section 18.6.3, you are almost there!

There are two reproducing kernels $k(s,t)$ that we need to consider, one for each of the function subspaces $\ker B$ and $\ker L$. We now show how these can be calculated, and we will put them to work in Chapter 21.

20.3.2 The reproducing kernel for $\ker B$

The reproducing kernel for the $\ker B$ subspace, consisting of functions that satisfy $Bx = 0$, has a simple relationship to the Green's function G. First, however, we need to explain what a reproducing kernel is in this context.

Given any two functions x and y in $\ker B$, let us define the L-inner product

$$\langle x, y\rangle_L = \langle Lx, Ly\rangle = \int Lx(s)Ly(s)\,ds.$$

Let G_I be the Green's function as defined in Section 20.2.3, and define a function $k_2(t,s)$ such that, for all t,

$$Lk_2(t,\cdot) = G_I(t;\cdot) \text{ and } Bk_2(t,\cdot) = 0. \tag{20.11}$$

By the defining properties of Green's functions, this means that

$$k_2(t,s) = \int G_I(s;w)G_I(t;w)\,dw. \tag{20.12}$$

The function k_2 has an interesting property. Suppose that ν is any function in $\ker B$, and consider the L-inner product of $k_2(t,\cdot)$ and ν. We have, for all t,

$$\langle k_2(t,\cdot),\nu\rangle_L = \int Lk_2(t,s)L\nu(s)\,ds = \int G_I(t;s)L\nu(s)\,ds = \nu(t) \tag{20.13}$$

by the key property (20.5) of Green's functions. Thus, in the space $\ker B$ equipped with the L-inner product, taking the L-inner product of k_2 using its second argument with any function ν yields the value of ν at its first argument. Overall, taking the inner product with k_2 reproduces the function ν, and k_2 is called the reproducing kernel for this function space and inner product.

Chapter 21 shows that the reproducing kernel is the key to the important question, "Is there an optimal set of basis functions for smoothing data?" To answer this question, we need to use the important property

$$\langle k_2(s,\cdot), k_2(t,\cdot)\rangle_L = k_2(s,t), \tag{20.14}$$

which follows at once from (20.13) setting $\nu(\cdot) = k_2(s, \cdot)$ and appealing to the symmetry of the inner product.

We can put the expression (20.12) in a slightly more convenient form for the purpose of calculation. Recalling the definitions of the vector-valued functions $\boldsymbol{\xi}$ and $\mathbf{v}$ in Section 20.2.3, we have from (20.9), assuming that $s \leq t$, that

$$k_2(s,t) = \int_0^s [u(s)'v(w)][v(w)'u(t)]\, dw = u(s)'\mathbf{F}(s)u(t), \tag{20.15}$$

where the order m symmetric matrix-valued function $\mathbf{F}(s)$ is

$$\mathbf{F}(s) = \int_0^s v(w)v(w)'\, dw. \tag{20.16}$$

To deal with the case $s > t$, we use the property that $k_2(s,t) = k_2(t,s)$.

The matrix analogue of the reproducing kernel $k_2(s,t)$ is

$$\mathbf{K}_2 = \mathbf{G}\mathbf{G}',$$

since we see that for any $\nu \in \ker B$

$$\begin{aligned} \mathbf{K}_2 L'L\nu &= \mathbf{N}(\mathbf{L}\mathbf{N})^{-1}(\mathbf{N}'\mathbf{L}')^{-1}\mathbf{N}'\mathbf{L}'\mathbf{L}\nu \\ &= \mathbf{N}(\mathbf{L}\mathbf{N})^{-1}\mathbf{L}\nu \\ &= \nu \end{aligned} \tag{20.17}$$

as required.

20.3.3 The reproducing kernel for $\ker L$

Suppose now that $f = \sum a_i\xi_i$ and $g = \sum b_i\xi_i$ are elements of $\ker L$. We can consider the B-inner product on the finite-dimensional space $\ker L$, defined by

$$\langle f, g\rangle_B = (Bf)'Bg = a'\mathbf{A}'\mathbf{A}b.$$

Define a function $k_1(t,s)$ by

$$k_1(t,s) = \boldsymbol{\xi}(t)'(\mathbf{A}'\mathbf{A})^{-1}u(s).$$

It is now easy to verify that, for any $f = \sum_i a_i\xi_i$,

$$\langle k_1(t,\cdot), f\rangle_B = \boldsymbol{\xi}(t)'(\mathbf{A}'\mathbf{A})^{-1}\mathbf{A}'\mathbf{A}a = \boldsymbol{\xi}(t)'a = a'\boldsymbol{\xi}(t) = f(t).$$

So k_1 is the reproducing kernel for the space $\ker L$ equipped with the B-inner product.

Finally, we consider the space of more general functions x equipped with the inner product $\langle\cdot,\cdot\rangle_{B,L}$ as defined in Section 18.6.3. It is easy to check from the properties we have set out that the reproducing kernel in this space is given by

$$k(s,t) = k_1(s,t) + k_2(s,t).$$

20.4 Further reading and notes

Although the theory of reproducing kernel Hilbert spaces is considered to be of relatively recent origin, and usually attributable to Aronszajn (1950), it is in fact grounded in the theory of Green's functions, a topic older by more than a century (Green, 1828). The concept of a reproducing kernel appears in most of the papers by G. Wahba, including Wahba (1990). More recently, reproducing kernels are used extensively in Gu (2002). The interested and highly motivated reader might want to consult a reference on functional analysis with an applied orientation, such as Aubin (2000).

21
More general roughness penalties

21.1 Introduction

A theme central to this book has been the use of roughness penalties to incorporate smoothing, whether in the context of using discrete data to define a smooth function in Chapter 5, functional principal components analysis in Chapter 9, or imposing regularity on estimated regression functions in the chapters on the functional linear model.

At the same time, the previous three chapters have dealt with the mathematical properties of linear differential operators L and with techniques for estimating them from data. Principal differential analysis provides a method of estimating low-dimensional functional variation in a sense analogous to principal components analysis, but by estimating an mth order differential operator L rather than a projection.

Moreover, we have seen that by coupling L with a suitable set of constraints on the m linearly independent functions ξ_j satisfying $L\xi_j = 0$, we can *partition* the space of smooth functions into two parts. This is achieved by defining a constraint operator B such that $B\xi_j \neq 0$, and the only function satisfying $Bx = Lx = 0$ is $x = 0$. Then any function x having m derivatives can be expressed uniquely as

$$x = \xi + e \text{ where } L\xi = 0 \text{ and } Be = 0. \tag{21.1}$$

We might call this the *partitioning principle*.

It is time to put these two powerful ideas together, to see what practical value there is in using the partitioning principle to define a roughness

penalty. We want to go beyond the standard practice of defining roughness in terms of $L = D^2$, and even beyond the slightly more general $L = D^m$, to consider what the advantages might be of using an arbitrary operator L, perhaps in conjunction with some constraints captured in the companion operator B. Specifically, when the goal is smoothing the data, we propose using the criterion

$$\texttt{PENSSE}(x) = \sum_j^n [y_j - x(t_j)]^2 + \lambda \times \texttt{PEN}_L(x), \tag{21.2}$$

where

$$\texttt{PEN}_L(x) = \int (Lx)^2(t)\, dt.$$

We begin with some examples.

21.1.1 The lip movement data

Consider the lip movement data introduced in Chapter 19 and plotted in Figure 21.1. We are interested in how these trajectories, all based on observations of a speaker saying "bob," vary from one replication to another. But in the experiment, the syllable was embedded in the phrase, "Say bob again," and it is clear that the lower lip enters and leaves the period during which the syllable is being formed at different heights. This is nuisance variation that we would be happy to eliminate.

Moreover, there was particular interest in the acceleration or second derivative of the lip, suggesting that we should penalize the fourth derivative by spline smoothing with $L = D^4$. Any cubic polynomial trend in the records is ignored if we do that. Now we want to define the *shape* component u and *endpoint* component ξ of each record x in such a way that the behavior of the record at the beginning and end of the interval of observation (normalized to be [0,1]) has minimal impact on the interior and more interesting portion of the curve. One way of achieving this objective is to require the shape components to satisfy the constraints

$$u(0) = Du(0) = 0 \text{ and } u(1) = Du(1) = 0.$$

This means that the constraint is defined by the boundary constraint operator B_B, defined as

$$B_B x = \begin{bmatrix} x(0) \\ Dx(0) \\ x(1) \\ Dx(1) \end{bmatrix}, \tag{21.3}$$

and the shape component u satisfies $B_B u = 0$.

We now have our two linear operators $L = D^4$ and $B = B_B$ in hand, and they are complementary in the sense that $\ker B \cap \ker L = 0$. That is,

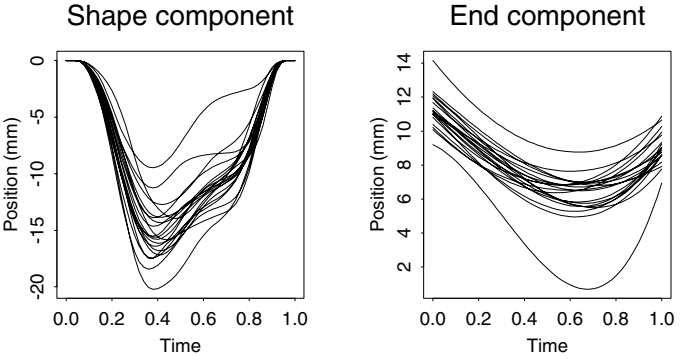

Figure 21.1. The right panel displays the 20 cubic polynomials ξ that match the lip position and derivative values at 0 and 1 for the smoothed versions of the curves in Figure 19.1. The left panel shows the shape components u that have zero endpoint positions and derivatives.

we have now unambiguously split any lip position record x into $x = \xi + u$, where $Bu = 0$, and ξ, a cubic polynomial because $L\xi = D^4\xi = 0$, picks up the endpoint variation by fitting the record's function and derivative values at both 0 and 1. Figure 21.1 displays the endpoint and shape components for all 20 records.

21.1.2 The weather data

We noted in the introduction that a rather large part of the mean daily or monthly temperature curve for any weather station can be captured by the simple function

$$T(t) = c_1 + c_2 \sin(\pi t/6) + c_3 \cos(\pi t/6) \tag{21.4}$$

and the same may be said for the log precipitation profiles. Functions of this form can be annihilated by the operator

$$L = (\pi/6)^2 D + D^3.$$

We could propose smoothing data using the criterion (21.2), where

$$\texttt{PEN}_L(x) = \int (Lx)^2(t)\, dt = \int [(\pi/6)^2 Dx(t) + D^3 x(t]^2(t)\, dt,$$

while paying attention to the periodic character of the data. What would we gain from this? For one thing, as we have already noted in Section 18.4.3, this procedure is likely to have considerable advantages in the estimation of curves x from raw data.

At the same time, the function L`Temp` in this example is interesting in itself, and Ramsay and Dalzell (1991) refer to this as the *harmonic acceleration* of temperature. They show by functional principal components

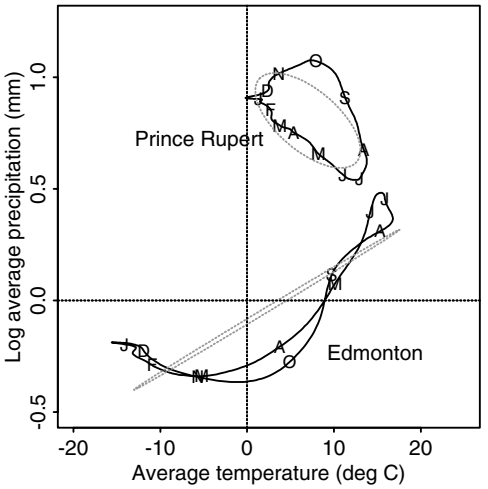

Figure 21.2. The solid cycles are the smoothed daily temperature and log precipitation data, plotted against each other, for two Canadian weather stations. The dotted curves are the estimated cycles based on strictly sinusoidal variation, taking the first three terms of the Fourier expansion of each observed temperature and log precipitation curve. Letters indicate the middle of each month.

and linear regression analyses that $L\texttt{Temp}$, and the harmonic acceleration of log precipitation, contain a great deal of information about the peculiarities of weather at any station. In order to identify the component e uniquely, though, we must choose a matching integral constraint operator B_I, and for this application they chose

$$B_I x = \begin{bmatrix} \int x(t)\,dt \\ \int x(t)\sin(\pi t/6)\,dt \\ \int x(t)\cos(\pi t/6)\,dt \end{bmatrix},$$

corresponding to the first three Fourier coefficients of the observed curves. The three functions ξ_i that span $\ker L$ are then 1, $\sin(\pi t/6)$ and $\cos(\pi t/6)$. Given any curve x, the partition (21.1) is achieved by setting the component $\boldsymbol{\xi}$ to be the first three terms in the Fourier expansion of x.

The solid curves in Figure 21.2 show, for two weather stations, plots of smoothed daily temperature against smoothed daily log precipitation through the year. The shifted sinusoidal components $\xi_j(t)$ for temperature and for log precipitation respectively become ellipses when plotted against each other and yield the dotted curves in the figure.

21.2 The optimal basis for spline smoothing

In Chapter 3 we reviewed the classic technique of representing functions by fitting a basis function expansion to the data. We took pains to point out that not all bases are equal: A good basis has basis functions which mimic the general features that we know apply to the data, such as periodicity, asymptotic linearity, and so on. When we get these features right, we can expect to do a good job with a smaller number of basis functions.

We also pointed out that when the number n of data points is large, computing an expansion in $O(n)$ operations is critical, and in order to achieve this, the basis functions should at least be nonzero only locally, or have compact support. The B-spline basis is especially attractive from this perspective.

In Section 5.6, we extended the basis function expansion concept to employ a partitioned basis (ϕ, ψ) along with a penalty on the size of the component expanded in terms of the basis functions ψ. But two properties, relevance to the data and convenience of computation, remain essential.

We now bring these elements together: Use the partitioning principle to define a set of basis functions that are optimal with respect to smoothing, provide a recipe for an $O(n)$ smoothing algorithm, and also show how these can be put into compact support form to give the appropriate analogue of B-splines. Further details are available in Heckman and Ramsay (2000).

We begin with a theorem that states that the optimal basis for spline smoothing in the context of operators (B, L) is defined by the reproducing kernel k_2 defined in Chapter 20.

Optimal Basis Theorem:

For any $\lambda > 0$, the function x minimizing the spline smoothing criterion (21.2) defined by a linear differential operator L of order m has the expansion

$$x(t) = \sum_{j=1}^{m} d_j \xi_j(t) + \sum_{i=1}^{n} c_i k_2(t_i, t). \tag{21.5}$$

Equation (21.5) can be put a bit more compactly. As before, let $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_m)'$; define another vector function

$$\tilde{k}(t) = \{k_2(t_1, t), k_2(t_2, t), \ldots, k_2(t_n, t)\}'.$$

Then the optimal basis theorem says that the function x has to be of the form $x = \mathbf{d}'\boldsymbol{\xi} + \mathbf{c}'\tilde{k}$, where $\mathbf{d}$ is a vector of m coefficients d_j and $\mathbf{c}$ is the corresponding vector of n coefficients c_i in (21.5). We give a proof of the optimal basis theorem, but as usual any reader prepared to take this on trust should simply skip to the next section.

Proof:

Suppose x^* is any function having square-integrable derivatives up to order m. The strategy for the proof is to construct a function $\tilde{x}$ of the form

(21.5) such that

$$\texttt{PENSSE}(\tilde{x}) \leq \texttt{PENSSE}(x^*)$$

with equality only if $\tilde{x} = x^*$. It then follows at once that we need never look beyond functions of the form (21.5) if we want to minimize the spline smoothing criterion PENSSE.

First of all, write $x^* = u^* + e^*$ where $u^* \in \ker L$ and $e^* \in \ker B$. Let $\mathcal{K}$ be the subspace of $\ker B$ spanned by the n functions $k_2(t_i, \cdot)$, and let $\tilde{e}$ be the projection of e^* onto $\mathcal{K}$ in the L-inner product. This means that $e^* = \tilde{e} + e^\perp$, where

$$\tilde{e} = c'\tilde{k}$$

for some vector c, and the residual $e^\perp$ in $\ker B$ satisfies the orthogonality relation

$$\langle e, e^\perp \rangle_L = \int (Le)(Le^\perp) = 0 \text{ for all } e \text{ in } \mathcal{K}. \tag{21.6}$$

We now define our function $\tilde{x} = u^* + \tilde{e}$, meaning that $\tilde{x}$ is necessarily of the required form (21.5), and $x^* - \tilde{x}$ is equal to the residual $e^\perp$.

To show that $\texttt{PENSSE}(\tilde{x}) \leq \texttt{PENSSE}(x^*)$, note first that, by the defining property of the reproducing kernel, for each i,

$$x^*(t_i) - \tilde{x}(t_i) = e^\perp(t_i) = \langle k_2(t_i, \cdot), e^\perp \rangle_L = 0$$

by property (21.6), since $k_2(t_i, \cdot)$ is of course a member of $\mathcal{K}$ and so is L-orthogonal to $e^\perp$.

Therefore x^* and $\tilde{x}$ agree at the arguments t_i, and so

$$\texttt{PENSSE}(x^*) - \texttt{PENSSE}(\tilde{x}) = \lambda\{\texttt{PEN}_L(x^*) - \texttt{PEN}_L(\tilde{x})\};$$

the residual sum of squares of the y_i is the same about each of the two functions x^* and $\tilde{x}$. Since $Lx^* = Le^*$ and $L\tilde{x} = L\tilde{e}$, we have

$$\begin{aligned} \texttt{PEN}_L(x^*) - \texttt{PEN}_L(\tilde{x}) &= \texttt{PEN}_L(e^*) - \texttt{PEN}_L(\tilde{e}) \\ &= \langle \tilde{e} + e^\perp, \tilde{e} + e^\perp \rangle_L - \langle \tilde{e}, \tilde{e} \rangle_L \\ &= \langle e^\perp, e^\perp \rangle_L + 2\langle \tilde{e}, e^\perp \rangle_L = \langle e^\perp, e^\perp \rangle_L \end{aligned}$$

since $\tilde{e}$ is in $\mathcal{K}$ and is therefore L-orthogonal to $e^\perp$. Therefore $\texttt{PEN}_L(e^*) \geq \texttt{PEN}_L(\tilde{e})$, and consequently $\texttt{PENSSE}(x^*) \geq \texttt{PENSSE}(\tilde{x})$. Equality holds only if $e^\perp \in \ker L$; since we already know that $e^\perp \in \ker B$, this implies that $e^\perp = 0$ and that $x^* = \tilde{x}$. This completes the proof of the theorem.

21.3 An $O(n)$ algorithm for L-spline smoothing

21.3.1 The need for a good algorithm

In principle, the optimal basis theorem should tell us exactly how to proceed. Since we know that the required function is of the form $x = d'u + c'\tilde{k}$,

we need only express $\texttt{PENSSE}(x)$ in terms of c and d and minimize to find the best values of c and d. How would this work out?

Let $\mathbf{K}$ be the matrix with values $k_2(t_i, t_j)$. From equation (20.14) it follows that

$$\texttt{PEN}_L(x) = \langle c'\tilde{k}, c'\tilde{k}\rangle_L = c'\mathbf{K}c.$$

The vector of values $x(t_i)$ is $\mathbf{U}d + \mathbf{K}c$, where $\mathbf{U}$ is the matrix with values $\xi_j(t_i)$. Hence, at least in principle, we can find x by minimizing the quadratic form

$$\texttt{PENSSE}(x) = (y - \mathbf{U}d - \mathbf{K}c)'(y - \mathbf{U}d - \mathbf{K}c) + \lambda c'\mathbf{K}c \tag{21.7}$$

to find the vectors c and d.

Unfortunately the matrix $\mathbf{K}$ is in practice usually extremely badly conditioned, that is to say, the ratio of its largest eigenvalue to its smallest explodes. A practical consequence of this is that the computations required to minimize the quadratic form (21.7) are likely to be unstable or impossible.

Furthermore, in smoothing long sequences of observations, it is critical to devise a smoothing procedure that requires a number of arithmetic operations that does not grow too quickly as the length of the sequence increases. For example, the handwriting data has $n = 1401$ and so an algorithm that was $O(n^2)$ would be impracticable and an $O(n^3)$ algorithm virtually impossible with current computing power. By adopting a somewhat different approach, we can set out an algorithm that requires only $O(n)$ operations, and furthermore avoids the numerical problems inherent in the direct minimization of (21.7).

The algorithm we use is based on the theoretical paper of Anselone and Laurent (1967), but is also known as the Reinsch algorithm because of the application to the cubic polynomial smoothing case ($L = D^2$) by Reinsch (1967, 1970). It was subsequently extended by Hutchison and de Hoog (1985). We do not attempt a full exposition of the rationale for this algorithm here, but Heckman and Ramsay (2000) and Ramsay, Heckman and Silverman (1997) can be consulted for details.

The algorithm requires the computation of values of two types of function that we have already encountered:

1. $\xi_j, j = 1, \ldots, m$: a set of m linearly independent functions satisfying $L\xi_j = 0$, that is, spanning $\ker L$. As before, we refer to these collectively as the vector-valued function $\boldsymbol{\xi}$.
2. k_2: the reproducing kernel function defined in Chapter 18 for the subspace of functions e satisfying $B_I e = 0$, where B_I is the initial value constraint operator.

The functions $\boldsymbol{\xi}$ and k_2 are the user-supplied components of the algorithm and are, of course, defined by the particular choice of operator L used in the smoothing application.

The algorithm splits into three phases:

1. an initial setup phase that does not depend on the smoothing parameter λ
2. a smoothing phase in which we smooth the data
3. a summary phase in which we compute performance measures for the smooth

This division of the task is of practical importance because we may want to try smoothing with many values of λ, and do not want to needlessly repeat either the initial setup phase or the final descriptive phase.

21.3.2 Setting up the smoothing procedure

In the initial phase, we define two symmetric $(n-m)\times(n-m)$ band-structured matrices $\mathbf{H}$ and $\mathbf{C}'\mathbf{C}$ where m is the order of operator L. Both matrices are band-structured with band width at most $2m+1$, which means that all entries more than m positions away from the main diagonal are zero. Because of symmetry, these band-structured matrices require only $(n-m)(m+1)$ storage locations.

We start by explaining how to construct the matrix $\mathbf{C}$. For each $i = 1,\ldots,n-m$, define the $(m+1)\times m$ matrix $\mathbf{U}^{(i)}$ to have (l,j) element $\xi_j(t_{i+l})$, for $l = 0,\ldots,m$. Thus $\mathbf{U}^{(i)}$ is the submatrix of $\mathbf{U}$ consisting only of rows $i, i+1,\ldots,i+l$. Find the QR decomposition (as discussed in Section A.3.3)

$$\mathbf{U}^{(i)} = \mathbf{Q}^{(i)}\mathbf{R}^{(i)},$$

where the matrix $\mathbf{Q}^{(i)}$ is square, of order $m+1$, and orthonormal, and where the matrix $\mathbf{R}^{(i)}$ is $(m+1)\times m$ and upper triangular. Let the vector $c^{(i)}$ be the last column of $\mathbf{Q}^{(i)}$; this vector is orthogonal to all the columns of $\mathbf{U}^{(i)}$. In fact any vector having this property will do, and in special cases the vector can be found by some other method. For polynomial spline smoothing, for instance, coefficients defining divided differences are used.

Now define the $n\times(n-m)$ matrix $\mathbf{C}$ so that its ith column has the $m+1$ values $c^{(i)}$ starting in row i; elsewhere the matrix contains zeroes. In practice, the argument sequence $t_1,\ldots,t_n$ is often equally spaced, and in this case it frequently happens that all the coefficient vectors $c^{(i)}$ are the same, and hence need be computed only once. The band structure of $\mathbf{C}$ immediately implies that $\mathbf{C}'\mathbf{C}$ has the required band structure, and can be found in $O(n)$ operations for fixed m.

The other setup-phase matrix $\mathbf{H}$ is the $(n-m)\times(n-m)$ symmetric matrix

$$\mathbf{H} = \mathbf{C}'\mathbf{K}\mathbf{C}, \tag{21.8}$$

where $\mathbf{K}$ is the matrix of values $k_2(t_i, t_j)$. It turns out that $\mathbf{H}$ is also band-structured with band width $2m-1$. This is a consequence of the expression (20.15) for the reproducing kernel, which yields the following two-part expression:

$$k_2(t_i, t) = \begin{cases} u(t_i)'\mathbf{F}(t)u(t) & \text{for } t_i \geq t \\ u(t_i)'\mathbf{F}(t_i)u(t) & \text{for } t_i \leq t, \end{cases} \tag{21.9}$$

for a certain matrix function $\mathbf{F}(t)$. This in turn implies that

$$\mathbf{K}_{ij} = \{\mathbf{U}\mathbf{F}(t_j)u(t_j)\}_i \quad \text{for } i \geq j. \tag{21.10}$$

Suppose $k \geq j$. Because $\mathbf{C}_{ik}$ is zero for $i < k$,

$$(\mathbf{C}'\mathbf{K})_{kj} = \sum_{i=k}^{n} \mathbf{C}_{ik}\mathbf{K}_{ij} = \sum_{i=k}^{n} \mathbf{C}_{ik}\{\mathbf{U}\mathbf{F}(t_j)u(t_j)\}_i,$$

substituting (21.10); notice that $i \geq j$ for all i within the range of summation $k \leq i \leq n$. It follows that for $k \geq j$ we have

$$(\mathbf{C}'\mathbf{K})_{kj} = \{\mathbf{C}'\mathbf{U}\mathbf{F}(t_j)u(t_j)\}_i = 0.$$

So $\mathbf{C}'\mathbf{K}$ is strictly upper-triangular. Because of the band structure of $\mathbf{C}$ this means that the matrix $\mathbf{H} = (\mathbf{C}'\mathbf{K})\mathbf{C}$ has zero entries for positions m or more below the main diagonal, and by symmetry $\mathbf{H}$ has the stated band structure.

21.3.3 The smoothing phase

The actual smoothing consists of two steps:

1. Compute the vector z, of length $n-m$, that solves

$$(\mathbf{H} + \lambda\mathbf{C}'\mathbf{C})z = \mathbf{C}'y, \tag{21.11}$$

   where the vector y contains the values to be smoothed.

2. Compute the vector of n values $\hat{y}_i = x(t_i)$ of the smoothing function x at the n argument values using

$$\hat{y} = y - \lambda\mathbf{C}z. \tag{21.12}$$

Because of the band structure of $(\mathbf{H} + \lambda\mathbf{C}'\mathbf{C})$ and of $\mathbf{C}$, both of these steps can be computed in $O(n)$ operators, and references on efficient matrix computation such as Golub and van Loan (1989) can be consulted for details.

21.3.4 The performance assessment phase

The vector of smoothed values $\hat{y}$ and the original values y that were smoothed are related as follows:

$$\hat{y} \quad = \quad y - \lambda\mathbf{C}(\mathbf{H} + \lambda\mathbf{C}'\mathbf{C})^{-1}\mathbf{C}'y$$

$$= \quad \{\mathbf{I} - \lambda\mathbf{C}(\mathbf{H} + \lambda\mathbf{C}'\mathbf{C})^{-1}\mathbf{C}'\}y. \tag{21.13}$$

The matrix $\mathbf{S}$ defined by

$$\mathbf{S} = \mathbf{I} - \lambda\mathbf{C}(\mathbf{H} + \lambda\mathbf{C}'\mathbf{C})^{-1}\mathbf{C}' \tag{21.14}$$

is often called the *hat matrix*, and in effect defines a linear transformation that maps the unsmoothed data into its smooth image by

$$\hat{y} = \mathbf{S}y.$$

Various measures of performance depend on the diagonal values in $\mathbf{S}$. Of these, the most popular are currently

$$\texttt{GCV} = \texttt{SSE}/(1 - n^{-1}\text{trace}\,\mathbf{S})^2, \tag{21.15}$$

where

$$\texttt{SSE} = \sum_{i=1}^{n}[y_i - x(t_i)]^2 = \|y - \hat{y}\|^2$$

and

$$\texttt{CV} = \sum_{i=1}^{n}[\{y_i - x(t_i)\}/\{1 - s_{ii}\}]^2, \tag{21.16}$$

where s_{ii} is the ith diagonal entry of $\mathbf{S}$. We can compute both measures GCV and CV in $O(n)$ operations given the band-structured nature of the matrices defining $\mathbf{S}$, using methods developed by Hutchison and de Hoog (1985).

One of the main applications of these two criteria, both of which are types of discounted error sums of squares, is as a guide for choosing the value of the smoothing parameter λ. It is relatively standard practice to look for the value that minimizes one of these two criteria, just as various variable selection procedures attempt to minimize discounted error sums of squares in standard regression analysis. Interestingly, the GCV measure was originally introduced by Craven and Wahba (1979) as an approximation to the CV criterion that could be computed in $O(n)$ operations; now CV is also available in n operations, but GCV still tends to be preferred in practice for other reasons. For example, various simulation studies have indicated that GCV tends to be a better basis for choosing the smoothing parameter λ, possibly because GCV makes use of smoothing itself by replacing the variable values $1 - s_{ii}$ by the average $1 - n^{-1}\text{trace}\,\mathbf{S}$.

Also of great value is a measure of the effective number of degrees of freedom of the smoothing operation. Two measures are

$$\texttt{DF}_1 = \text{trace}\,\mathbf{S} \text{ and } \texttt{DF}_2 = \text{trace}\,\mathbf{S}'\mathbf{S} = \text{trace}\,\mathbf{S}^2. \tag{21.17}$$

These dimensionality measures were introduced and discussed by Buja et al. (1989). It can be shown that in the limit as $\lambda \to \infty$, both measures become simply m, and similarly, as $\lambda \to 0$, both measures converge to

n. In between, they give slightly different impressions of how much of the variation in the original unsmoothed data remains in the smoothed version.

21.3.5 Other $O(n)$ algorithms

There is an intimate connection between the theory of splines and the theory of stochastic differential equations (Wahba, 1978, 1990, Weinert, Bird and Sidhu, 1980). This leads to the possibility of using the Kalman filter, a technique widely used in engineering and other fields to extract an estimate of a signal from noisy data, to compute a smoothing spline. Ansley, Kohn and Wong (1993), using a Kalman filtering algorithm described in Ansley and Kohn (1989), give some examples of computing an L-spline in $O(n)$ operations. However, except for fairly simple cases, this algorithm appears to be difficult to implement, and its description involves substantial mathematical detail. Nevertheless, we feel that it is important to call the reader's attention to this stimulating literature on smoothing by state-space methods.

21.4 A compact support basis for L-splines

In this section our concern is the construction of compact support basis functions from the reproducing kernel basis functions $k_2(t_i, \cdot)$. A basis made up of such functions may, for example, be useful for techniques such as the regularized principal components analysis described in Section 9.4.1, and has many numerical advantages, analogous to those of B-splines.

For any fixed $i = 1, \ldots, n-2m$, consider the sequence of $2m+1$ basis functions based on the reproducing kernel:

$$k_2(t_{i+\ell}, \cdot), \ell = 0, \ldots, 2m.$$

Let $b_\ell^{(i)}, \ell = 0, \ldots, 2m$ be a corresponding sequence of weights defining a new basis function

$$\psi_i = \sum_{\ell=0}^{2m} b_\ell^{(i)} k_2(t_{i+\ell}, \cdot). \tag{21.18}$$

The properties we are seeking are

$$\psi_i(t) = 0, t \le t_i \text{ and } \psi_i(t) = 0, t \ge t_{i+2m}.$$

But from the first line of (21.9), we see the first of these is achieved if

$$\sum_{\ell=0}^{2m} b_\ell^{(i)} \xi_{j_1}(t_{i+\ell}) = 0, \ j_1 = 1, \ldots, m \tag{21.19}$$

and at the same time the second line of (21.9) indicates that the second property is satisfied if

$$\sum_{\ell=0}^{2m} b_\ell^{(i)} [\sum_{j_1=1}^{m} \xi_{j_1}(t_{i+\ell}) f_{j_1,j_2}(t_{i+\ell})] = 0, \ j_2 = 1, \ldots, m, \tag{21.20}$$

where $f_{j_1,j_2}(t_{i+\ell})$ is entry (j_1, j_2) of $\mathbf{F}(t_{i+\ell})$.

Now these are two sets of m linear constraints on the $2m+1$ coefficients $b_\ell^{(i)}$, and we know that in general we can always find a coefficient vector $b^{(i)}$ that satisfies them. The reason that there are only $2m$ constraints for $2m+1$ coefficients is that the linear constraints can only define the vector $b^{(i)}$ up to a change of scale.

Let the $(2m+1) \times 2m$ matrix $\mathbf{V}^{(i)}$ have in its first m columns the values $\xi_{j_1}(t_{i+\ell}), j_1 = 1, \ldots, m$ and in its second set of m columns the values $\sum_{j_1=1}^{m} \xi_{j_1}(t_{i+\ell}) f_{j_1,j_2}(t_{i+\ell}), j_2 = 1, \ldots, m$. Then the constraints (21.19) and (21.20) can be written in the matrix form

$$(b^{(i)})' \mathbf{V}^{(i)} = 0.$$

As in the calculation of the vectors $c^{(i)}$ in Section 21.3.2, the required vector $b^{(i)}$ is simply the last column of the $\mathbf{Q}$ matrix in the QR decomposition of $\mathbf{V}^{(i)}$.

If the argument values are unequally spaced, this calculation of the coefficient vectors $b^{(i)}$ must be carried for each value of i from 1 to $n-2m$. However, in the frequently encountered case where the t_i values are equally spaced, only one coefficient calculation is required, and the resulting set of coefficients b serves for all $n-2m$ compact support splines ψ_i.

Observant readers may note that we have lost $2m$ basis functions by this approach. We may deal with this difficulty in various ways. One approach is to say that a little bit of fitting power has been lost, but that if n is large, this may have little impact on the smoothing function, and what little impact it has is at the ends of the interval. Alternatively, however, we can use a technique employed in defining polynomial B-splines, and add m additional argument values at each end of the interval. For computational convenience in the equally spaced argument case, we can make these simply a continuation of the sequence in both directions. This augments the basis in order to retain the full fitting power of the original reproducing kernel basis.

21.5 Some case studies

21.5.1 The gross domestic product data

The gross domestic product data introduced in Chapter 18 share with many economic indicators the overall tendency for exponential growth. If we wish

to smooth the de-seasonalized GDP record of the United States displayed in Figure 18.3, the operator $L = -\gamma D + D^2$ annihilates $\xi_1(t) = 1$ and $\xi_2(t) = e^{\gamma t}$, so these are obvious choices for the functions spanning $\ker L$. A reasonable choice for the matching constraint operator is simply B_I, such that $B_I u = \{u(0), Du(0)\}'$, implying that the coefficients of ξ_1 and ξ_2 are specified by the initial value and slope of the smoothed record.

In this case, we might decide to estimate parameter γ by estimating the slope of the relationship between log GDP and time by ordinary regression analysis. Another possibility is to fit all or part of the data by nonlinear least squares regression using the two functions ξ_1 and ξ_2. That is, we minimize the error sum of squares with respect to the coefficients c_1 and c_2 of $c_1\xi_1 + c_2\xi_2$ and with respect to γ which, of course, determines ξ_2. Since for any fixed γ value, the minimizing values of the coefficients can be computed directly by linear least squares, it makes sense to use a one-dimensional function minimizing routine such as Brent's method (Press et al. 1992) to find the optimal γ value; each new value of γ within the iterative method implies a linear regression to get the associated values of c_1 and c_2. The resulting least squares estimate of γ for the U.S. data, based on the values from 1980 to 1989, when the growth was more exponential, is 0.054.

Using this value of γ, we used the method of Section 21.3 to find the L-smoothing spline shown in Figure 21.3. We minimized the GCV criterion to obtain the value $\lambda = 0.053$. The $\mathtt{DF}_1$ measure of equivalent degrees of freedom was 39.6, so we purchased the excellent fit of the spline at the price of a rather large number of degrees of freedom.

By comparison, the cubic smoothing spline that minimizes GCV produces almost identical results in terms of GCV and $\mathtt{DF}_1$ values. This is perhaps not too surprising since the curve is only slightly more exponential than linear. But the results are rather different when we smooth with the fixed value of $\mathtt{DF}_1 = 10$, corresponding to $\lambda = 22.9$. The L-spline fit using this more economical model is just barely visible in Figure 21.3, and $\mathtt{GCV} = 0.00068$. The cubic polynomial spline with $\mathtt{DF}_1 = 10$ yields $\mathtt{GCV} = 0.00084$, and its poorer fit reflects the fact that some of its precious degrees of freedom were used up in fitting the mild exponential trend.

21.5.2 The melanoma data

These data, displayed in Figure 17.5, represent a more complex relationship, with a cyclic effect superimposed on a linear development. The interesting operator is

$$L = \omega^2 D^2 + D^4 \tag{21.21}$$

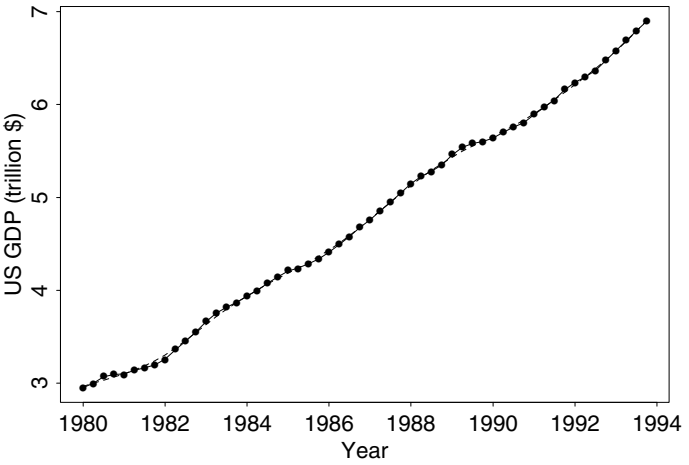

Figure 21.3. The line indicates the spline smooth of the U.S. GDP data using $L = -0.054D + D^2$ and the minimum GCV value for smoothing parameter λ. The dashed line indicates the L-spline fit corresponding to $\mathtt{DF}_1 = 10$.

for some appropriate constant ω, since this would annihilate the four functions

$$u(t) = (1, t, \sin \omega t, \cos \omega t)'.$$

Using the techniques of Chapter 18, the reproducing kernel is

$$\begin{aligned} k_2(s,t) \quad = \quad & \omega^{-7}[(\omega s)^2(\omega t/2 - \omega s/6) - \omega t + \omega s + \omega t \cos \omega s \\ & + \omega s \cos \omega t + \sin \omega s - \sin \omega t + \sin(\omega t - \omega s) \\ & - (\sin \omega s \cos \omega t)/2 + s \cos(\omega t - \omega s)/2], \\ & s \le t. \end{aligned} \tag{21.22}$$

We estimated the parameter ω to be 0.650 by the nonlinear least squares approach. This corresponds to a period of 9.66 years, roughly the period of the sunspot cycle affecting solar radiation and consequently melanoma. When we smooth the data with the spline defined by the operator (21.21) and select λ so as to minimize GCV, it turns out that λ becomes arbitrarily large, corresponding to a parametric fit using only the basis functions $\boldsymbol{\xi}$, consuming four degrees of freedom, and yielding $\mathtt{GCV} = 0.076$. The polynomial smoothing spline with order $m = 4$, displayed in Figure 17.5, is a minimum-GCV estimate corresponding to $\mathtt{DF}_1 = 12.0$ and $\mathtt{GCV} = 0.095$. Thus, polynomial spline smoothing required three times the degrees of freedom to produce a fit that was still worse in GCV terms than the L-spline

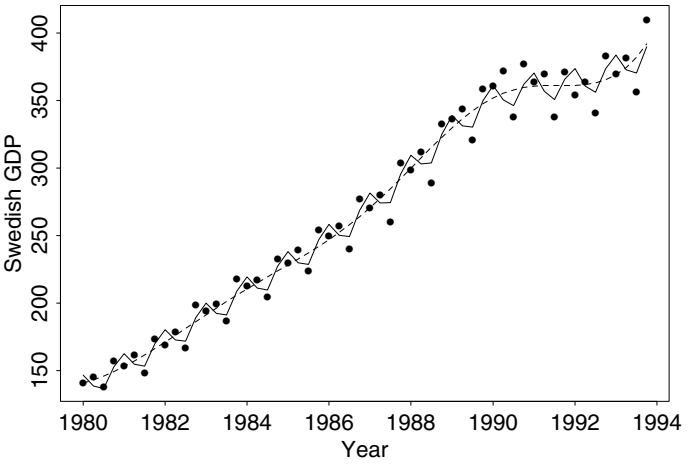

Figure 21.4. The gross domestic product for Sweden with seasonal variation. The solid line is the smooth using operator $L = (-\gamma D+D^2)(\omega^2 I+D^2)$, and the dashed line is the smooth for $L = D^4$, the smoothing parameter being determined by minimizing the GCV criterion in both cases.

smooth. Of the two order-4 methods, the operator (21.21) is much to be preferred to $L = D^4$.

21.5.3 The GDP data with seasonal effects

In the data provided by the U.S. and most other countries, the within-year variation in GDP that is a normal aspect of most economies was removed. But the data for Sweden, displayed in Figure 21.4, does retain this seasonal variation. This suggests composing the operator $-\gamma D + D^2$ used for the U.S. GDP data with the de-seasonalizing operator $\omega^2 I + D^2$ to obtain the composite operator of order four

$$L = (-\gamma D + D^2)(\omega^2 I + D^2) = -\gamma\omega^2 D + \omega^2 D^2 - \gamma D^3 + D^4. \quad (21.23)$$

This annihilates the four linearly independent functions given by the components of

$$u(t) = (1, \exp\gamma t, \sin\omega t, \cos\omega t)'.$$

In this application we know that $\omega = 2\pi$ for time measured in years, and the nonlinear least squares estimate for γ was 0.078.

The minimum GCV L-spline for these data is the solid line in the figure, and corresponds to $\texttt{GCV} = 142.9, \texttt{SSE} = 5298$, and $\texttt{DF}_1 = 10.4$. This fairly low-dimensional spline tracks both the seasonal and long-term

variation rather well. By contrast, the minimum GCV polynomial spline corresponding to $L = D^4$ is shown by the dashed line, and corresponds to $\texttt{GCV} = 193.8, \texttt{SSE} = 8169$, and $\texttt{DF}_1 = 7.4$. As both the curve itself and the GCV value indicate, the polynomial spline was completely unable to model the seasonal variation, and treated it as noise. On the other hand, reducing the smoothing parameter λ to the point where SSE was reduced to the same value as was attained for the L-spline required $\texttt{DF}_1 = 28.2$, or nearly three times the degrees of freedom. Again we see that building the capacity to model important sources of variation into the operator L pays off handsomely.

21.5.4 Smoothing simulated human growth data

One of the triumphs of nonparametric regression techniques has been their capacity to uncover previously unsuspected aspects of growth in skeletal height (Gasser, Müller, Köhler, Molinari and Prader, 1984; Ramsay, Bock and Gasser, 1995). In this illustration, spline smoothing using an estimated differential operator was applied to simulated smoothing data. The objective was to see whether estimating the smoothing operator improves the estimation of the height and height acceleration growth functions over an a priori "off-the-rack" smoother.

To investigate how the performance of the L-spline would compare with a polynomial spline in practice, we simulated data to resemble as much as possible actual human growth curve records. We generated two samples: a training sample of 100 records that was analyzed in a manner representative of actual practice, and a validation sample of 1000 records to see how these analyses would perform on data for which the analyses were not tuned.

The simulated data for both the training and validation samples consisted of growth records generated by using the triple logistic parametric nine-parameter growth model proposed by Bock and Thissen (1980). According to this model, height $h_i(t)$ at age t for individual i is

$$h_i(t) = \sum_{j=1}^{3} c_{ij}/[1 + \exp(-a_{ij}(t - b_{ij}))]. \tag{21.24}$$

This model, although not completely adequate to account for actual growth curves, does capture their salient features rather well. The actual number of parameters in the model turns out to be only eight, since parameter $a_{i,1}$ can be expressed as a function of the other parameters.

We generated each record by first sampling from a population of coefficient vectors having a random distribution estimated from actual data for males in the Fels Growth Study (Roche, 1992). We computed the errorless growth curves (in cm) for the 41 age values ranging from 1 to 21 in half-yearly steps, and generated the simulated data by adding independent normal error with mean 0 and standard deviation 0.5 to these values.

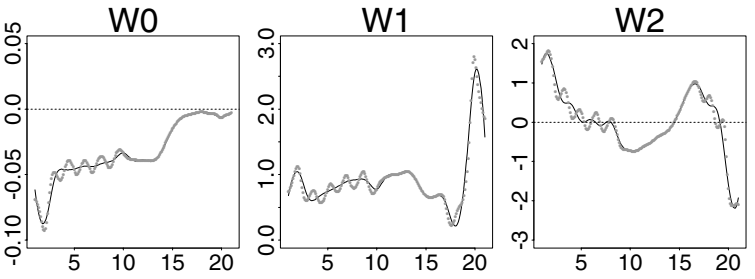

Figure 21.5. The three weight functions w_0, w_1, and w_2 for the operator $L = w_0 I + w_1 D + w_2 D^2 + D^3$: The points indicate the point-wise-approximation, and the solid line indicates the basis function expansion.

These simulated data had roughly the same variability as actual growth measurements.

The first step was to use the training sample to estimate the order three L-spline that comes as near as possible to annihilating the curves. To this end, the first analysis consisted of polynomial spline smoothing of the simulated data to get estimates of the first three derivatives. The smoothing operator used for this purpose was D^5, implying that the smoothing splines were piecewise polynomials of degree 9. This permitted us to control the roughness of the third derivative in much the same way as a cubic smoothing spline controls the roughness of the smoothing function itself. The smoothing parameter was chosen to minimize the `GCV` criterion, and with this amount of replicated data, this value of its minimum is sharply defined. Since our principal differential analysis estimate of the operator L required numerical integration, we also obtained function and derivative estimates at 201 equally-spaced values 1(.1)21.

We estimated a third-order differential operator L using both the point-wise technique and the basis function expansion approach described in Chapter 19. For the latter approach, we used the 23 order 4 B-splines defined by positioning knots at the integer values of age. Figure 21.5 displays the estimated weight functions w_0, w_1, and w_2 for the operator $L = w_0 I + w_1 D + w_2 D^2 + D^3$. Although these are difficult to interpret, we can see that w_0 is close to 0, suggesting that the operator could be simplified by dropping the first term. On the other hand, w_1 is close to one until the age of 15 when the growth function has strong curvature as the pubertal growth spurt ends, and its strong variation after 15 helps the operator to deal with this pronounced curvilinearity. The acceleration weight w_2 varies substantially over the whole range of ages.

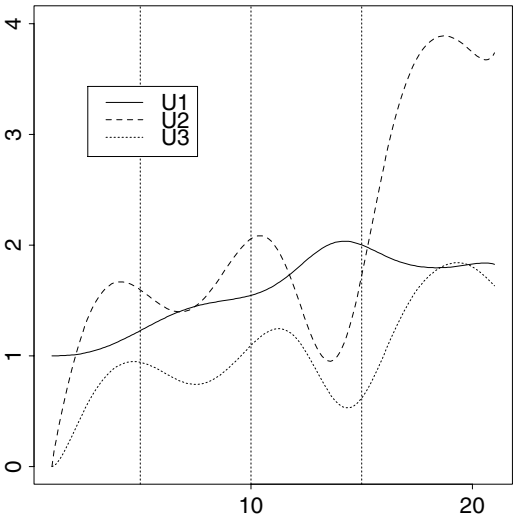

Figure 21.6. The three solutions to the homogeneous equation $Lu = 0$ corresponding to the linear differential operator L estimated for the simulated human growth data.

Figure 21.6 shows three linearly independent solutions ξ_j to $Lu = 0$. Linear combinations of these three functions can produce good approximations to actual growth curves.

The next step was to use the estimated functions ξ_j and the techniques of Chapter 18 to estimate the Green's function G and the reproducing kernel k_2 associated with this operator. We approximated the integrals involved using the trapezoidal rule applied to the values at the 201 argument values.

Now we were ready to carry out the actual smoothing of the training sample data by using the two techniques, L-spline and polynomial spline smoothing, both of order three, just much as one would in practice. For both techniques, we relied on the GCV criterion to choose the smoothing parameter. The polynomial smooth gave values of $\texttt{GCV}, \texttt{DF}_1$ and λ of 487.9, 9.0 and 4.4, respectively, and the L-spline smooth produced corresponding values of 348.2, 11.2 and 0.63.

How well would these two smoothing techniques approximate the curves generating the data? To answer this question, we generated 1,000 new simulated curves using the same generation process, and applied these two smoothers using the training sample values of λ. Since we knew the values of the true curves, we could estimate the root-mean-squared error criterion

$$\texttt{RMSE}(t) = \sqrt{\mathrm{E}\{\hat{x}(t) - x(t)\}^2}$$

by averaging the squared error across the 1,000 curves for a given specific age t, and then taking the square root. This yielded the two RMSE curves

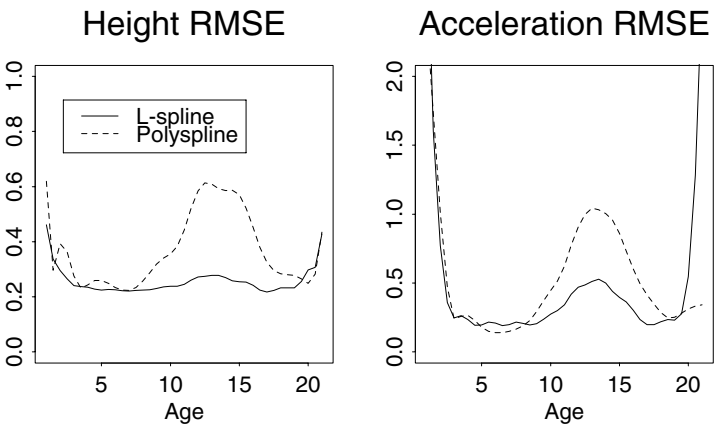

Figure 21.7. The left panel displays root-mean-squared error (RMSE) as a function of age for the simulated growth data. The solid line is for smoothing using the estimated differential operator L, and the dashed line is for polynomial smoothing using $L = D^3$. The right panel shows these results for the estimated height acceleration.

displayed in Figure 21.7. We see that the estimate of both the growth curve itself and its acceleration by the L-spline procedure is much better for all but the final adult period, where the L-spline estimate of the acceleration curve becomes rather noisy and unstable. The improvement in the estimation of both curves is especially impressive prior to and during the pubertal growth spurt: The mean square error for the polynomial smooth is about four times that of the L-spline smooth. That is, using the L-spline is roughly equivalent to using the polynomial smooth with quadruple the sample size. same generation process, and applied these two smoothers using the training sample values of λ. Since we knew the values of the true curves, we could estimate the root-mean-squared error criterion

$$\texttt{RMSE}(t) = \sqrt{\mathrm{E}\{\hat{x}(t) - x(t)\}^2}$$

by averaging the squared error across the 1,000 curves for a given specific age t, and then taking the square root. This yielded the two `RMSE` curves displayed in Figure 21.7. We see that the estimate of both the growth curve itself and its acceleration by the L-spline procedure is much better for all but the final adult period, where the L-spline estimate of the acceleration curve becomes rather noisy and unstable. The improvement in the estimation of both curves is especially impressive prior to and during the pubertal growth spurt: The mean square error for the polynomial smooth is about four times that of the L-spline smooth. That is, using the L-spline is roughly equivalent to using the polynomial smooth with quadruple the sample size.

22
Some perspectives on FDA

22.1 The context of functional data analysis

We conclude this volume with some historical remarks and pointers to bibliographic references which have not been included in the main course of our development. We are, of course, acutely aware that many branches of statistical science consider functional models and the data that go with them. FDA has a long historical shadow, extending at least back to the attempts of Gauss and Legendre to estimate a comet's trajectory (Gauss, 1809; Legendre, 1805). So what we offer here is perhaps little more than a list of personal inspirations. In addition we suggest some directions for further research.

22.1.1 Replication and regularity

While we want to leave the question of exactly what constitutes FDA soft around the edges, functional data problems as we have described them have two general features: replication and regularity. These are intimately related. Both permit the use of information from multiple data values to identify patterns; replication implies summaries taken across different observations, while regularity allows us to exploit information across argument values.

Replication is closely bound up with the key concept of a functional observation as a single entity, rather than a set of individual numbers or values. The availability of a sample of N related functional observations

then leads to an interest in structure and variability in the data that requires more than one observation to detect. This is in contrast with much of the literature on nonparametric regression or curve estimation, where the focus is on estimating a single curve.

Functional principal components analysis, regression analysis, and canonical correlation, like their multivariate counterparts, characterize variation in terms of features that have stability across replicates. Likewise, principal differential analysis and the use of an estimated linear differential operator for smoothing presume a model structure that belongs to the entire sample. Even curve registration aims to remove one important source of inter-curve variation so as to render more obvious the structure that remains.

Regularity implies that we exploit the smoothness of the processes generating the data, even though these data usually come to us in discrete form. The assumption that a certain number of derivatives exist has been used in most of the analyses that we have considered. The roughness penalty approach used throughout the book controls the size of derivatives and mixtures of derivatives of the functional parameters that we have estimated. In this way we stabilize estimated principal components, regression functions, monotone transformations, canonical weight functions, and linear differential operators.

Are there more general concepts of regularity that would aid FDA? For example, wavelet approaches to smoothing briefly discussed in Section 3.6.1 are probably relevant, because of their ability to accommodate notions of regularity that, nevertheless, allow certain kinds of local misbehavior.

Independent identically distributed observations are only one type of regularity. For example, can we use the replication principle implicit in stationary time series and where the values of the process are functions, to define useful FDAs? Besse and Cardot (1997) offer an interesting first step in this direction.

22.1.2 Some functional aspects elsewhere in statistics

Analysis of variance is often concerned with within-replication treatments. While an ANOVA design does not assume as a rule that these treatments correspond to variation over time or some other continuum, in practice this is often the case. Consequently texts on ANOVA such as Maxwell and Delaney (2003) pay much attention to topics that arise naturally when treatments correspond to events such as days, related spatial positions, and so on. Modifications taking account of a more complex correlational structure for the residuals and the use of contrasts to make inferences about linear, quadratic, and other types of trend across treatments are examples.

As we indicated at the end of Chapter 5, fields such as longitudinal data analysis (Diggle et al., 1994), analysis of repeated measurements (Keselman and Keselman, 1993 and Lindsey, 1993) and growth curve analysis are cognate to functional data analysis. Two classic papers that use principal

components analysis to describe the modes of variation among replicated curves are Rao (1958) and Tucker (1958); Rao (1987) offers a summary of his and others' work on growth curves. Two more recent applications are Castro, Lawton and Sylvestre (1986) and Grambsch et al. (1995).

But these and the many other studies of curve structure do not give the regularity of the phenomena a primary role, placing more emphasis instead on replication. Likewise, empirical Bayes, hierarchical linear model, or multilevel linear model approaches do treat functional data in principle, with the added feature of using prior information, but the nature of the prior structure tends to be multivariate rather than functional. In particular, as we noted in Chapter 5, the estimation of a between–curve variance-covariance matrix whose order is equal to the number of basis functions used to represent the curves places severe limits on the the complexity of the functional variation.

Nevertheless, we expect that further research will show that the experience gained and tools developed in these collateral disciplines can be put to good use in FDA.

22.1.3 Functional analytic treatments of statistical methods

One topic clearly within the scope of FDA is the description of statistical methods using functional analysis. For example, principal components analysis is a technique that lends itself naturally to many types of generalization. The notion of the eigenanalysis of a symmetric matrix was extended to integral operators with symmetric kernels in the last century, and the Karhunen-Loève decomposition of more general linear operators (Karhunen, 1947; Loève, 1945) is essentially the singular value decomposition in a wider context.

Parzen's papers (1961, 1963) are classics, and have had a great influence on the spline smoothing literature by calling attention to the important role played by the reproducing kernel. Grenander (1981) contributed further development, Eaton (1983) provided a systematic coverage of multivariate analysis using inner product space notation, and Stone (1987) also proposed a coordinate-free treatment.

Applied mathematicians and statisticians in France have been particularly active in recasting procedures originally developed in a conventional discrete or multivariate setting into a functional analytic notational framework. Deville (1974) considered the PCA of functional observations with values in $\mathcal{L}^2$. Cailliez and Pagès (1976) wrote an influential textbook on multivariate statistics that was both functional analytic in notation and coordinate-free in a geometrical sense. This was a courageous attempt to present advanced concepts to a mathematically unsophisticated readership, and it deserves to be better known. Dauxois and Pousse (1976) produced a comprehensive and sophisticated functional analytic exposition of PCA and CCA that unhappily remains in unpublished form.

While the exercise of recasting the usual matrix treatments of multivariate methods into the more general language of functional analysis is intrinsically interesting to those with a taste for mathematical abstraction, it also defined directly the corresponding methods for infinite-dimensional or functional data. Some facility in functional analysis is a decided asset for certain aspects of research in FDA, as it already is in many other branches of applied mathematics.

22.2 Challenges for the future

We now turn to a few areas where there is clearly need for further research. These should be be seen as a small selection of the wide range of topics that a functional data analytic outlook opens up.

22.2.1 Probability and inference

The presence of replication inevitably invites some consideration of random functions and probability distributions on function spaces. Of course, there is a large literature on stochastic processes and random functions, but because of our emphasis on data analysis we have not emphasized these topics in the present volume.

We note, in passing, that functional observations can be random in a rather interesting variety of ways. We pointed out in Section 21.3 that the problem of spline smoothing is intimately related to the theory of stochastic processes defined by the nonhomogeneous linear differential equation $Lx = f$ where L is a deterministic linear differential operator and f is white noise. Should we allow for some stochastic behavior or nonlinearity in L? Is white noise always an appropriate model for f? Should we look more closely at the behavior of an estimate of f in defining smoothing criteria, FDAs, and diagnostic analyses and displays, exploiting this estimate in ways analogous to our use of residuals in regression analysis? There is a large literature on such *stochastic differential equations*; see, for example Øksendal (1995). Though stochastic differential equations are of great current interest in financial mathematics, they have had relatively little impact on statistics more generally. This seems like a way to go.

We discussed the extension of classical inferential tools such as the t-test or F-ratio to the functional domain. We often need simulation to assess the significance of statistics once we move beyond the context of inference for a fixed argument value t. For a rather different approach to inference that incorporates both theoretical arguments and simulation, see Fan and Lin (1998).

Because of the infinite-dimensional nature of functional variation, the whole matter of extending conventional methods of inference—whether

parametric or nonparametric, Bayesian or frequentist—is one that will require considerable thought before being well understood. We consider that there is much to do before functional data analysis will have an inferential basis as developed as that of multivariate statistics.

22.2.2 Asymptotic results

There is an impressive literature on the asymptotic and other theoretical properties of smoothing methods. Although some would argue that theoretical developments have not always had immediate practical interest or relevance, there are many examples clearly directed to practical concerns. For a recent paper in the smoothing literature that addresses the issue of the positive interaction between theoretical and practical research, see, for example, Donoho et al. (1995).

Some investigations of various asymptotic distributional aspects of FDA are Dauxois et al. (1982), Besse (1991), Pousse (1992), Leurgans et al. (1993), Pezzulli and Silverman (1993), Silverman (1996) and Kneip and Engel (1995), for example. Nevertheless, theoretical aspects of FDA have not been researched in sufficient depth, and it is hoped that appropriate theoretical developments will feed back into advances in practical methodology.

22.2.3 Multidimensional arguments

Although we have touched multivariate functions of a single argument t, coping with more than one dimension in the domain of our functions has been mainly beyond our scope. But of course there is a rapidly growing number of fields where data are organized by space instead of or as well as time. Consider, for example, the great quantities of satellite and medical image data, where spatial dimensionality is two or three and temporal dependence is also of growing importance.

There is a large and growing literature on spatial data analysis; see, for example, Cressie (1991) and Ripley (1991). Likewise smoothing over two or more dimensions of variation is a subject of active research (Scott, 1992). In particular, Wahba (1990) has pioneered the extension of regularization techniques to multivariate arguments. In principle, there is no conceptual difficulty in extending our own work on FDA to the case of multivariate arguments by using the roughness penalties relevant to tensor or thin-plate splines. Indeed, the paper by Hastie et al. (1995) reviewed in Section 11.7 uses roughness penalty methods to address a functional data analysis problem with a spatial argument. However, there are questions about multivariate roughness penalty methods in FDA that require further research.

22.2.4 Practical methodology and applications

Clearly, much research is needed on numerical methods, as is evident when one considers the work on something as basic as the point-wise linear model underlying spline smoothing. We think that regularization techniques will play a strong role, in part because they are so intuitively appealing. But of course there are often simpler approaches that may work more or less as well.

It is our hope that this book will give impetus to the wider dissemination and use of FDA techniques. More importantly than any of the detailed methodological issues raised in this chapter, the pressing need is for the widespread use of functional data analytic techniques in practice.

22.2.5 Back to the data!

Finally, we say simply that it is the data that we have analyzed, and our colleagues who collected them, that are responsible for our understanding of functional data analysis. If what this book describes is found to deserve a name for itself, it will be because we have discovered, with each new set of functional data, challenges and invitations to develop new methods. Statistics shows its finest aspects when exciting data find existing statistical technology not entirely satisfactory. It is this process that informs this book, and ensures that unforeseen adventures in research await us all.

Appendix

Some algebraic and functional techniques

This appendix covers various topics involving matrix decompositions, projections in vector and function spaces, and the constrained maximization of certain quadratic forms through the solution of appropriate eigenequations.

A.1 Inner products $\langle x, y \rangle$

An advance in mathematical notation occurs when we separate the name for an operation from explicit instructions on how to carry it out. Consider, for example, the operation $+$. Suppose one opens a mathematics book at a random page, and discovers the expression $x + y$. One might imagine that everyone would always mean the same by $x + y$, but a moment's thought shows that computing the sum can involve very different techniques depending on whether x and y are real numbers, complex numbers, vectors, matrices of the same dimensions or functions. What really counts is that any author who uses the symbol $+$ can be assumed to mean an operation that obeys the basic properties of addition, $x+y = y+x$ and $(x+y)+z = x+(y+z)$, and that this operation also interlocks with the multiplication operation $\times$ through $(x + y) \times z = x \times z + y \times z$ and $x \times (y + z) = x \times y + x \times z$. The author assumes that we can actually carry out the operation involved ourselves, or else in some exotic situations he or she furnishes us with detailed instructions. The notation $x+y$ allows the basic structure of addition to be assumed, almost subconsciously, leaving the

details to be supplied in any particular case if necessary. Hiding the details focusses our attention on what really matters.

A.1.1 Some specific examples

We now discuss a generic notation for inner products, extending the familiar idea of the inner product of two vectors $\mathbf{x}$ and $\mathbf{y}$. Consider the *Euclidean inner product* operation $\mathbf{x}'\mathbf{y}$, where $\mathbf{x}$ and $\mathbf{y}$ are vectors of the same length. The operation has the following simple properties:

Symmetry: $\mathbf{x}'\mathbf{y} = \mathbf{y}'\mathbf{x}$ for all $\mathbf{x}$ and $\mathbf{y}$,

Positivity: $\mathbf{x}'\mathbf{x} \geq 0$ for all $\mathbf{x}$, with $\mathbf{x}'\mathbf{x} = 0$ if and only if $\mathbf{x} = 0$, and

Bilinearity: for all real numbers a and b, $(a\mathbf{x} + b\mathbf{y})'\mathbf{z} = a\mathbf{x}'\mathbf{z} + b\mathbf{y}'\mathbf{z}$ for all vectors $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$.

Of course, these properties follow from the instructions implied in the definition

$$\mathbf{x}'\mathbf{y} = \sum_i x_i y_i. \tag{A.1}$$

However, it is important to note that the Euclidean inner product operation, and the instructions defining it, are of critical importance in multivariate data analysis *because* of the properties of symmetry, positivity and bilinearity, which can therefore be considered of more fundamental significance than the definition (A.1) itself.

This basic role of symmetry, positivity and bilinearity is further emphasized when we realize that $\mathbf{x}'\mathbf{W}\mathbf{y}$, where $\mathbf{W}$ is a positive definite matrix of appropriate order, also has these properties and, indeed, can be used almost anywhere that we use $\mathbf{x}'\mathbf{y}$. So, for example, we use $\mathbf{x}'\mathbf{\Sigma}^{-1}\mathbf{y}$, where $\mathbf{\Sigma}$ is a population covariance matrix, to define the multivariate normal distribution, to compute Mahalanobis distances, to define generalized least squares estimates instead of ordinary least squares, and many other useful things.

Now suppose that x and y are not vectors, but rather functions with values $x(t)$. The natural functional counterpart to $x'y$ is $\int x(t)y(t)\,dt$, replacing the sum in (A.1) by an integral. Again we have an operation on two functions x and y that is denoted by presenting the instructions for computing its value, but we know that this, too, is symmetric in x and y, linear in either function, and satisfies the positivity requirement. The same conclusions can be drawn for the operation $\int \omega(t)x(t)y(t)\,dt$, where ω is a strictly positive weight function, and indeed for the more general operation $\int\int \omega(s,t)x(s)y(t)\,ds\,dt$ if ω is strictly positive-definite, which simply means that the positivity requirement for the inner product is satisfied.

It should by now be clear that we can achieve a great leap forward in generality by using a common notation for these various real-valued

operations that is understood to imply symmetry, positivity and bilinearity, without bothering with the details of the computation. We call such an operation an *inner product*, and we use the generic notation $\langle x, y\rangle$ for the inner product of x and y. The fundamental properties of an inner product are:

Symmetry: $\langle x, y\rangle = \langle y, x\rangle$ for all x and y;

Positivity: $\langle x, x\rangle \geq 0$ for all x, with $\langle x, x\rangle = 0$ if and only if $x = 0$;

Bilinearity: for all real numbers a and b, $\langle ax + by, z\rangle = a\langle x, z\rangle + b\langle y, z\rangle$ for all vectors x, y and z.

Note that bilinearity in the second argument follows from symmetry and bilinearity in the first.

A.1.2 General properties: association, size, angle, distance

We can think of the inner product as defining a scalar measure of *association* between pairs of quantities x and y. The symmetric nature of the measure means that it is, as we would usually require, invariant with respect to the order of the quantities, and its bilinearity means that changing the scale of either argument and/or using a sum as either argument leaves the measure unchanged in its essential properties.

Positivity means that the inner product of any x with itself is essentially a measure of its *size*. The positive square root of this size measure is called the *norm* of x, written $\|x\|$, so that

$$\|x\|^2 = \langle x, x\rangle \tag{A.2}$$

with $\|x\| \geq 0$. In the special case where x is an n-vector, and the inner product is the Euclidean inner product (A.1), the norm of x is simply the length of the vector measured in n-dimensional space. In the case of a function f, a basic type of norm is $\|f\| = \sqrt{\int f^2}$, and is called its $\mathcal{L}^2$ norm.

Whatever inner product is used, the standard properties of inner products lead to the following properties of the norm:

1. $\|x\| \geq 0$ and $\|x\| = 0$ if and only if $x = 0$
2. $\|ax\| = |a|\|x\|$ for all real numbers a
3. $\|x + y\| \leq \|x\| + \|y\|$.

From the properties of the inner product also follows the *Cauchy-Schwarz inequality*:

$$|\langle x, y\rangle| \leq \|x\|\|y\| = \sqrt{\langle x, x\rangle\langle y, y\rangle}.$$

This inequality links the inner product with the derived size measure or norm, and also leads to the *cosine inequality*:

$$-1 \leq \langle x, y\rangle/(\|x\|\|y\|) \leq 1.$$

The cosine inequality links the inner product to the geometrical concept of *angle*; the angle between x and y can be defined to be the angle θ such that

$$\cos\theta = \frac{\langle x, y \rangle}{\|x\|\|y\|}.$$

Where x and y are n-vectors and the inner product is Euclidean inner product, θ is the angle between x and y in the usual geometric sense. Similarly, the cosine of the angle between two functions f and g can be defined as $\int fg / \sqrt{(\int f^2)(\int g^2)}$. The use of the cosine inequality to justify the idea of the angle between two vectors or functions further illuminates the notion that $\langle x, y \rangle$ is a association measure. Once we have obtained a scale invariant coefficient by dividing by $\|x\|\|y\|$, we have a useful index of the extent to which x and y are measuring the same thing.

The particular relation $\langle x, y \rangle = 0$, called *orthogonality*, implies that x and y can be considered as being at right angles to one another. Because of bilinearity, orthogonality remains unchanged under any rescaling of either quantity. Orthogonality plays a key role in the operation of *projection* that is discussed in Section A.2.1.

From the inner product, we also derive a measure of *distance* between x and y

$$d_{xy} = \|x - y\| = \sqrt{\langle x - y, x - y \rangle}$$

that has extremely wide applications; again, in the Euclidean case, distance corresponds to the usual geometric definition.

Thus, the simple algebraic properties of symmetry, positivity and bilinearity of the inner product lead easily to very useful definitions of the size of a quantity x, and of the angle and distance between x and y. We can be confident that, no matter how we define $\langle x, y \rangle$ in a particular application, the essential characteristics of these three measures remain unchanged.

The nature of the inner product depends on something more fundamental about x and y: They are elements of a *vector space* in which elements can be added, can be multiplied by real numbers to yield new vectors, and in which addition distributes with respect to scalar multiplication. The ensemble of a vector space and an associated inner product is called an *inner product* space.

Finally, of the three properties, only symmetry and bilinearity are really crucial. We can often get by with relaxing positivity to the weaker condition that $\langle x, x \rangle \geq 0$, so that $\langle x, x \rangle$ may be zero for some x's that are not themselves zero. Then the inner product is called a *semi-inner product* and the norm a *seminorm*. Most properties of inner products remain true for semi-inner products.

A.1.3 Descriptive statistics in inner product notation

As an example of how inner products can work for us, we consider how standard descriptive statistics can be expressed in inner product notation. Consider the space of possible univariate samples $x = (x_1, \ldots, x_N)$ of size N. Define the inner product to be the Euclidean inner product

$$\langle x, y\rangle = \sum_i x_i y_i = x'y.$$

Let 1 indicate the vector of size N all of whose elements are unity. Then some familiar univariate descriptive statistics become

Mean: $\bar{x} = N^{-1}\langle x, 1\rangle$. Note that $\bar{x}$, being a multiple of an inner product, is a scalar and not a vector. The vector of length N all of whose elements are $\bar{x}$ is $\bar{x}1$.

Variance: $s_x^2 = N^{-1}\langle x - \bar{x}1, x - \bar{x}1\rangle = N^{-1}\|x - \bar{x}1\|^2$

Covariance: $s_{xy} = N^{-1}\langle x - \bar{x}1, y - \bar{y}1\rangle$

Correlation: $r_{xy} = s_{xy}/(s_x s_y)$.

It is easy to show that the covariance s_{xy} is itself a semi-inner product between x and y. It is then an immediate consequence of the cosine inequality that the correlation coefficient satisfies the well-known *correlation inequality*

$$-1 \leq r_{xy} \leq 1.$$

Now suppose that we stop using the Euclidean inner product, but instead go for

$$\langle x, y\rangle = \sum_i w_i x_i y_i,$$

where w_i is a nonnegative weight to be applied to observation i. What difference would this make? None at all, except of course we must now divide by the constant $\sum_i w_i$ instead of N in defining $\bar{x}, s_x^2$, and s_{xy}. The essential characteristics of these statistics depend on the characteristics of the inner product, and not on precisely how the inner product is actually calculated. Of course, the precise weighting affects the values of the statistics, but the essential meanings of the various descriptive statistics, for example as measures of location, scale and dependence remain basically unchanged.

We can generalize this idea further: Suppose that the sequence of observations is known to be correlated, with covariance matrix $\mathbf{\Sigma}$. Then we can use $\langle x, y\rangle = x'\mathbf{\Sigma}^{-1}y$ to provide a basis for descriptive statistics that compensate for the known covariance structure on the observations.

Now consider these same statistics in the context of x being a function with values $x(t)$, where argument t takes values within some real interval

such as $[0, T]$. Thus the index i taking N possible values has been replaced by the index t taking an infinity of values. Define the inner product as

$$\langle x, y\rangle = \int_0^T x(t)y(t)\,dt,$$

where we assume that the functions are sufficiently well behaved that the integral is always defined and finite. Then the various descriptive statistics continue to be defined as above, except that we divide by $\int_0^T dt = T$ instead of N and the vector 1 is replaced by the function $1 = 1(t)$ which takes the value of unity for all t. In the functional case, $\bar{x}$ becomes the mean level of the function x, s_x^2 becomes a measure of its variation about its mean level, and s_{xy} and r_{xy} measure the correspondence between the variation of x and y. Moving to

$$\langle x, y\rangle = \int_0^T \omega(t)x(t)y(t)\,dt,$$

for some positive weight function ω, and dividing by $\int \omega(t)\,dt$ really wouldn't change these interpretations in any essential way, except that different parts of the range of t would be regarded as being of different importance.

Finally, we note that even the divisors in these statistics can be defined in inner product terms, meaning that our fundamental descriptive statistics can be written in the unifying form

$$\begin{aligned} \bar{x} &= \langle x, 1\rangle/\|1\|^2 \\ s_x^2 &= \|x - \bar{x}1\|^2/\|1\|^2 \\ s_{xy} &= \langle x - \bar{x}1, y - \bar{y}1\rangle/\|1\|^2. \end{aligned}$$

A.1.4 Some extended uses of inner product notation

In this book, we take the somewhat unorthodox step of using inner product notation to refer to certain *linear operations* that, strictly speaking, do not fall within the rubric of inner products.

So far in our discussion, the result of an inner product has always been a single real number. One way in which we extend our notation is the following. Let $x = (x_1, \ldots, x_m)'$ be a vector of length m, each element of which is an element of some vector space, whether finite dimensional or functional. Then the notation $\langle x, y\rangle$, where y is a single element of the same space, indicates the m-vector whose elements are $\langle x_1, y\rangle, \ldots, \langle x_m, y\rangle$. Furthermore, if y is similarly a vector of length, say, n, then the notation $\langle x, y'\rangle$ defines the *matrix* with m rows and n columns containing the values $\langle x_i, y_j\rangle, i = 1, \ldots, m; j = 1, \ldots, n$. We only use this convention in situations where the context should make clear whether x and/or y are vectors of elements of the space in question.

In the functional context, we sometimes write

$$\langle z, \beta \rangle = \int z(s)\beta(s)\, ds$$

even when the functions z and β are not in the same space. We hope that the context of this use of inner product notation will make clear that a true inner product is not involved in this case. The alternative would have been the use of different notation such as (z, β), but we considered that the possibilities of confusion justified avoiding this convention.

An important property is that $\langle z, \beta \rangle$ is always a *linear operator* when regarded as a function of either of its arguments; generally speaking a linear operator on a function space is a mapping A such that, for all f_1 and f_2 in the space, and for all scalars a_1 and a_2, we have $A(a_1 f_1 + a_2 f_2) = a_1 A f_1 + a_2 A f_2$.

A.2 Further aspects of inner product spaces

We briefly review two further aspects of inner product spaces that are useful in our later development.

A.2.1 Projections

Let $u_1, \ldots, u_n$ be any n elements of our space, and let $\mathcal{U}$ be the subspace consisting of all possible linear combinations of the u_i. We can characterize the subspace $\mathcal{U}$ by using suitable vector notation. Let u be the n-vector whose elements are the $u_1, \ldots, u_n$. Then every member of $\mathcal{U}$ is of the form $u'c$ for some real n-vector c.

Associated with the subspace $\mathcal{U}$ is the *orthogonal projection onto* $\mathcal{U}$, which is defined to be a linear operator P with the following properties:

1. For all z, the element Pz falls in $\mathcal{U}$, and so is a linear combination of the functions $u_1, \ldots, u_n$.

2. If y is in $\mathcal{U}$ already, then $Py = y$.

3. For all z, the *residual* $z - Pz$ is orthogonal to all elements v of $\mathcal{U}$.

From the first two of these properties, it follows at once that $PP = P^2 = P$. From the third property, it is easy to show that the operator P maps each element z to its *nearest* point in $\mathcal{U}$, distance being measured in terms of the norm. This makes projections very important in statistical contexts such as least squares estimation.

A.2.2 Quadratic optimization

Some of our functional data analysis methodology require the solution of a particular kind of constrained optimization problem. Suppose that A is a linear operator on a function space satisfying the condition

$$\langle x, Ay\rangle = \langle Ax, y\rangle \text{ for all } x \text{ and } y.$$

Such an operator is called a *self-adjoint* operator.

Now consider the problem of maximizing $\langle x, Ax\rangle$ subject to the constraint $\|x\| = 1$. In Section A.5, we set out results relating this optimization problem to the eigenfunction/eigenvalue problem $Au = \lambda u$. We then go on to consider the more general problem of maximizing $\langle x, Ax\rangle$ subject to a constraint on $\langle x, Bx\rangle$ for a second self-adjoint operator B.

A.3 Matrix decompositions and generalized inverses

We describe two important matrix decompositions, the singular value decomposition and the QR decomposition. Both of these are standard techniques in numerical linear algebra, and can be carried out within packages such as S-PLUS and MATLAB®. We do not give any details of the way the decompositions are computed; for these see, for example, Golub and Van Loan (1989) or the standard numerical linear algebra package LINPACK (Dongarra et al., 1979).

A.3.1 Singular value decompositions

Suppose $\mathbf{Z}$ is an $m \times n$ matrix. For many purposes it is useful to carry out a *singular value decomposition* (SVD) of $\mathbf{Z}$. This expresses $\mathbf{Z}$ as the product of three matrices

$$\mathbf{Z} = \mathbf{U}\mathbf{D}\mathbf{V}' \tag{A.3}$$

where, for some integer $q \leq \min(m, n)$,

- $\mathbf{U}$ is $m \times q$ and $\mathbf{U}'\mathbf{U} = \mathbf{I}_q$, where $\mathbf{I}_q$ is the identity matrix of order q;
- $\mathbf{D}$ is a $q \times q$ diagonal matrix with strictly positive elements on the diagonal;
- $\mathbf{V}$ is $n \times q$ and $\mathbf{V}'\mathbf{V} = \mathbf{I}_q$.

The diagonal elements $d_1, d_2, \ldots, d_q$ of $\mathbf{D}$ are called the *singular values* of $\mathbf{Z}$, and the SVD can always be carried out in such a way that the diagonal elements $d_1, d_2, \ldots, d_q$ satisfy

$$d_1 \geq d_2 \geq \ldots \geq d_q > 0. \tag{A.4}$$

In this case, the number q is equal to the rank of the matrix $\mathbf{Z}$, i.e., the maximum number of linearly independent rows or columns of $\mathbf{Z}$.

In the special case where $\mathbf{Z}$ is square and symmetric, the requirement that the diagonal elements of $\mathbf{D}$ are necessarily positive is usually dropped, but the matrices $\mathbf{U}$ and $\mathbf{V}$ are chosen to be identical. Furthermore we may allow $q \geq$ rank $\mathbf{Z}$. The d_i then include all the nonzero eigenvalues of $\mathbf{Z}$, together with some or all of the zero eigenvalues if there are any. We have

$$\mathbf{Z} = \mathbf{U}\mathbf{D}\mathbf{U}' \text{ with } \mathbf{U}'\mathbf{U} = \mathbf{I}. \tag{A.5}$$

If, in addition, $\mathbf{Z}$ is positive semi-definite, so that $x'\mathbf{Z}x \geq 0$ for all vectors x, then

$$d_1 \geq d_2 \geq \ldots \geq d_q \geq 0. \tag{A.6}$$

A.3.2 Generalized inverses

Given any $m \times n$ matrix $\mathbf{Z}$, we can define a *generalized inverse* or *g-inverse* of $\mathbf{Z}$ to be any $n \times m$ matrix $\mathbf{Z}^-$ such that

$$\mathbf{Z}\mathbf{Z}^-\mathbf{Z} = \mathbf{Z}. \tag{A.7}$$

If $m = n$ and $\mathbf{Z}$ is an invertible matrix, then it follows from (A.7) that $\mathbf{Z}^{-1}$ is a g-inverse of $\mathbf{Z}$. Furthermore, by pre and post multiplying (A.7) by $\mathbf{Z}^{-1}$, we see that $\mathbf{Z}^{-1}$ is the *unique* g-inverse of $\mathbf{Z}$ in this case.

In the more general case, the matrix $\mathbf{Z}^-$ is not generally unique, but a particular g-inverse, called the *Moore-Penrose g-inverse* $\mathbf{Z}^+$ can always be calculated using the singular value decomposition (A.3) of the matrix $\mathbf{Z}$. Set

$$\mathbf{Z}^+ = \mathbf{V}\mathbf{D}^{-1}\mathbf{U}'. \tag{A.8}$$

It is easy to check that $\mathbf{Z}^+$ is a g-inverse of $\mathbf{Z}$ and also that

$$\mathbf{Z}^+\mathbf{Z}\mathbf{Z}^+ = \mathbf{Z}^+ \text{ and } \mathbf{Z}\mathbf{Z}^+ = \mathbf{U}\mathbf{U}'. \tag{A.9}$$

A.3.3 The QR decomposition

The *QR decomposition* of an $m \times n$ matrix $\mathbf{Z}$ is a different decomposition that yields the expression

$$\mathbf{Z} = \mathbf{Q}\mathbf{R},$$

where $\mathbf{Q}$ is an $m \times m$ orthogonal matrix (so that $\mathbf{Q}'\mathbf{Q} = \mathbf{Q}\mathbf{Q}' = \mathbf{I}$) and $\mathbf{R}$ is an $m \times n$ upper-triangular matrix (so that $\mathbf{R}_{ij} = 0$ if $i > j$).

If $m > n$ then the last $(m - n)$ rows of $\mathbf{R}$ will be zero, and each of the last $(m - n)$ columns x of $\mathbf{Q}$ will satisfy $x'\mathbf{Z} = 0$. Dropping these rows and columns will yield a *restricted QR decomposition* $\mathbf{Z} = \mathbf{Q}_1\mathbf{R}_1$ where $\mathbf{R}_1$ is an $n \times n$ upper-triangular matrix and $\mathbf{Q}$ is an $m \times n$ matrix of orthonormal columns.

A.4 Projections

In discussing the key concept of *projection*, we first consider projection matrices in m-dimensional spaces, and then go on to consider more general inner product spaces.

A.4.1 Projection matrices

Suppose that an $m \times m$ matrix $\mathbf{P}$ has the property that $\mathbf{P}^2 = \mathbf{P}$. Define $\mathcal{P}$ to be the subspace of R^m spanned by the columns of $\mathbf{P}$. The matrix $\mathbf{P}$ is then called a *projection matrix* onto the subspace $\mathcal{P}$. The following two properties, which are easily checked, give the reason for this definition:

1. Every m-vector $\mathbf{z}$ is mapped by $\mathbf{P}$ into the subspace $\mathcal{P}$.

2. If $\mathbf{z}$ is already a linear combination of columns of $\mathbf{P}$, so that $\mathbf{z} = \mathbf{P}\mathbf{u}$ for some vector $\mathbf{u}$, then $\mathbf{P}\mathbf{z} = \mathbf{z}$.

If $\mathbf{P}$ is a symmetric matrix, then $\mathbf{P}$ is called an *orthogonal* projection matrix, and will have several nice properties. For example, for any vector $\mathbf{z}$ we have

$$(\mathbf{P}\mathbf{z})'\{(\mathbf{I}-\mathbf{P})\mathbf{z}\} = \mathbf{z}'\mathbf{P}'(\mathbf{I}-\mathbf{P})\mathbf{z} = \mathbf{z}'(\mathbf{P}\mathbf{z}-\mathbf{P}^2\mathbf{z}) = 0.$$

This means that the projected vector $\mathbf{P}\mathbf{z}$ and the 'residual vector' $\mathbf{z}-\mathbf{P}\mathbf{z}$ are orthogonal to one another, in the usual Euclidean sense. Furthermore, suppose $\mathbf{v}$ is any vector in $\mathcal{P}$. Then, by a very similar argument,

$$\mathbf{v}'(\mathbf{z}-\mathbf{P}\mathbf{z}) = (\mathbf{P}\mathbf{v})'(\mathbf{I}-\mathbf{P})\mathbf{z} = \mathbf{v}'\mathbf{P}(\mathbf{I}-\mathbf{P})\mathbf{z} = 0,$$

so that the residual vector is orthogonal to all vectors in $\mathcal{P}$.

Suppose that $\mathbf{w}$ is any vector in $\mathcal{P}$ other than $\mathbf{P}\mathbf{z}$. Then $\mathbf{w}-\mathbf{P}\mathbf{z}$ is also in $\mathcal{P}$ and therefore is orthogonal to $\mathbf{z}-\mathbf{P}\mathbf{z}$. Defining $\langle x, y\rangle = x'y$ and $\|x\|$ to be the usual Euclidean inner product and norm, we then have

$$\begin{aligned} \|\mathbf{z}-\mathbf{w}\|^2 &= \|\mathbf{z}-\mathbf{P}\mathbf{z}\|^2 + \|\mathbf{P}\mathbf{z}-\mathbf{w}\|^2 + 2\langle \mathbf{z}-\mathbf{P}\mathbf{z}, \mathbf{P}\mathbf{z}-\mathbf{w}\rangle \\ &= \|\mathbf{z}-\mathbf{P}\mathbf{z}\|^2 + \|\mathbf{P}\mathbf{z}-\mathbf{w}\|^2 > \|\mathbf{z}-\mathbf{P}\mathbf{z}|^2. \end{aligned} \tag{A.10}$$

This means that $\mathbf{P}\mathbf{z}$ is the closest point to $\mathbf{z}$ in the subspace $\mathcal{P}$. Thus orthogonal projections onto a subspace have the property of mapping each vector to the nearest point in the subspace.

More generally, if the inner product is $\langle \mathbf{x}, \mathbf{y}\rangle = \mathbf{x}'\mathbf{W}\mathbf{y}$, and if $\mathbf{P}$ is a projection onto the space $\mathcal{P}$ such that $\mathbf{W}\mathbf{P}$ is symmetric, then $\mathbf{P}$ is orthogonal with respect to this inner product, meaning that $\langle \mathbf{P}\mathbf{z}, \mathbf{z}-\mathbf{P}\mathbf{z}\rangle = 0$ and $\langle \mathbf{v}, \mathbf{z}-\mathbf{P}\mathbf{z}\rangle = 0$ for all $\mathbf{v}$ in $\mathcal{P}$.

A.4.2 Finding an appropriate projection matrix

Now suppose we are not given a projection matrix, but instead we are given a subspace $\mathcal{U}$ of R^m, and we wish to find an orthogonal projection matrix $\mathbf{P}$ that projects onto $\mathcal{U}$.

Let $\mathbf{Z}$ be any matrix whose columns are m-vectors that span the subspace $\mathcal{U}$. There is no need for the columns to be linearly independent. Define $\mathbf{P}$ by

$$\mathbf{P} = \mathbf{Z}\mathbf{Z}^{-}.$$

It is straightforward to show that $\mathbf{P}$ is a projection onto the subspace $\mathcal{U}$ as required.

In order to get an *orthogonal* projection, define $\mathbf{P}$ using the Moore-Penrose g-inverse $\mathbf{Z}^{+}$. Then, in terms of the SVD of $\mathbf{Z}$, we have $\mathbf{P} = \mathbf{U}\mathbf{U}'$, so that $\mathbf{P}$ is a symmetric matrix and hence an orthogonal projection.

A.4.3 Projections in more general inner product spaces

We can extend these ideas to projections in more general inner product spaces as discussed in Section A.2.1. As in that section, let $u_1, \ldots, u_n$ be any n elements of our space, and let u be the n-vector whose elements are the $u_1, \ldots, u_n$. Let $\mathcal{U}$ be the subspace consisting of all possible linear combinations $c'u$ for real n-vectors c. Suppose that P is an orthogonal projection onto $\mathcal{U}$ as specified in Section A.2.1. The proof that P maps each element z to the nearest member $P\mathbf{z}$ of $\mathcal{U}$ is identical to the argument given in (A.10) because that depends only on the defining properties of an inner product and associated norm.

How are we to find an orthogonal projection of this kind? Extend our notation to define $\mathbf{K} = \langle u, u' \rangle$ to be the symmetric $n \times n$ matrix with elements $\langle u_i, u_j \rangle$. Given any real n-vector x, we have $x'\mathbf{K}x = \langle x'u, u'x \rangle = \|x'u\|^2 \geq 0$, so the matrix $\mathbf{K}$ is positive semi-definite.

Define the operator P by

$$Pz = u'\mathbf{K}^{+}\langle u, z \rangle$$

for all z. By definition Pz is a linear combination of the elements of u and hence is in $\mathcal{P}$. We shall show that P is an orthogonal projection onto $\mathcal{P}$.

If y is in $\mathcal{P}$, then $y = u'c$ for some real vector c, so that $Py = u'\mathbf{K}^{+}\mathbf{K}c$, and $y - Py = u'd$ where $d = (\mathbf{I} - \mathbf{K}^{+}\mathbf{K})c$. Therefore, since $\mathbf{K}\mathbf{K}^{+}\mathbf{K} = \mathbf{K}$,

$$\|y - Py\|^2 = d'\mathbf{K}d = d'(\mathbf{K} - \mathbf{K}\mathbf{K}^{+}\mathbf{K})c = 0,$$

implying that $\|y - Py\|^2 = 0$ and $Py = y$.

Finally, given any v in $\mathcal{P}$, and any z, use the property (A.9) to show that

$$\begin{aligned} \langle Pz - v, Pz \rangle &= \langle P(z - v), Pz \rangle = \langle z - v, u' \rangle \mathbf{K}^{+}\mathbf{K}\mathbf{K}^{+}\langle u, z \rangle \\ &= \langle z - v, u' \rangle \mathbf{K}^{+}\langle u, z \rangle = \langle Pz - v, z \rangle \end{aligned}$$

and therefore that $\langle Pz - v, z - Pz\rangle = 0$, completing the proof that P is the required orthogonal projection onto $\mathcal{P}$.

A.5 Constrained maximization of a quadratic function

A.5.1 The finite-dimensional case

Suppose that $\mathbf{A}$ is a symmetric $p \times p$ matrix. An important result in linear algebra concerns the constrained maximization problem

$$\max \mathbf{x}'\mathbf{A}\mathbf{x} \text{ for } p\text{-vectors } \mathbf{x} \text{ subject to } \mathbf{x}'\mathbf{x} = 1. \tag{A.11}$$

Let $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$ be the eigenvalues of $\mathbf{A}$, and let $\mathbf{u}_i$ be the corresponding eigenvectors, each normalized to have $\|\mathbf{u}_i\| = 1$. Let $\mathbf{U}$ be the matrix whose columns are the eigenvectors $\mathbf{u}_i$ and $\mathbf{D}$ be the diagonal matrix with diagonal elements λ_i. We then have $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{U}'$, and $\mathbf{U}\mathbf{U}' = \mathbf{U}'\mathbf{U} = \mathbf{I}$.

Set $\mathbf{y} = \mathbf{U}'\mathbf{x}$ in (A.11), so that $\mathbf{x} = \mathbf{U}\mathbf{y}$. We have $\mathbf{x}'\mathbf{x} = \mathbf{y}'\mathbf{U}'\mathbf{U}\mathbf{y} = \mathbf{y}'\mathbf{y}$, so the constraint $\mathbf{x}'\mathbf{x} = 1$ is equivalent to $\mathbf{y}'\mathbf{y} = 1$. Therefore, in terms of $\mathbf{y}$, the maximization problem (A.11) can be rewritten as

$$\max \mathbf{y}'\mathbf{D}\mathbf{y} \text{ for } p\text{-vectors } \mathbf{y} \text{ subject to } \mathbf{y}'\mathbf{y} = 1. \tag{A.12}$$

This is clearly solved by setting $\mathbf{y}$ to be the vector $(1, 0, \ldots, 0)'$, so that $\mathbf{x}$ is the first column of $\mathbf{U}$, in other words the leading normalized eigenvector $\mathbf{u}_1$ of $\mathbf{A}$.

By an extension of this argument, we can characterize all the eigenvectors of $\mathbf{A}$ as solutions of successive optimization problems. The jth normalized eigenvector $\mathbf{u}_j$ solves the problem (A.11) subject to the additional constraint of being orthogonal to all the solutions found so far:

$$\max \mathbf{x}'\mathbf{A}\mathbf{x} \text{ subject to } \mathbf{x}'\mathbf{x} = 1 \text{ and } \mathbf{x}'\mathbf{u}_1 = \mathbf{x}'\mathbf{u}_2 = \ldots = \mathbf{x}'\mathbf{u}_{j-1} = 0. \tag{A.13}$$

Setting $\mathbf{x} = \mathbf{u}_j$, we have $\mathbf{x}'\mathbf{A}\mathbf{x} = \lambda_j \mathbf{u}_j'\mathbf{u}_j = \lambda_j$, the jth eigenvalue.

A.5.2 The problem in a more general space

Now suppose we are working within a more general inner product space. The role of a symmetric matrix is now played by a self-adjoint linear operator A, that is, one satisfying the condition

$$\langle x, Ay\rangle = \langle Ax, y\rangle \text{ for all } x \text{ and } y.$$

We shall assume that A is a completely continuous (or compact) symmetric transformation on a Hilbert space; there is no need at all for the reader to understand what this means, but anyone interested is referred to Aubin (2000) or any other standard text on functional analysis. The reader can

always take it on trust that the assumptions are satisfied when we appeal to the results of this section.

The problem

$$\max\langle x, Ax\rangle \text{ subject to } \|x\| = 1 \tag{A.14}$$

corresponds to the maximization problem (A.11), and we can define a sequence u_j as the solutions to the succession of optimization problems

$$\max\langle x, Ax\rangle \text{ subject to } \|x\| = 1 \text{ and } \langle x, u_i\rangle = 0 \text{ for } i < j. \tag{A.15}$$

Under the conditions referred to above, these optimization problems can be solved by considering the eigenfunction problem

$$Au = \lambda u$$

and normalizing the eigenfunctions u to satisfy $\|u\| = 1$. Suppose the eigenvalues are $\lambda_1 \geq \lambda_2 \geq \ldots$ with eigenfunctions $u_1, u_2, \ldots$. Then the leading eigenfunction u_1 solves the optimization problem (A.14) and the value of the maximum is λ_1. The successive eigenfunctions u_j solve the constrained problem (A.15), and the maximum at the jth stage is $\langle u_j, Au_j\rangle = \lambda_j \|u_j\|^2 = \lambda_j$.

A.5.3 Generalized eigenproblems

We sometimes wish to modify the optimization problems we have considered by the introduction of a positive definite symmetric matrix $\mathbf{B}$ into the constraints, replacing the constraint $\|\mathbf{x}\| = 1$ by $\mathbf{x}'\mathbf{B}\mathbf{x} = 1$ or, more generally, $\langle \mathbf{x}, \mathbf{B}\mathbf{x}\rangle = 1$, and similarly defining orthogonality with respect to the matrix $\mathbf{B}$.

Consider the solutions of the *generalized eigenproblem*

$$Av = \rho Bv,$$

where v is either a function or a vector, and A and B are corresponding linear operators acting on V. We normalize the solutions to satisfy $\langle v, Bv\rangle = 1$. Suppose the solutions are $v_1, v_2, \ldots$, with corresponding generalized eigenvalues $\rho_1 \geq \rho_2 \geq \ldots$. Under suitable conditions, which are always satisfied in the finite-dimensional case, and are analogous to those noted above for more general spaces, the leading generalized eigenvector or function v_1 solves the problem

$$\max\langle v, Av\rangle \text{ subject to } \langle v, Bv\rangle = 1, \tag{A.16}$$

and the maximizing value is equal to ρ_1. The jth generalized eigenvector or function v_j solves the problem

$$\max\langle v, Av\rangle \text{ subject to } \langle v, Bv\rangle = 1 \text{ and } \langle v, Bv_i\rangle = 0 \text{ for } i < j$$

and the maximizing value is ρ_j.

Finally, we note that the problem of maximizing the ratio

$$\frac{\langle v, Av\rangle}{\langle v, Bv\rangle} \tag{A.17}$$

for $v \neq 0$ is equivalent to that of maximizing $\langle v, Av\rangle$ subject to the constraint $\langle v, Bv\rangle = 1$. To see this, note that scaling any v to satisfy the constraint does not affect the value of the ratio (A.17), and so the maximum of the ratio is unaffected by the imposition of the constraint. Once the constraint is imposed, the denominator of (A.17) is equal to 1, and so maximizing the ratio subject to $\langle v, Bv\rangle = 1$ is exactly the same as the original maximization problem (A.16).

A.6 Kronecker Products

Let $\mathbf{A}$ be an m by n matrix and let $\mathbf{B}$ be a p by q matrix. The Kronecker product $\mathbf{A}\otimes\mathbf{B}$ is the super or composite matrix of order mp by nq consisting of sub-matrices $a_{ij}\mathbf{B}$. That is,

$$\mathbf{A}\otimes\mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \dots & a_{1n}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \dots & a_{2n}\mathbf{B} \\ \vdots & \vdots & & \vdots \\ a_{m1}\mathbf{B} & a_{m2}\mathbf{B} & \dots & a_{mn}\mathbf{B} \end{bmatrix}.$$

One of the most common applications of the Kronecker product is to express a linear equation of the form

$$\mathbf{AXB}' = \mathbf{C},$$

which cannot be solved for $\mathbf{X}$ by conventional matrix algebra, in the form

$$(\mathbf{B}\otimes\mathbf{A})\text{vec}\,(\mathbf{X}) = \text{vec}\,(\mathbf{C}),$$

where vec $(\mathbf{X})$ indicates the vector of length nq obtained by writing matrix $\mathbf{X}$ as a vector column-wise, and, in the same way, vec $(\mathbf{C})$ indicates the vector of length mp obtained by writing matrix $\mathbf{C}$ as a vector column-wise. Then we can express the solution directly as

$$\text{vec}\,(\mathbf{X}) = (\mathbf{B}\otimes\mathbf{A})^{-1}\text{vec}\,(\mathbf{C}),$$

provided that, of course, matrix $\mathbf{B}\otimes\mathbf{A}$ is nonsingular.

The Kronecker product is *bilinear* in the sense that

$$\text{vec}\,(\mathbf{A}_1\mathbf{XB}_1' + \mathbf{A}_2\mathbf{XB}_2') = (\mathbf{B}_1\otimes\mathbf{A}_1 + \mathbf{B}_2\otimes\mathbf{A}_2)\text{vec}\,(\mathbf{X}).$$

Other useful relations for simplifying expressions involving Kronecker products are

$$\mathbf{A}\otimes(\mathbf{B}\otimes\mathbf{C}) \quad = \quad (\mathbf{A}\otimes\mathbf{B})\otimes\mathbf{C}$$

$$\begin{aligned}
(\mathbf{A} \otimes \mathbf{B})' &= \mathbf{A}' \otimes \mathbf{B}' \\
(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) &= (\mathbf{AC}) \otimes (\mathbf{BD}) \\
(\mathbf{A} + \mathbf{B}) \otimes \mathbf{C} &= (\mathbf{A} \otimes \mathbf{C}) + (\mathbf{B} \otimes \mathbf{C}) \\
\mathbf{A} \otimes (\mathbf{B} + \mathbf{C}) &= (\mathbf{A} \otimes \mathbf{B}) + (\mathbf{A} \otimes \mathbf{C}) \\
\operatorname{trace}(\mathbf{A} \otimes \mathbf{B}) &= (\operatorname{trace}\mathbf{A})(\operatorname{trace}\mathbf{B}),
\end{aligned}$$

Finally, if both $\mathbf{A}$ and $\mathbf{B}$ are nonsingular, then

$$(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}.$$

A.7 The multivariate linear model

We now return to a more statistical topic. A review of the multivariate linear model may be helpful, both to fix ideas and notation, and because some of the essential concepts transfer without much more than a change of notation to functional contexts. But a slight change of perspective is helpful on what the design matrix means. Moreover, a notion that is used repeatedly for functional data is *regularization*, and we introduce regularization in Section A.8 within the multivariate context.

A.7.1 Linear models from a transformation perspective

Let $\mathbf{Y}$ be a $N \times p$ matrix of dependent variable observations, $\mathbf{Z}$ be a $N \times q$ matrix, and $\mathbf{B}$ be a $q \times p$ matrix. In classical terminology, $\mathbf{Z}$ is the *design matrix* and $\mathbf{B}$ is a matrix of *parameters*.

The multivariate linear model is

$$\mathbf{Y} = \mathbf{ZB} + \mathbf{E}. \tag{A.18}$$

The rows of the disturbance or residual matrix $\mathbf{E}$ are often thought of, at least at the population level, as independent samples from a common population of p-variate observations with mean 0 and finite covariance matrix $\mathbf{\Sigma}$.

Although in many contexts it is appropriate to think of the columns of $\mathbf{Z}$ as corresponding to variables, it is better for our purposes to take the more general view that $\mathbf{Z}$ represents a linear transformation that maps matrices $\mathbf{B}$ into matrices with the dimensions of $\mathbf{Y}$. This can be indicated by the notation

$$\mathbf{Z} : R^{q \times p} \to R^{N \times p}.$$

The space of all possible transformed values $\mathbf{ZB}$ then defines a subspace of $R^{N \times p}$, denoted by $R(\mathbf{Z})$, and is called the *range space* of $\mathbf{Z}$.

A.7.2 The least squares solution for $\mathbf{B}$

When it is assumed that the rows of the disturbance matrix $\mathbf{E}$ are independent, each with covariance matrix $\mathbf{\Sigma}$, the natural inner product to use in the observation space $R^{N\times p}$ is

$$\langle \mathbf{X}, \mathbf{Y}\rangle = \operatorname{trace} \mathbf{X}\mathbf{\Sigma}^{-1}\mathbf{Y}' = \operatorname{trace} \mathbf{Y}'\mathbf{X}\mathbf{\Sigma}^{-1} \tag{A.19}$$

for $\mathbf{X}$ and $\mathbf{Y}$ in $R^{N\times p}$. We then measure the goodness of fit of any parameter matrix $\mathbf{B}$ to the observed data $\mathbf{Y}$ making use of the corresponding norm

$$\texttt{LMSSE}(\mathbf{B}) = \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|^2 = \operatorname{trace}\,(\mathbf{Y} - \mathbf{Z}\mathbf{B})'\mathbf{\Sigma}^{-1}(\mathbf{Y} - \mathbf{Z}\mathbf{B}). \tag{A.20}$$

Suppose, for the moment, that the matrix $\mathbf{Z}$ is of full column rank, or that $N \geq q$ and the columns of $\mathbf{Z}$ are independent. A central result on the multivariate linear model is that the matrix $\hat{\mathbf{B}}$ that minimizes $\texttt{LMSSE}(\mathbf{B})$ is given by

$$\hat{\mathbf{B}} = (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{Y}. \tag{A.21}$$

The corresponding predictor of $\mathbf{Y}$ is given by

$$\hat{\mathbf{Y}} = \mathbf{Z}\hat{\mathbf{B}} = \mathbf{Z}(\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{Y}. \tag{A.22}$$

The matrix $\hat{\mathbf{Y}}$ can be thought of as the matrix in the subspace $R(\mathbf{Z})$ that minimizes $\|\mathbf{Y} - \hat{\mathbf{Y}}\|^2$ over all possible approximations $\hat{\mathbf{Y}} = \mathbf{Z}\mathbf{B}$ falling in $R(\mathbf{Z})$.

Note that the least squares estimator $\hat{\mathbf{B}}$ and the best linear predictor $\hat{\mathbf{Y}}$ do not depend on the variance matrix $\mathbf{\Sigma}$, even though the fitting criterion $\texttt{LMSSE}(\mathbf{B})$ does. It turns out that when the details of the minimization of $\texttt{LMSSE}(\mathbf{B})$ are carried through, the variance matrix $\mathbf{\Sigma}$ cancels out. But if there are covariances among errors or residuals *across* observations, contained in a variance-covariance matrix $\mathbf{\Gamma}$, say, then the inner product (A.19) becomes

$$\langle \mathbf{X}, \mathbf{Y}\rangle = \operatorname{trace}\, \mathbf{Y}'\mathbf{\Gamma}^{-1}\mathbf{X}\mathbf{\Sigma}^{-1}.$$

Using this inner product in the definition of goodness of fit, the estimator of $\mathbf{B}$ and the best predictor of $\mathbf{Y}$ becomes

$$\hat{\mathbf{B}} = (\mathbf{Z}'\mathbf{\Gamma}^{-1}\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{\Gamma}^{-1}\mathbf{Y}$$

and

$$\hat{\mathbf{Y}} = \mathbf{Z}(\mathbf{Z}'\mathbf{\Gamma}^{-1}\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{\Gamma}^{-1}\mathbf{Y}.$$

Thus, the optimal solution does depend on how one treats errors *across* observations.

A.8 Regularizing the multivariate linear model

One of the major themes of this book is *regularization*, and for readers familiar with multivariate analysis, it may be helpful to introduce this idea in the multivariate context first. Others, especially those who are familiar with curve estimation already, may prefer to omit this section.

Suppose now that we are dealing with an under-determined problem, where $q > N$ and the matrix $\mathbf{Z}$ is of full row rank N. This means that the range space $R(\mathbf{Z})$ is the whole of $R^{N\times p}$.

A.8.1 Definition of regularization

Regularization involves attaching a *penalty term* to the basic squared error fitting criterion:

$$\texttt{LMSSE}_\lambda(\mathbf{B}) = \|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|^2 + \lambda \times \texttt{PEN}(\mathbf{B}). \tag{A.23}$$

The purpose of the penalty term $\texttt{PEN}(\mathbf{B})$ is to require that the estimated value of $\mathbf{B}$ not only yields a good fit in the sense of small $\|\mathbf{Y} - \mathbf{Z}\mathbf{B}\|^2$, but also that some aspect of $\mathbf{B}$ captured in the function PEN is kept under control. The positive penalty parameter λ quantifies the relative importance of these two aims. If λ is large, then we are particularly concerned with keeping $\texttt{PEN}(\mathbf{B})$ small, and getting a good fit to the data is only of secondary importance; if λ is small, then we are not so concerned about the value of $\texttt{PEN}(\mathbf{B})$.

One example of this type of regularization is the *ridge regression* technique, often used to stabilize regression coefficient estimates in the presence of highly collinear independent variables. In this case, what is penalized is the size of the regression coefficients themselves, in the sense that $\texttt{PEN}(\mathbf{B}) = \text{trace}(\mathbf{B}'\mathbf{B})$, the sum of squares of the entries of $\mathbf{B}$. The solution to the minimization of $\texttt{LMSSE}_\lambda(\mathbf{B})$ is then

$$\mathbf{B} = (\mathbf{Z}'\mathbf{Z} + \lambda\mathbf{I})^{-1}\mathbf{Z}'\mathbf{Y}.$$

As λ approaches zero, $\mathbf{B}$ approaches the least squares solution described in Section A.7, but as λ grows, $\mathbf{B}$ approaches zero. Thus, ridge regression is said to shrink the solution towards zero.

A.8.2 Hard-edged constraints

One way to obtain a well-determined problem is to place constraints on the matrix $\mathbf{B}$. For example, consider the model where it is assumed that the coefficients in each column of $\mathbf{B}$ are a constant vector, so all we have to do is to estimate a single number for each column. If we define the $(q-1)\times q$ matrix $\mathbf{L}$ to have $L_{ii} = 1$ and $L_{i,i+1} = -1$ for each i, and all other entries

zero, then our assumption about $\mathbf{B}$ can be written as the constraint

$$\mathbf{LB} = 0. \tag{A.24}$$

In order for the elements of $\mathbf{B}$ to be identifiable on the basis of the observed data, the design matrix $\mathbf{Z}$ has to satisfy the condition

$$\mathbf{Z1} \neq 0, \tag{A.25}$$

where $\mathbf{1}$ is a vector of q unities.

The transformation $\mathbf{L}$ reduces multiples of the vector $\mathbf{1}$ exactly to zero. The identifiability condition (A.25) can be replaced by the condition that the zero vector is the only q-vector $\mathbf{b}$ such that both $\mathbf{Lb}$ and $\mathbf{Zb}$ are zero. Equivalently, the matrix $[\mathbf{Z}' \ \ \mathbf{L}']$ is nonsingular.

A.8.3 Soft-edged constraints

Instead of enforcing the hard-edged constraint $\mathbf{LB} = 0$, we may wish to let the coefficients in any column of $\mathbf{B}$ vary, but not more than really necessary, by exploring compromises between the rank-one extreme implied by (A.24) and a completely unconstrained underdetermined fit. We might consider this a soft-edged constraint, and it can be implemented by a suitable regularization procedure. If we define

$$\texttt{PEN}(\mathbf{B}) = \|\mathbf{LB}\|^2 = \text{trace}(\mathbf{B}'\mathbf{L}'\mathbf{LB}) \tag{A.26}$$

then the penalty $\texttt{PEN}(\mathbf{B})$ quantifies how far the matrix $\mathbf{B}$ is from satisfying the constraint $\mathbf{LB} = 0$.

The regularized estimate of $\mathbf{B}$, obtained by minimizing the criterion (A.23), now satisfies

$$(\mathbf{Z}'\mathbf{Z} + \lambda\mathbf{L}'\mathbf{L})\mathbf{B} = \mathbf{Z}'\mathbf{Y}. \tag{A.27}$$

For any $\lambda > 0$, a unique solution for $\mathbf{B}$ requires the nonsingularity of the matrix $[\mathbf{Z}'\,\mathbf{L}']$, precisely the condition for identifiability of the model subject to the constraint (A.24).

In the limit as the parameter $\lambda \to \infty$, the penalized fitting criterion (A.23) automatically enforces on $\mathbf{B}$ the one-dimensional structure $\mathbf{LB} = 0$. On the other hand, in the limit $\lambda \to 0$, no penalty at all is applied, and $\mathbf{B}$ takes on whatever value results in minimizing the error sum of squares to zero, due to the underdetermined character of the problem. Thus, from the regularization perspective, the constrained estimation problem $\mathbf{LB} = 0$ that arises frequently in linear modelling designs is simply an extreme case of the regularization process where $\lambda \to \infty$.

We have concentrated on a one-dimensional constrained model, corresponding to a $(q-1)\times q$ matrix $\mathbf{L}$, but of course the ideas can be immediately extended to nonsingular $s \times q$ constraint matrices $\mathbf{L}$ that map a q-vector into a space of vectors of dimension $s \leq q$. In this case, the constrained model is of dimension $q - s$. Note also that the specification of the matrix $\mathbf{L}$

corresponding to any particular constrained model is not unique, and that if $\mathbf{L}$ is specified differently the regularized estimates are in general different.

Finally, we note in passing that Bayesian approaches to regression, in which a multivariate normal prior distribution is proposed for $\mathbf{B}$, can also be expressed in terms of a penalized least squares problem of the form (A.23). For further details see, for example, Kimeldorf and Wahba (1970), Wahba (1978) or Silverman (1985).

References

Abraham, C., Cornillion, P. A., Matzner–Lober, E. and Molinari, N. (2003) Unsupervised curve-clustering using b-splines. *Scandinavian Journal of Statistics,* **30,** 581–595.

Aguilera, A. M., Ocaña, F. A. and Valderrama, M. J. (1999) Forecasting with unequally spaced data by a functional principal component approach, *Test,* **8,** 233-253.

Anderson, T. W. (1984) *An Introduction to Multivariate Statistical Analysis.* Second edition. New York: Wiley.

Anselone, P. M. and Laurent, P. J. (1967) A general method for the construction of interpolating or smoothing spline-functions. *Numerische Mathematik*, **12**, 66–82.

Ansley, C. F. and Kohn, R. (1990) Filtering and smoothing in state space models with partially diffuse initial conditions. *Journal of Time Series Analysis,* **11,** 275–293.

Ansley, C. F., Kohn, R. and Wong, C-M. (1993) Nonparametric spline regression with prior information. *Biometrika,* **80,** 75–88.

Aronszajn, N. (1950) Theory of reproducing kernels. *Transactions of the American Mathematical Society,* **68,** 337–404.

Atteia, M. (1965) Spline-fonctions généralisées. *Comptes Rendus de l'Académie des Sciences Série I: Mathématique*, **261**, 2149–2152.

Aubin, J-P. (2000) *Applied Functional Analysis, Second Edition.* New York: Wiley-Interscience.

Basilevsky, A. (1994) *Statistical Factor Analysis and Related Methods.* New York: Wiley.

Besse, P. (1979) Etude descriptive des processus: Approximation et interpolation. Thèse de troisième cycle, Université Paul-Sabatier, Toulouse.

Besse, P. (1980) Deux exemples d'analyses en composantes principales filtrantes. *Statistique et Analyse des Données*, **3**, 5–15.

Besse, P. (1988) Spline functions and optimal metric in linear principal component analysis. In J. L. A. van Rijkevorsel and J. de Leeuw (eds.) *Component and Correspondence Analysis: Dimensional Reduction by Functional Approximation*. New York: Wiley.

Besse, P. (1991) Approximation spline de l'analyse en composantes principales d'une variable aléatoire hilbertienne. *Annales de la Faculté des Sciences de Toulouse,* **12,** 329–349.

Besse, P. and Ramsay, J. O. (1986) Principal components analysis of sampled functions. *Psychometrika*, **51**, 285–311.

Besse, P. and Cardot, H (1997) Approximation spline de la prévision d'un processus fonctionnel autorégressif d'ordre 1. *Canadian Journal of Statistics,* **24,** 467–487.

Besse, P., Cardot, H. and Ferraty, F. (1997). Simultaneous nonparametric regressions of unbalanced longitudinal data. *Computational Statistics and Data Analysis,* **24,** 255–270.

Besse, P., Cardot, H. and Stephenson, D. (2000) Autoregressive forecasting of some functional climatic variations. *Scandinavian Journal of Statistics,* **27,** 673–687.

Bock, R. D. and Thissen, D. (1980) Statistical problems of fitting individual growth curves. In F. E. Johnston, A. F. Roche and C. Susanne (eds.) *Human Physical Growth and Maturation: Methodologies and Factors*. New York: Plenum.

Bookstein, F. L. (1991) *Morphometric Tools for Landmark Data: Geometry and Biology*. Cambridge: Cambridge University Press.

Bosq, D. (2000) *Linear processes in function spaces*. New York: Springer.

Brumback, B. and Rice, J. A. (1998) Smoothing spline models for the analysis of nested and crossed samples of curves (with discussion). *Journal of the American Statistical Association,* **93,** 961–994.

Buckheit, J. B, Olshen, R. A., Blouch, K. and Myers, B. D. (1997) Modeling of progressive glomerular injury in humans with lupus nephritis. *American Journal of Physiology*, **237**, 158–169.

Buja, A., Hastie, T. and Tibshirani, R. (1989) Linear smoothers and additive models (with discussion). *Annals of Statistics*, **17**, 453–555.

Cai, Z., Fan, J. and Li, R. (2000) Efficient estimation and inferences for varying-coefficient models, *Journal of the American Statistical Association,* **95,** 888–902.

Cai, Z., Fan, J. and Yao, Q. (2000) Functional-coefficient fregression models for non–linear time series, *Journal of the American Statistical Association,* **95,** 941–956.

Cailliez, F. and Pagès, J. P. (1976) *Introduction à l'analyse des données*. Paris: SMASH (Société de Mathétiques Appliquées et de Sciences Humaines.

Cardot, H. (2002) Spatially adaptive splines for statistical linear inverse problems. *Journal of Multivariate Analysis,* **81,** 100–119.

Cardot, H. (2004) Nonparametric estimation of smoothed principal components analysis of sampled noisy functions. *Journal of Nonparametric Statistics,* to appear.

Cardot, H., Ferraty, F. and Sarda, P. (1999) Functional linear model, *Statistics & Probability Letters,* **45,**, 11–22.

Cardot, H., Ferraty, F. and Sarda, P. (2003) Spline estimators for the functional linear model. *Statistica Sinica,* **13,**, 571–591.

Cardot, H., Faivre, R. and Goulard, M. (2003). Functional approaches for predicting land use with the temporal evolution of coarse resolution remote sensing data. *Journal of Applied Statistics,* **30,** 1185–1199.

Cardot, H., Ferraty, F., Mas, A. and Sarda, P. (2003). Testing Hypotheses in the functional linear model. /em Scandinavian Journal of Statistics, **30,** 241–255.

Cardot, H., Faivre, R. and Maisongrande, P. (2004). Random Effects Varying Time Regression Models: Application to Remote Sensing. J. Antoch (ed.) /em Compstat 2004 proceedings, Physica-Verlag, 777–784.

Cardot, H., Crambes, C. and Sarda, P. (2004). Estimation spline de quantiles conditionnels pour variables explicatives fonctionnelles. *Comtes Rendu de l'Académie des Sciences. Paris, serie I,* **339,** 141–144.

Cardot H., Goia, A. and Sarda, P. (2004). Testing for no effect in functional linear regression models, some computational approaches. /em Communications in Statistics – Simulation and Computation, **33,** 179–199.

Cardot, H., Ferraty, F., Mas, A. and Sarda, P. (2004) Testing hypotheses in the functional linear model. *Scandinavian Journal of Statistics,* **30,** 241–255.

Castro, P. E., Lawton, W. H. and Sylvestre, E. A. (1986) Principal modes of variation for processes with continuous sample curves. *Technometrics*, **28**, 329–337.

Char, B. W., Geeddes, K. O., Gonnet, G. H., Leong, B. L., Monagan, M. B. and Watt, S. M. (1991) *MAPLE V Language Reference Manual*. New York: Springer.

Chiou, J.-M., Müller, H.-G. and Wang, J.-L. (2003) Functional quasi-likelihood regression models with smooth random effects. *Journal of the Royal Statistical Society, Series B.,* **65,** 405–423.

Chui, C. K. (1992) *An Introduction to Wavelets*. San Diego: Academic Press.

Cleveland, W. S. (1979) Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, **74**, 829–836.

Coddington, E. A. (1989) *An Introduction to Ordinary Differential Equations*. New York: Dover.

Coddington, E. A. and Levinson, N. (1955) *Theory of Ordinary Differential Equations*. New York: McGraw-Hill.

Cook, R. D. and Weisberg, S. (1982) *Residuals and Influence in Regression*. New York: Chapman and Hall.

Cox, D. R. and Lewis, P. A. W. (1966) *The Statistical Analysis of Series of Events.* London: Methuen.

Craven, P. and Wahba, G. (1979) Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation, *Numerische Mathematik*, **31**, 377–403.

Cressie, N. (1991) *Statistics for Spatial Data.* New York: Wiley.

Cueva, A., Febrero, M. and Fraiman, R. (2002) Linear functional regression: The case of fixed design and functional response. *The Canadian Journal of Statistics,* **30,** 285–300.

Dalzell, C. J. and Ramsay, J. O. (1993) Computing reproducing kernels with arbitrary boundary constraints. *SIAM Journal of Scientific Computing*, **14**, 511–518.

Daubechies, I. (1992) *Ten Lectures on Wavelets.* CBMS-NSF Series in Applied Mathematics, **61**. Philadelphia: Society for Industrial and Applied Mathematics.

Dauxois, J. and Pousse, A. (1976) Les analyses factorielles en calcul des probabilité et en statistique: Essai d'étude synthètique. Thèse d'état, Université Paul-Sabatier, Toulouse.

Dauxois, J., Pousse, A. and Romain, Y. (1982) Asymptotic theory for the principal component analysis of a vector random function: Some applications to statistical inference. *Journal of Multivariate Analysis*, **12**, 136–154.

Dauxois, J. and Nkiet, G. M. (2002) Measure of association for Hilbertian subspaces and some applications. *Journal of Multivariate Analysis,* **82,** 263–297.

de Boor, C. (2001) *A Practical Guide to Splines.* Revised Edition. New York: Springer.

Delves, L. M. and Mohamed, J. L. (1985) *Computational Methods for Integral Equations.* Cambridge: Cambridge University Press.

Deville, J. C. (1974) Méthodes statistiques et numériques de l'analyse harmonique. *Annales de l'INSEE*, **15**, 7–97.

Diggle, P. J., Liang, K.-Y. and Zeger, S. L. (1994) *Analysis of Longitudinal Data.* New York: Oxford University Press.

Dongarra, J. J., Bunch, J. R., Moler, C. B. and Stewart, G. W. (1979) *LINPACK Users' Guide.* Philadelphia: Society for Industrial and Applied Mathematics.

Donoho, D. L., Johnstone, I. M., Kerkyacharian, G. and Picard, D. (1995) Wavelet shrinkage: asymptopia? (with discussion). *Journal of the Royal Statistical Society*, Series B, **57**, 301–369.

Draper, N. R. and Smith, H. (1998) *Applied Regression Analysis. Third Edition.* New York: Wiley.

Dreesman, J. M. and Tutz, G. (2001) Non–stationary conditional models for spatial data based on varying coefficients, *The Statistician,* **50,** 1–15.

Eaton, M. L. (1983) *Multivariate Statistics: A Vector Space Approach.* New York: Wiley.

Eaton, M. L. and Perlman, M. D. (1973) The non-singularity of generalized sample covariance matrices. *Annals of Statistics*, **1**, 710–717.

Efron, B. and Tibshirani, R. J. (1992) *An Introduction to the Bootstrap.* London: Chapman & Hall.

Eilers, P. H. C. and Marx, B. D. (1996) Flexible smoothing with B-splines and penalties, with comments. *Statistical Science,* **11**, 89–121.

Engle, R. F., Granger, C. W. J., Rice, J. A. and Weiss, A. (1986) Semiparametric estimates of the relation between weather and electricity sales. *Journal of the American Statistical Association*, **81**, 310–320.

Escabias, M., Aguilera, A. M. and Valderrama, M. J. (2004) Principal component estimation of functional logistic regression: Discussion of two different approaches. *Nonparametric Statistics,* In press.

Eubank, R. L. (1999) *Spline Smoothing and Nonparametric Regression, Second Edition.* New York: Marcel Dekker.

Eubank, R. L., Muñoz Maldonado, Y., Wang, N. and Wang, S. (2004) Smoothing spline estimation in varying coefficient models. *Journal of the Royal Statistical Society, Series B.,* **66,** 653–667.

Falkner, F. (ed.) (1960) *Child Development: An International Method of Study.* Basel: Karger.

Fan, J. and Gijbels, I. (1996) *Local Polynomial Modelling and its Applications.* London: Chapman and Hall.

Fan, J. and Lin, S.-K. (1998) Tests of significance when data are curves. *Journal of the Americal Statistical Association,* **93,** 1007–1021.

Fan, J., Yao, Q. and Cai, Z. (2003) Adaptive varying-coefficient linear models, *Journal of the Royal Statistical Society, Series B,* **65,** 57–80.

Faraway, J. J. (1997) Regression analysis for a functional response, *Technometrics,* **39,** 254–261.

Ferraty, F. and Vieu, P. (2001) The functional nonparametric model and its applications to spectrometric data. *Computational Statistics,* **17,** 545–564.

Ferraty, F., Goia, A. and Vieu, P. (2002) Functional nonparametric model for time series: A fractal approach for dimension reduction. *Test,* **11,** 317–344.

Frank, I. E. and Friedman, J. H. (1993) A statistical view of some chemometrics regression tools. *Technometrics,* **35**, 109–135.

Friedman, J. H. and Silverman, B. W. (1989) Flexible parsimonious smoothing and additive modeling (with discussion). *Journal of the American Statistical Association*, **31**, 1–39.

Gasser, T. and Kneip, A. (1995). Searching for structure in curve samples. *Journal of the American Statistical Association*, **90**, 1179–1188.

Gasser, T., Kneip, A., Binding, A., Prader, A. and Molinari, L. (1991a) The dynamics of linear growth in distance, velocity and acceleration. *Annals of Human Biology*, **18**, 187–205.

Gasser, T., Kneip, A., Ziegler, P., Largo, R., Molinari, L., and Prader, A. (1991b) The dynamics of growth of height in distance, velocity and acceleration. *Annals of Human Biology*, **18**, 449–461.

Gasser, T., Kneip, A., Ziegler, P., Largo, R. and Prader, A. (1990) A method for determining the dynamics and intensity of average growth. *Annals of Human Biology*, **17**, 459–474.

Gasser, T. and Müller, H.-G., (1979) Kernel estimation of regression functions. In T. Gasser and M. Rosenblatt (eds.) *Smoothing Techniques for Curve Estimation*. Heidelberg: Springer, pp. 23–68.

Gasser, T. and Müller, H.-G., (1984) Estimating regression functions and their derivatives by the kernel method. *Scandinavian Journal of Statistics*, **11**, 171–185.

Gasser, T. , Müller, H.-G., Köhler, W., Molinari, L. and Prader, A. (1984). Nonparametric regression analysis of growth curves. *Annals of Statistics*, **12**, 210–229.

Gauss, C. F. (1809) *Theoria motus corporum celestium*. Hamburg: Perthes et Besser.

Gelfand, A. E., Kim, H-J., Sirmans, C.F. and Banerjee, S. (2003) Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association,* **98,** 387–396.

Gervini, D. and Gasser, T. (2004) Self–modeling warping functions. *Journal of the Royal Statistical Society, Series B,* **66,** 959–971.

Golub, G. and Van Loan, C. F. (1989) *Matrix Computations*. Second edition. Baltimore: Johns Hopkins University Press.

Grambsch, P. M., Randall, B. L. Bostick, R. M., Potter, J. D. and Louis, T. A. (1995) Modeling the labeling index distribution: an application of functional data analysis. *Journal of the Americal Statistical Association*, **90**, 813–821.

Green, G. (1828) *An Essay on the Mathematical Analysis to the Theories of Electricity and Magnetism.* Privately printed.

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

Greenhouse, S. W. and Geisser, S. (1959) On methods in the analysis of profile data. *Psychometrika*, **24**, 95–112.

Grenander, U. (1981) *Abstract Inference*. New York: Wiley.

Gu, C. (2002) *Smoothing Spline ANOVA Models*. New York: Springer.

Guo, W. (2002) Functional mixed effects models. *Biometrics,* **58,** 121–128.

Hall, P. and Heckman, N. E. (2002) Estimating and depicting the structure of a distribution of random runctions. *Biometrika,* **89,** 145–158.

Hanson, M. H. and Kooperberg, C. (2002) Spline adaptation in extended linear models, with discussion. *Statistical Science,* **17,** 2–51.

Härdle, W. (1990) *Applied Nonparametric Regression*. Cambridge: Cambridge University Press.

Hastie, T. and Tibshirani, R. (1990) *Generalized Additive Models*. New York: Chapman and Hall.

Hastie, T. and Tibshirani, R. (1990) Varying-coefficient models. *Journal of the Royal Statistical Society, Series B,* **55,** 757–796.

Hastie, T., Buja, A. and Tibshirani, R. (1995) Penalized discriminant analysis. *Annals of Statistics*, **23**, 73–102.

He, G. Z., Müller, H-G. and Wang, J. I. (2003) Functional canonical analysis for square integrable stochastic processes. *Jurnal of Multivariate Analysis,* **85,** 54–77.

Heckman, N. E. and Ramsay, J. O. (2000). Penalized regression with model based penalties. *Canadian Journal of Statistics.* **28,** 241–258.

Hermanussen, M., Thiel, C., von Büren, E., de los Angeles Rol. de Lama, M., Pérez Romero, A., Ariznaverreta Ruiz, C., Burmeister, J., and Tresguerres, J. A. F. (1998). Micro and macro perspectives in auxology: Findings and considerations upon the variability of short term and individual growth and the stability of population derived parameters. *Annals of Human Biology*, **25**, 359–395.

Huynh, H. S. and Feldt, L. (1976) Estimation of the Box correction for degrees of freedom from sample data in randomized block and split-plot designs. *Journal of Educational Statistics*, **1**, 69–82.

Houghton, A. N., Flannery, J. and Viola, M. V. (1980) Malignant melanoma in Connecticut and Denmark. *International Journal of Cancer*, **25**, 95–104.

Hutchison, M. F. and de Hoog, F. R. (1985) Smoothing noisy data with spline functions. *Numerische Mathematik*, **47**, 99–106.

Ince, E. L. (1956) *Ordinary Differential Equations.* New York: Dover.

James, G. M. (2002) Generalized linear models with functional predictors, *Journal of the Royal Statistical Society, Series B.*, **64,** 411-432.

James, G. M. and Hastie, T. J. (2001) Functional linear discriminant analysis for irregularly sampled curves, *Journal of the Royal Statistical Society, Series B.*, **63,** 533–550.

James, G. M. and Sugar, C. A. (2003) Clustering sparsely sampled functional data. *Journal of the Americal Statistical Association,* **98,** 397–408.

James, G. M., Hastie, T. J. and Sugar, C. A. (2000) Principal component models for sparse functional data, *Biometrika*, **87,** 587–602.

Johnson, R. A. and Wichern, D. A. (1988) *Applied Multivariate Statistical Analysis.* Englewood Cliffs, N. J.: Prentice Hall.

Johnstone, I. M. and Silverman, B. W. (1997) Wavelet threshold estimators for data with correlated noise. *Journal of the Royal Statistical Society, Series B*, **59**, 319–351.

Jolicoeur, P., Pontier, J., Abidi, H. (1992) Asymptotic models for the longitudinal growth of human stature. *American Journal of Human Biology*, **4**, 461–468.

Karhunen, K. (1947) Über linear Methoden in der Warscheinlichkeitsrechnung. *Annales Academiae Scientiorum Fennicae*, **37**, 1–79.

Keselman, H. J. and Keselman, J. C. (1993) Analysis of repeated measurements. In L. K. Edwards (ed.) *Applied analysis of Variance in Behavioral Science*, New York: Marcel Dekker, 105–145.

Kim, Y-J. and Gu, G. (2004) Smoothing spline Gaussian regression: More scalable computation via efficient approximation. *Journal of the Royal Statistical Association, Series B.,* **66,** 337–356.

Kimeldorf, G. S. and Wahba, G. (1970) A correspondence between Bayesian estimation on stochastic processes and smoothing by splines. *Annals of Mathematical Statistics*, **41**, 495–502.

Kneip, A. and Engel, J. (1995) Model estimation in nonlinear regression under shape invariance. *Annals of Statistics*, **23**, 551–570.

Kneip, A. and Gasser, T. (1988) Convergence and consistency results for self-modeling nonlinear regression. *Annals of Statistics*, **16**, 82–112.

Kneip, A. and Gasser, T. (1992) Statistical tools to analyze data representing a sample of curves. *Annals of Statistics*, **20**, 1266–1305.

Kneip, A., Li, X., MacGibbon, K. B. and Ramsay, J. O. (2000) Curve registration by local regression. *The Canadian Journal of Statistics,* **28,** 19–29.

Kneip, A. and Utikal, K. J. (2001) Inference for density families using functional principal components analysis. *Journal of the Americal Statistical Association,* **96,** 519–542.

Koenker, R., Ng, P. and Portnoy, S. (1994) Quantile smoothing splines. *Biometrika,* **81,** 673–680.

Krommer, A. R. and Überhuber, C. W. (1998) *Computational Integration.* Philadelpha: Society for Industrial and Applied Mathematics.

Lancaster, H. O. (1969) *The Chi-squared Distribution.* New York: Wiley.

Largo, R. H., Gasser, T., Prader, A., Stützle, W. and Huber, P. J. (1978) Analysis of the adolescent growth spurt using smoothing spline functions. *Annals of Human Biology*, **5**, 421–434.

Legendre, A. M. (1805) *Nouvelles Méthodes pour la Détermination des Orbites des Comètes*. Paris: Courcier.

Leurgans, S. E., Moyeed, R. A. and Silverman, B. W. (1993) Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society*, Series B, **55**, 725–740.

Li, K-C. (1991) Sliced inverse regression for dimension reduction (with discussion). *Journal of the American Statistical Association,* **86**, 316–342.

Li, K-C., Aragon, Y., Shedden, K. and Thomas Agnan, C. (2003) Dimension reduction for multivariate response data. *Journal of the American Statistical Association,* **98**, 99–109.

Liggett, W., Cazares, L. and Semmes, O. J. (2003) A look at mass spectral measurement. *Chance,* **16**, 24-28.

Lindsey, J. K. (1993) *Models for Repeated Measurements*. New York: Oxford University Press.

Lindstrom, M. J. (2002) Bayesian estimation of free-knot splines using reversible jumps. *Computational Statistics and Data Analysis,* **41,** 255–269.

Lindstrom, M. J. and Kotz, S. (2004) Free-knot splines. In S. Kotz, N. L. Johnson and B. R. Campbell (eds.) *Encyclopedia of Statistics.* New York: Wiley.

Liu, X. and Müller, H-G. (2004) Functional convex averaging and synchronization for time-warped random curves. *Journal of the American Statistical Association,* **99,** 687–699.

Locantore, N., Marron, J. S., Simpson, D. G., Tripoli, N., Zhang, J. T. and Cohen, K. L. (1999) Robust principal component analysis for functional data. *Test,* **8,** 1–73.

Loève, M. (1945) Fonctions aléatoires de second ordre. *Comptes Rendus de l'Académie des Sciences, Série I: Mathématique*, **220**, 469.

Mao, W. and Zhao, L. H. (2003) Free–knot polynomial splines with confidence intervals. *Journal of the Royal Statistical Society, Series B.,* **65,** 901–919.

Maxwell, S. E. and Delaney, H. D. (2003) *Designing Experiments and Analyzing Data: A Model Comparison Perspective*. Belmont, CA: Wadsworth.

Moore, R. E. (1985) *Computational Functional Analysis*. Chichester: Wiley.

Mulaik, S. A. (1972) *The Foundations of Factor Analysis*. New York: McGraw-Hill.

Müller, H-G. and Stadtmüller, U. (2004) Generalized functional linear models. *Annals of Statistics,* to appear.

Muñoz Maldonado, Y., Staniswalis, J. G., Irwin, L. N. and Byers, D. (2002) A similarity analysis of curves. *The Canadian Journal of Statistics,* **30,** 373–381.

Nadaraya, E. A. (1964) On estimating regression. *Theory of Probability and its Applications*, **10**, 186–190.

Nason, G. P. and Silverman, B. W. (1994) The discrete wavelet transform in S. *Journal of Computational and Graphical Statistics*, **3**, 163–191.

Nielsen, H. A., Nielsen, T. S., Joensen, A. K., Madsen, H. and Holst, Ja. (2000) Tracking time-varying coefficient functions. *International Journal of Adaptive Control and Signal Processing,* **14**, 813–828.

Ocaña, F. A., Aguilera, A. M. and Valderrama, M. J. (1999) Functional principal components analysis by choice of norm. *Journal of Multivariate Analysis,* **71,** 262–276.

OECD (1995) *Quarterly National Accounts*, **3**.

Øksendal, B. (1995) *Stochastic Differential Equations: An Introduction with Applications*. New York: Springer.

Olshen, R. A., Biden, E. N., Wyatt, M. P. and Sutherland, D. H. (1989) Gait analysis and the bootstrap. *Annals of Statistics*, **17**, 1419–1440.

O'Sullivan, F. (1986) A statistical perspective on ill-posed linear inverse problems. *Statistical Science*, **1**, 502–527.

O'Sullivan, F., Yandell, B. and Raynor, W. (1986) Automatic smoothing of regression functions in generalized linear models. *Journal of the American Statistical Association,* **81,** 441–455.

Parzen, E. (1961) An approach to time series analysis. *Annals of Mathematical Statistics*, **32**, 951–989.

Parzen, E. (1963) Probability density functionals and reproducing kernel Hilbert spaces, In M. Rosenblatt (ed.) *Proceedings of the Symposium on Time Series Analysis*, Providence, RI.: Brown University, 155–169.

Pezzulli, S. D. and Silverman, B. W. (1993) Some properties of smoothed principal components analysis for functional data. *Computational Statistics*, **8**, 1–16.

Pinheiro, J. C. and Bates, D. M. (2000) *Mixed-effects Models in S and S-PLUS.* New York: Springer.

Pousse, A. (1992) Etudes asymptotiques. In J.-J. Droesbeke, B. Fichet and P. Tassi (Eds.) *Modèles pour l'Analyse des Données Multidimensionelles.* Paris: Economica.

Press W. H., Teukolsky S. A., Vetterling W. T. and Flannery B. P. (1999) *Numerical recipes in C.* Second edition. Cambridge: Cambridge University Press.

Ramsay, J. O. (1982). When the data are functions. *Psychometrika*, **47**, 379–396.

Ramsay, J. O. (1989) The data analysis of vector-valued functions. In E. Diday (ed.) *Data Analysis, Learning Symbolic and Numeric Knowledge.* Commack, N. Y.: Nova Science Publishers.

Ramsay, J. O. (1996a) Principal differential analysis: data reduction by differential operators. *Journal of the Royal Statistical Society*, Series B, **58**, 495–508.

Ramsay, J. O. (1996b) Estimating smooth monotone functions. *Journal of the Royal Statistical Society, Series B.*, **60**, 365–375.

Ramsay, J. O. (1996c) Pspline: An Splus module for polynomial spline smoothing. Computer software in the `statlib` archive.

Ramsay, J. O. (2000). Functional components of variation in handwriting. *Journal of the American Statistical Association*, **95**, 9–15.

Ramsay, J. O., Altman, N. and Bock, R. D. (1994) Variation in height acceleration in the Fels growth data. *Canadian Journal of Statistics*, **22**, 89–102.

Ramsay, J. O., Bock. R. D. and Gasser, T. (1995) Comparison of height acceleration curves in the Fels, Zurich, and Berkeley growth data. *Annals of Human Biology*, **22**, 413–426.

Ramsay, J. O. and Dalzell, C. J. (1991) Some tools for functional data analysis (with Discussion). *Journal of the Royal Statistical Society*, Series B, **53**, 539–572.

Ramsay, J. O, Heckman, N. and Silverman, B. W. (1997) Some general theory for spline smoothing. *Behavioral Research: Instrumentation, Methods, and Computing*, **29,** 99–106.

Ramsay, J. O. and Li, X. (1996) Curve registration. *Journal of the Royal Statistical Society, Series B.*, **60,** 351–363.

Ramsay, J. O., Munhall, K., Gracco, V. L. and Ostry, D. J. (1996) Functional data analyses of lip motion. *Journal of the Acoustical Society of America*, **99**, 3718–3727.

Ramsay, J. O. and Silverman, B. W. (2002) *Applied Functional Data Analysis.* New York: Springer.

Ramsay, J. O., Wang, X. and Flanagan, R. (1995) A functional data analysis of the pinch force of human fingers. *Applied Statistics*, **44**, 17–30.

Rao, C. R. (1958) Some statistical methods for comparison of growth curves. *Biometrics*, **14**, 1–17.

Rao, C. R. (1987) Prediction in growth curve models (with discussion). *Statistical Science*, **2**, 434–471.

Ratcliffe, S. J., Leader, L. R. and Heller, G. Z. (2002) Functional data analysis with application to periodically stimulated foetal heart rate data. I: Functional regression. *Statistics in Medicine,* **21,** 1103–1114.

Ratcliffe, S. J., Heller, G. Z. and Leader, L. R. (2002) Functional data analysis with application to periodically stimulated foetal heart rate data. II: Functional logistic regression. *Statistics in Medicine,* **21,** 1115–1127.

Reinsch, C. (1967) Smoothing by spline functions. *Numerische Mathematik*, **10**, 177–183.

Reinsch, C. (1970) Smoothing by spline functions II. *Numerische Mathematik*, **16**, 451–454.

Rice, J. A. and Wo, C. O. (2001) Nonparametric mixed effects models for unequally sampled noisy curves. *Biometrics,* **57,** 253–259.

Rice, J. A. and Silverman, B. W. (1991) Estimating the mean and covariance structure nonparametrically when the data are curves. *Journal of the Royal Statistical Society*, Series B, **53**, 233–243.

Ripley, B. D. (1991) *Statistical Inference for Spatial Processes.* Cambridge: Cambridge University Press.

Roach, G. F. (1982) *Green's Functions.* Second edition. Cambridge: Cambridge University Press.

Roche, A. (1992) *Growth, Maturation and Body Composition: The Fels Longitudinal Study 1929–1991.* Cambridge: Cambridge Press.

Rønn, B. B. (2001) Nonparametric maximum likelihood estimation for shifted curves. *Journal of the Royal Statistical Society, Series B.,* **63,** 243–259.

Rossi, N., Wang, X. and Ramsay, J. O. (2002) Nonparametric item response function estimates with the EM algorithm. *Journal of the Behavioral and Educational Sciences,* **27,** 291–317.

Sakoe, H. and Chiba, S. (1978) Dynamic programming algorithm for optimizating for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal processing,* **26**, 43–49.

Scott, D. W. (1992) *Multivariate Density Estimation.* New York: Wiley.

Schumaker, L. (1981) *Spline Functions: Basic Theory.* New York: Wiley.

Scott, D. W. (1992) *Multivariate Density Estimation.* New York: Wiley.

Searle, S. R., Casella, G. and McCulloch, C. E. (1992) *Variance Components.* New York: Wiley.

Seber, G. A. F. (1984) *Multivariate Observations.* New York: Wiley.

Silverman, B. W. (1982) On the estimation of a probability density function by the maximum penalized likelihood method. *Annals of Statistics*, **10**, 795–810.

Silverman, B. W. (1985) Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society*, Series B, **47**, 1–52.

Silverman, B. W. (1986) *Density Estimation for Statistics and Data Analysis.* London: Chapman and Hall.

Silverman, B. W. (1994) Function estimation and functional data analysis. In A. Joseph, F. Mignot, F. Murat, B. Prum and R. Rentschler (eds.) *First European Congress of Mathematics.* Basel: Birkhäuser. vol II, pp. 407–427.

Silverman, B. W. (1995). Incorporating parametric effects into functional principal components analysis. *Journal of the Royal Statistical Society*, Series B, **57**, 673–689.

Silverman, B. W. (1996) Smoothed functional principal components analysis by choice of norm. *Annals of Statistics*, **24**, 1–24.

Silverman, B. W. (1999) Wavelets in statistics: Beyond the standard assumptions. *Philosophical Transactions of the Royal Society of London, Series A.* **357,** 2459–2473.

Silverman, B. W. (2000) Wavelets for regression and other statistical problems. In M. G. Schimek (Ed.) *Smoothing and Regression: Approaches, Computation and Application.* New York: Wiley.

Silverman, B. W. and Vassilicos, J. C. (1999) Wavelets: The key to intermittent information? *Philosophical Transactions of the Royal Society of London, Series A.* **357,** 2393–2395.

Snyder, D. L. and Miller, M. I. (1991) *Random Point Processes in Time and Space.* New York: Springer.

Simonoff, J. S. (1996) *Smoothing Methods in Statistics.* New York: Springer.

Spitzner, D. J., Marron, J. S. and Essick, G. K. (2003) Mixed–model functional ANOVA for studying human tactile perception. *The Journal of the American Statistical Association,* **98,** 263–272.

Statistical Sciences (1995) *S-PLUS Guide to Statistical and Mathematical Analysis, Version 3.3.* Seattle: StatSci, a division of MathSoft, Inc.

Stoer, J. and Bulirsch, R. (2002) *Introduction to Numerical Analysis.* Third Edition. New York: Springer.

Stone, M. (1987) *Coordinate-Free Multivariable Statistics.* Oxford: Clarendon Press.

Tarpey, T. and Kinateder, K. K. J. (2003) Clustering functional data. *Journal of Classification,*, **20,** 93–114.

Tenenbaum, M. and Pollard, H. (1963) *Ordinary Differential Equations.* New York: Harper and Row.

Tucker, L. R. (1958) Determination of parameters of a functional relationship by factor analysis. *Psychometrika*, **23**, 19–23.

Tuddenham, R. D. and Snyder, M. M. (1954) Physical growth of California boys and girls from birth to eighteen years. *University of California Publications in Child Development* **1**, 183–364.

Valderrama, M. J., Aguilera, A. M. and Ocaña, F. A. (2000) *Predicción Dinámica Mediante Análisis de Datos Funcionales.* Madrid: Hespérides.

Viele, K. (2001) Evaluating fit in functional data analysis using model embeddings. *The Canadian Journal of Statistics,* **29,** 51–66.

Vinod, H. D. (1976) Canonical ridge and econometrics of joint production. *Journal of Econometrics*, **4**, 147–166.

Viviani, R., Grön, G. and Spitzer, M. (2005) *Human Brain Mapping,* **24,** 109–129.

Wahba, G. (1978) Improper priors, spline smoothing and the problem of guarding against model errors in regression. *Journal of the Royal Statistical Society*, Series B, **40**, 364–372.

Wahba, G. (1990) *Spline Models for Observational Data.* Philadelphia: Society for Industrial and Applied Mathematics.

Wand, M. P. and Jones, M. C. (1995) *Kernel Smoothing.* London: Chapman and Hall.

Wang, K. and Gasser, T. (1997) The alignment of curves by dynamic time warping. *The Annals of Statistics,* **25,** 1251–1276.

Wang, K. and Gasser, T. (1998) Asymptotic and boostrap confidence bounds for the structural average of curves. *The Annals of Statistics,* **26,** 972–991.

Wang, K. and Gasser, T. (1999) Synchronizing sample curves nonparametrically. *The Annals of Statistics,* **27,** 439–460.

Watson, G. S. (1964) Smooth regression analysis. *Sankhyā*, Series A, **26**, 101–116.

Weinert, H. L., Byrd, R. H. and Sidhu, G. S. (1980) A stochastic framework for recursive computation of spline functions: Part II, smoothing splines. *Journal of Optimization Theory and Applications,* **2,** 255–268.

West, M. and Harrison, P. J. (1989) *Bayesian Forecasting and Dynamic Models.* New York: Springer.

West, M., Harrison, P. J. and Migon, H. S. (1985) Dynamic generalized linear models and Bayesian forecasting (with discussion). *Journal of the American Statistical Association,* **80,** 73–97.

Wilson, A. M., Seelig, T. J., Shield, R. A. and Silverman, B. W. (1996) The effect of imposed foot imbalance on point of force application in the equine. Technical report, Department of Veterinary Basic Sciences, Royal Veterinary College.

Wolfram, S. (1991) *The Mathematica Book, fifth edition.* Champaign, Illinois: Wolfram Research Inc.

Wu, C. O., Chiang, C.-T. and Hoover, D. R. (1998) Asymptotic confidence regions for kernel smoothing of a varying coefficient model with longitudinal data. *Journal of the American Statistical Association,* **93,** 1388–1418.

Yao, F., Müller, H.-G., Clifford, A. J., Dueker, S. R., Follet, J., Lin, Y., Bucholz, B. A. and Vogel, J. S. (2003) Shrinkage estimation for functional principal component scores with application to the population kinetics of plasma folate. *Biometrics,* **59,** 676–685.

Yao, F., Müller, H.-G. and Wang, J.-L. (2004) Functional data analysis for sparse longitudinal data. *Journal of the American Statistical Association,* to appear.

Zhang, C. (2003) Calibrating the degrees of freedom for automatic data smoothing and effective curve checking. *The Journal of the American Statistical Association,* **98,** 609–628.

Zhang, W. and Lee, S.-K. (2000) Variable bandwidth selecton in varying-coefficient models, *Journal of Multivariate Analysis,* **74**, 116–134.

Zhang, W., Lee, S.-K. and Song, X. (2002) Semiparametric smooth coefficient models, *Journal of Business and Economic Statistics,* **20,** 412–422.

Index