Dissertation Second Draft

190005680

2023-03-02

Contents

1 Estimating Functions From Observed Data

1.1 How is the Observed Data Related To The Functional Data?

If a statistician wishes to work with functional data they must estimate these functions from their observed data. In order to do this a relationship must be established between the observed data and the functional data. Within FDA, the following relationship is assumed between the observed data and the function underlying it.

$$Y_i = f(t_i) + \epsilon_i \tag{1}$$

where Y_i is the observed value of the variable of interest at time t_i, $f(t_i)$ is the value of the function underlying the data evaluated at time t_i and $\epsilon_i \sim N(0, \sigma^2)$ is some random error in observation. It is assumed that these errors are independent identically distributed random variables (σ is a constant).

Within this dissertation, it will be assumed that the error in each observation are independent and identically distributed normal random variables with constant variance. There is some literature which attempts to relax these assumptions about the errors in observation (Liu (2016)).

An example is given in Figure 1 of some data generated from an underlying function which fulfills the relationship given in Equation (1).

1.2 Introductory Example: UK Alcohol Mortality Rates

In order to easily introduce the stages of analysis and analytical techniques used in FDA I will use a simple example. In this example, I wish to use FDA to attempt to answer two questions. These questions are:

1. Is there a difference in the number of alcohol related deaths between countries in the United Kingdom (UK)?
2. Is there a correlation between the unemployment rate and the number of alcohol related deaths in a country?

The two datasets I will use to answer these questions are taken from the Office of National Statistics (ONS). The first dataset I will use (Breen and Butt (2022)) contains the "age-standardised alcohol-specific death rates per 100,000 people, in UK constituent countries, for deaths registered between 2001 and 2021". These "age-standardized death rates" are calculated according to the guidance laid out in European Standard Population

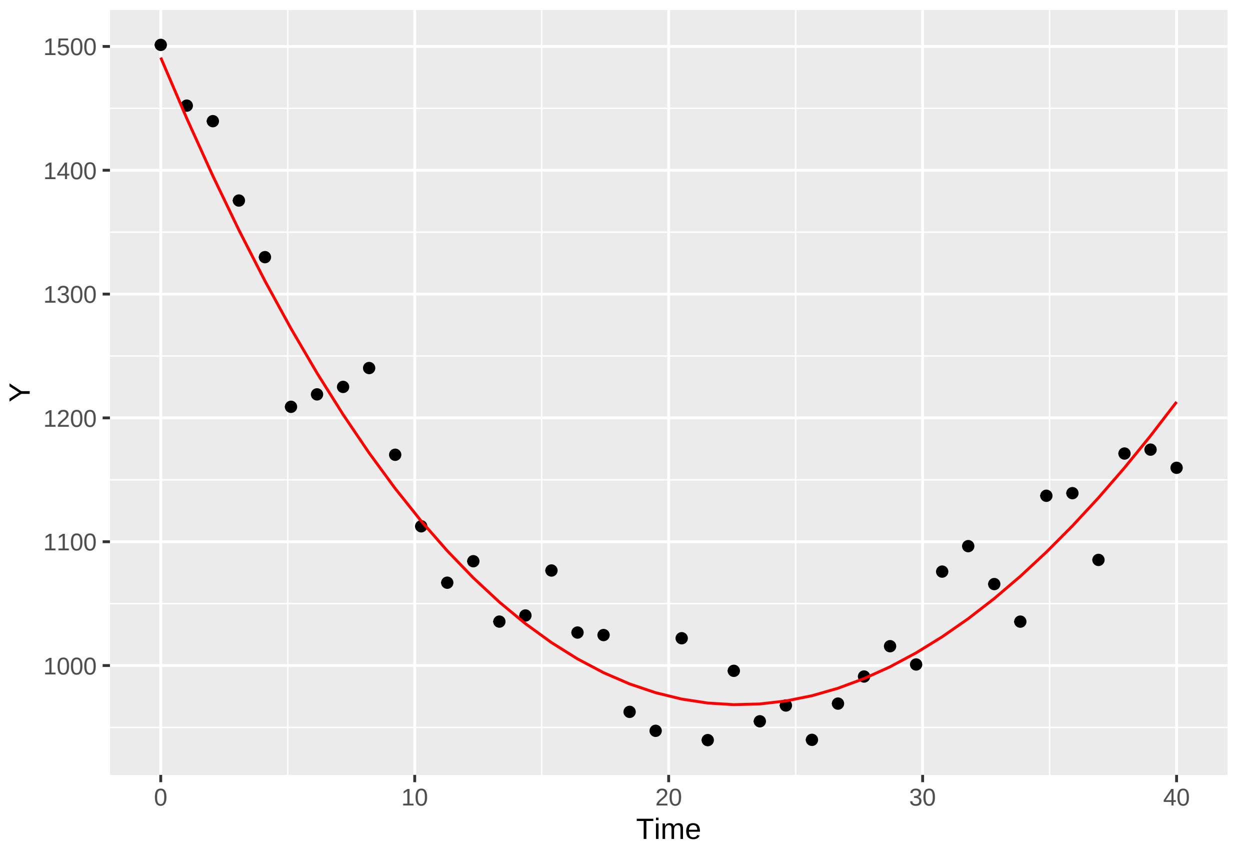


Figure 1: An Example of Data Simulated From a Function According to the Assumptions of FDA. The Red Line Shows the Function Y(t) which Generates the Data ($\sigma^2 = 30$)

2013 (ESP 2013) (Pace M (2013)) and adjust the death rates of a country to avoid over representing larger age categories within the population. An initial plot of this data can be seen in Figure 2 below.

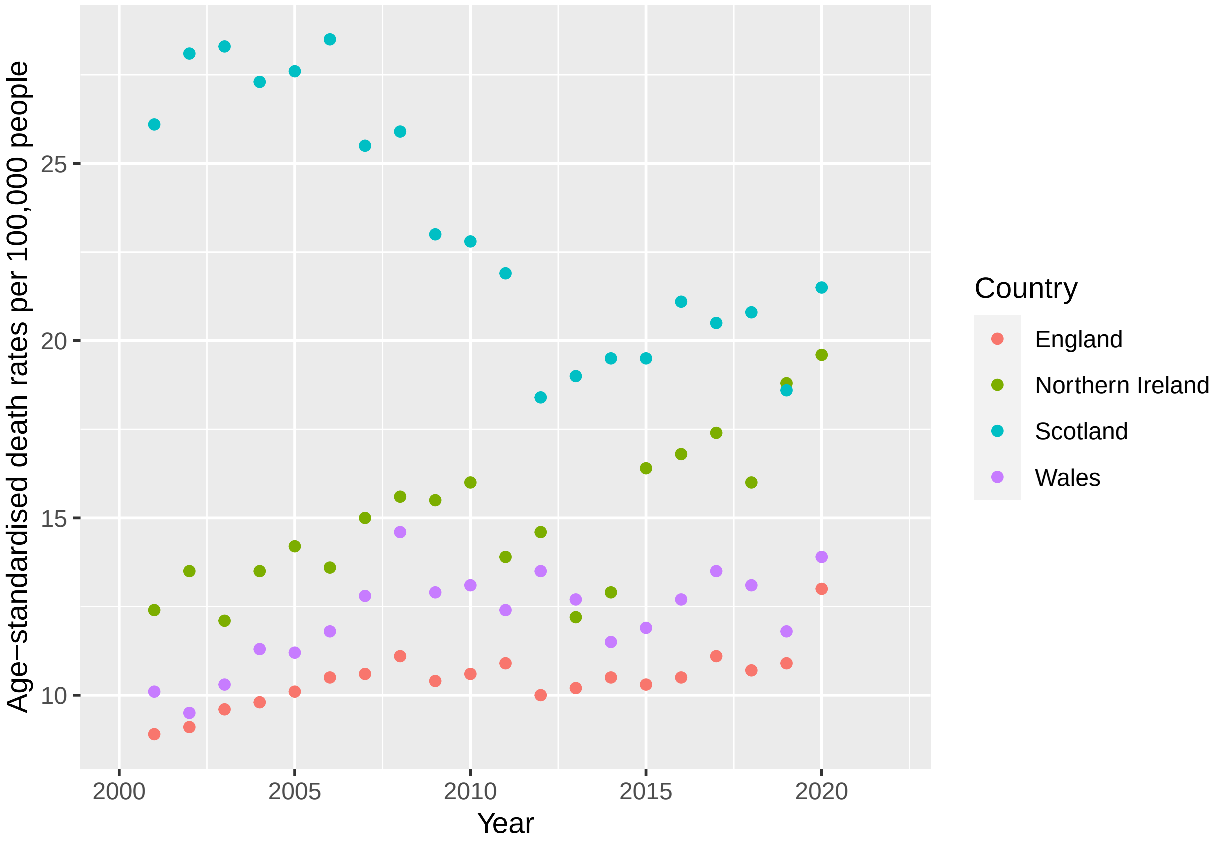


Figure 2: Age Standardised Alcohol-Specific Death Rates For Each Country within the UK from 2001 to 2022

The second dataset I will use is taken from the Labour Force Survey (LFS) ("Labour Market Overview, UK: February 2023" (2023)). This dataset gives the unemployment rate, as estimated using the methodologies laid out within the LFS, as the percentage of the population of a country which is unemployed. This dataset gives the unemployment rate for 1992 to 2022. I will only be using the data from the years 2001 to 2022 as that is the range of years that the alcohol related deaths data covers. An initial plot of the unemployment data can be seen in Figure 3 below.

In order to use FDA methods to analyse this data I will have to estimate a function which underlies the alcohol mortality rate and another function which underlies for the unemployment rate for each country and use these to answer the questions presented. The analytical form of these functions is not known in this example and, therefore, they must be estimated from the observed data. These functions will be estimated using linear combinations of basis functions.

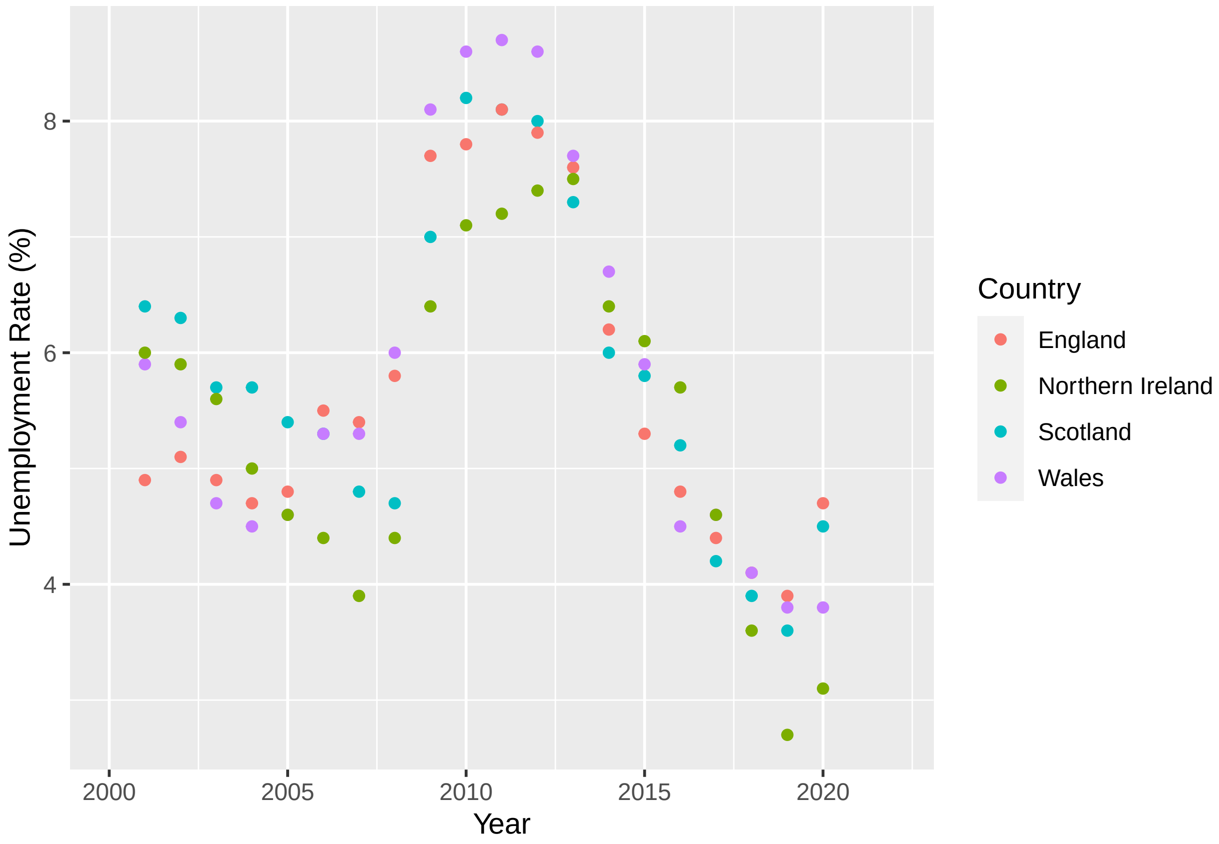


Figure 3: Unemployment Rates for Each Country in The UK From 2001 to 2022

1.3 Basis Functions

The functions which underlie the data we observe are not known when we receive our data. We must estimate them from our observed data. In order to estimate these functions basis functions must be used.

To understand what basis functions are let us first consider a set of three functions Ψ defined below.

$$\Psi = \{1, x, x^2\} \tag{2}$$

The span of these functions, that being the set of functions produced by taking all linear combinations of these 3 functions, is the set of all polynomials of degree 2 (or less) with coefficients in the real numbers, $\mathcal{P}_2(\mathbb{R})$. This means that every function within $\mathcal{P}_2(\mathbb{R})$ can be written as a linear combination of these 3 functions. A general function $F \in \mathcal{P}_2(\mathbb{R})$ can be written as shown below.

$$F(x) = a \times 1 + b \times x + c \times x^2 \tag{3}$$

for $a, b, c \in \mathbb{R}$.

The span of Ψ being P_2 makes Ψ a basis of P_2. Each of the functions contained within Ψ is called a basis function.

We wish to find a basis which spans a set of functions that the functions we are trying to estimate are contained in. If we can find such a basis then we can estimate our functions by taking a general linear combination of the functions in this basis and estimating the coefficients which provide the best fit based on the observed data.

An arbitary function f can be estimated by a linear combination of a set of basis functions which form a basis Ω for the functional space F with $f \in F$ as shown below. This linear combnation of basis functions which estimates the function is known as the basis expansion of the function.

$$f(t) \approx \sum_{k=1}^{K} c_k \phi_k(t) = \boldsymbol{c'}\boldsymbol{\phi}(t) \tag{4}$$

c_k is the coefficient of the k_{th} basis function $\phi_k(t) \in \Omega$ and there are K functions in Ω. $\boldsymbol{\phi}(t)$ is a K-dimensional vector of all of the basis functions in Ω evaluated at time t and $\boldsymbol{c}$ is a K-dimensional vector of the coefficients of these basis functions. This basis expansion of f is an estimation, not an equality, as the coefficients of the basis functions must be estimated through smoothing techniques (discussed in Smoothing Techniques section of this dissertation). It is not possible to obtain the actual coefficients without knowledge of the analytical form of f.

There is an infinite number of bases which can be used to estimate our functions. Therefore, we must assume some things about the functions we wish to estimate in order to choose the basis that we wish to use. There are a few standard sets of basis functions which can be used to estimate functions from data with various different properties. One such example is the Fourier basis, a set of basis functions, defined by their period T, which can be linearly combined to create any T-periodic function. The functions which I am trying to estimate in the alcohol mortality rates example appear to be non-periodic in nature, as shown by their being no obvious repeated patterns in the data over time, and so I must use a set of basis functions which can estimate non-periodic functions. There are several bases which can do this but I will be estimating my functions using a B-Spline basis.

1.4 B-Spline Basis Functions

A B-Spline basis is a set of piece-wise polynomial functions defined by their domain, $[a, b] \subset \mathbb{R}$, their knots/break points and their order/degree. The span of a set of degree n B-Spline basis functions with domain $[a, b]$ is the set of all spline functions of degree n (or less) with domain $[a, b]$. A spline function of degree n is a piecewise continuous curve made of polynomial segments all of degree n or less.

B-Spline basis functions can be linearly combined to estimate functions which are non-periodic in nature. To do this a particular set of B-Spline functions must be chosen to estimate the functions of interest. This means choosing the domain, knots/break points and degree of the B-Spline basis functions to be used.

Choosing the domain of the basis functions is a relatively trivial task as it is often obvious from the data observed. Typically we do not want to define our functions to have a domain that is outside of that of the observed data as any values of the function estimated outside of the domain of the observed data will be based not on an inference but on an extrapolation from the data. Therefore, in almost all cases the domain of the B-Spline functions is chosen to be that of the observed data. In the case of the alcohol mortality rates example, the first year that the death rate is measured in is 2001 and the final year is 2020 so the domain of the B-Spline functions I will be using is $[2001, 2020]$.

Choosing the degree/order of our basis functions is a slightly more challenging task. The degree of a polynomial is the exponent of the highest order term in the polynomial e.g. if $f(x) = x^{24} + x^3 + 6$ then the degree of $f(x)$ is 24. The order of a polynomial is one more than the degree of the polynomial e.g. the order of $f(x)$ is $24 + 1 = 25$. The function we estimate from the observed data will never match the function that underlies the data exactly as we must estimate the coefficients of the basis functions using smoothing. We only need our function to be estimated well enough that it provides the information we need from it. For example, if we wish to use the functions 1st derivative we would need our estimate of the function to be smooth when differentiated once. Therefore, our choice of degree of the B-Spline functions is dependent on what we wish to do with them. It is good practice in these type of situations to fix the order of B-splines used to be two higher than than the highest order of derivative to be used in the analysis (Ramsay J. O. (2009)). In the alcohol mortality rates example, the derivatives of the functions do not need to be used to answer either of the two questions posed. In the case where no derivatives are to be used often polynomial splines will not suffice in estimating the function and cubic splines should be used in this case, e.g. splines of order 4. I will, therefore, use order 4 basis splines.

The final thing to decide is where to place the knots/break points defining the splines on the domain $[a, b]$. The break points of a B-Spline basis are the points where each of the B-Spline functions in the basis meet. The knots are related to break points such that each break point has at least one knot situated at it. The number of knots placed at a break point indicates the number of derivatives of the function that should appear smooth across that break point with k knots at a point indicating the first $n - k - 1$ derivatives should join smoothly across that point. In all of the examples of FDA I will share within this dissertation, I will only place 1 knot at each break point as I will not be using the derivatives of functions. The choice of where break points go can be an incredibly simple task as the optimum break point placement is placing a break point on each point on the domain where there is an observation within the original data. However, this is often not computationally feasible. This is because the number of knots that a basis is defined by also determines the number of basis functions within that basis. The relation between the two is given below.

$$number\ of\ basis\ functions\ =\ order\ +\ number\ of\ interior\ knots \tag{5}$$

Lets consider a situation in which we wished to estimate splines of order 4, the original dataset used consisted of 4000 data points and we wanted to place a knot at each of them. We would then have $3998 + 4 = 4002$ basis functions within our basis which would mean we would have to estimate 4002 coefficients, one for each these basis functions when estimating our function which underlies the data. The question of optimal knot placement for these types of situations will be revisited in the final example of this dissertation. In the alcohol mortality rates example, I will place a break point at each observation as there are only 20 observations within the data for each country.

I have now chosen chosen the domain, order and break point placements of the B-Spline basis I wish to use to estimate my functions for each country within the alcohol mortality rates example. This means I can now create this B-Spline basis within R. I will do this using the `create.bspline.basis` function within the `fda` package in R (Ramsay, Graves, and Hooker (2022)). To create a B-Spline basis of functions with domain $[2001, 2022]$, order 4 and a break point at each year I used the following function call.

```
basis_mortality_rates <- create.bspline.basis(rangeval=range(Years),breaks=Years,norder=4)
```

I now have the basis I will be using to estimate my functions within the alcohol mortality rates example. Referring back to Equation (4), I have my basis functions but I am still yet to find the coefficients of the linear combination of these basis functions which provides the best estimate of the underlying functions. To estimate these coefficients a process called smoothing must be used.

1.5 Smoothing Techniques

1.5.1 What is Smoothing?

1.5.2 Smoothing using a Roughness Penalty

The generally preferred method of finding the coefficients for a functional data object is smoothing using a roughness penalty. This method of smoothing seeks to minimise the sum of squared errors of the fitted function while also penalising the roughness of the function.

Lets first consider the sum of squared errors (SSE) of a function estimated from observed data via smoothing. The SSE of a function gives a measure of the difference between the function and the observed data without regard to the direction of the difference. The SSE of a functional fit can be derived from Equation (1). This derivation is given below.

$$Y_i = f(t_i) + \epsilon_i \tag{6}$$

$$\Rightarrow \epsilon_i = Y_i - f(t_i) \tag{7}$$

$$SSE(f) = \sum_{j=1}^{n} [\epsilon_i]^2 \tag{8}$$

$$\Rightarrow SSE(f) = \sum_{j=1}^{n} [Y_j - f(t_j)]^2 \tag{9}$$

An SSE of 0 implies that the function passes through all of the observed data points. An example is given below in Figure 4 of a function with SSE 0. While an SSE of 0 can initially sound like the function fits perfectly to the data we must bear in mind that one of the main assumptions of FDA is that there is some observation error in the observed data points, as stated in Equation (1). This is why the function with SSE 0, given by the dashed line, in Figure 4 estimates the actual function, given by the red line, so badly. It does not take into account this inherent error in observation.

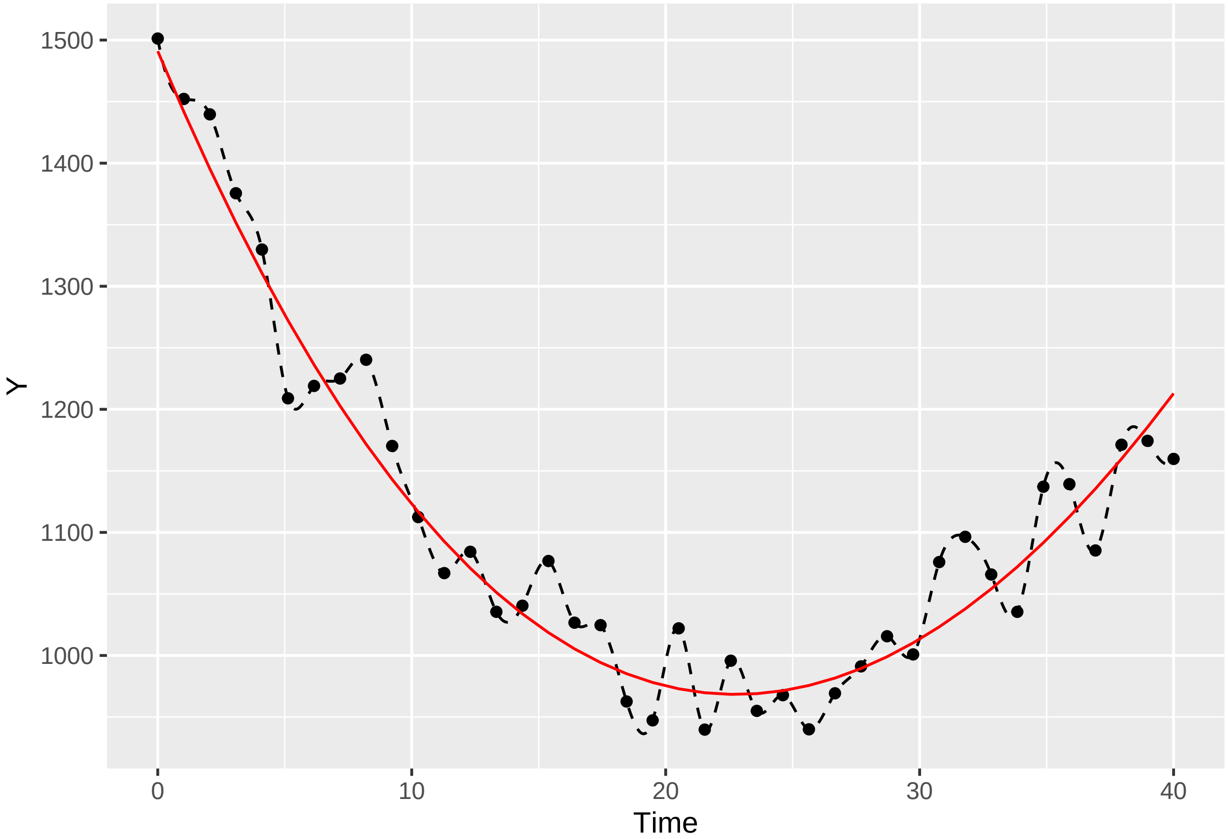
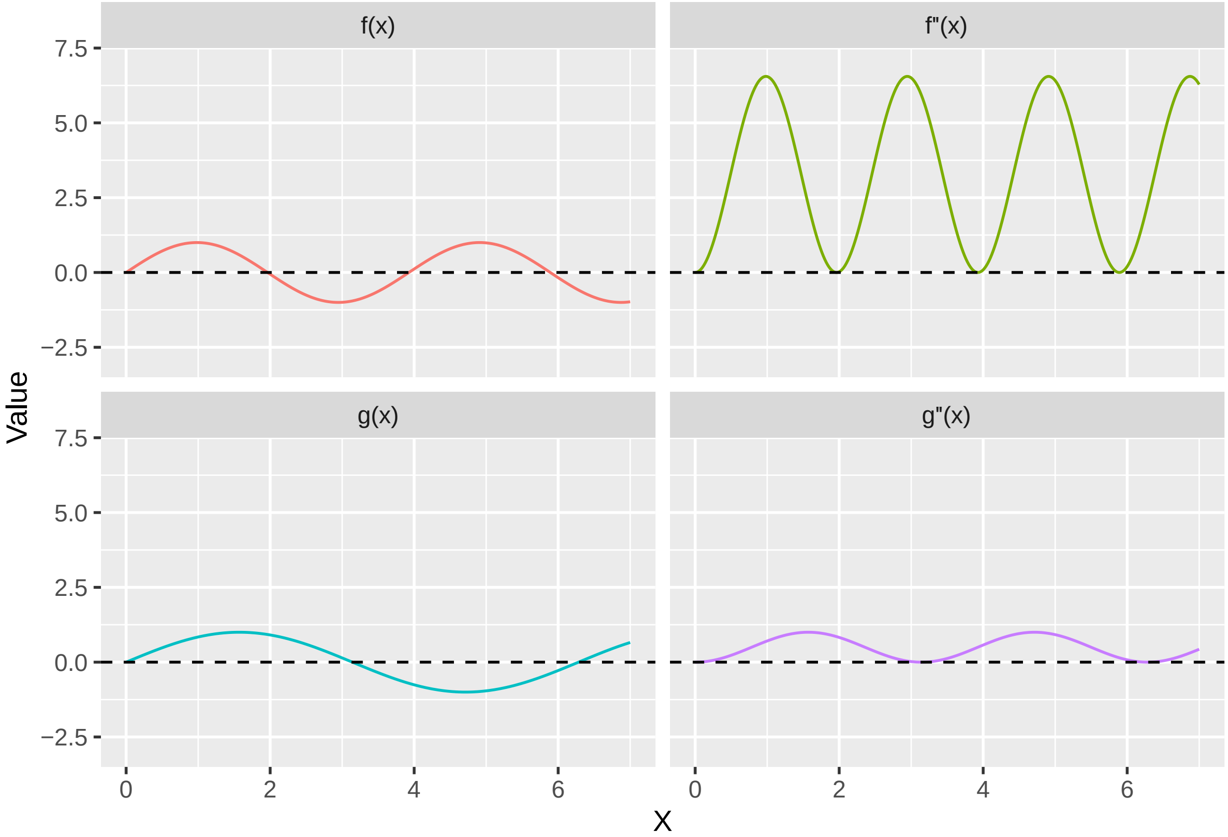


Figure 4: Estimate of Function with SSE=0 (Dashed Line) With True Function (Red Line) and the Data Used to Estimate It Overlayed

The "roughness" of a function is not a clearly defined concept and is often situational. In this case, we define roughness to be synonymous with the wigglyness of the function e.g. how steep the slopes of the function are. The steepness of the slopes of a function can be examined by looking at derivatives of the function. We do not want our function to interpolate the observed data. We want it to have a relatively smooth fit such that it allows for some error within the observations. Therefore, we must penalise our functions roughness when choosing the coefficients.

Within smoothing, many different derivatives of the function can be used as a measure of "roughness".

Within this dissertation I will be using the second derivative of the function. The area under the squared second derivative of a function is a good measure of how rough a function is. A straight line would have a squared second derivative of 0 for all values of x. Therefore, the area under the squared second derivative is effectively a measure of how far away a function is from being a straight line. Shown below are two functions $f(x)$ and $g(x)$, their second derivatives and their second derivatives squared. $f(x)$ is clearly "rougher" than $g(x)$. We can clearly see that the area under $(f''(x))^2$ is larger than that of $(g''(x))^2$.



We wish to find the coefficients which minimise the SSE of our function while also allowing for some error in observtaions. Let us consider the objective function shown below which linearly combines the SSE with our measure of roughness. The function we are trying to find is also rewritten using it's basis expansion as a reminder that we are trying to find the coefficients which minimise this objective function.

$$F(f) = \sum_j [Y_j - f(t_j)]^2 + \lambda \int_{t_0}^{t_1} [f''(t)]^2 dt \tag{10}$$

$$\Rightarrow F(\boldsymbol{c}) = \sum_j [Y_j - \boldsymbol{c'}\boldsymbol{\phi}(t_j)]^2 + \lambda \boldsymbol{c'}[\int_{t_0}^{t_1} \boldsymbol{\phi}''(t)\boldsymbol{\phi}''(t)^T dt]\boldsymbol{c} \tag{11}$$

The parameter λ featured in Equation (11) is known as the smoothing parameter of the objective function F. λ can be thought of as determining how smooth we want out function to be. If $\lambda = 0$ then the objective function becomes the SSE, therefore, providing no smoothing. As $\lambda \to \infty$ the roughness penalty approaches 0 implying that a straight line is fit to the data.

Assuming a value of lambda is chosen which gives the best weighting of SSE and roughness penalty, a set of coefficients which minimises this objective function would provide the optimal fit of the function to the data. The function would be as close to the observed data as possible while also not overfitting the original data by interpolating them.

As stated above, the ideal lambda must be found which gives the best balance between minimising the SSE and minimising the roughness of the function. The ideal λ, e.g. the ideal weighting of the roughness penalty and the SSE, is found using the generalized cross-validation (GCV) criteria as developed by Craven P. and G. Wahba (Craven and Wahba (1978)). This criteria is given below for an arbitrary function f and measures how well a function fits the observed data. The lower a functions GCV criterion value the better the function estimates the original underlying function.

$$GCV(f) = \left(\frac{n}{n - df(f)}\right)\left(\frac{SSE(f)}{n - df(f)}\right) \tag{12}$$

The degrees of freedom of f are as defined in page 65 of Ramsay J. O. (2009).

In order to find the value of the smoothing parameter which minimises this equation many functions are fit. Each function is fit by finding the coefficients which minimise the objective function for a different value fo the smoothing parameter. The GCV scores for these functions are then compared to see which value of the smoothing parameter provides the function with the best fit to the data. The value of the smoothing parameter that gives the lowest GCV is taken as the ideal smoothing parameter to be used in fitting the final estimate of the function.

1.6 Implementing Smoothing using a Roughness Penalty In R

We now know how to find the optimum choice of smoothing parameter. This means we now have all the components needed to conduct smoothing using a roughness penalty. Within the alcohol mortality rates example, I wish to estimate the functions which underlie the age standardized alcohol mortality rates for each country. I wish to estimate these function using the B-Splines basis functions I previously defined but I do not know the coefficients of these functions in the basis expansion. I will find these coefficients by finding the coefficients which minimise the objective function defined above. The code below finds these coefficients and fits the resulting basis expansion estimations of the function for the 4 countries in the data set. The value for the smoothing parameter has been set to 0.01 in this example and this smoothing parameter is the same across for all objective functions used to find the coefficients for the 4 countries.

```
observation_matrix <- data.matrix(mortality_rates_wide %>% dplyr::select(-c(Year)))

minimum_lambda <- 0.01

fd_par_obj <- fdPar(basis_mortality_rates,2,minimum_lambda)

smooth_basis_output <- smooth.basis(Years,observation_matrix,fd_par_obj)

sample_of_functions <- smooth_basis_output$fd

gcv <- smooth_basis_output$gcv

sample_of_functions$fdnames$time <- "Years"

sample_of_functions$fdnames$values <- "Mortality Rate"
```

The `fdPar` object defined above supplies the basis functions, smoothing parameter and order of derivative to be used to the function `smooth.basis` which fits the functions. In the example above, I supplied the number 2 as the second argument of the function to indicate that the second derivative of the function is to be used to define the roughness penalty however any positive integer can be supplied (or an `Lfd` object which defines a linear differential operator). The final two lines of the above give names to the domain of the functions estimated and the range of the functions estimated. In this case, our domain is measured in years and the output of our functions, e.g. the range, is the mortality rate of the country. The `gcv` variable, which is extracted from the output of the `smooth.basis` function, gives the GCV score for the function fit. When multiple functions are to be fit using the same smoothing parameter we wish to find the value of the smoothing parameter which minimises the sum of the functions GCVs. The code below gives a function I have written. This function takes as an input the log (base 10) of a smoothing parameter, as well as the basis functions, observations matrix and the order of derivative, to be used in the smoothing process and returns the sum of the GCV scores of the functions. This function can be used to perform a grid search over different values of the smoothing parameter to find which smoothing parameter minimises the sum of the GCVs. The log of the smoothing parameter is used as the GCV score of the functions tends to change only slightly for small changes in the smoothing parameter.

```r
GCV_func <- function(log_lambda,basis,observations,time_points,penalty){

  lambda <- 10^log_lambda

  fd_par_obj <- fdPar(basis,penalty,lambda)

  smoothbasisobj <- smooth.basis(time_points,observations,fd_par_obj)

  return(sum(smoothbasisobj$gcv))
}
```

The following code performs a grid search which the optimal lambda for the objective functions for the estimation alcohol mortality rates functions. The function `optimise` performs a grid search across values of the log of the smoothing parameter ranging from 1 to 10 as given in the `lower` and `upper` arguments. The final two lines of code extract the optimal log lambda and then finds the minimum lambda from this.

```r
optimised_function <- optimise(GCV_func,lower=0,upper=10,
                               basis=basis_mortality_rates,observations=observation_matrix,
                               time_points=Years,penalty=2)

minimum_log_lambda <- optimised_function$minimum

minimum_lambda <- 10^minimum_log_lambda
```

I now have the lambda which gives the best weighting between the SSE and the roughness of the function. I can now find the coefficients using the model fitting procedure laid out above. I now have my sample of functions that give the age standardized mortality rate of each of the 4 countries within the UK. A plot of these functions is given below. How these plots were created as well as the full code for the fitting procedure for these mortality rate functions can be found in the Appendix Section of this dissertation.

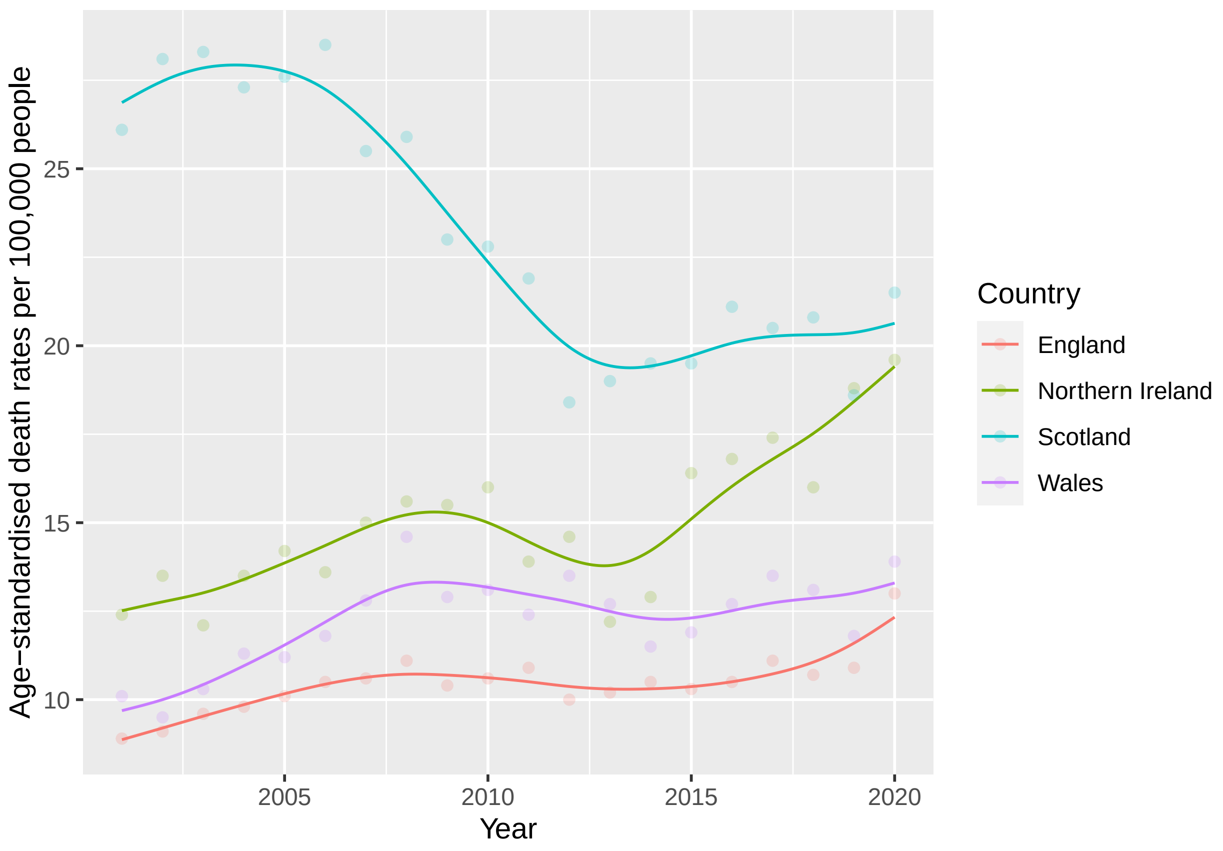


Figure 5: The Estimated Functions of Age Standardised Mortality Rate For Each Country in the UK (Dots Represent the Original Data)

Now that I have found estimates of the functions which underlie the alcohol mortality rates I can use functional data analysis to analyse their trends. However, to do this I first need to produce some functions which summarise the main characteristics of the functions.

2 Summary Functions Of Functional Data

I now have a sample of four functions, one for each country, for my alcohol mortality rates example. Before I conduct any formal analysis of this sample I want to summarise the sample in order to visually identify the main trends in alcohol related deaths. In typical data analysis of time series, in which a typical sample would be a set of vectors of observations of a quantity over time, a statistician may look at the mean vector of the observations, which gives the average value of the sample at each observed time, as a summary measure. The mean function of a functional sample is analogous to this in FDA. This function gives the mean value of all of the functions in the sample at a time t. The exact form of this function for a sample of function $\{x_1(t), ..., x_n(t)\}$ is given below.

$$\bar{x}(t) = N^{-1} \sum_{i}^{n} x_i(t) \tag{13}$$

The mean function of my sample of functions for the alcohol mortality rates example, as well as the original sample, is shown in Figure 6 below.

The mean function of the alcohol mortality rates data shows that there was a slight increase in the number of alcohol related deaths across the 4 countries between 2001 and 2008. The death rates then fell between 2008 and 2013 and then from there started to increase again until 2020.

When working with a sample of time series data, a statistician would typically also want to look at the standard deviation in values. This shows how varied the sample is at each time point. The standard deviation function of a sample of functions is analogous to this in FDA. This function gives the standard deviation in the sample of functions at a time t. The exact form of this function for a sample of function $\{x_1(t), ..., x_n(t)\}$ is given below.

$$\sigma(t) = (N-1)^{-1} \sum_{i}^{n} [x_i(t) - \bar{x}(t)] \tag{14}$$

The standard deviation function of my sample of functions for the alcohol mortality rates example is shown in Figure 7 below. The units of the standard deviation are the same as those of the observation, in this case

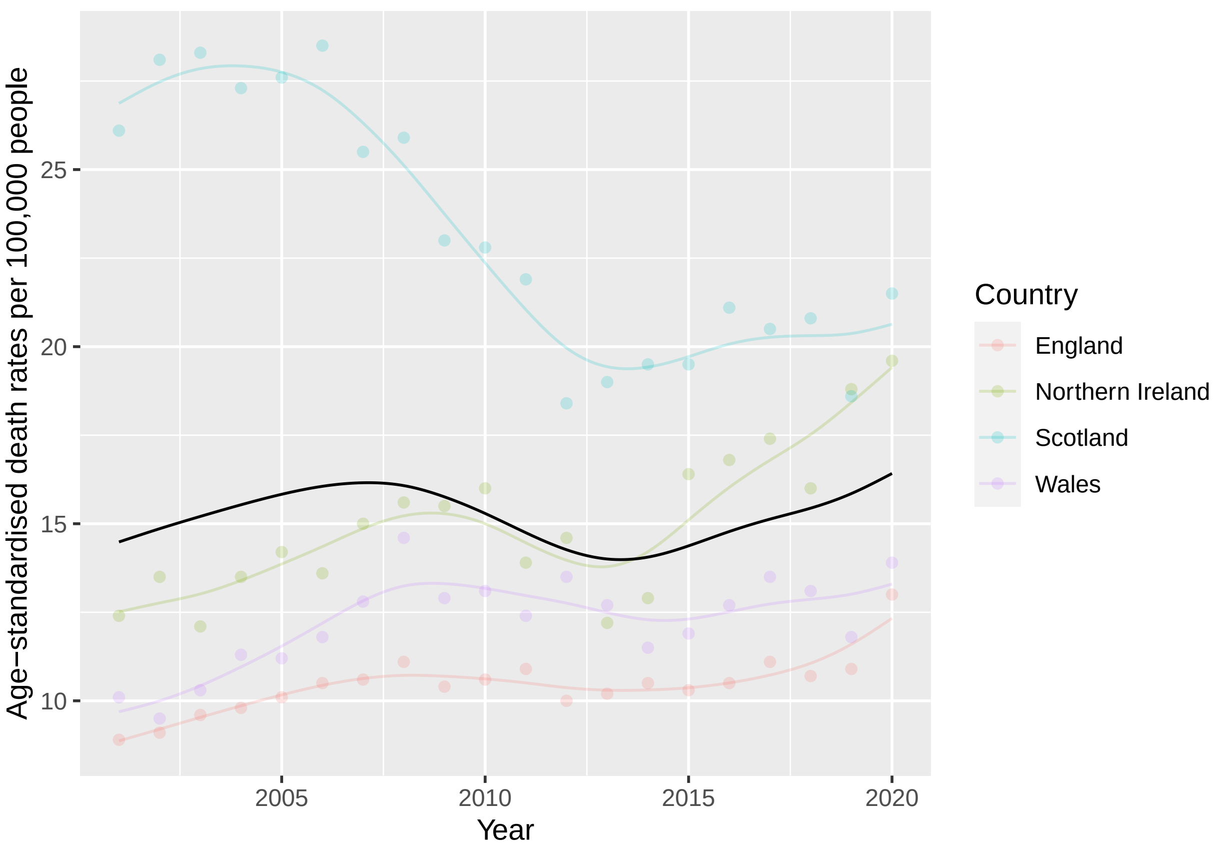


Figure 6: The mean function of the sample of alcohol mortality rates functions (shown in black) with original sample of functions overlayed

the units are the age-standardized death rate per 100,000 people.

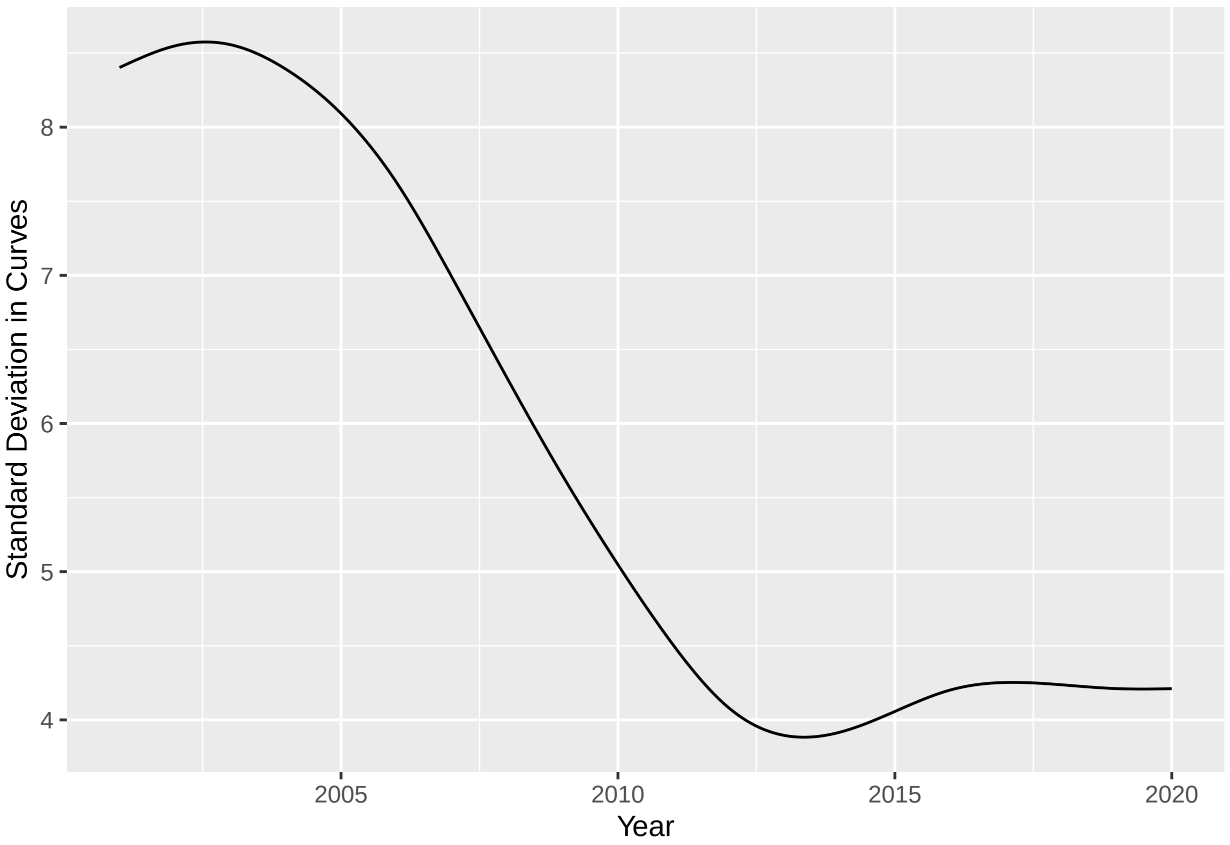


Figure 7: The standard deviation function of the sample of alcohol mortality rates functions

This standard deviation function shows that there was a large difference in the number of alcohol related deaths between the 4 countries in 2001. This difference between the countries dramatically decreased between 2002 and 2013. However the difference in death rates increased again between 2015 and 2017 until it plateaued in 2017.

3 Appendix

References

Breen, Paul, and Asim Butt. 2022. "Alcohol-Specific Deaths in the UK: Registered in 2021." *Office of National Statistics (ONS) Statistical Bulletin (ONS Website)*. https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/deaths/bulletin/alcoholspecificdeathsintheuk/2021registrations.

Craven, Peter, and Grace Wahba. 1978. "Smoothing Noisy Data with Spline Functions." *Numerische Mathematik*. https://doi.org/10.1007/BF01404567.

"Labour Market Overview, UK: February 2023." 2023. *Office of National Statistics (ONS) Statistical Bulletin (ONS Website)*. https://www.ons.gov.uk/employmentandlabourmarket/peopleinwork/employmentandemployeetypes/bulletins/uklabourmarket/february2023.

Liu, Haiyan. 2016. "Dissertation Title." PhD thesis, Universität Konstanz. https://kops.uni-konstanz.de/server/api/core/bitstreams/978acb23-b57e-43d2-8e3d-3f7674df1bb4/content.

Pace M, Glickman M, Lanzieri G. 2013. "Revision of the European Standard Population." *Methodologies & Working Papers*. https://doi.org/10.2785/11470.

Ramsay J. O., Graves S, Hooker G. 2009. *Functional Data Analysis with r and MATLAB*. 233 Spring Street, New York: Springer Science; Business Media.

Ramsay, J. O., Spencer Graves, and Giles Hooker. 2022. *Fda: Functional Data Analysis*. https://CRAN.R-project.org/package=fda.