Chapter 1

Background Material

1.1 Preliminaries

1.1.1 Functional Data Analysis

Functional data analysis (FDA) is a field of statistics where it is assumed that the data observed at a given set of independent observation times (or coordinates etc.) represent noisy observations of some underlying function.[29]

The approach taken here is to assume that an unknown differential equation can adquately - though not necessarily exactly - describe the process producing the data.

1.1.1.1 Specification of Function Spaces

The functions in question are generally all assumed to be members of some countably infinite dimensional vector space, such as the set of all functions $f(\cdot)$ such that $\int_0^T |f''(t)|^2 dt < \infty$ over some interval $[0, T]$.

This assumption implies that any given function can be represented as a countably infinite combination of basis elements, which are themselves functions. This means for a chosen set of basis elements $\{\phi_1(t), \phi_2(t), \dots\}$ and any given function $f(t)$, there is a set of coefficients $\{a_1, a_2, \dots\}$ such that:

$$f(t) = a_1\phi_1(t) + a_2\phi_2(t) + \dots$$

Functional Data Analysis can thus be regarded as a generalisation of multivariate statistics where the number of dimensions is potentially infinite.

Substantial complications are introduced into the statistical analysis because functions are generally much richer objects than real numbers or vectors. A function will generally have a different value for each input value, and the number of non-integer numbers on any interval - and hence potential inputs - is infinite. Functions cannot be trivially represented on paper or in computer memory in a similar fashion as real numbers or vectors.

For the purposes of this thesis, it is assumed that the functions in question are continous mappings.

In practice one attempts to resolve this difficulty by finding or otherwise constructing a discrete problem that resembles the functional problem, and then solve this approximate problem.

It might be that case that the approximate problem can itself only be solved approximately using numerical methods.

Statistical problems that involve differential equations are particularly difficult. More naive approaches force the practitioner to solve the ordinary differential equation (ODE) numerically everytime it is desired to evaluate the goodness of fit. For these situations, it is necesary by definition to use numerical analytic techniques to construct a proxy problem that resembles the original problem sufficiently well and that is sufficiently easy to work with.

For example, consider the problem of parametric estimation for a stochastic differential equation (SDE) of the form

$$dX = f(X;\theta)dt + \sigma dW.$$

Here $X(t)$ is the stochastic process being modelled, $f(\cdot;\theta)$ is a known function with a parameter θ to be estimated, σ is a volatility parameter, and $W(t)$ is a standard Brownian motion.

This SDE is equivalent to asserting for any time t and increment h that

$$X(t+h) = X(t) + \int_t^{t+h} f(X(s);\theta)ds + \sigma[W(t+h) - W(t)].$$

Suppose there are observations $X_1, X_2, \ldots, X_N$ of $X(t)$ at evenly spaced times, and that h is the distance between the time points. The integral formulation of the SDE suggests that if h is small enough, then

$$X_{k+1} \approx X_k + f(X_k;\theta) + \sigma h Z_{k+1}.$$

The Z_k here are i.i.d standard Normal random variables. This is known as the *Euler-Maruyama Approximation.*

Instead of attempting to estimate parameters for the SDE, we can fit parameters for a non-linear AR(1) process that acts as a proxy problem for the original SDE. This is a much more tractable problem than the original SDE.

In FDA, the assumption is usually made that all the functions can be represented as a linear combination from some choosen *finite* set of basis functions. Rather than discretise the differential operator as in the above example, the space of functions is discretised instead.

A differential equation (or a similar problem) over some finite dimesional space of functions with n dimensions can be represented as a problem over the Euclidean space R^n, this is a discrete problem.

The modelling process for functional data as described in Figure 1.1 can be more complex than standard statistical problems.

Formulate a Model: As is the case for any statistical problem, the first step is to formulate a model . Here one must only be certain that the model at used is sufficiently broad or well-specified to be able to actually capture the phenomona at hand.

Construct a Discretised Model that Approximates the Original Model: Unless the statistical model is trivial, the next step is to construct a proxy model. This generally requires ideas from Numerical Analysis.

Conduct Statistical Analysis Using the Discretised Model: While the discretised model tends to be simpler than the original model, this task is not necesarily trivial as shall be seen.

Check the Approximation Error in Discretised Model: If the discretised model is too poor an approximation, then the results of any statistical analysis conducted could be substantially biased as a result of the approximation error introduced, even if the original model were perfectly sound. If the orginal model is biased, then the approximate one might be even more so.

Therefore, one should consider conducting post hoc checks. For example, running the analysis again with a more accurate approximate model and comparing the results with the original model. If both agree, it is evidence the approximate models are both reasonably accurate. Constructing a more accurate approximation is generally a straightforward and intuitive process, with the exact approach depending on the situation at hand.

In the context of FDA, this generally entails increasing the number of basis functions so that the associated approximation error is smaller.

For example, suppose that one were attempting to estimate the parameters of an ODE by means of least squares, and one was using a finite difference solver to compute the fitted values, and hence to determine the goodness-of-fit.

Once the fitting alogrithm had converged, one might run the solver again with a smaller stepsize and the same parameters and check if this has made a substantial change in the the sum of squared residuals.

If there has been a substantial change as a result of the stepsize reduction, then one would have to consider running the entire fitting procedure again starting from the previously computed parameter estimate, except with the smaller stepsize, taking the new parameter estimates, and then checking again if decreasing the stepsize yet again produces substantial change in the goodness-of-fit statistic.

This procedure can even be automated. The Implicit Filtering alogrithm computes an approximate gradient using finite differnences and uses this to perform optimisation. If the alogorithm cannot produce a decrease in the objective function, or it cannot be

certain that the true gradient isn't in fact zero, it reduces the stepsize. The algorithm terminates when the change in the objective function between changes in the stepsize has fallen below a chosen tolerance level.

If the fitting method used is slow however, then these such approaches can potentially be very slow due to the need to solve the same problem over and over again at increasing levels of precision.

Fortunately, Functional Data Analysis does not always require the recomputation of the curve in such a fashion whenever the parameters are changed. Instead of being implicitly represented as solutions of an ODE, functions are explicitly represented as elements in some finite dimesional vector space. As shall be seen, the objective function is generally a mapping from some vector space R^n to R that can often be evaluated reasonably easily, or at least more easily than having to run an ODE solver.

Check If Results of Statistical Analysis Are Consistent With Discretised Model. In the previous step, one checked that the approximate model was actually acting as a proxy for the original model. One must then check that the statistical analysis conducted using the approximate model is valid in its own right. For example, it will be seen throughout this thesis that many statistical problems involving functions can be approximated by nonlinear regression models. These constructed nonlinear regression models should be checked for statistical validity.

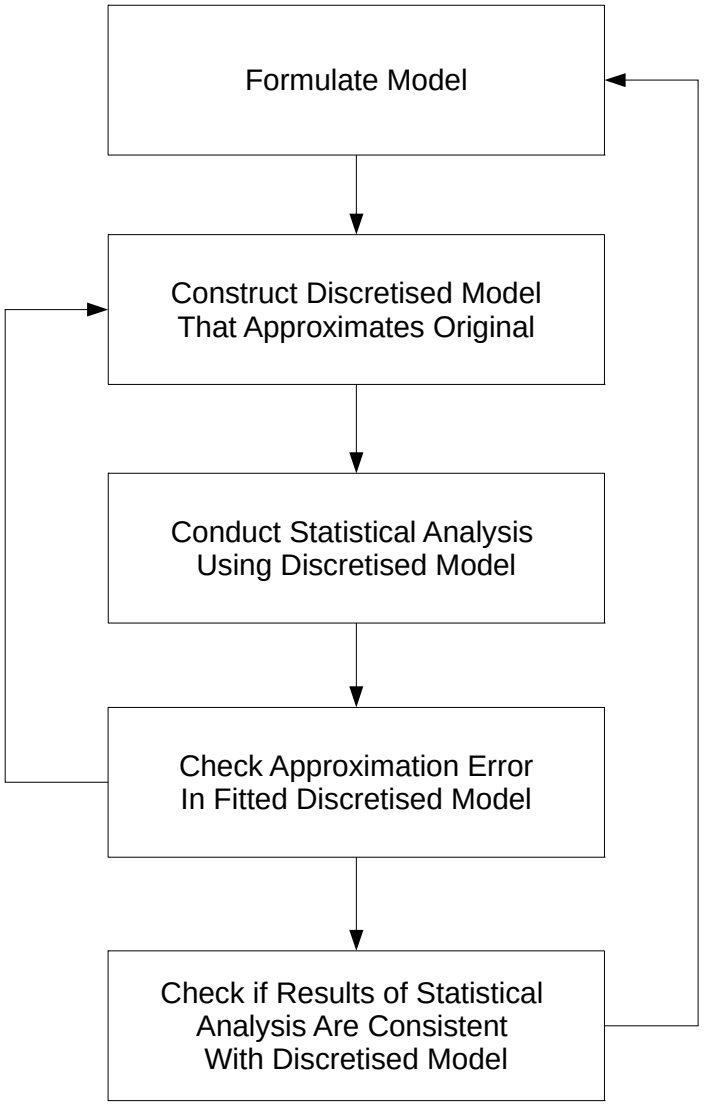

Figure 1.1: Statistcal Modelling Process For Functions

1.2 Penalised Regression

Suppose we have N noisy observations y_i at times t_i from some function $f(t)$, and we wish to estimate $f(t)$, from the data. A naive approach would be to estimate $f(t)$ by minimising a least squares criterion:

$$SSE(f) = \sum_{i=1}^{N} [y_i - f(t_i)]^2$$

Here, $SSE(\cdot)$ is a function that assigns a real number to every real-valued function that is defined for all the t_i.

There is an obvious problem with this criterion - it does not have a unique minimiser. Any function $g(t)$ such that $g(t_i) = y_i$ is will minimise $SSE(\cdot)$. There are an infinite number of degrees of freedom, but only a finite number of observations.

To ensure uniqueness, it is necessary to impose a further condition to discriminate between different candidates, a way to choose between different functions that interpolate a given set of points.

1.2.1 Smoothing Splines

One potential criterion is to introduce a second order penalty. If two functions fit the observed data equally well, the more regular or less "wiggly" function is chosen. There are several ways of translating this intuition into a formal fitting procedure.

A common choice is to measure the degree of irregularity by using the integral of the second derivative over a chosen interval $[0, T]$. The upper limit T should be choosen to allow for all observation times to be included.

$$\int_0^T |f''(t)|^2 dt.$$

For a given set of points, the smooth interpolating curve that minimises the energy integral above is given by an interpolating cubic spline.

Choosing the most regular interpolating curve is not necessarily a very good estimation strategy however because it strongly prioritises goodness-of-fit above all other considerations. If the data is noisy, there is a risk of overfitting and poor predictive power. There is a trade-off between bias and variance.

In practice, a joint estimation strategy is pursued that attempts to find a good balance between fidelity to the observed data and reasonably regular behavior. This involves minimising the following penalised least squares criterion:

$$PENSSE(f; \lambda) = \sum_{i=1}^{N} (y_i - f(t_i))^2 + \lambda \int_0^T |f''(t)|^2 dt$$

The λ term dictates the trade-off between fidelity to the data and regularity.

Suppose there were a candidate function $g(t)$, then by taking the cubic spline such that its value at t_i is equal to $g(t_i)$, we can produce a curve $s(t)$ that has the same least-squares error as $g(t)$, but with $\int [s''(t)]^2 dt \leq \int [g''(t)]^2 dt$. Thus, the curve that minimises $PENSSE$ can be assumed to be a cubic spline.

To find the minimiser of $PENSSE(\cdot; \lambda)$ first, assume that $f(t)$ can be represented as a linear combination of K cubic spline functions $\phi_i(t)$ that can represent any cubic spline with knots at the t_i. This implies that

$$f(t) = \sum_{i=1}^{K} c_i \phi_i(t).$$

Note that it is only required that the set of basis splines only possess enough resolution to represent the function that minimises $PENSSE$, it is not required that this set of splines is minimal.

Let the design matrix $\mathbf{\Phi}$ be defined by $\mathbf{\Phi}_{ij} = \phi_i(t_j)$, and let the weight matrix $\mathbf{R}$ be defined by $\mathbf{R}_{ij} = \int_0^T \phi_i''(t)\phi_j''(t)$. Then $PENSSE$ can be written in terms of the vector of coefficients $\mathbf{c}$ and observations $\mathbf{y}$ as:

$$PENSSE(\mathbf{c}; \lambda) = \|\mathbf{y} - \mathbf{\Phi c}\|^2 + \lambda \mathbf{c}'\mathbf{R}\mathbf{c}$$

The problem has been discretised into one on R^K.

The optimal value of $\mathbf{c}$ is given by

$$\hat{\mathbf{c}} = (\mathbf{\Phi}'\mathbf{\Phi} + \lambda \mathbf{R})^{-1}\mathbf{\Phi}'\mathbf{y}$$

This is an exact solution to the original problem because the span of the $\{\phi_i(t)\}$ contains the function that minimises $PENSSE$. The coefficient vector $\hat{\mathbf{c}}$ is the set of coordinates of the optimal function within this finite-dimensional vector space.

1.2.2 Piecewise Trigonometric Interpolation

Consider a more difficult penalised regression problem:

$$PENSSE(f; \lambda) = \sum_{i=1}^{N} (y_i - f(t_i))^2 + \lambda \int_0^T |f''(t) - f(t)|^2 dt$$

The penalty against $f''(t)$ has been replaced with a penalty against $f''(t) - f(t)$. $PENSSE$ can be minimised in this case taking by a piecewise function consisting of linear combinations of $\sin(t)$ and $\cos(t)$ over each interval, and matching them together.

Note that a function of the form

$$a_0 + a_1 \cos(t) + b_1 \sin(t) + a_2 \cos(2t) + b_2 \sin(2t) + \ldots$$

can be written as a polynomial in e^{it} and e^{-it}. For this reason, such a piecewise trignometric function can also be referred to as a piecewise trignometric polynomial or a piecewise trignometric spline.[33]

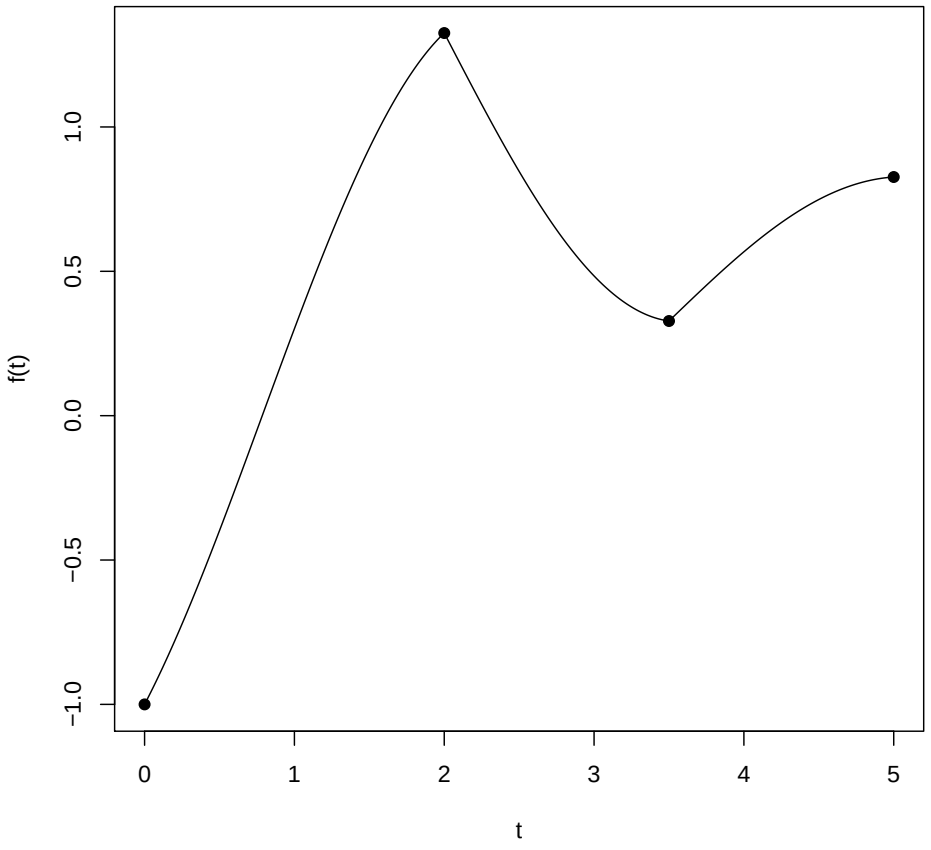

Figure 1.2: Plot of a Piecewise Trigonometic Curve. Note the kinks between segments.

As can be seen in Figure 1.2, a piecewise trignometric polynomial of second degree generally fails to be smooth at the boundary points, and thus has a kinked appearance. For the purposes of statistical modelling, it is strongly desireable to impose the additional constraint that $f(t)$ must be everywhere differentiable. This cannot be achieved for a pieceswise basis formed from the functions $\{\sin(t), \cos(t)\}$ because there are only two free parameters on each segement and they are needed to ensure continuity.

1.3 Finte Dimensionalisation: the General Case

To find an exact solution to the two problems in Section 1.2, it was necessary to construct a finite dimensional function space that contained the minimal function. However it is not guaranteed that this is always possible. In practice, one would hope that the optimal function can be approximated sufficiently well by taking a linear combination from some choosen set of functions. Spline bases tend to be a reliable workhorse that are effectively the default choice. They provide a good balance between being well behaved as objects for regression and having good approximating power.

For comparison, Chebyshev Polynomials can often provide better approximation power for a given number of basis functions.[2] Unfortunately, it was found that they can be poorly behaved statistically because they consist of high order polynomials that are difficult to fit to data.

Functional Data Analysis thus consists of the following steps, illustrated in Figures 1.3 and :

1. Formulate a model for $f(t)$. Usually, this takes the form of a penalised regression model, where $f(t)$ is defined as the function that minimises some kind of penalised error

2. Assume that $f(t)$ can be written as a finite combination of chosen basis functions. In practice, this is only approximately true, so it is important to ensure that our basis can actually approximate the optimal $f(t)$ sufficently well. The function $f(t)$ can thus be written:

$$\begin{aligned} f(t) &= \sum_{i=1}^{K} c_i \phi_i(t) \\ &= [c_1, \ldots, c_K]'[\phi_1(t), \ldots, \phi_K(t)] \\ &= \mathbf{c}'\phi(t) \end{aligned}$$

   Note that $f(t)$ is now defined by the coefficent vector $\mathbf{c}$.

3. Formulate the model in terms of the coefficient vector $\mathbf{c}$. A statistical problem over some given functional space has been transformed into a statistical problem over R^K.

For a given choice of $\mathbf{c}$, one gets a goodness-of-fit statistic of some kind back. It's important to note that the problem of fitting the coefficients $\mathbf{c}$ is a problem in nonlinear regression as a result of the finite dimesionalisation. Besides formulating an FDA model, one needs to consider the questions of constructing a finite dimensional approximation and then solving the associated nonlinear regression. The situation is sketched in Figure 1.4.

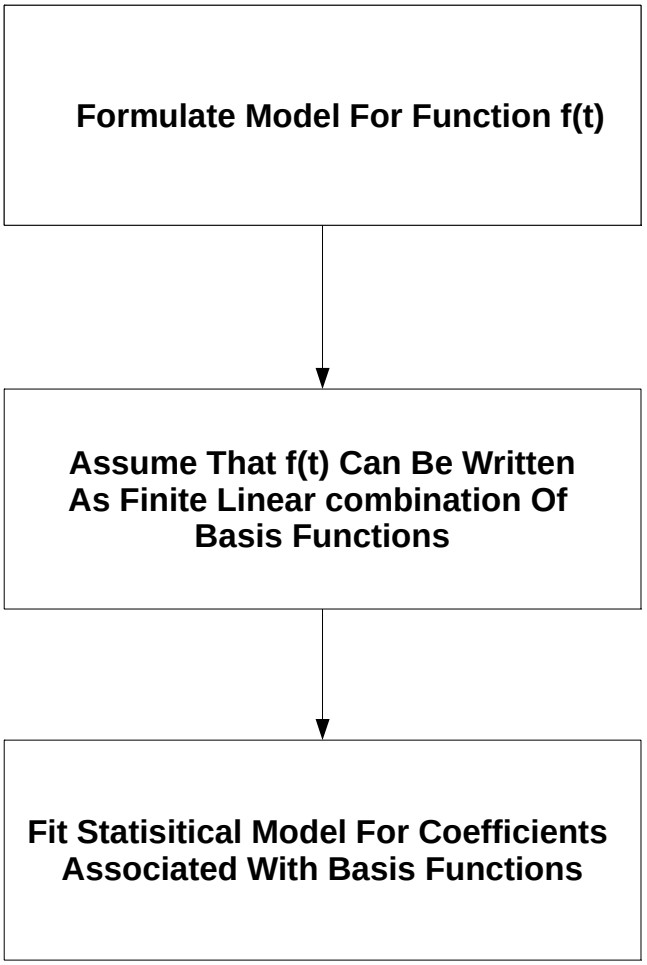

Figure 1.3: Statistical Modelling Process For Functional Data Analysis

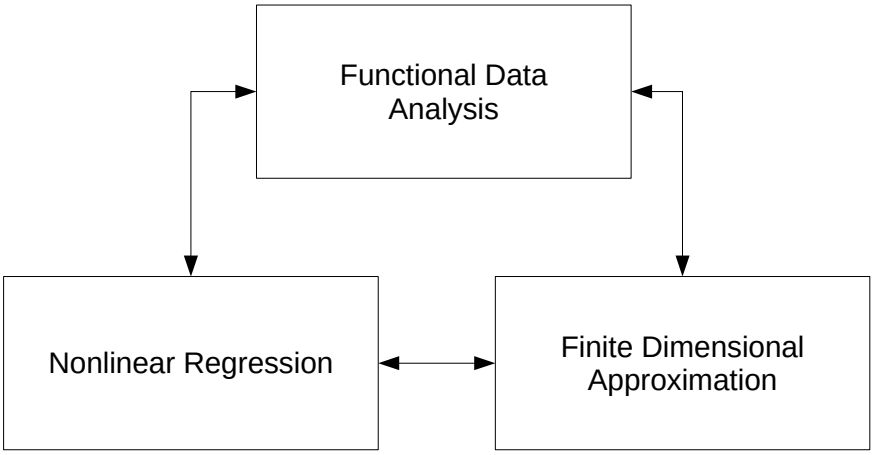

Figure 1.4: Elements of Functional Data Analysis

1.3.1 FDA With A Quadratic Basis

As is done in some texts[1], we will immediately provide an example with a very small basis to illustrate these steps. Consider the following penalised regression problem:

$$PENSSE(f;\lambda) = \sum_{i=1}^{N}[y_i - f(t_i)]^2 + \lambda \int_0^1 |t^2 f'' - 0.5f|^2 dt$$

The differential equation associated with the penalty term is known as an Euler's Equation, and can be regarded here as a toy equation representative of equations such as Bessel's equation. The solution is given by $f(t) = at^{r_1} + bt^{r_2}$, where r_1 and r_2 are the roots of the quadratic equation $r^2 - r - 0.5 = 0$. Thus, $r_1 \approx -0.36$ and $r_2 \approx 1.36$.

For the sake of illustration it will be assumed that that $f(t)$ can be written as a quadratic - a linear combination of the basis functions $\{1, t, t^2\}$:

$$f(t) = at^2 + bt + c$$

Then:

$$\begin{aligned}
\int_0^1 |t^2 f'' - 0.5f|dt &= \int_0^1 \left| at^2 - \frac{1}{2}(at^2 + bt + c) \right|^2 dt \\
&= \int_0^1 \left| \frac{1}{2}\left(at^2 - bt - c\right) \right|^2 dt \\
&= \frac{1}{4}\int_0^1 |at^2 - bt - c|^2 dt \\
&= \frac{1}{4}[a \ -b \ -c]'\mathbf{H}[a \ -b \ -c] \\
&= \frac{1}{4}[a \ b \ c]'(\mathbf{A}'\mathbf{H}\mathbf{A})[a \ b \ c] \\
&= [a \ b \ c]'\mathbf{K}[a \ b \ c]
\end{aligned}$$

Here $\mathbf{K} = \frac{1}{4}\mathbf{A}'\mathbf{H}\mathbf{A}$, the elements of the matrix $\mathbf{H}$ are defined by $\mathbf{H}_{ij} = \int_0^1 t^i t^j dt = 1/(i+j+1)$, and elements of the matrix $\mathbf{A}$ are given by:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

Thus, the penalised error is given by:

$$PENSSE(a,b,c;\lambda) = \sum_{i=1}^{N}(y_i - at_i^2 - bt_i - c)^2 + \lambda[a \ b \ c]'\mathbf{K}[a \ b \ c] \tag{1.1}$$

[1] [2] for example.

We have now gone from a problem specified in terms of functions, to a penalised least squares problem in the three coefficents a, b and c. The quality of this approximate model as λ gets larger and larger depends on how well the functions $t^{-0.36}$ and $t^{1.36}$ can be respectively approximated by quadratics over the interval $[0, 1]$.

To illustrate this example further, the method was fitted to simulated data. A solution to the ODE $t^2 f'' - f = 0$ was generated over the interval $[0, 1]$, samples were taken at various points before being corrupted by Gaussian noise. The quadratic that minimised (1.1) with $\lambda = 100$ was then found. For comparison, the data was also fitted to a quadratic using ordinary least squares. The original function $f(t)$, the perturbed data, and the two fitted functions are all shown in Figure 1.5

It's already been noted that the quality of the model depends partially on how well $f(t)$ can ever be approximated by a quadratic over $[0, 1]$ in the first place. Therefore, the quadratic $q(t)$ that minimises $\int_0^1 |f(t) - q(t)|dt$ was found numerically and also plotted in Figure 1.5.

Figure 1.5 suggests that $f(t)$ can be approximated reasonably well by quadratics for so long as one stays away from the point $t = 0$. This is consistent with theory. The ODE $t^2 f'' - f = 0$ behaves degenerately at the origin. When $t = 0$, the ODE has what is known as a singular point, the term in front of f'' becomes zero so that the ODE reduces to $(0)^2 f'' - f = 0$. Additionally, it is always the case that the second derivative diverges to infinity at 0 if $f(t)$ is of the form $at^{-0.36} + bt^{1.36}$. As a result of both the singular point and infinite curvature at $t = 0$, polynomial approximation is predicted to be execptionally tricky around this point.[13, 37]

Comparing the two fits in Figure 1.5, it is fair to argue that the penalised regression model captures the shape of $f(t)$ better than ordinary least squares away from $t = 0$. Both models seem to have similar predictive power on average. The penalised fit is being heavily influenced by the singularity at $t = 0$ and probably would have performed better if a more robust loss function than least squares were used.

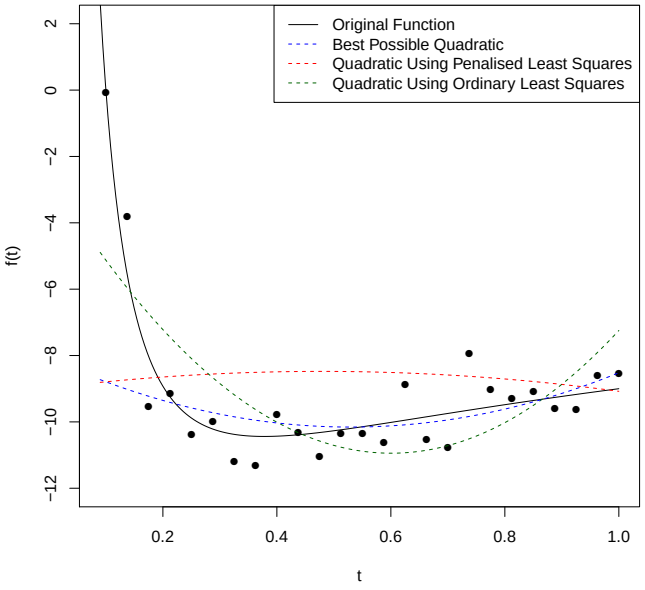

Figure 1.5: Performing FDA with the differential operator $Lf = t^2 f'' - 0.5f$ and the basis set $\{1, t, t^2\}$.

1.4 The FDA Package

Instead of having to develop develop FDA algorithms from scratch as done in Section 1.3, the `FDA` package was developed[28, 31] to tackle penalised problems of the form:

$$PENSSE(f) = \sum_{i=1}^{N} [y_i - f(t_i)]^2 + \lambda \int |Lf(t)|^2 dt \tag{1.2}$$

Here Lf is a parameterised linear differential operator of the form $\sum_{j=0}^{n} \beta_j D^j$ where the β_j are constants. The result of fitting the differential operator $Lf = f - \omega^2 f^{(4)}$ with $\omega = 0.65$ is shown in Figure 1.6.

The `FDA` packages is not as powerful as the `Data2LD` package, which will be introudced later on. It has the advantage of simplicity and ease of use though, and is used throughout this thesis to fit FDA models unless `Data2LD` is essential. A deficiency of the `FDA` package is that it provides no guidence on the best choice of the parameters β_i nor the smoothing parameter λ.[2]

[2]The `FDA` package has a command called `lambda2gcv` whose documentation claims it '[finds] the smoothing parameter that minimizes GCV' [28]. Inspection of the code for this function shows that it only performs a fit based on the value of λ passed and then reports the GCV. Incorrect or unclear documentation is unfortunately not an uncommon problem with FDA codes.

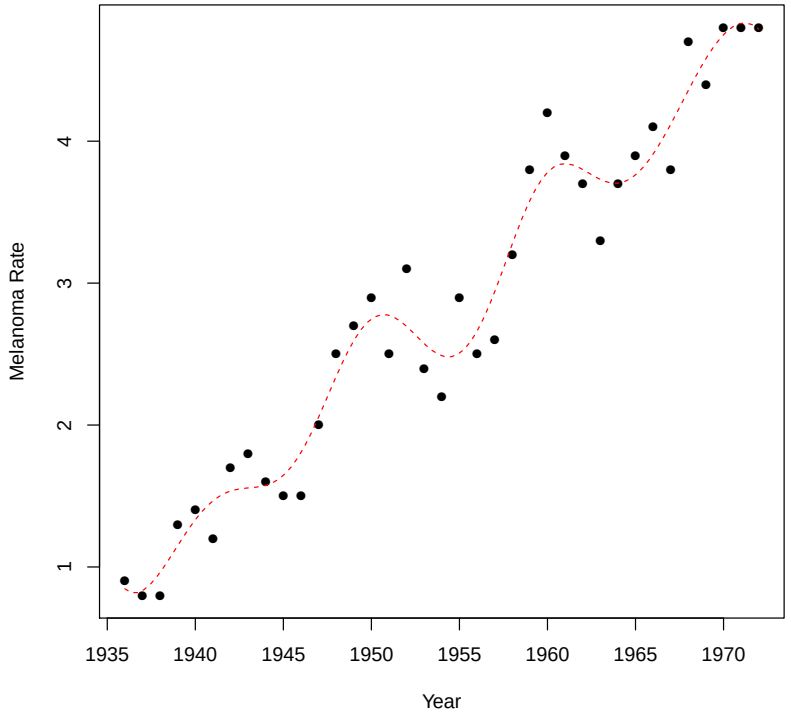

Figure 1.6: Using the `FDA` package to smooth the melanoma data with the differential operator $Lf = f - \omega^2 f^{(4)}$.

1.5 The Data2LD Package

The Data2LD package is an R package intended to perform smoothing using general linear differential operators with a forcing function, that is, ODEs of the form:

$$\sum \beta_i(t) D^i f(t) = u(t) \tag{1.3}$$

The $\beta_i(t)$ are paremeter functions for the linear differential opertor on the lefthand side, and $u(t)$ is a forcing function.

More generally, Data2LD can model a system of inhomogenous linear differntial equations:

$$\mathbf{y}(t)' + \mathbf{B}(t)\mathbf{y} = \mathbf{u}(t) \tag{1.4}$$

Each element of $\mathbf{B}(t)$ is a time-varying linear parameter function of the the form $\beta_{ij}(t)$ and each element of $\mathbf{u}(t)$ denotes the forcing function applied to the ith equation.

A further advantage of Data2LD over the FDA package is that not only can it smooth ODEs with functional parameters, but it estimate the associated parameters even if they are functions.

While Data2LD can estimate parameters for the differential operator, it does not provide a means for finding the optimal smoothing parameter.[3]

1.6 Modelling the Reflux Data: A Parametric Approach vs Data2LD

The Reflux data, plotted in Figure 1.7, describes the output of an oil refining system. A given fraction of oil is being distilled into a specific tray, at which point it flows out through a valve. At a given time, the valve is switched off, and distillate starts to accumulate in the tray [29]. The Reflux data was taken from the Data2LD package used for FDA, which will be discussed in more detail later. The authors of the Data2LD package model the data using the following ODE:

$$\begin{cases} y'(t) = -\beta y(t) & t \leq t_0 \\ y'(t) = -\beta y(t) + u_0 & t \geq t_0 \\ y(0) = 0 \end{cases} \tag{1.5}$$

Up until the point t_0, the function satisfies the ODE $y' = -\beta y$. At the breakpoint, a constant forcing function u_0 is turned on to model the valve being switched off, so that the ODE then becomes $y' = -\beta y + u_0$.

This ODE admits an exact solution. Letting $\gamma = u_0/\beta$ and C be an arbitray constant, then the solution is given by

[3]For Data2LD, the smoothing parameter is written in terms of $\rho = \lambda/(1+\lambda)$.

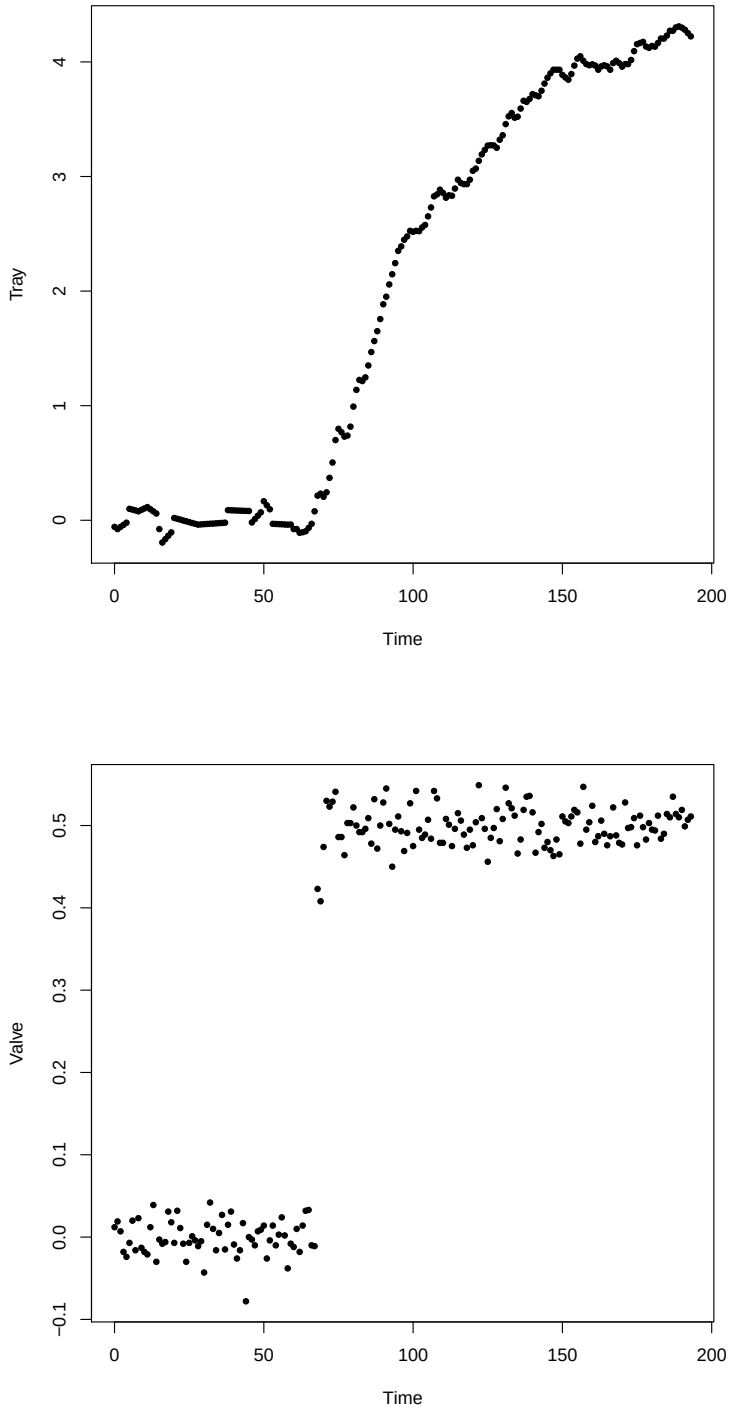

Figure 1.7: Reflux Data

$$y(t) = \begin{cases} 0 & t \leq t_0 \\ \gamma + Ce^{-\beta(t-t_0)} & t \geq t_0 \end{cases}$$

Without loss of generality the exponential term $Ce^{-\beta(t-t_0)}$ can be replaced with one of the is of the form $Ce^{-\beta t}$. This is the case because $Ce^{-\beta(t-t_0)} = Ce^{-\beta t}e^{-\beta t_0} = [Ce^{-\beta t_0}e^{-\beta t}]$, the $e^{-\beta t_0}$ term is thus absorbed into the constant term.

In order to ensure that $y(t)$ is continous at t_0 and monotone increasing, we require that $\gamma + C = 0$ and that $\beta > 0$

1.6.1 Parametric Approach

It turns out that the constraint $C = -\gamma$ is unsuitable from the point of view of numerical parameter estimation. R's `nls` command reports errors when this constraint is imposed.

However, if we allow t_0 to vary, we can allow C to assume any negative value while preserving monotonicity and continuity.

Assume that $y(t)$ is instead given by:

$$\tilde{y}(t) = \max(0, \gamma + Ce^{-\beta(t-t_0)})$$

The function $\tilde{y}(t)$ satisfies the same ODE and initial conditions as $y(t)$ except that the change point t_0 is shifted to t_0' defined by:

$$t_0' = \max\left(t_0, t_0 - \frac{1}{\beta}\ln\left(\frac{-\gamma}{C}\right)\right)$$

The function $\tilde{y}(t)$ is a combination of simpler functions, joined together using the maximum operator instead of the addition operator, see Figure 1.8.

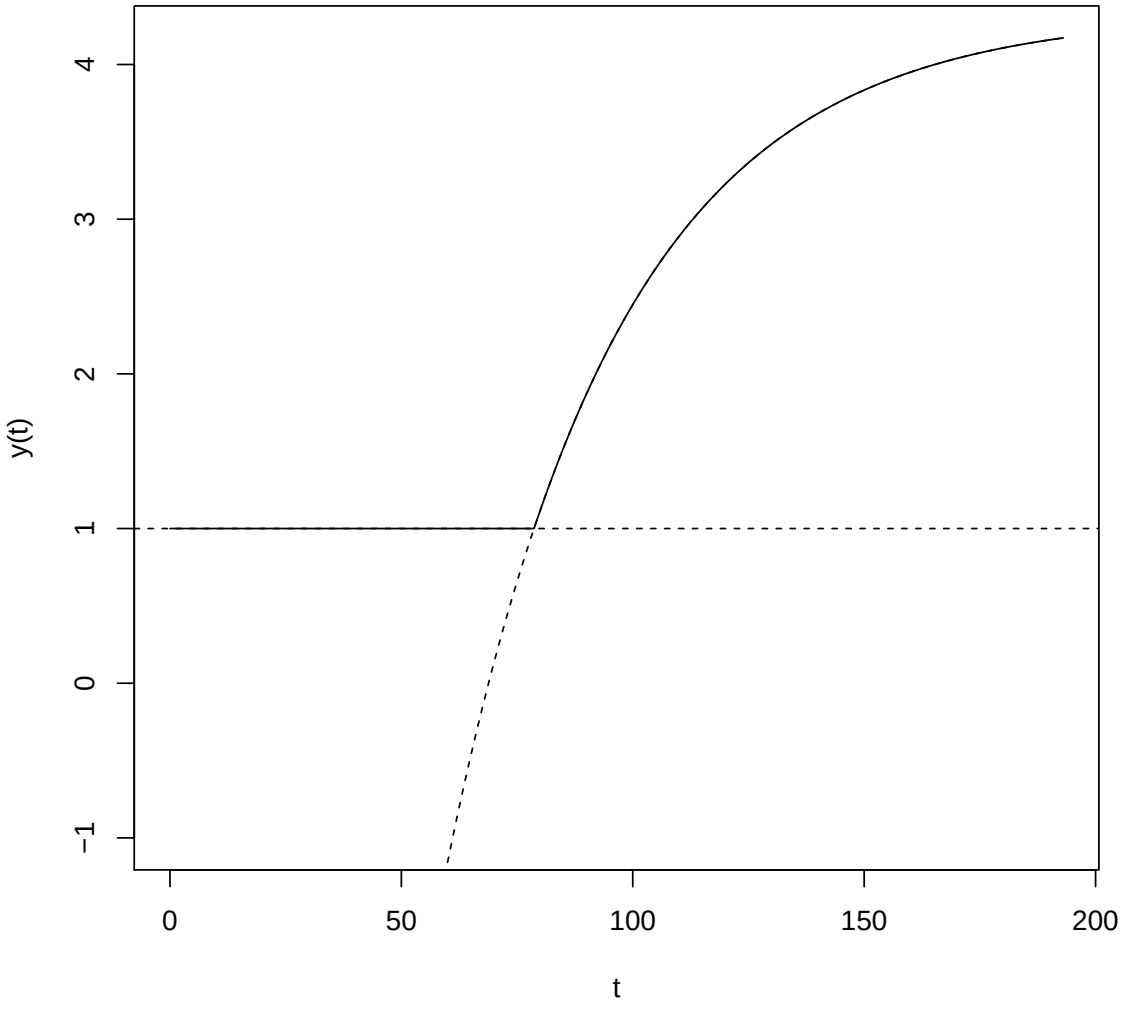

Figure 1.8: Plot of $\tilde{y}(t)$ and its constituent functions

1.6.1.1 Parametric Fitting

Instead of approximately solving an associated problem as discussed in Section 1.3, a purely parametric approach to fitting the ODE (1.5) will be employed. The question of modelling the Reflux data using FDA will be discussed in a later chapter.

We assume that the breakpoint t_0 is known in advance. Then our model for $y(t)$

$$y(t) = \begin{cases} 0 & t \leq t_0 \\ \beta_0 + \beta_1 e^{\beta_2 t} & t \geq t_0 \end{cases} \tag{1.6}$$

Note that this function might not be well defined at t_0, we will adress the question of matching later on. We must estimate the three unknown coefficients $\beta_0, \beta_1, \beta_2$.

Estimating β_0 from the data: Figure 1.7 suggests that $\beta_2 < 0$, and $\beta_1 < 0$, under this assumption, we have that:

$$\lim_{t \to \infty} y(t) = \beta_0$$

Where the convergence happens monotonically from below

So an initial estimate for β_0 is given by $\hat{\beta}_0 = \max(y_i)$

Estimating β_1 and β_2 from β_0 and the data: For $t \geq t_0$, the model in Equation 1.6 can be rearranged so that:

$$\log(\beta_0 - y(t)) = \log|\beta_1| + \beta_2 t$$

This equation is only valid so long as the left hand side is well defined however. It is necessary to exclude the largest observed value of y.

The values of $\log|\beta_1|$ and β_2 can be estimated by performing linear regression against $\log(\beta_0 - y(t))$, with the largest value of y observed excluded. It was assumed that $\beta_1 < 0$, so $\hat{\beta}_1$ can be found from the estimate of $\log|\beta_1|$.

Simulataneous Estimation of Parameters: Now that we have reasonable estimates for β_0, β_1, and β_2, we can use non linear regression to estimate all three jointly.

Matching: For $t < t_0$, it is estimated that $\hat{y}(t) = 0$. For $t \geq t_0$. the estimate is given by $\hat{y}(t) = \hat{\beta}_0 + \hat{\beta}_1 e^{\hat{\beta}_2 t}$. There are distinct estimates for $y(t)$ at $t \leq t_0$ and $t \geq t_0$, which do not necessarily agree at $t = t_0$. This is the case for the estimates produce here since $\hat{y}(t_0) = 0.029$.

To stitch the two functions together, let $\hat{y}(t) = \max(0, \hat{\beta}_0 + \hat{\beta}_1 e^{\hat{\beta}_2 t})$. This is a continous function that entirely satisfies the orginal ODE, except for the precise location of the breakpoint.

The resulting fit is presented in Figure 1.9.

Breakpoint Estimation: The value of t_0 used for the fit is given by $t_0 = 68$. A statistical estimate of the breakpoint can be found from finding the point where $\hat{\beta}_0 + \hat{\beta}_1 e^{\hat{\beta}_2 t}$ is zero:

$$\hat{t}_0 = \left| \frac{1}{\hat{\beta}_2} \log \left(-\frac{\hat{\beta}_0}{\hat{\beta}_1} \right) \right|$$

Using this formula, it was estimated that $t_0 = 67.71$. This new value will produce the same results as for $t_0 = 68$ because it doesn't change the set of observation points used to estimate β_0, β_1, and β_2.

1.6.1.2 Discussion

The parametric approach taken to estimation here is somewhat *ad hoc.* Instead of devising a formal estimation strategy in advance, the fitting approach evolved organically alongside the problems of solving the ODE and fitting the data. Use was made of properties tied to the ODE model to compute estimates. While this has produced an effective fit, there are obvious concerns about generalising this approach to other ODEs. Futhermore, since the fitting model was devised by peeking at the data, it is not obvious that one can find a valid p-value for the fit without getting an entirely new set of data.

This issue is difficult to resolve using purely parametric methods. It is often the case in Applied Mathematics that one can't fully investigate an ODE model until one has a rough grasp of its behaviour. It has been demonstrated that the associated Statistical fitting problem inherits this tendency.

1.6.2 Fitting the Reflux Data with Data2LD

While the parametric approach employed in Section 1.6.1 requires a considerable amount of domain-specific knowledge, the functional model can be more generally employed. The FDA approach doesn't rely on individual features of the specific differential equation at hand,[4] and produces a similar fit to the Reflux data as the parametric approach.

The functional model asserts that

$$y'(t) \approx -\beta y(t) + u(t)$$

Where $y(\cdot)$ and $u(\cdot)$ are functions to be estimated, and β is a single scalar parameter.

It is assumed that $u(t)$ is a step function of the form

$$u(t) = a\mathbb{I}_{[0,t_0)}(t) + b\mathbb{I}_{[t_0,\infty)}(t)$$

[4]The FDA approach does rely on more general features of course, such as whether or not the differential equation is linear.

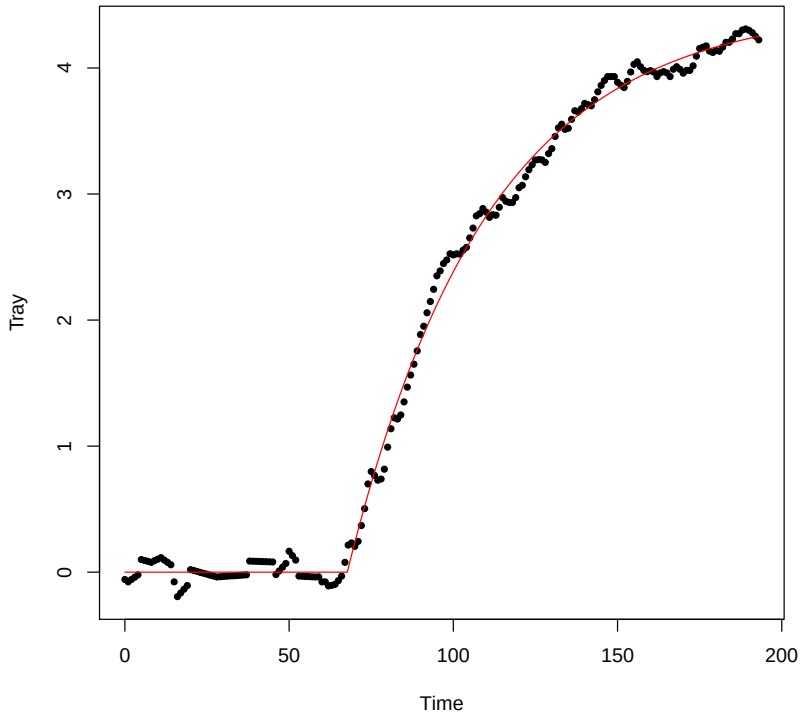

Figure 1.9: Fitting the Reflux data to the ODE model parametrically.

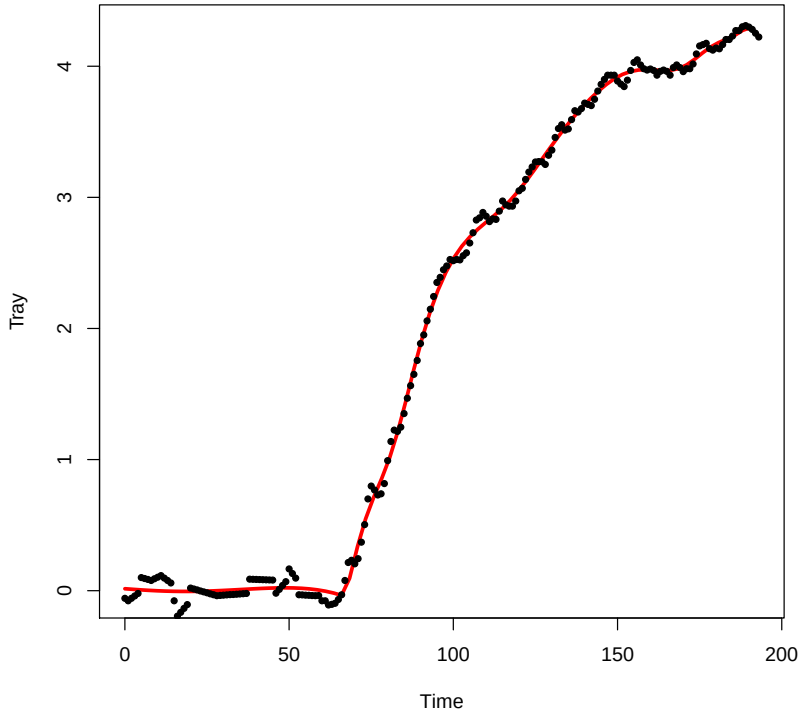

Figure 1.10: Modelling the Reflux data using `Data2LD`.

As in the parametric case, the breakpoint t_0 is fixed in advance. It is further assumed that $y(t)$ can be expanded as a linear combination of B-Splines. The knots are duplicated at t_0 so that the first derivative at the breakpoint is discontinous.

This model was fitted using the `Data2LD` package, and the results are plotted in Figure 1.10. It can be seen that the fit is quite similar to the parametric one presented in Figure 1.9. The main disadvantage of the FDA approach compared to the parametric one is that `Data2LD` can be complex and unintuitive to use.

Bibliography

[1] Jonathan Barzilai and Aharon Ben-Tal. Nonpolynomial and inverse interpolation for line search: synthesis and convergence rates. *SIAM Journal on Numerical Analysis*, 19(6):1263–1277, 1982.

[2] John P Boyd. *Chebyshev and Fourier spectral methods*. Courier Corporation, 2001.

[3] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[4] Richard P Brent. *Algorithms for minimization without derivatives*. Courier Corporation, 2013.

[5] Jiguo Cao and James O Ramsay. Parameter cascades and profiling in functional data analysis. *Computational Statistics*, 22(3):335–351, 2007.

[6] Kwun Chuen Gary Chan. Acceleration of expectation-maximization algorithm for length-biased right-censored data. *Lifetime data analysis*, 23(1):102–112, 2017.

[7] Edwin KP Chong and Stanislaw H Zak. *An introduction to optimization*, volume 76. John Wiley & Sons, 2013.

[8] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[9] Ronald Aylmer Fisher. The wave of advance of advantageous genes. *Annals of eugenics*, 7(4):355–369, 1937.

[10] Bengt Fornberg. Generation of finite difference formulas on arbitrarily spaced grids. *Mathematics of computation*, 51(184):699–706, 1988.

[11] PR Graves-Morris, DE Roberts, and A Salam. The epsilon algorithm and related topics. *Journal of Computational and Applied Mathematics*, 122(1-2):51–80, 2000.

[12] David R Hunter and Kenneth Lange. A tutorial on MM algorithms. *The American Statistician*, 58(1):30–37, 2004.

[13] Eugene Isaacson and Herbert Bishop Keller. *Analysis of numerical methods*. Courier Corporation, 2012.

[14] Carl T Kelley. *Iterative methods for linear and nonlinear equations*, volume 16. Siam, 1995.

[15] Carl T Kelley. *Implicit filtering*, volume 23. SIAM, 2011.

[16] C.T. Kelley. A brief introduction to implicit filtering. `https://projects.ncsu.edu/crsc/reports/ftp/pdf/crsc-tr02-28.pdf`, 2002. [Online; accessed 12-October-2019].

[17] Jack Kiefer. Sequential minimax search for a maximum. *Proceedings of the American mathematical society*, 4(3):502–506, 1953.

[18] Kenneth Lange. *Optimization*. Springer, 2004.

[19] Kenneth Lange. *Numerical analysis for statisticians*. Springer Science & Business Media, 2010.

[20] Kenneth Lange. The MM algorithm. `https://www.stat.berkeley.edu/~aldous/Colloq/lange-talk.pdf`, April 2007. [Online, accessed 18-September-2019].

[21] Randall J LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*, volume 98. Siam, 2007.

[22] Steve McConnell. *Code complete*. Pearson Education, 2004.

[23] Geoffrey McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*, volume 382. John Wiley & Sons, 2007.

[24] J Nocedal and SJ Wright. *Numerical Optimisation*. Springer verlag, 1999.

[25] Naoki Osada. *Acceleration methods for slowly convergent sequences and their applications*. PhD thesis, Nagoya University, 1993.

[26] Yudi Pawitan. *In all likelihood: statistical modelling and inference using likelihood*. Oxford University Press, 2001.

[27] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. ISBN 3-900051-07-0.

[28] J. O. Ramsay, Hadley Wickham, Spencer Graves, and Giles Hooker. *fda: Functional Data Analysis*, 2018. R package version 2.4.8.

[29] James Ramsay. Functional data analysis. *Encyclopedia of Statistics in Behavioral Science*, 2005.

[30] Jim O Ramsay, Giles Hooker, David Campbell, and Jiguo Cao. Parameter estimation for differential equations: a generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(5):741–796, 2007.

[31] JO Ramsay, G Hooker, and S Graves. *Functional data analysis with R and MATLAB*. Springer Science & Business Media, 2009.

[32] Walter Rudin et al. *Principles of mathematical analysis*, volume 3. McGraw-hill New York, 1964.

[33] Larry Schumaker. *Spline functions: basic theory*. Cambridge University Press, 2007.

[34] Christopher G Small. A survey of multidimensional medians. *International Statistical Review/Revue Internationale de Statistique*, pages 263–277, 1990.

[35] Karline Soetaert, Thomas Petzoldt, and R. Woodrow Setzer. Solving differential equations in R: Package deSolve. *Journal of Statistical Software*, 33(9):1–25, 2010.

[36] Arie Tamir. Line search techniques based on interpolating polynomials using function values only. *Management Science*, 22(5):576–586, 1976.

[37] Gerald Teschl. *Ordinary differential equations and dynamical systems*, volume 140. American Mathematical Soc., 2012.

[38] Keller Vandebogart. Method of quadratic interpolation. `http://people.math.sc.edu/kellerlv/Quadratic_Interpolation.pdf`, September 2017. [Online; accessed 13-September-2019].

[39] Jet Wimp. *Sequence transformations and their applications*. Elsevier, 1981.

[40] Stephen Wright. Optimization for data analysis. In Michael W. Mahoney, John C. Duchi, and John C. Duchi, editors, *The Mathematics of Data*, chapter 2, pages 49–98. American Mathematical Society and IAS/Park City Mathematics Institute and Society for Industrial and Applied Mathematics, 2018.

[41] Tong Tong Wu, Kenneth Lange, et al. The MM alternative to EM. *Statistical Science*, 25(4):492–505, 2010.