# Chapter 1
# Functional data structures

Statistics is concerned with obtaining information from observations $X_1, X_2, \ldots, X_N$. The $X_n$ can be scalars, vectors or other objects. For example, each $X_n$ can be a satellite image, in some spectral bandwidth, of a particular region of the Earth taken at time $n$. Functional Data Analysis (FDA) is concerned with observations which are viewed as functions defined over some set $T$. A satellite image processed to show surface temperature can be viewed as a function $X$ defined on a subset $T$ of a sphere, $X(t)$ being the temperature at location $t$. The value $X_n(t)$ is then the temperature at location $t$ at time $n$. Clearly, due to finite resolution, the values of $X_n$ are available only at a finite grid of points, but the temperature does exist at every location, so it is natural to view $X_n$ as a function defined over the whole set $T$.

Some functional data belong to the class of high dimensional data in the sense that every data object consists of a large number of scalar values, and the number of measurements per objects may be larger than the sample size $N$. If there are $m$ measurements per object, such data falls into the "large $m$ small $N$" paradigm. However, for functional data, the values within one functional object (a curve or surface) for neighboring arguments are similar. Typical functional objects are thus smooth curves or surfaces that can be approximated by smooth functions. Thus, to perform computations, a functional object can be replaced by a few smooth standard building blocks. The central idea of this book is to study the approximations of functional objects consisting of large number of measurements by objects that can be described using only $p$ coefficients of the standard building blocks, with $p$ being much smaller than $N$. Such approximations give rise to many interesting and challenging questions not encountered in statistical inference for scalars or vectors.

## 1.1 Examples of functional data

The data that motivated the research presented in this book is of the form $X_n(t),\ t \in [a, b]$, where $[a, b]$ is an interval on the line. Each observation is thus a curve. Such

L. Horváth and P. Kokoszka, *Inference for Functional Data with Applications*,
Springer Series in Statistics 200, DOI 10.1007/978-1-4614-3655-3_1,

curves can arise in many ways. Figure 1.1 shows a reading of a magnetometer over a period of one week. A magnetometer is an instrument that measures the three components of the magnetic field at a location where it is placed. There are over 100 magnetic observatories located on the surface of the Earth, and most of them have digital magnetometers. These magnetometers record the strength and direction of the field every five seconds, but the magnetic field exists at any moment of time, so it is natural to think of a magnetogram as an approximation to a continuous record. The raw magnetometer data are cleaned and reported as averages over one minute intervals. Such averages were used to produce Figure 1.1. Thus $7 \times 24 \times 60 = 10,080$ values (of one component of the field) were used to draw Figure 1.1. The dotted vertical lines separate days in Universal Time (UT). It is natural to view a curve defined over one UT day as a single observation because one of the main sources influencing the shape of the record is the daily rotation of the Earth. When an observatory faces the Sun, it records the magnetic field generated by wind currents flowing in the ionosphere which are driven mostly by solar heating. Thus, Figure 1.1 shows seven consecutive functional observations.

Many important examples of data that can be naturally treated as functional come from financial records. Figure 1.2 shows two consecutive weeks of Microsoft stock prices in one minute resolution. In contrast to the magnetic field, the price of an asset exists only when the asset is traded. A great deal of financial research has been done using the closing daily price, i.e. the price in the last transaction of a trading day. However many assets are traded so frequently that one can practically

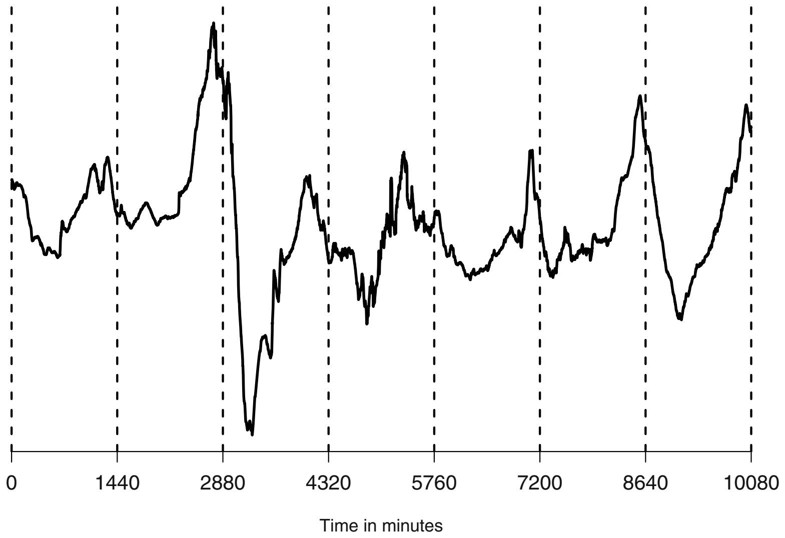

**Fig. 1.1** The horizontal component of the magnetic field measured in one minute resolution at Honolulu magnetic observatory from 1/1/2001 00:00 UT to 1/7/2001 24:00 UT.

think of a price curve that is defined at any moment of time. The Microsoft stock is traded several hundred times per minute. The values used to draw the graph in Figure 1.2 are the closing prices in one-minute intervals. It is natural to choose one trading day as the underlying time interval. If we do so, Figure 1.2 shows 10 consecutive functional observations. From these functional observations, various statistics can be computed. For example, the top panels of Figure 1.3 show the mean functions for the two weeks computed as $\hat{\mu}(t) = 5^{-1} \sum_{i=1}^{5} X_i(t)$, where $X_i(t)$ is the price at time $t$ on the $i$th day of the week. We see that the mean functions have roughly the same shape (even though they have different ranges), and we may ask if it is reasonable to assume that after adjusting for the ranges, the differences in these curves can be explained by chance, or these curves are really different. This is clearly a setting for a statistical hypothesis test which requires the usual steps of model building and inference. Most chapters of this book focus on inferential procedures in models for functional data. The bottom panels of Figure 1.3 show the five curves $X_i(t) - \hat{\mu}(t)$ for each week. We will often work with functional data centered in this way, and will exhibit the curves using the graphs as those in the bottom panels of Figure 1.3.

Functional data arise not only from finely spaced measurements. For example, when measurements on human subjects are made, it is often difficult to ensure that they are made at the same time in the life of a subject, and there may be different numbers of measurements for different subjects. A typical example are growth curves, i.e. $X_n(t)$ is the height of subject $n$ at time $t$ after birth. Even though every

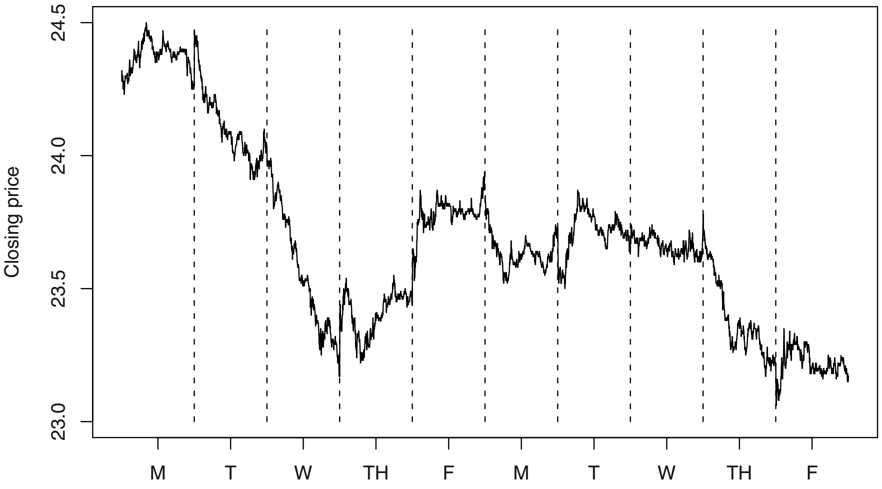

**Fig. 1.2** Microsoft stock prices in one-minute resolution, May 1-5, 8-12, 2006

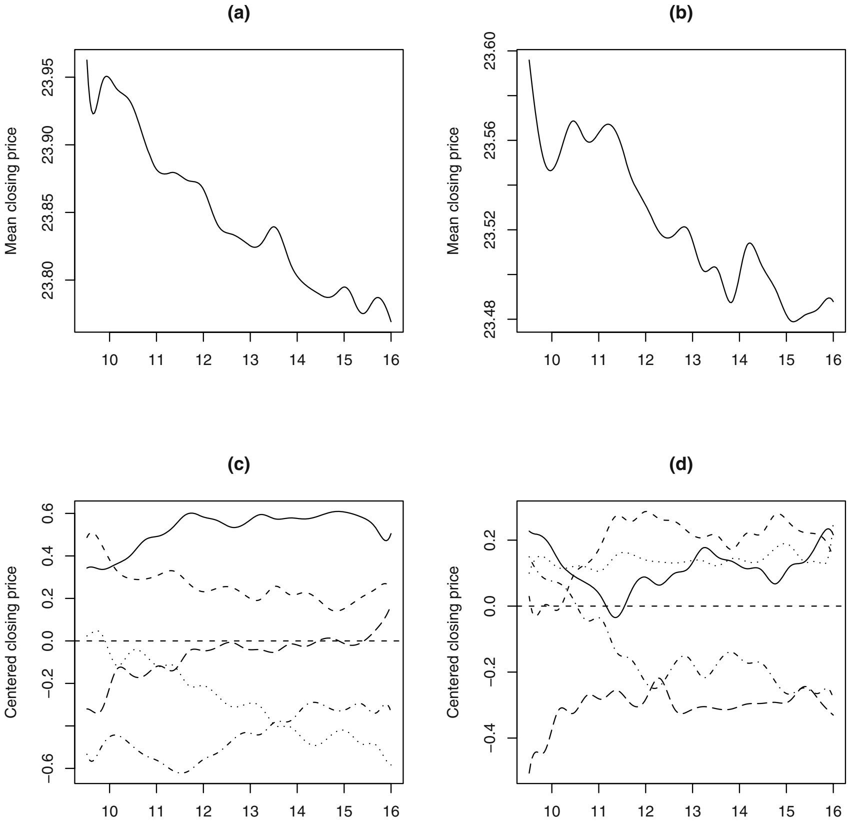

**Fig. 1.3** (a) Mean function of Microsoft stock prices, May 1-5, 2006; (b) Mean function of Microsoft stock prices, May 8-12, 2006; (c) Centered prices of Microsoft stock, May 1-5, 2006; (d) Centered prices of Microsoft stock, May 8-12, 2006.

individual has a height at any time $t$, it is measured only relatively rarely. Thus it has been necessary to develop methods of estimating growth curves from such sparse unequally spaced data, in which smoothing and regularization play a crucial role. Examples and methodology of this type are discussed in the monographs of Ramsay and Silverman (2002, 2005).

It is often useful to treat as functional data measurements that are neither sparse nor dense. Figure 1.4, shows the concentration of nitrogen oxide pollutants, referred to as $NO_x$, measured at Barcelona's neighborhood of Poblenou. The $NO_x$ concentration is measured every hour, so we have only 24 measurements per day. It is nevertheless informative to treat these data as a collection of daily curves because the pattern of pollution becomes immediately apparent. The pollution peaks in morning hours, declines in the afternoon, and then increases again in the evening. This

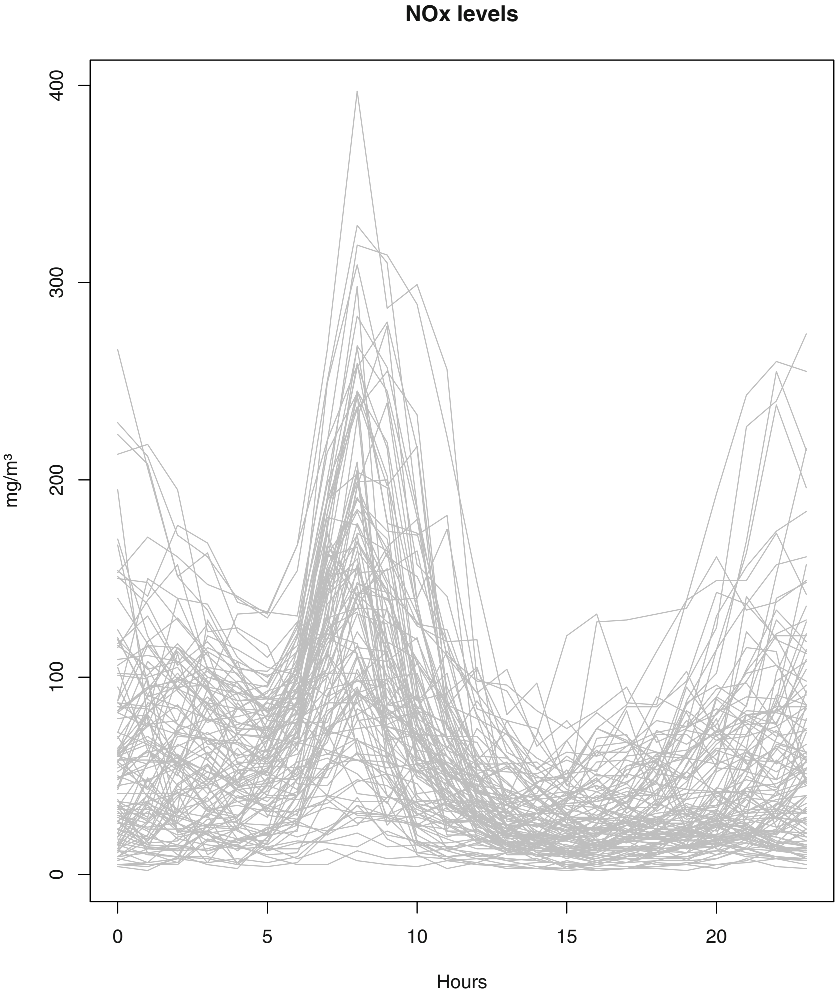

**Fig. 1.4** Hourly levels of $NO_x$ pollutants measured in Poblenou, Spain. Each curve represents one day.

pattern is easy to explain because the monitoring station is in a city center, and road traffic is a major source of $NO_x$ pollution. Broadly speaking, for functional data the information contained in the *shape* of the curves matters a great deal. The above data set was studied by Febrero *et al.* (2008), Jones and Rice (1992) study ozone levels In Upland, California.

The usefulness of the functional approach has been recognized in many other fields of science. Borggaard and Thodberg (1992) provide interesting applications of the functional principal component analysis to chemistry. A spectrum is a sampling of a continuous function at a set of fixed wavelengths or energies. Borggaard and Thodberg (1992) point out that multivariate linear regression often fails because the number of input variables is very large. Their simulations and examples show that functional regression provides much better results. Spectra are studied in detail in Ferraty and Vieu (2006). Starting with Kirkpatrick and Heckman (1989), it has been recognized that evolutionary important traits are better modeled as infinite–dimensional data. The examples in Griswold *et al.* (2008) are organismal growth trajectories, thermal performance curves and morphological shapes. Griswold *et al.* (2008) argue that the functional approach provides a more convenient framework than the classical multivariate methods. Many recent applications of functional data analysis are discussed in Ferraty (2011).

In the remaining sections of this chapter, we present a few analyses which illustrate some ideas of FDA. The discussion in this chapter is informal, in the following chapters the exposition will be more detailed and rigorous. In Section 1.2, we discuss in the functional context the concepts of the center of a distribution and outliers. Section 1.3 show how temporal dependence in functional data can be modeled. Finally, Section 1.4 focuses on modeling the dependence between two samples.

## 1.2 Detection of abnormal NO$_x$ pollution levels

In this section, based on the work of Febrero *et al.* (2008), we show how the fundamental statistical concepts of the center of a distribution and of an outlier can be defined in the functional context. A center of a sample of scalars can be defined by the median, the mean, the trimmed mean, or other similar measures. The definition of an outlier is less clear, but for relatively small samples even visual inspection may reveal suspect observations. For a collection of curves, like those shown in Figure 1.4, it is not clear how to define central curves or outlying curves. The value of a function at every point $t$ may not be an outlier, but the curve itself may be a functional outlier. Generally speaking, once incorrectly recorded curves have been removed, a curve is an outlier if it comes from a populations with a different distribution in a function space than the majority of the curves. An outlier may be far away from the other curves, or may have a different shape. The concept of depth of functional data offers a possible framework for identifying central and outlying observations; those with maximal depth are central, and those with minimal depth are potential outliers.

The depth of a scalar data point can be defined in many ways, see Zuo and Serfling (2000). To illustrate, suppose $X_1, X_2, \dots X_N$ are scalar observations, and

$$F_N(x) = \frac{1}{N} \sum_{n=1}^{N} \mathbf{I}\{X_n \leq x\}$$

is their empirical distribution function. The halfspace depth of the observation $X_i$ is defined as

$$HSD_N(X_i) = \min\{F_N(X_i), 1 - F_N(X_i)\}\,.$$

If $X_i$ is the median, then $F_N(X_i) = 1/2$, and so $HSD_N(X_i) = 1/2$, the largest possible depth. If $X_i$ is the largest point, then $F_N(X_i) = 1$, and so $HSD_N(X_i) = 0$, the least possible depth. Another way of measuring depth is to define

$$D_N(X_i) = 1 - \left| \frac{1}{2} - F_N(X_i) \right|.$$

The largest possible depth is now 1, and the smallest $1/2$.

Suppose now that we have a sample of functions $\{X_n(t),\ t \in [a,b],\ n = 1, 2, \ldots, N\}$. We define the empirical distribution function at point $t$ by

$$F_{N,t}(x) = \frac{1}{N} \sum_{n=1}^{N} \mathbf{I}\{X_n(t) \le x\},$$

and we can define a functional depth by integrating one of the univariate depths. For example, Fraiman and Muniz (2001) define the functional depth of the curve $X_i$ as

$$FD_N(X_i) = \int_a^b \left[ 1 - \left| \frac{1}{2} - F_{N,t}(X_i(t)) \right| \right] dt.$$

There are also other approaches to defining functional depth, an interested reader is referred to Febrero *et al.* (2008) and López-Pintado and Romo (2009).

Once a measure of a functional depth, denote it generically by $FD_N$, has been chosen, we can use the following algorithm to identify outliers:

1. Calculate $FD_N(X_1), FD_N(X_2), \ldots, FD_N(X_N)$.
2. Remove curves with depth smaller than a threshold $C$ from the sample and classify them as outliers. If there are no such curves, the procedure ends here.
3. Go back to step 1 and apply it to the sample without outliers removed in step 2.

The critical element of this procedure is determining the value of $C$ which should be so small that only a small fraction, say 1%, of the curves are classified as outliers, if there are in fact no outliers. The value of $C$ can then be computed from the sample using some form of bootstrap, two approaches are described in Febrero *et al.* (2008). Step 3 is introduced to avoid masking, which takes place when "large" outliers mask the presence of other outliers.

Febrero *et al.* (2008) applied this procedure with three measures $FD_N$ to the data shown in Figure 1.4, but split into working and non-working days. The two samples containing 76 working and 39 nonworking days between February 23 and June 26, 2005 are shown in Figure 1.5, with outliers identified by black lines. For working

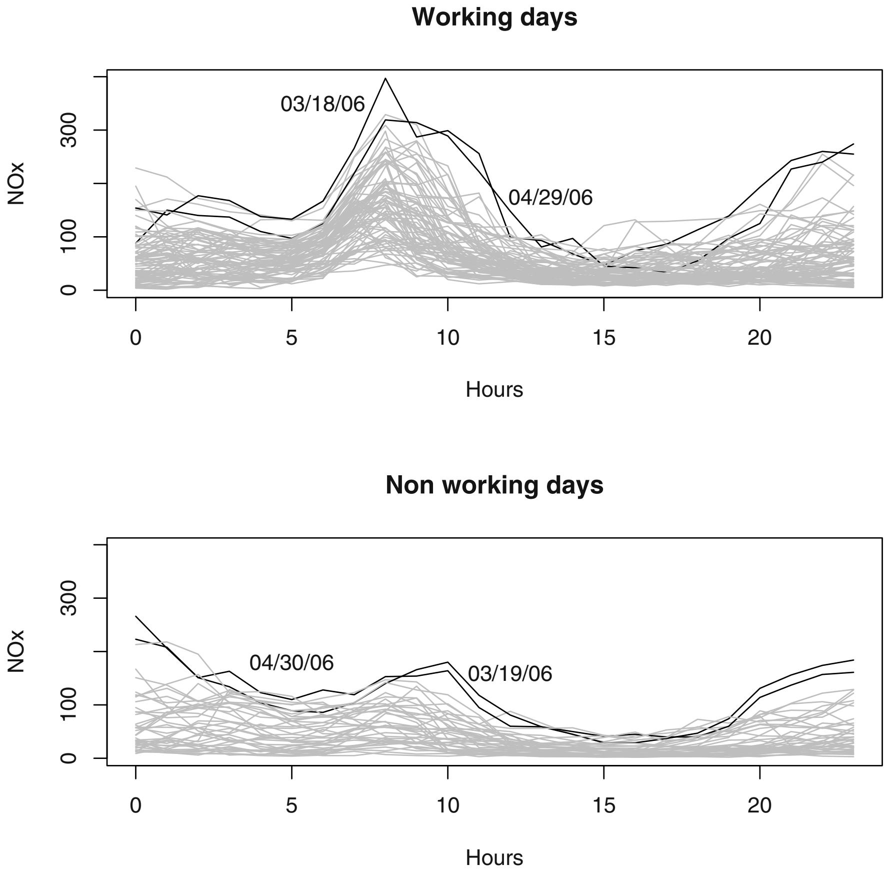

**Fig. 1.5** Outliers in NO$_x$ concentration curves in the samples of working and nonworking days.

days, these are Friday, March 18, and Friday, April 29. For non-working days, the outliers are the following Saturdays, March 19 and April 30. These days are the beginning of long weekend holidays in Spain. This validates the identification of the NO$_x$ curves on these days as outliers, as the traffic pattern can be expected to be different on holidays.

Febrero *et al.* (2008) did not attempt to develop an asymptotic justification for the procedure described in this section. Its performance is assessed by application to a real data set. Such an approach is common. In this book, however, we focus on statistical procedures whose asymptotic validity can be established. Resampling procedures for functional data taking values in a general measurable space are reviewed by McMurry and Politis (2010).

## 1.3 Prediction of the volume of credit card transactions

In this section, based on the work of Laukaitis and Račkauskas (2002), we describe the prediction of the volume of credit card transactions using the functional autoregressive process, which will be studied in detail in Chapter 13.

The data available for this analysis consists of all transactions completed using credit cards issued by Vilnius Bank, Lithuania. Details of every transaction are documented, but here we are interested only in predicting the daily pattern of the volume of transactions. For our exposition, we simplified the analysis of Laukaitis and Račkauskas (2002), and denote by $D_n(t_i)$ the number of credit card transactions on day $n$, $n = 1, \dots, 200$, (03/11/2000 – 10/02/2001) between times $t_{i-1}$ and $t_i$, where $t_i - t_{i-1} = 8$ min, $i = 1, \dots, 128$. We thus have $N = 200$ daily curves, which we view as individual observations. The grid of 8 minutes was chosen for ease of exposition, Laukaitis and Račkauskas (2002) divide each day into 1024 intervals of equal length. The transactions are normalized to have time stamps in the interval $[0, 1]$, which thus corresponds to one day. The left most panel of Figure 1.6 shows the $D_n(t_i)$ for two randomly chosen days. The center and right panels show smoothed functional versions $D_n(t)$ obtained, respectively, with 40 and 80 Fourier basis functions as follows. Each vector $[D_n(t_1), D_n(t_2), \dots, D_n(t_{128})]$ is approximated using sine and cosine functions $B_m(t)$, $t \in [0, 1]$, whose frequencies increase with $m$. We write this approximation as

$$D_n(t_i) \approx \sum_{m=1}^{M} c_{nm} B_m(t_i), \quad n = 1, 2, \dots, N.$$

The trigonometric functions are defined on the whole interval $[0, 1]$, not just at the points $t_i$, so we can continue to work with truly functional data

$$Y_n(t) = \sum_{m=1}^{M} c_{nm} B_m(t), \quad n = 1, 2, \dots, N.$$

In this step, we reduced the the number of scalars needed to represent each curve from 128 to $M$ (40 or 80). If the original data are reported on a scale finer than 8 minutes, the computational gain is even greater. The step of expanding the data with respect to a fixed basis is however often only a preliminary step to further dimension reduction. The number $M$ is still too large for many matrix operations, and the choice of the trigonometric basis is somewhat arbitrary, a spline or a wavelet basis could be used as well. The next step will attempt to construct an "optimal" basis.

Before we move on to the next step, we remove the weekly periodicity by computing the differences $X_n(t) = Y_n(t) - Y_{n-7}(t)$, $n = 8, 9, \dots, 200$. Figure 1.7 displays the first three weeks of these data. The most important steps of the analysis are performed on the curves $X_n(t), n = 8, 9, \dots, 200$. which we view as a stationary functional time series. Thus, while each $X_n$ is assumed to have the same distribution in a function space, they are dependent. We assume that each $X_n$ is an

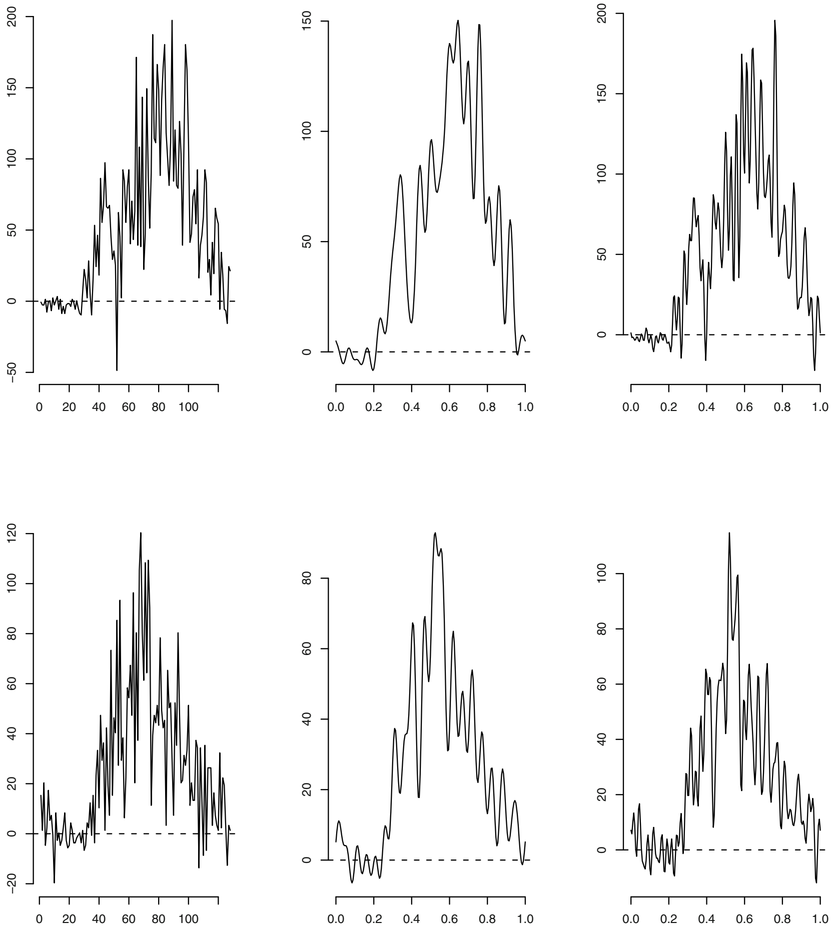

**Fig. 1.6** Two functional observations $X_n$ derived from the credit card transactions (left–most panel) together with smooths obtained by projection on 40 and 80 Fourier basis functions.

element of the space $L^2 = L^2([0, 1])$ of square integrable functions on $[0, 1]$, and that there is a function $\psi(t, s),\ t \in [0, 1], s \in [0, 1]$, such that

$$X_n(t) = \int_0^1 \psi(t, s) X_{n-1}(s) ds + \varepsilon_n(t),$$

where the errors $\varepsilon_n$ are iid elements of $L^2$. The above equation extends to the functional setting the most popular model of time series analysis, the AR(1) model, in which the scalar observations $X_i$ are assumed to satisfy $X_i = \psi X_{i-1} + \varepsilon_i$, see e.g.

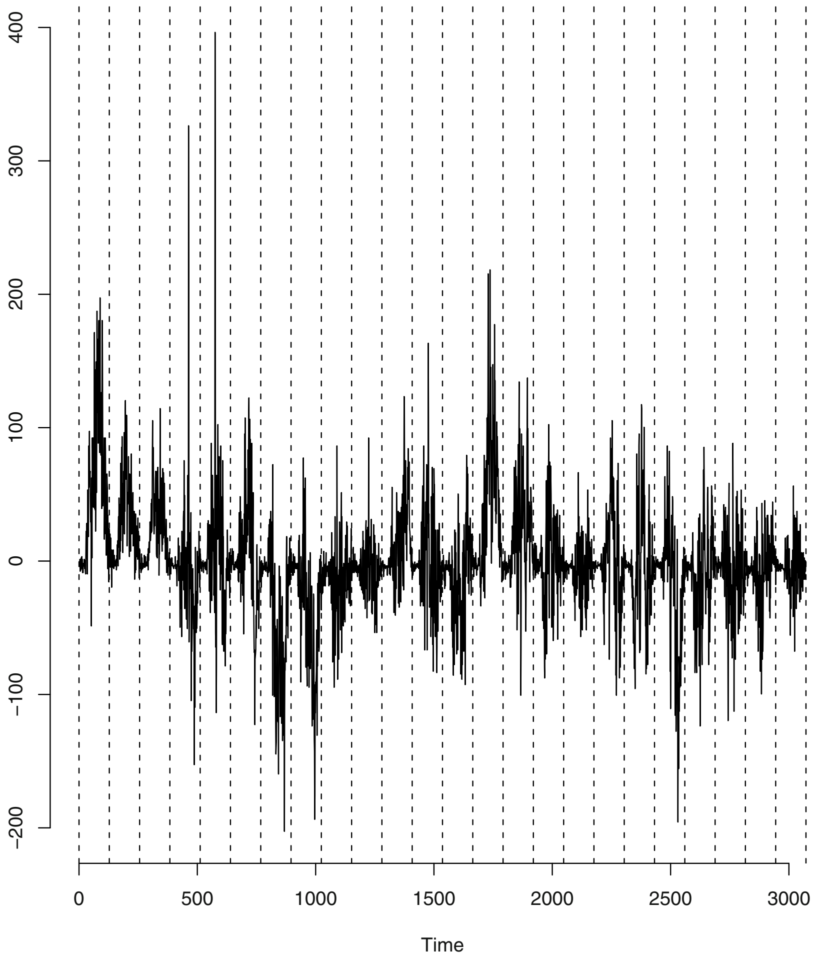

**Fig. 1.7** Three weeks of centered time series of $\{X_n(t_i)\}$ derived from credit card transaction data. The vertical dotted lines separate days.

Chapter 3 of Box *et al.* (1994). To compute an estimate of the kernel $\psi(t,s)$, the curves $X_n$ are approximated by an expansion of the form

$$X_n(t) \approx \sum_{k=1}^{p} \xi_{kn} v_k(t),$$

where the $v_k$ are the functional principal components (FPC's) of the $X_n$, $n = 8, 9, \dots, 200$. The idea of expansion with respect to FPC's will be taken up in

Chapter 3. Here we note that $p$ is generally much smaller than the number of the points at which the curves are evaluated (128 in this example) or the number $M$ of basis functions (40 or 80 in this example). The $v_k$ are orthonormal, and form an "optimal" system for expressing the observations. Laukaitis and Račkauskas (2002) recommend using $p = 4$ FPC's. Once an estimator $\hat{\psi}$ has been constructed, we can predict $X_n$ via $\hat{X}_n = \int_0^1 \hat{\psi}(t,s)X_{n-1}(s)ds$ and the transaction volume curves via

$$\hat{Y}_{n+1}(t) = Y_{n-6}(t) + \int_0^1 \hat{\psi}(t,s)[Y_n(s) - Y_{n-7}(s)]ds.$$

Figure 1.8 shows examples of two curves $Y_n$ ($n = 150$ and $n = 190$) and their predictions $\hat{Y}_n$. In general, the predictions tend to underestimate the transaction volume. This is because even for the scalar AR(1) process, the series of prediction $\hat{X}_n = \hat{\phi} X_{n-1}$ has a smaller range than the observations $X_n = \phi X_{n-1} + \varepsilon_n$. The problem of prediction of functional time series is studied in detail in Chapter 13.

## 1.4 Classification of temporal gene expression data

This section, based on the work of Leng and Müller (2006), introduces one of many formulations of the functional linear model. We introduce such models in Chapter 8, and study them in Chapters 9, 11 and 10. Our presentation focuses only on the central idea and omits many details, which can be found in Leng and Müller (2006) and Müller and Stadtmüller (2005).

Figure 1.9 shows expression time courses of 90 genes. The expressions are measured at 18 time points $t_i$ with $t_i - t_{i-1} = 7$ minutes. The genes can be classified as G1 phase and non–G1 phase. A classification performed using traditional methods yielded 44 G1 and 46 non–G1 genes. Leng and Müller (2006) proposed a statistical method of classifying genes based exclusively on their expression trajectories. Their approach can be summarized as follows.

After rescaling time, each trajectory is viewed as a smooth curve $X_n(t)$, $t \in [0,1]$, observed, with some error, at discrete time points $t_i$. It is assumed that the curves are independent and identically distributed with the mean function $\mu(t) = EX_n(t)$ and the FPC's $v_k$, so that they admit a representation

$$X_n(t) = \mu(t) + \sum_{k=1}^{\infty} \xi_{kn} v_k,$$

with

$$\xi_{kn} = \int_0^1 (X_n(t) - \mu(t))v_k(t)dt.$$

The unknown curves $X_n$ must be estimated, as outlined below, but the idea is that the scalars

$$\eta_n = \alpha + \int_0^1 \beta(t)\,(X_n(t) - \mu(t))\,dt,$$

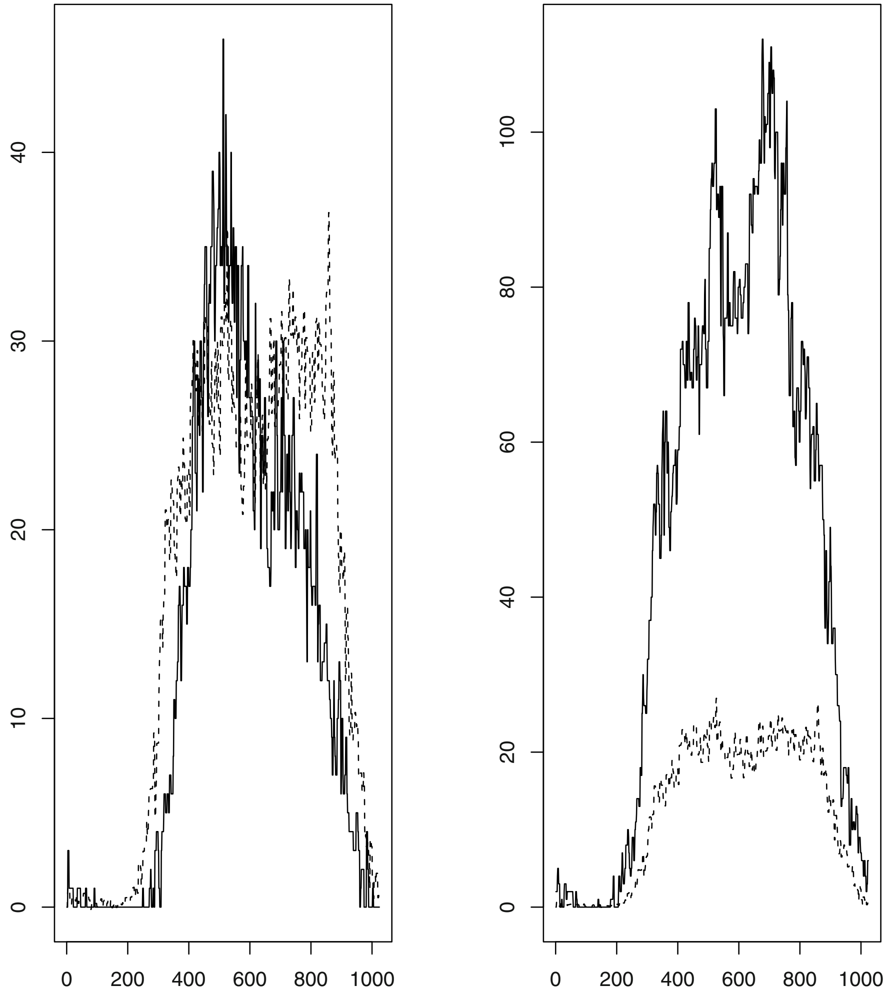

**Fig. 1.8** Two credit card transaction volume curves $Y_n$ (solid lines) and their predictions $\hat{Y}_n$ (dotted lines)

for some parameters $\alpha$ and $\beta(t), t \in [0, 1]$, can be used to classify the genes as G1 or non–G1. Note that the parameter $\beta$ is a smooth curve. The idea of classification is that we set a cut–off probability $p_1$, and classify a gene as G1 phase if $h(\eta_n) > p_1$, where

$$h(\eta) = \frac{e^\eta}{1 + e^\eta}.$$

The central issue is thus to approximate the linear predictors $\eta_n$, and this involves the estimation of the curves $X_n$ and the parameters $\alpha$ and $\beta$.

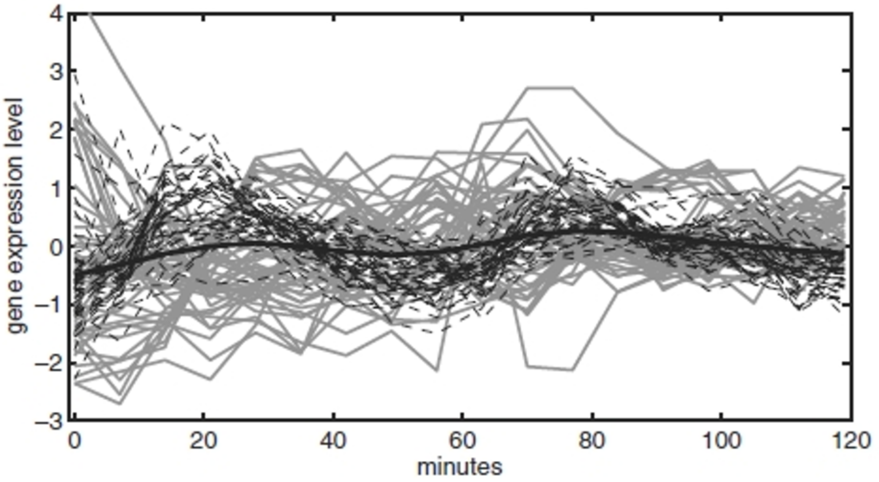

**Fig. 1.9** Temporal gene expression profiles of yeast cell cycle. Dashed lines: G1 phase; Gray solid lines: non-G1 phases; Black solid line: overall mean curve.

The curves $X_n$ are estimated by smooth curves

$$X_n^{(p)}(t) = \hat{\mu}(t) + \sum_{k=1}^{p} \hat{\xi}_{kn} \hat{v}_k(t).$$

For the curves shown in Figure 1.9, using $p = 5$ is appropriate. Estimation of the FPC's $v_k$ involves obtaining a smooth estimate of the covariance surface

$$c(t,s) = E\left\{(X_n(t) - \mu(t))(X_n(s) - \mu(s))\right\}, \quad t, s \in [0,1].$$

Inserting $X_n^{(p)}$ into the equation defining $\eta_n$ yields

$$\eta_n^{(p)}(\alpha, \beta) = \alpha + \int_0^1 \beta(t) \left( \sum_{k=1}^{p} \hat{\xi}_{kn} \hat{v}_k(t) \right) dt,$$

i.e

$$\eta_n^{(p)}(\alpha, \beta) = \alpha + \sum_{k=1}^{p} \beta_k \hat{\xi}_{kn},$$

where

$$\beta_k = \int_0^1 \beta(t) \hat{v}_k(t) dt, \quad k = 1, 2, \ldots, p.$$

The parameters $\alpha, \beta_1, \ldots, \beta_p$ are estimated using the generalized linear model

$$Y_n = h\left( \alpha + \sum_{k=1}^{p} \beta_k \hat{\xi}_{kn} \right) + e_n,$$

where $Y_n = 1$ if a gene is classified as G1 using traditional methods, and $Y_n = 0$ otherwise. This is done by solving an appropriate score equation. Denoting the estimates by $\hat{\alpha}, \hat{\beta}_1, \ldots, \hat{\beta}_p$, we can compute the linear predictor

$$\hat{\eta}_n = \hat{\alpha} + \sum_{k=1}^{p} \hat{\beta}_k \hat{\xi}_{kn}$$

for any trajectory, and classify the gene as G1 phase if $h(\hat{\eta}) > p_1$.

Leng and Müller (2006) applied this method to the time courses of 6,178 genes in the yeast cell cycle, and found that their method compares favorably with an earlier method. In the training sample of the 90 trajectories, they found 5 genes which their method classified as non–G1, but the traditional method as G1. They argued that the traditional method may have classified some of these 5 genes incorrectly.

## 1.5 Statistical packages, bases, and functional objects

All procedures described in this book can be implemented in readily available statistical software without writing additional code in FORTRAN or C++. We have implemented them using the `R` package `fda`. When applied to a single data sets, these procedures are reasonably fast, and never take more then a few minutes on a single processor laptop or desktop. Some simulations, which require running the same procedure thousands of times can however take hours, or days if bootstrap is involved.

Ramsay *et al.* (2009) provide a solid introduction to computational issues for functional data, and numerous examples. Their book describes not only the `R` package `fda`, but contains many examples implemented in `Matlab`. Clarkson *et al.* (2005) describes the implementation in `S+`.

Throughout this book we often refer the choice of a *basis* and the number of basis functions. This is an important step in the analysis of functional data, which is often not addressed in detail in the subsequent chapters, so we explain it here. This is followed by brief comments on sparsely observed data.

We assume that the collected raw data are already cleaned and organized. Let $t$ be the one-dimensional argument. Functions of $t$ are observed at discrete sampling values $t_j$, $j = 1, \ldots, J$, which may or may not be equally spaced. We work with $N$ functions with indexes $i = 1, \ldots, N$; these are our functional data. These data are converted to the functional form, i.e. a functional object is created. In order to do this, we need to specify a basis. A basis is a system of basis functions, a linear combination of which defines the functional objects. The elements of a basis may or may not be orthogonal. We express a functional observation $X_i$ as

$$X_i(t) \approx \sum_{k=1}^{K} c_{ik} \phi_k(t),$$

where the $\phi_k$, $k = 1, \dots, K$, are the basis functions. One of the advantages of this approach is that instead of storing all the data points, one stores the coefficients of the expansion, i.e. the $c_{ik}$. As indicated in Section 1.3, this step thus involves an initial dimension reduction and some smoothing. It is also critical for all subsequent computations which are performed on the matrices built from the coefficients $c_{ik}$. The number $K$ of the basis functions impacts the performance of some procedures, but other are fairly insensitive to its choice. We discuss this issue in subsequent chapters on a case by case basis. We generally choose $K$ so that the plotted functional objects resemble original data with some smoothing that eliminates the most obvious noise. If the performance of a test depends on $K$, we indicate what values of $K$ give correct size. The choice of the basis is typically important. We work in this book with two systems: the *Fourier basis* and the *B–spline* basis. The Fourier basis is usually used for periodic, or nearly periodic, data. Fourier series are useful for expanding functions with no strong local features and a roughly constant curvature. They are inappropriate for data with discontinuities in the function itself or in low order derivatives. The B-spline basis is typically used for non-periodic locally smooth data. Spline coefficients are fast to compute and B–splines form a very flexible system, so a good approximation can be achieved with a relatively small $K$.

In R, bases are created with calls like:

```
minutebasis<-create.fourier.basis(rangeval=c(0,1440),nbasis=49)
```

```
minutebasis<-create.bspline.basis(rangeval=c(0,1440),nbasis=49)
```

The parameter `rangeval` is a vector containing the initial and final values of the argument $t$. The bases created above will be used for magnetometer data, which consist of 1440 data points per day. These data are in one minute resolution, and there are 1440 minutes in a day. The argument `nbasis` is the number of basis functions.

Once a basis is created, the data are converted into functional objects. This is needed to reduce the computational burden; only the coefficients $c_{ik}$ are used after this conversion. In our example, the reduction is from 1440 to 49 numbers. In order to convert raw data into a functional object the function `data2fd` is used. The code below produces Figure 1.10. The data are the daily records of the magnetic intensity stored in the matrix `data`.

```
minutetime<-seq(from = 1, to = 1440, by = 1)
minutebasis<-create.bspline.basis(rangeval=c(0,1440),nbasis=69)
data.fd<-data2fd(data, minutetime, basisobj=minutebasis)
plot.fd(data.fd, col="black")
title("Functional data, March -- April, 2001")
mean.function<-mean.fd(data.fd)
lines(mean, lw=7)
```

The `fda` package contains a variety of display functions and summary statistics such as `plot.fd`, `mean.fd`, `var.fd`, `sd.fd`, `center.fd`, etc. All these functions use functional objects as input.

The data we work with in this book are available at very densely spaced (typically equispaced) and numerous points $t_j$ (often over a thousand per curve). We

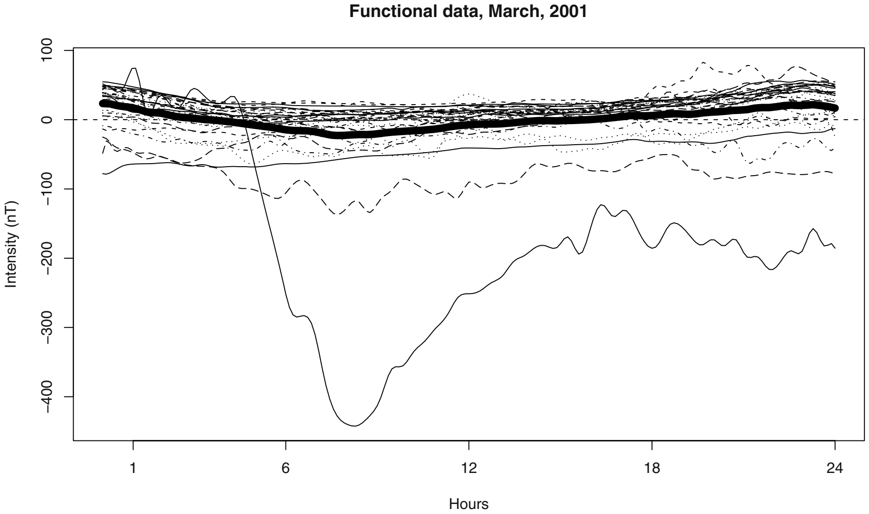

**Fig. 1.10** 31 magnetic intensity functions with the mean function (thick line)

need smoothing with a basis expansion as a means to make further calculations feasible. The measurement errors for our data are typically very small relative to the magnitude of the curves, and so are negligible. In many application, the data are available only at a few sparsely distributed points $t_j$, which may be different for different curves, and the data are available with non–negligible measurement errors, Yao *et al.* (2005a) introduce such data structures. For such data, smoothing with a basis expansion is at best inappropriate, and often not feasible. Different smoothing techniques are required to produce smooth curves which can be used as input data for the procedures described in this book. These techniques are implemented in the `Matlab` package `PACE` developed at the University of California at Davis, available at `http://anson.ucdavis.edu/$\sim$mueller/data/software.html`, at the time of writing.

After the data have been represented as functional objects, we often construct a more sparse representation by expanding them with respect to the orthonormal system formed by the functional principal components $v_k$, as illustrated in Section 1.3 and 1.4. In `R`, this is done by using the function `pca.fd`. For the procedures described in this book, the argument `centerfns` must be set to `TRUE`. This means that the sample mean function $\bar{X}_N(t) = N^{-1}\sum_{i=1}^{N} X_i(t)$ is subtracted from each $X_i(t)$ before the $v_k$ are estimated. The FPC's are thus computed for the *centered* data. Further details are presented in Section 3.4.