Spline Regression

MAST90083 Computational Statistics and Data Mining

Karim Seghouane

School of Mathematics & Statistics
The University of Melbourne

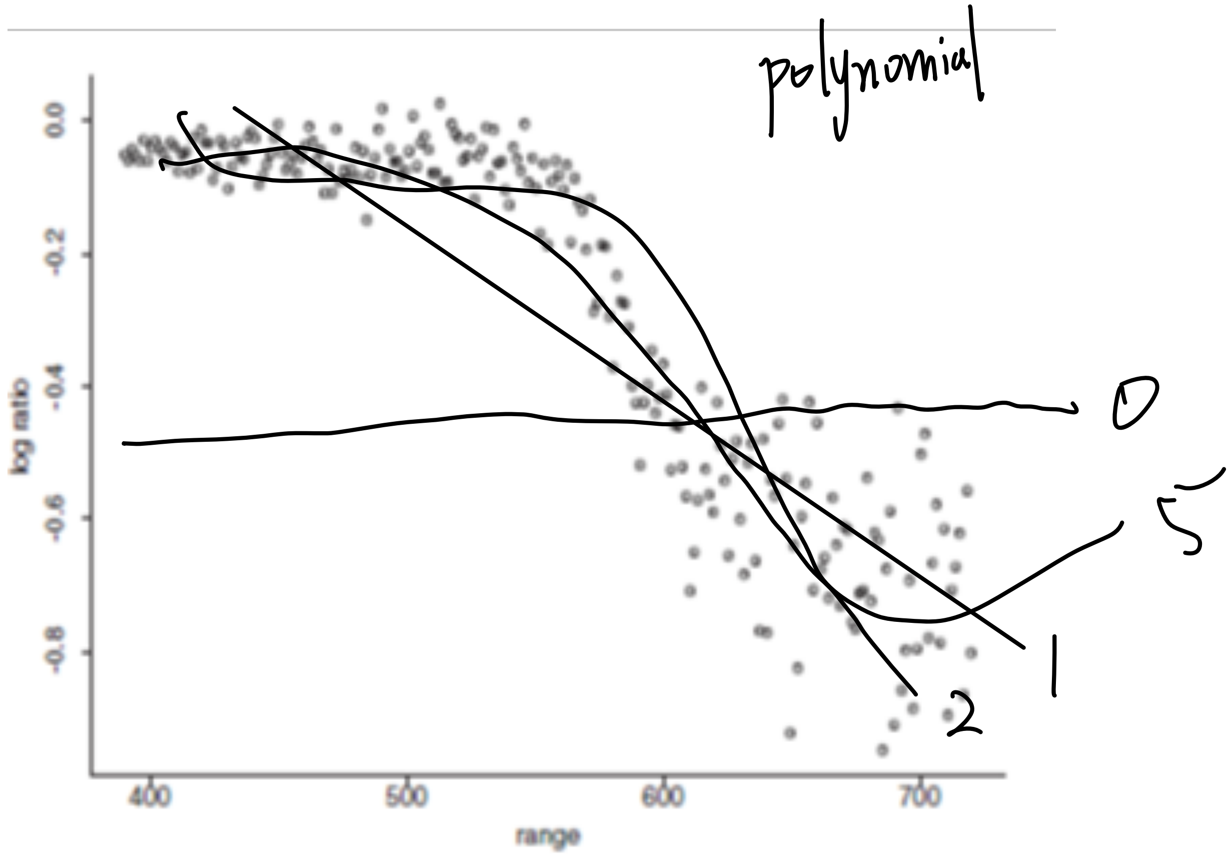
Outline

Introduction

- Some data sets are hard or impossible to model using traditional parametric techniques
- Many data sets also involve nonlinear effects that are difficult to model parametrically
- There is a need for flexible techniques to handle complicated nonlinear relationships
- Here we look at some ways of freeing oneself of the restrictions of parametric regression models

Introduction

The interest is the discovery of the underlying trend in the observed data which are treated as a collection of points on the plane



Introduction

- Alternatively, we could think of the vertical axis as a realization of a random variable y conditional on the variable x
- The underlying trend would then be a function

mean of y of particular points

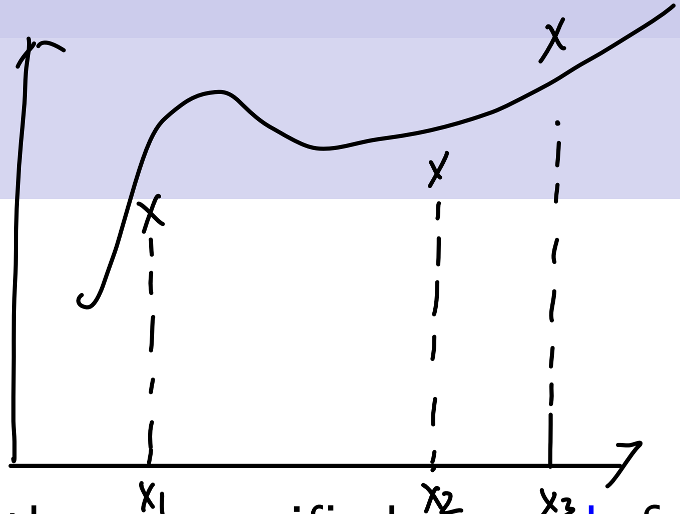
$$f(x) = E(y|x)$$

- This can also be written as

$$y_i = f(x_i) + \epsilon_i, \quad E(\epsilon_i) = 0$$

- and the problem is referred as nonparametric regression

Introduction



$f(x)=E(y|x)$

$y_{ij}=f(x_i)+\xi_{ij}$

$j=1,\dots,m$

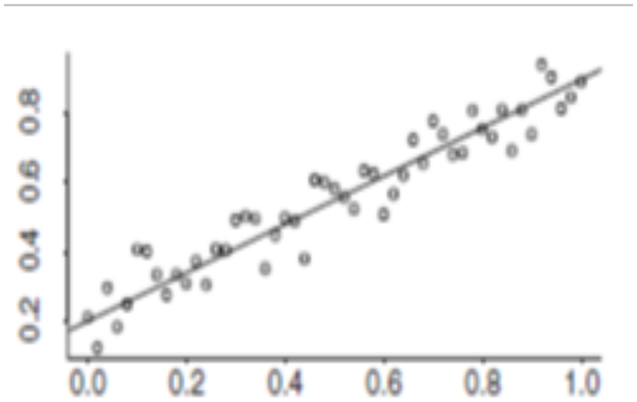
$f(x_i)=\frac{1}{m}\sum_{j=1}^{m} y_{ij}$

- **Aim** Estimate the unspecified smooth function from the pairs (x_i, y_i), $i = 1, \dots, n$.
- x here will be considered univariate
- There are several available methods, here we focus first on penalized splines
- It is an extension of linear regression modeling

Motivation

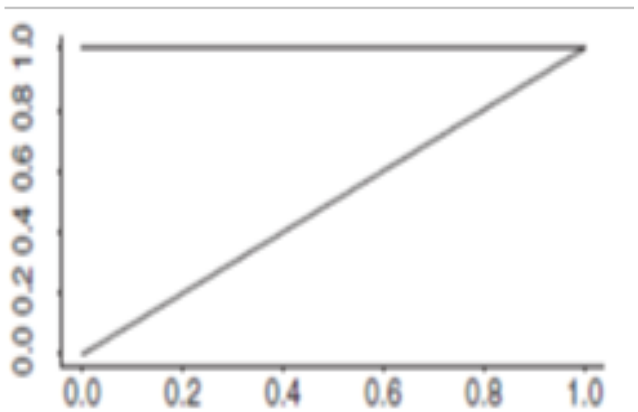
- Let's start with the straight line regression model

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$



Motivation

- The corresponding basis for this model are the functions: 1 and x



- The model is a linear combination of these functions which is the reason for use of the world basis

Motivation

- The basis functions correspond to the columns of X for fitting the regression

$$X = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}$$

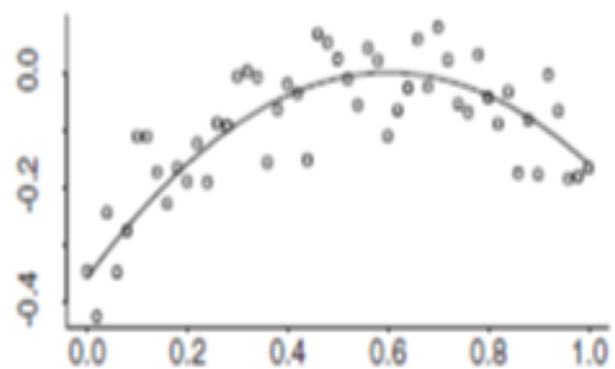
- The vector of fitted values

$$\hat{\mathbf{y}} = X\left(X^\top X\right)^{-1} X^\top \mathbf{y} = H\mathbf{y}$$

Motivation

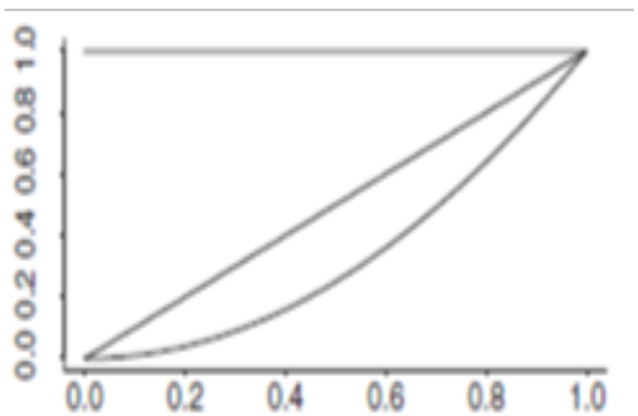
- The quadratic model is a simple extension of the linear model

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \epsilon_i$$



Motivation

- There is an extra basis function x^2 corresponding to the addition of the $\beta_2 x_i^2$ term to the model



- The quadratic model is an example of how the simple linear model might be extended to handle nonlinear structure

Motivation

- The basis functions correspond to the columns of X for fitting the regression in the case of a quadratic model is given by

$$X = \begin{bmatrix} 1 & x_1 & x_1^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix}$$

- The vector of fitted values

$$\hat{\mathbf{y}} = X\left(X^\top X\right)^{-1} X^\top \mathbf{y} = H\mathbf{y}$$

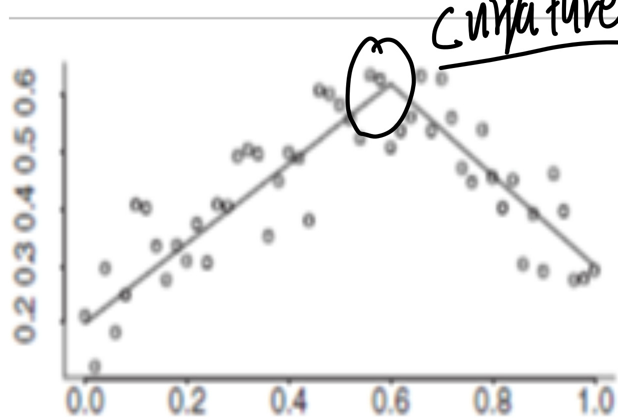
Spline basis function

成大事者不纠结 成大事者不纠结 成大事者不纠结 成大事者不纠结

成大事者不纠结

- We know look at how the model can be extended to accommodate a different type of nonlinear structure

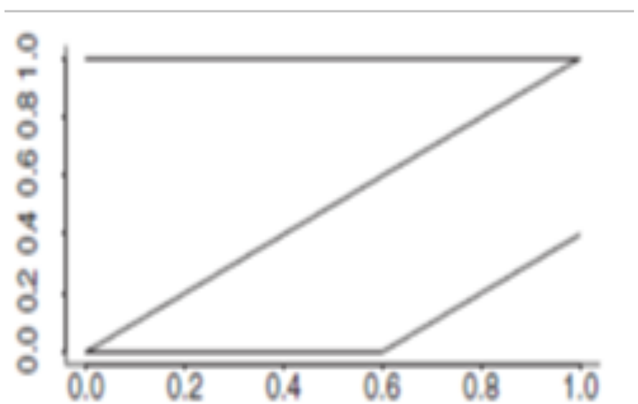
curvature



- Broken line model: it consists of two differently sloped lines that join together

Spline basis function

- Broken line: A linear combination of three basis functions



- where we have $(x - 0.6)_+$ with $V_+ \begin{cases} V & V > 0 \\ 0 & V \leq 0 \end{cases}$

$$u_+ = \begin{cases} u & u > 0 \\ 0 & u \leq 0 \end{cases}$$

Spline basis function

- Broken line model is

$$y_i = \beta_0 + \beta_1 x_i + \beta_{11}\left(x_i - 0.6\right)_+ + \epsilon_i$$

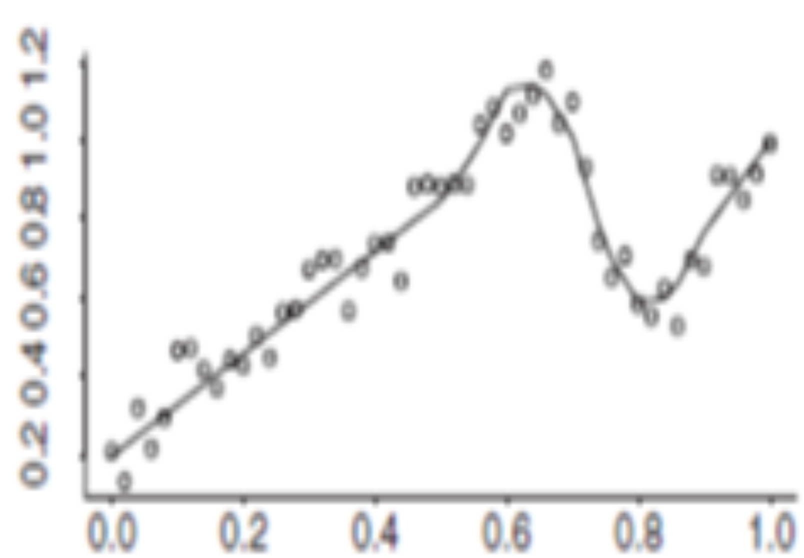
- which can be fit using the least square estimator with

$$X = \begin{array}{cccc} & & & \text{new term} \\ & & & \\ \end{array}$$

$$X = \begin{bmatrix} 1 & x_1 & (x_1 - 0.6)_+ \\ \vdots & \vdots & \vdots \\ 1 & x_n & (x_n - 0.6)_+ \end{bmatrix}$$

Spline basis function

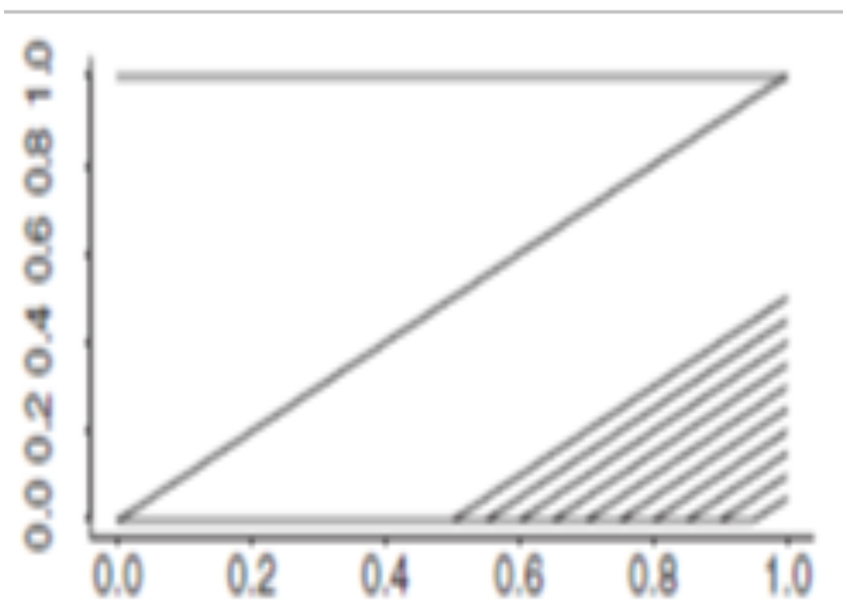
- Assume a more complicated structure



- Straight line structure in the left-hand half but the right-hand is prone to a high amount of detailed structure (whip model)

Spline basis function

- If we have good reason to believe that our underlying structure is of this basic, we could change the basis ?



一个趋 一个趋 一个趋

- where the functions: $(x-0.5)_+$, $(x-0.55)_+, \ldots, (x-0.95)_+$

Spline basis function

- The basis can do a reasonable job with a linear portion between $x = 0$ and $x = 0.5$
- We can use least square to fit such model with

new linear term

$$X = \begin{bmatrix} 1 & x_1 & (x_1 - 0.5)_+ & (x_1 - 0.55)_+ & \dots & (x_1 - 0.95)_+ \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & (x_n - 0.5)_+ & (x_n - 0.55)_+ & \dots & (x_n - 0.95)_+ \end{bmatrix}$$

Spline basis function

- It is possible to handle any complex type of structure by simply adding functions of the form $(x - k)_+$ to the basis
- This is equivalent to adding a column of values to the X matrix
- The value k corresponding to the function $(x - k)_+$ is referred to as a knot
- This is because the function is made up of two lines that are tied together at $x = k$

Spline basis function

- The function $(x - 0.6)_+$ is called a linear spline basis function
- A set of such functions is called a linear spline basis
- Any linear combination of linear spline basis functions 1, x, $(x_i - k_1)_+, \ldots, (x_i - k_K)_+$ is a piecewise linear function with knots k_1, $k_2, \ldots, k_K$ and called spline

~~points~~

K: knots

C

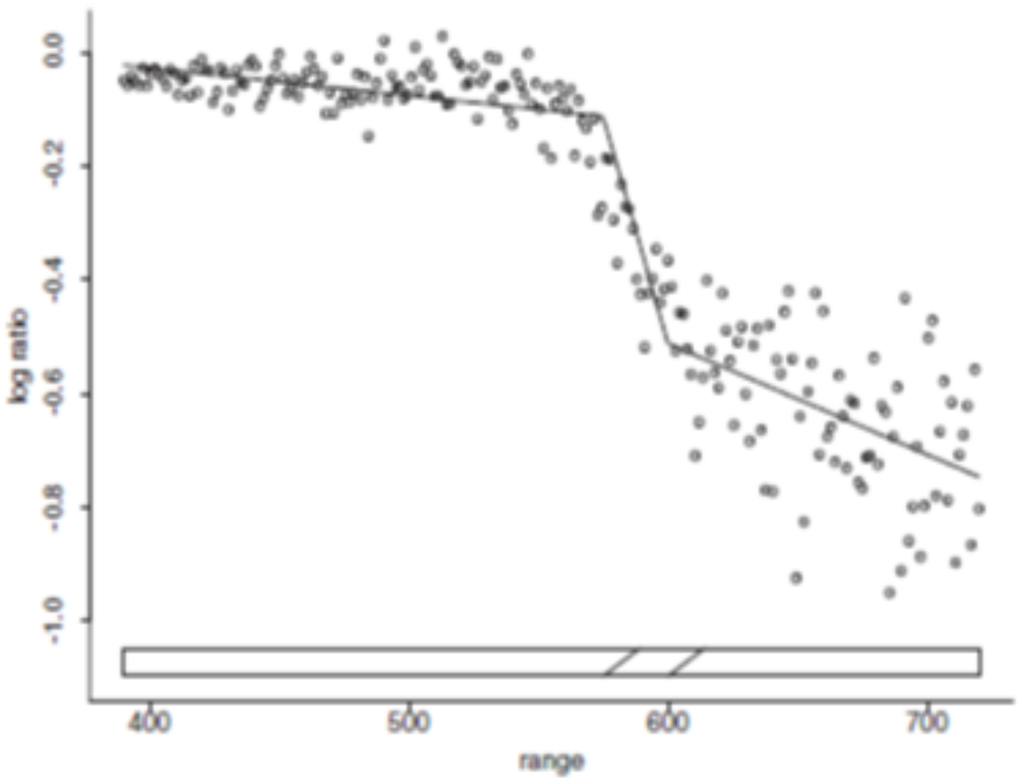
Spline basis function

- Rather than referring to the spline basis function $(x-k)_+$ it is common to simply refer to it knots k
- We say the model has a knot at 0.35 it the function $(x-0.35)_+$ is in the basis (another linear term start at 0.35) .
- The spline model for a function f is

$$f(x) = \underbrace{\beta_0 + \beta_1 x}_{\text{global part}} + \underbrace{\sum_{i=1}^{K} \beta_{ki} (x - k_i)_+}_{\text{help you model local variation that will not be captured by global part.}}$$

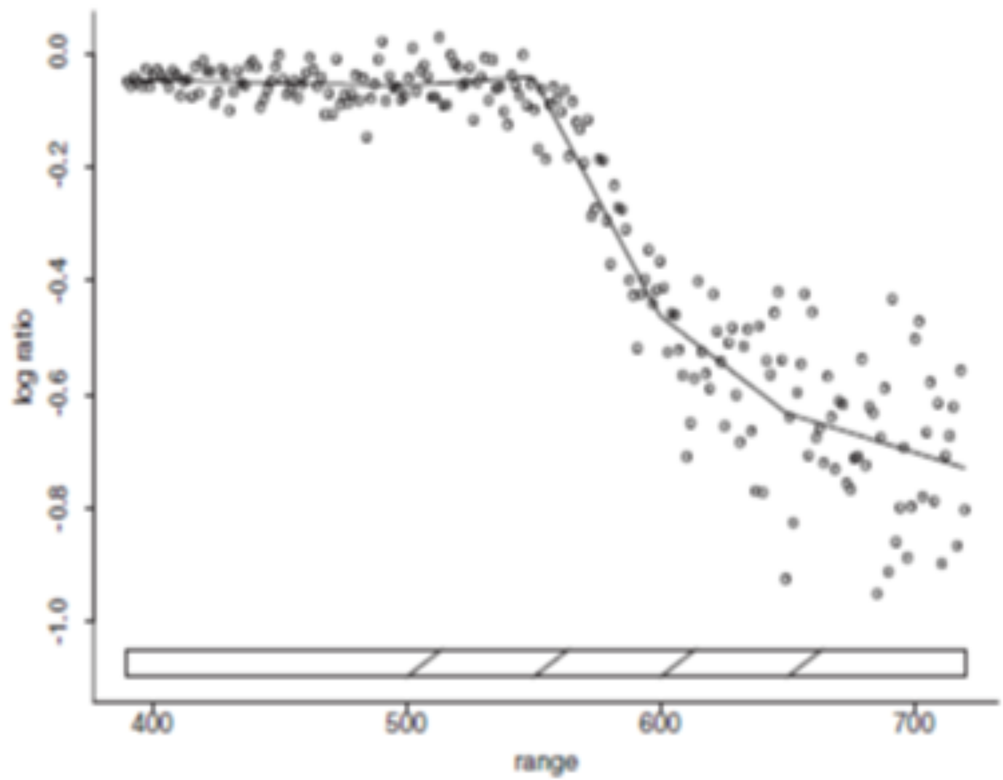
Illustration

- The selection of a good basis is usually challenging
- Start by trying to choose knots by trial (at range 575 and 600)



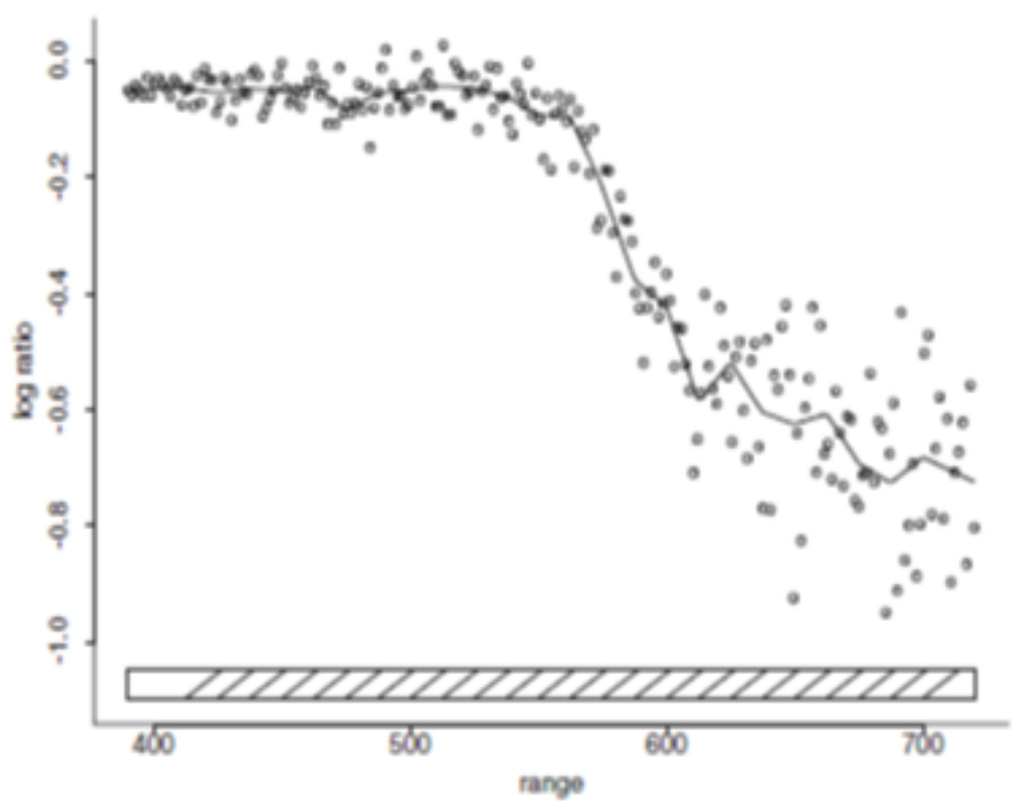
Illustration

- The fit lacking in quality for low values of range
- An obvious remedy is to use more knots (at range 500, 550, 600 and 650)



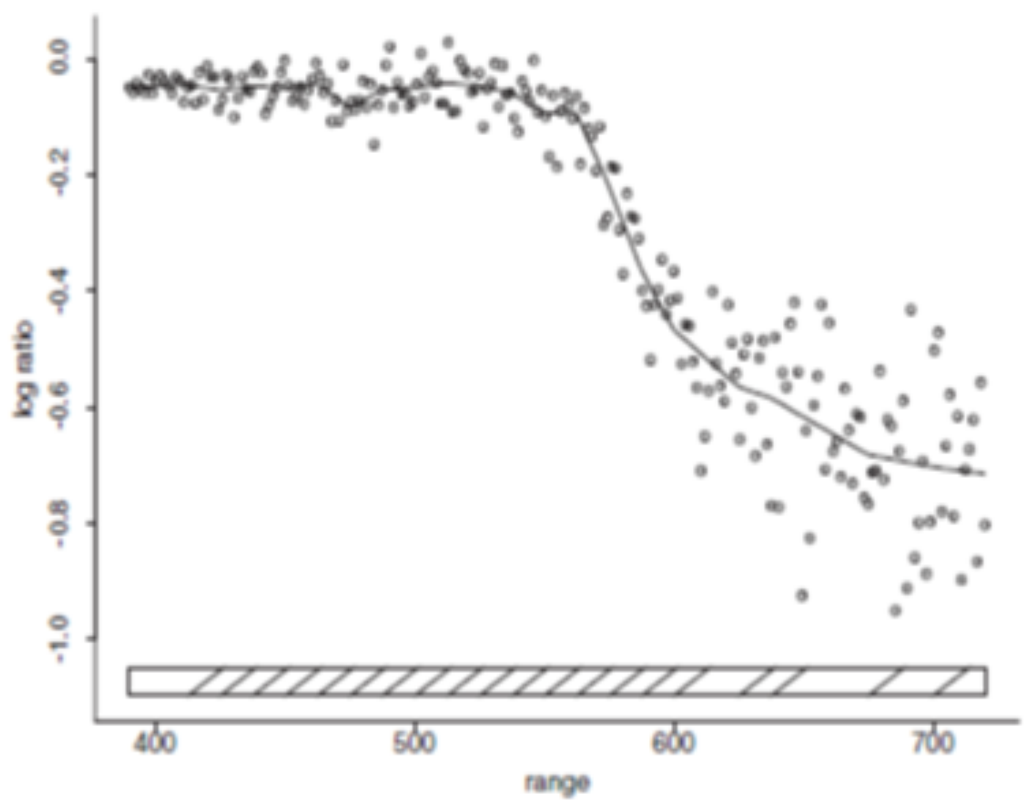
Illustration

- Larger set of knots (at every 12.5), the fitting procedure has much more flexibility
- The plots is heavily overfitted



Illustration

- Pruning the knots (at 612.5, 650, 662.5 and 687.5) to overcome the overfitting issue
- This fits the data well without overfitting
- Obtained, after a lot of time consuming trial and error



Knot selection

- A natural attempt at automatic selection of the knots is to use a model selection criterion
- If there are K candidate knots then there are 2^K possible models assuming the overall intercept and linear term are always present

  K knots $\Rightarrow 2^K$ possible models
- Highly computational intensive

Penalized spline regression

retain all knots but constrain the influence

- Too many knots in the model induces roughness of the fit
- An alternative approach: retain all the knots but constrain their influence
- Hope: this will result in a less variable fit
- Consider a general spline model with K knots, K large

Penalized spline regression

- The ordinary least square fit is written as

$$\hat{\mathbf{y}} = X\hat{\boldsymbol{\beta}} \text{ where } \hat{\boldsymbol{\beta}} \text{ minimizes } \|\mathbf{y} - X\boldsymbol{\beta}\|^2$$

- and $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_{11}, ..., \beta_{1K}]$ with β_{1k} the coefficient of the k^{th} knot.
- Unconstrained estimation of the $\boldsymbol{\beta}$ leads to a wiggly fit

摇摆性

Penalized spline regression

Constraints on the β_{1k} that might help avoid this situation are

- $\max|\beta_{1k}| < C$
- $\sum|\beta_{1k}| < C$
- $\sum\beta_{1k}^2 < C$

With an appropriate choice of C each of these will lead to a smoother fit, however the last constraint is much simpler to implement

$$f(x) = \underbrace{\beta_0 + \beta_1 x}_{\text{linear part}} + \underbrace{\sum_{i=1}^{K} \beta_{ki} (x - k_i)_+}_{\text{knots}} \rightarrow \text{penalize this only.}$$

Penalized spline regression

Define the matrix **D** if size $(K+2) \times (K+2)$

$$
\mathbf{D} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{2\times 2} & \mathbf{0}_{2\times K} \\ \mathbf{0}_{K\times 2} & \mathbf{I}_{K\times K} \end{bmatrix}
$$

Penalized spline regression

- The third constraint is easier to implement than the first two
- The minimization problem

only penalize on the spline term

$$\text{Minimize } \|\mathbf{y} - X\boldsymbol{\beta}\|^2 \text{ subject to } \boldsymbol{\beta}^\top \mathbf{D}\boldsymbol{\beta} \leq C$$

- This is equivalent to choosing $\boldsymbol{\beta}$ to minimize

$$\|\mathbf{y} - X\boldsymbol{\beta}\|^2 + \lambda\boldsymbol{\beta}^\top \mathbf{D}\boldsymbol{\beta}$$

- for $\lambda \geq 0$ and has solution

$$\hat{\boldsymbol{\beta}}_\lambda = \left(X^\top X + \lambda \mathbf{D}\right)^{-1} X^\top \mathbf{y}$$

$$y = \beta_0 + \beta_1 X + \sum_{i=1}^{2} \beta_{1i}(X - b_i) + \varepsilon_i$$

$$= f(x_i) + \varepsilon_i$$

$$\min_{\beta} \|y - f(x_i; \beta)\|_2^2 + \lambda \beta^T D \beta$$

$$\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_{11} \\ \beta_{12} \end{bmatrix}$$

$$\beta^T = \begin{bmatrix} \beta_0 & \beta_1 & \beta_{11} & \beta_{12} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & \beta_{11} & \beta_{12} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_{11} \\ \beta_{12} \end{bmatrix} = \beta_{11}^2 + \beta_{12}^2$$

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Penalized spline regression

estimation more rigid ⇒ less prone to anything

- The term $\lambda\boldsymbol{\beta}^\top\mathbf{D}\boldsymbol{\beta}$ is called a roughness penalty since it penalizes fits that are too rough, thus yielding smoother result
- The amount of smoothness is controlled by λ, which is therefore referred to as a smoothing parameter
- The fitted values for penalized spline regression are

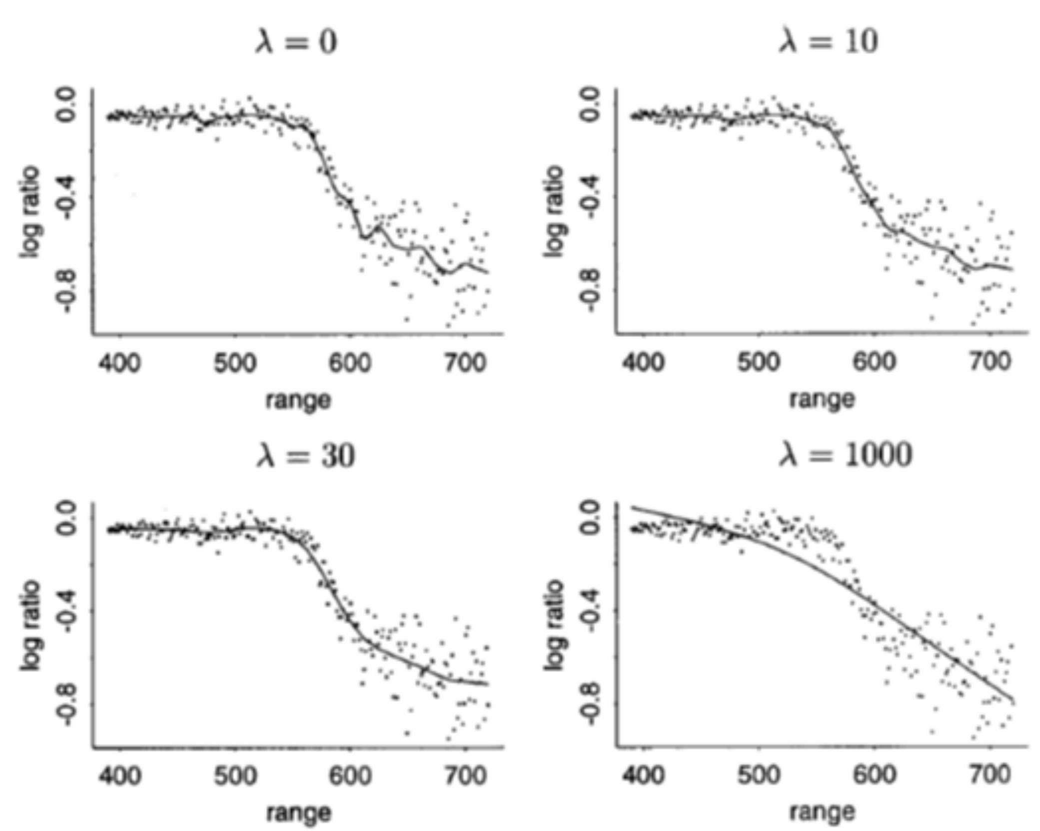
$$\hat{\mathbf{y}} = X \underbrace{\left(X^\top X + \lambda\mathbf{D}\right)^{-1} X^\top \mathbf{y}}$$

$$\Downarrow$$

$$\hat{\boldsymbol{\beta}}_\lambda = \left(X^\top X + \lambda\mathbf{D}\right)^{-1} X^\top \mathbf{y}$$

Illustration

- Linear penalized spline regression fits for different values of the smoothing parameter (depends on λ)



Quadratic spline bases

- We have discussed linear splines, that is continuous, piecewise linear functions
- The reason for the piecewise linear nature of the functions ?

Quadratic spline bases

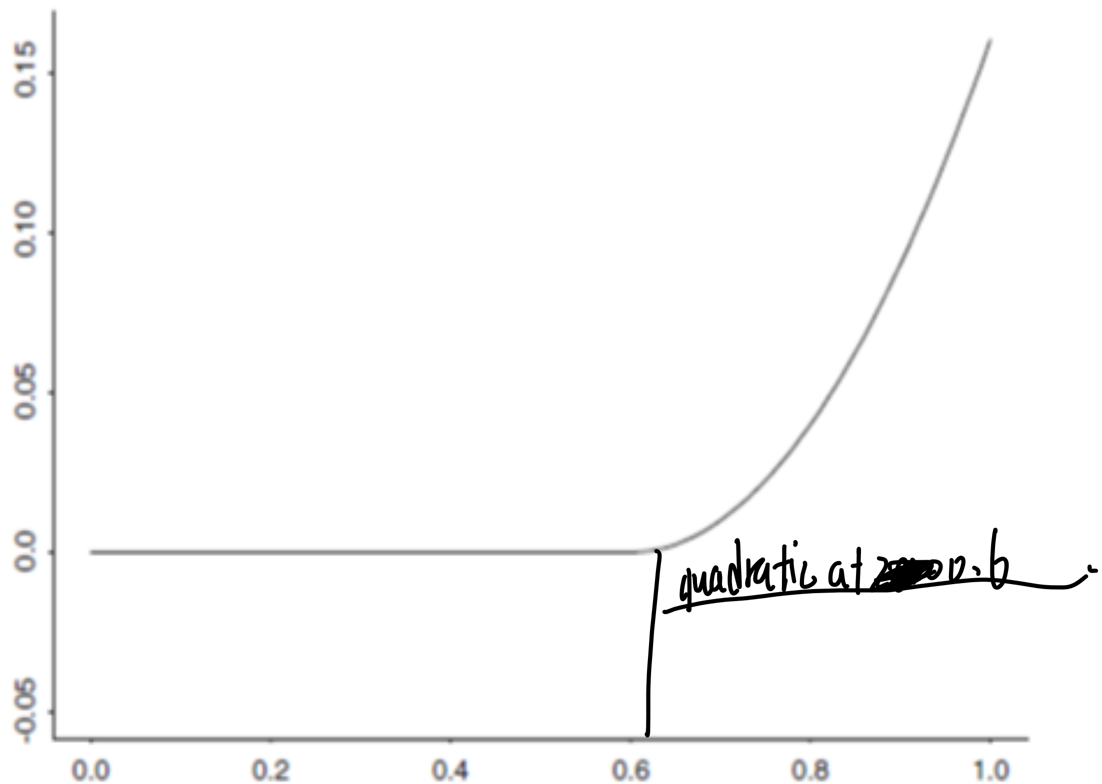
- We have discussed linear splines, that is continuous, piecewise linear functions
- The reason for the piecewise linear nature of the functions ?
- is that they are a linear combination of piecewise linear functions of the form $(x-k)_+$
- A simple way of escaping from piecewise linearity ?

Quadratic spline bases

- We have discussed linear splines, that is continuous, piecewise linear functions
- The reason for the piecewise linear nature of the functions ?
- is that they are a linear combination of piecewise linear functions of the form $(x-k)_+$
- A simple way of escaping from piecewise linearity ?
- is to add x^2 to the basis and also to replace each $(x-k)_+$ by its square $(x-k)_+^2$

Illustration of a quadratic spline basis function

- Illustration of the function $(x - 0.6)_+^2$



Quadratic spline bases

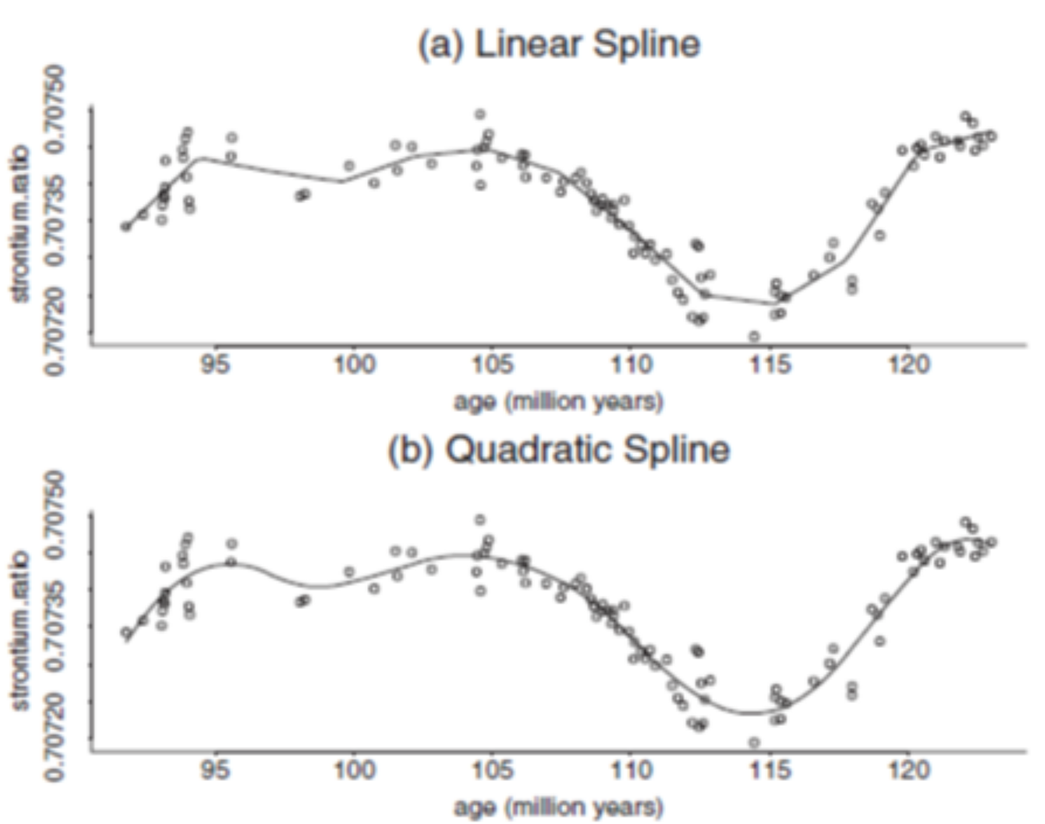
- The function doesn't have a sharp corner like $(x-0.6)_+$ does
- The function $(x-0.6)_+^2$ has a continuous first derivative
- Any linear combination of the functions

$$1, x, x^2, (x-k_1)_+^2, \ldots, (x-k_K)_+^2$$

- also have a continuous first derivative and not have any sharp corner
- This result in better fit
- This is called a quadratic spline basis with knots at $k_1, \ldots, k_K$

Illustration of quadratic spline basis functions

- Quadratic spline do a better job of fitting peaks and valleys



Other spline bases

- We discussed linear and quadratic spline models
- One reason for considering other models is to achieve smoother fits $\rightarrow$ important if one plans to differentiate the fit to estimate derivative of the regression function
- In principle a change of basis does not change the fit but some bases are more stable and allow computation of a fit with better accuracy
- Besides numerical stability: ease of implementation is another reason for selecting one basis over another
- An obvious generalization is given by

$$1, x, \ldots, x^p, (x - k_1)_+^p, \ldots, (x - k_K)_+^p$$

- know as the truncated power basis of degree p

Other spline bases

- since the function $(x-k)_+^p$ has $p-1$ continuous derivatives, higher values of p lead to smoother spline functions
- The p^{th} degree spline model is

$$f(x) = \beta_0 + \beta_1 x + \ldots + \beta_p x^p + \sum_{i=1}^{K} \beta_{ki} (x-k_i)_+^p$$

- The expression for the fitted values is given by

$$\hat{\mathbf{y}} = X\left(X^\top X + \lambda \mathbf{D}\right)^{-1} X^\top \mathbf{y}$$

λ很小时 $\Rightarrow$ inverse X^TX

什么时候好 inserve X^TX

when we have orthorgnal matrix $X^TX = I$

$$\mathbf{D} = \mathrm{diag}\left(\mathbf{0}_{p+1}, \mathbf{1}_K\right)$$

B-Spline bases

Truncated power bases can be used in practice

- if the knots are selected carefully or
- a penalized fit is used

Truncated power bases have the practical disadvantage that they are far from orthogonal $X^T X = |$

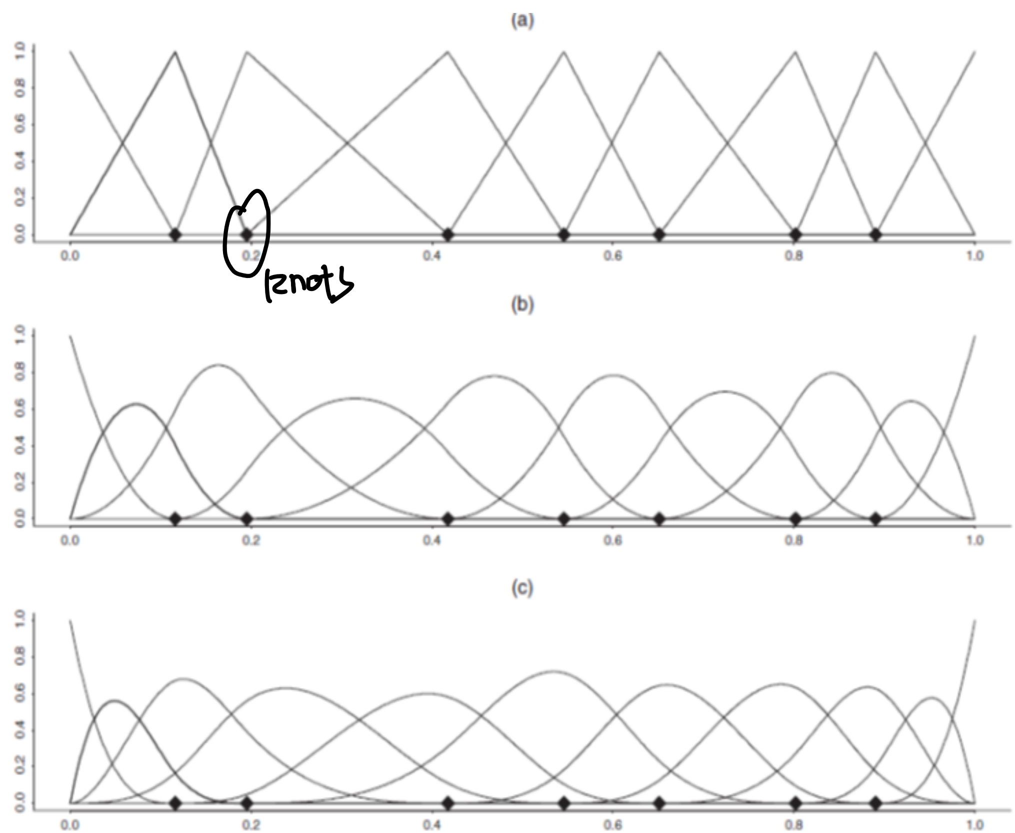
- this lead to numerical instability
- particularly when there is a large number of knots (λ is small or zero)

B-Spline bases

- In practice, especially for OLS fitting, it is advisable to work with equivalent bases with more stable numerical properties.
- The most common choice is the B-spline basis

B-Spline bases

B-spline bases of degree 1, 2 and 3 for the case of seven irregularly spaced knots



B-Spline bases

- Each of these are equivalent to the truncated power basis of the same degree
- In regression, this means using B-spline for the columns of X or truncated power basis of similar degree produce identical fits (knots at the same locations).

B-Spline bases

Mathematically, this equivalence is quantified as follows

- Let X_T be the X matrix with columns obtained with a power truncated basis and X_T是 matrix
  $\Rightarrow X_B^T X_B = I$ X_B与X_T不同
- let X_B be the X matrix corresponding to the B-spline basis of the same degree and same knots locations then

$n \times (k+p+1) \quad (k+p+1) \times (k+p+1)$

$$X_B = X_T L_p \text{ where } L_p \text{ is square } \cancel{\text{invertible}} \text{ matrix}$$

The penalized spline fit of degree p in terms of $B-$spline

$$\Rightarrow \hat{y} = X_T (X_T^T X_T + \lambda D)^{-1} X_T^T y = X_T (L_p L_p^{-1}) (X_T^T X_T + \lambda D)^{-1} (L_p^T)^{-1} L_p^T X_T^T y$$

$$\hat{\mathbf{y}} = X_B \left(X_B^\top X_B + \lambda L_p^\top \mathbf{D} L_p \right)^{-1} X_B^\top \mathbf{y}$$

$X_B^T X_B = I$ can be inversed faster when λ is small

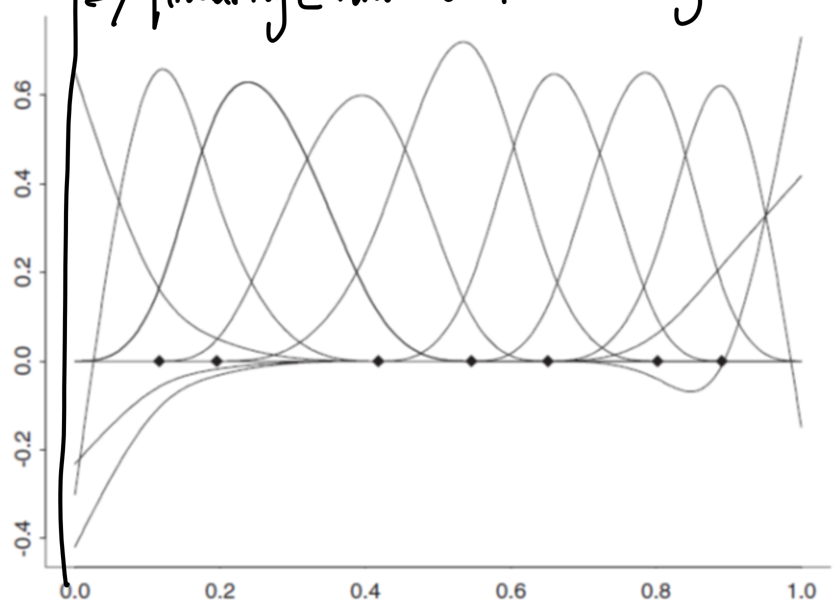
$\rightarrow$ basis used in regression packages

Note 1

Natural Cubic Spline

- Nature cubic spline is a modification of cubic spline that adds a linearity constraint beyond the boundary knots (0 and 1)
- The other knots are called interior knots
- The linearity is enforced through the constraints that the spline f satisfy $f'' = f''' = 0$ at the boundary knots

linearity constrain on the boundary



Cubic smoothing spline : don't need to select knots, put cubic spline at each knot

- Spline basis method that avoids the knot selection issue by using a maximal set of knots (or n knots) (add penalty on all knots)
- Among all functions $f(x)$ with two continuous derivatives, select $\hat{f}(x)$ that minimizes

Cubic smoothing spline : $\underset{f(x)}{\operatorname{argmin}}$
$$\sum_{i=1}^{n} \{y_i - f(x_i)\}^2 + \lambda \int \{f''(x)\}^2 dx$$
put cubic spline at every observation

$\hat{f}(x)$: cubic smoothing spline, cubic spline with knot at x_i

- The regularization controls the complexity of the fit by penalizing the curvature of the function f
- The minimizer of this penalized sum of squares is a natural cubic spline with knots at the x_i

Cubic smoothing spline

The smoothing parameter λ controls the tradoff between closeness to the data and complexity and there are two special cases

- $\lambda = 0$: f can be any function that interpolates the data (very rough)
- $\lambda = \infty$: least square line fit (since no second derivative can be tolerated)
- The function is over-parametrized since there are n knots which implies n degrees of freedom
- The penalty term translates to a penalty on the spline coefficients which are shrunk toward the linear fit

Cubic smoothing spline

$\underbrace{x_1, \ldots, x_n}_{\text{knots}}, \underbrace{y_1, \ldots, y_n}_{f(x)=\sum_{i=1}^{n} B_i(x)\beta_i}$ $n=3$

Since the solution is a natural spline, it can take the form

$$f(x) = \sum_{i=1}^{n} B_i(x)\,\beta_i$$

basis function B_i at x_i

$B_j(x)$ are an $n-$dimensional set of basis functions for representing this family of natural spline

With this representation, the criterion for smoothing spline reduces

$$RSS(\beta, \lambda) = (\mathbf{y} - \mathbf{B}\beta)^\top (\mathbf{y} - \mathbf{B}\beta) + \lambda\beta^\top \Omega\beta$$

where $\mathbf{B}_{ij} = B_i(x_j)$ and $\{\Omega\}_{lm} = \int B_l''(x)B_m''(x)dx$

$$f(x)=\sum_{i=1}^{3} B_i(x)\beta_i \qquad n=3$$

$$=\beta_1 B_1(x)+\beta_2 B_2(x)+\beta_3 B_3(x)$$

$$B_1(x)=\begin{bmatrix} B_1(x_1)\\ B_1(x_2)\\ B_1(x_3)\end{bmatrix} \qquad \beta=\begin{bmatrix}\beta_1\\ \beta_2\\ \beta_3\end{bmatrix}$$

$$B=\begin{bmatrix} B_1(x_1) & B_2(x_1) & B_3(x_1)\\ B_1(x_2) & B_2(x_2) & B_3(x_2)\\ B_1(x_3) & B_2(x_3) & B_3(x_3)\end{bmatrix} \qquad D=\{(x_i,y_i),\ i=1,2,3\}$$

B

$$=\frac{\partial\,(y-B\beta)^T(y-B\beta)+\lambda\beta^T w\beta}{\partial\beta}$$

$$=-B^T(y-B\beta)+\lambda w\beta$$

$$=-B^Ty+B^TB\beta+\lambda w\beta$$

$$\beta=(B^TB+\lambda w)^{-1}B^Ty$$

Cubic smoothing spline

The fitted smoothing spline is given by

- The solution is

$$\hat{\beta} = \left(\mathbf{B}^\top \mathbf{B} + \lambda\Omega\right)^{-1} \mathbf{B}^\top \mathbf{y}$$

- The fitted smoothing spline is given by

$$\hat{f}(x) = \sum_{i=1}^{n} B_i(x)\,\hat{\beta}_i$$

- Efficient computation in $O(n)$ operations can be realized using a Cholesky decomposition

$$\left(\mathbf{B}^\top \mathbf{B} + \lambda\Omega\right)\beta = \mathbf{B}^\top \mathbf{y}$$

General form of penalized spline

The general definition of penalized spline is $\mathbf{B}(x)\boldsymbol{\beta}$ and

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} \left[y_i - \mathbf{B}(x_i)\boldsymbol{\beta}\right]^2 + \lambda \boldsymbol{\beta}^\top \mathbf{D} \boldsymbol{\beta}$$

where $\mathbf{D}$ is symmetric positive semidefinite and $\lambda > 0$

- In case of spline basis $\mathbf{D} = \mathrm{diag}\left(\mathbf{0}_{p+1}, \mathbf{1}_K\right)$
- In smoothing splines $\mathbf{D}$ defines the penalty

General form of penalized spline

When applying splines, there are two basic choices to make

- The spline model: the degree and knot locations
- The penalty: the form of the penalty

Once the choices have been made, there follow two secondary choices

- The basis functions: truncate power functions or B-splines
- The basis functions used in the computations

Linear smoothers

Penalized spline is a linear function of the data $\mathbf{y}$

$$\hat{\mathbf{y}} = S_\lambda \mathbf{y} \text{ with } S_\lambda = X\left(X^\top X + \lambda \mathbf{D}\right)^{-1} X^\top$$

- where X corresponds for example to the p^{th} degree truncated spline basis
- S_λ is usually called the smoother matrix

In general

$$\hat{\mathbf{y}} = L\mathbf{y}$$

where L is an $n \times n$ matrix that doesn't depend on $\mathbf{y}$ directly (but does through λ). This is also called linear smoother.

Error of the smoothers

Let $\hat{f}$ be an estimator of f obtained from

$$y_i = f(x_i) + \epsilon_i$$

An important quantity of interest is the error incurred by an estimator with respect to a given target. The most common measure of error is the mean square error MSE

$$MSE\left\{\hat{f}(x)\right\} = E\left[\left\{\hat{f}(x) - f(x)\right\}^2\right]$$

which has the advantage of admitting the decomposition

$$MSE\left\{\hat{f}(x)\right\} = \left[E\left\{\hat{f}(x)\right\} - f(x)\right]^2 + \mathrm{var}\left\{\hat{f}(x)\right\}$$

which represents the squared bias and variance of the error.

Error of the smoothers

- The entire curve is of interest $\rightarrow$ so it is common to measure the error globally across several values of x
- Mean integrated squared error (MISE) is a possibility

$$\mathrm{MISE}\left\{\hat{f}(.)\right\} = \int_{\mathcal{X}} \mathrm{MSE}\left\{\hat{f}(x)\right\} dx$$

- when only error at the observations are considered

$$\mathrm{MSSE}\left\{\hat{f}(.)\right\} = E\sum_{i=1}^{n}\left\{\hat{f}(x_i) - f(x_i)\right\}^2$$

Error of the smoothers

- Let $\hat{\mathbf{f}} = \left[\hat{f}(x_1), ..., \hat{f}(x_n)\right]^\top$ denotes the vector of fitted values and
- let $\mathbf{f} = [f(x_1), ..., f(x_n)]$ denotes the vector of unknown values

$$\mathrm{MSSE}\left(\hat{\mathbf{f}}\right) = E\|\hat{\mathbf{f}} - \mathbf{f}\|^2$$

- For linear smoother $\hat{\mathbf{f}} = L\mathbf{y}$

$$\mathrm{MSSE}\left(\hat{\mathbf{f}}\right) = \sum_{i=1}^{n}\left(E\hat{f}(x_i) - f(x_i)\right)^2 + \mathrm{var}\left\{\hat{f}(x_i)\right\}$$

$$\mathrm{MSSE}\left(\hat{\mathbf{f}}\right) = \|(L - I)\mathbf{f}\|^2 + \sigma_\epsilon^2 \mathrm{tr}\left(LL^\top\right)$$

Note 2

Error of the smoothers

- The bias is given by

$$\mathrm{Bias}\left(\hat{\mathbf{f}}\right) = \mathbf{f} - E\left(\hat{\mathbf{f}}\right) = \mathbf{f} - L\mathbf{f}$$

- The covariance

$$\mathrm{cov}\left(\hat{\mathbf{f}}\right) = L\mathrm{cov}\left(\mathbf{y}\right)L^{\top} = \sigma_{\epsilon}^{2}LL^{\top}$$

- The diagonal contains the pointwise variances at the x_i

Degrees of freedom of a smoother

- For penalized spline

$$df_{fit} = \mathrm{tr}\left[\left(X^\top X + \lambda D\right)^{-1} X^\top X\right] = \mathrm{tr}\left(S_\lambda\right)$$

- For K knots and degree p

$$\mathrm{tr}\left(S_0\right) = p + 1 + K$$

- At the other extreme

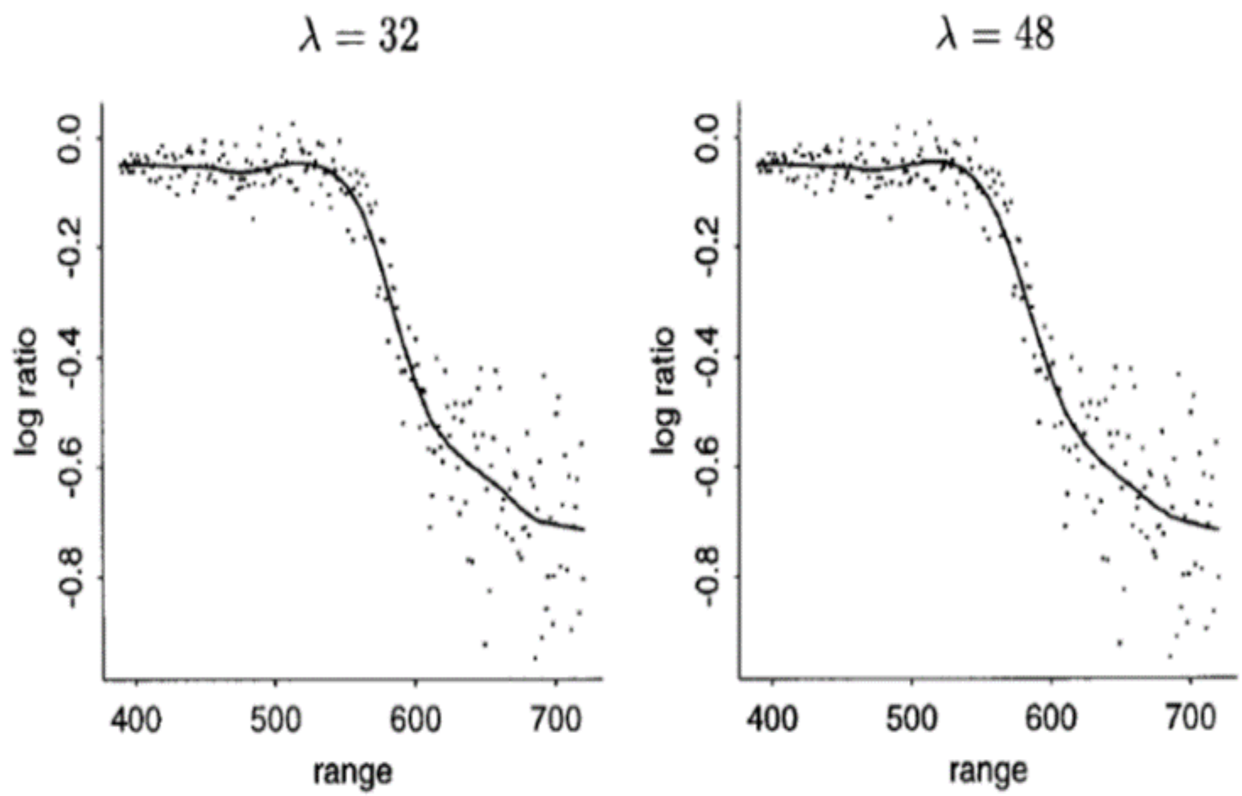
$$\mathrm{tr}\left(S_\lambda\right) \to p + 1 \text{ as } \lambda \to \infty$$

- So for $\lambda > 0$

$$p + 1 < \mathrm{df} < p + 1 + K$$

Degrees of freedom of a smoother

Different values lead to similar appearance. They have roughly the same degree of freedom



Cross validation

- The most common measure for the goodness of fit of a regression curve

$$\mathrm{RSS} = \frac{1}{N}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{1}{N}\|\mathbf{y} - \hat{\mathbf{y}}\|^2$$

- It is minimized for $\lambda = 0$ for which $\hat{y}_i = y_i$, $1 < i \leq n$
- Solution close to interpolation

Cross validation

- Cross-validation allows the estimation of λ when $\hat{f}(x, \lambda)$ is used as a nonparametric regression at x

$$\mathrm{RSS}(\lambda) = \frac{1}{N}\sum_{i=1}^{n}\left(y_i - \hat{f}_{-i}(x_i, \lambda)\right)^2$$

- The cross-validation criterion is

$$\mathrm{CV}(\lambda) = \frac{1}{N}\sum_{i=1}^{n}\left(y_i - \hat{f}_{-i}(x_i, \lambda)\right)^2 = \frac{1}{N}\sum_{i=1}^{n}\left(\frac{y_i - \hat{f}_\lambda(x_i)}{1 - S_\lambda(i,i)}\right)^2$$

- The choice of λ is the one that minimizes $CV(\lambda)$ over $\lambda \geq 0$

Cross validation

A computationally efficient variant can be obtained using a simplified version where $S_\lambda(i,i)$ are replaced by their average

$$\frac{1}{n}\sum_{i=1}^{n} S_\lambda(i,i) = \frac{1}{n}\mathrm{tr}\left(S_\lambda\right)$$

This leads to the generalization cross validation

$$\mathrm{GCV}(\lambda) = \frac{1}{N}\sum_{i=1}^{n}\left(\frac{[(I - S_\lambda)\,\mathbf{y}]_i}{1 - n^{-1}\mathrm{tr}(S_\lambda)}\right)^2 = \frac{RSS(\lambda)}{(1 - n^{-1}\mathrm{tr}(S_\lambda))^2}$$

Selection with a criterion

Our interest

$$\mathrm{MSSE}\left(\hat{\mathbf{f}}\right) = \|\left(L - I\right)\mathbf{f}\|^2 + \sigma_\epsilon^2 \mathrm{tr}\left(LL^\top\right)$$

where $S_\lambda = L$, however

$$E\left(RSS\right) = E\|\hat{\mathbf{f}} - \mathbf{y}\|^2 = \mathrm{MSSE}\left(\hat{\mathbf{f}}\right) + \sigma_\epsilon^2\left(n - 2df_{fit}\right)$$

where $df_{fit} = \mathrm{tr}\left(S_\lambda\right) = \mathrm{tr}\left(L\right)$.

Note 3

Selection with a criterion

It follows that if $\hat{\sigma}_\epsilon^2$ is an unbiased estimate of σ_ϵ^2 then

$$\mathrm{IC} = \mathrm{RSS} + 2\hat{\sigma}_\epsilon^2 df_{fit}$$

is an unbiased estimator of

$$\mathrm{MSSE}\left(\hat{\mathbf{f}}\right) + n\sigma_\epsilon^2$$

but $n\sigma_\epsilon^2$ doesn't depend on $S_\lambda \rightarrow$ then minimizing IC is approximately similar to minimizing $\mathrm{MSSE}\left(\hat{\mathbf{f}}\right)$

Selection with a criterion

For penalized splines this leads to

$$C_p(\lambda) = \mathrm{RSS}(\lambda) + 2\hat{\sigma}_\epsilon^2 df_{fit}(\lambda)$$

for selecting λ. We represent $\hat{\lambda}_{C_p}$ the smoothing parameter obtained by minimizing $C_p(\lambda)$.

As estimate of $\hat{\sigma}_\epsilon^2$ we take

$$\hat{\sigma}_\epsilon^2 = \frac{\mathrm{RSS}(\lambda)}{df_{res}(\lambda)}$$

Note 4

Selection with a criterion

GCV can approximately take a different form

$$GCV\,(\lambda) = \frac{RSS(\lambda)}{\left(1 - n^{-1}\mathrm{tr}(S_\lambda)\right)^2}$$

$$= RSS(\lambda) + 2\hat{\sigma}^2_\epsilon\,(\lambda)\,df_{fit}$$

The main difference is that GCV estimates σ^2_ϵ using $RSS(\lambda)$ where as $C_p(\lambda)$ requires a prior estimate of σ^2_ϵ.

Selection with a criterion

This can be extended to other selection criteria for example

$$AIC(\lambda) = \log\left(RSS(\lambda)\right) + \frac{2}{n} df_{fit}$$

Other basis

Assume f is defined on the unit interval, under some regularity conditions, f admits a Fourier series representation

$$f(x) = \beta_0 + \sum_{j=1}^{\infty} \{\beta_j^s \sin(j\pi x) + \beta_j^c \cos(j\pi x)\}$$

For higher values of j, the functions $\sin(j\pi x)$ and $\cos(j\pi x)$ become more oscillatory $\rightarrow$ account for the finer structure in f

Other basis

For smoother f, the corresponding coefficients are small

$$\hat{f}(x) = \hat{\beta}_0 + \sum_{j=1}^{J} \left\{ \hat{\beta}_j^s \sin(j\pi x) + \hat{\beta}_j^c \cos(j\pi x) \right\}$$

$\hat{\beta}_j^s$, $\hat{\beta}_j^c$, $(1 \leq j \leq J)$ and $\hat{\beta}_0$ are obtained by least squares.

The values of J is the smoothing parameter in this case.

Radial Basis functions

An extension of the truncated power functions

$$1, x, \ldots, x^p, |x - k_1|^p, \ldots, |x - k_K|^p$$

where

$$|x - k_i|^p = r\left(|x - k_i|\right) \text{ where } r(u) = u^p$$

This shows that this basis $|x - k_i|^p$ $(1 \leq i \leq K)$ depends only on the distance $|x - k_i|$ and the univariate function r

Radial Basis functions

Extension to multivariate cases $\mathbf{x} \in \mathbb{R}^d$ and $k_1, \ldots, k_K \in \mathbb{R}^d$ is straightforward

$$r\left(\|\mathbf{x} - k_i\|\right)$$

- where $\|\mathbf{v}\| = \sqrt{\mathbf{v}^\top \mathbf{v}}$ is the vector length
- These functions are radially symmetric
- They are called radial basis functions

Cubic approximation

A cubic smoothing spline approximation can be written as

$$\hat{f}(x) = \hat{\beta}_0 + \hat{\beta}_1 x + \sum_{j=1}^{n} \hat{\beta}_{1j} |x - x_j|^3$$

where $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_{11}, \ldots, \hat{\beta}_{1n}$ minimize

$$\|\mathbf{y} - X_0 \boldsymbol{\beta}_0 - X_1 \boldsymbol{\beta}_1\|^2 + \lambda \boldsymbol{\beta}_1^\top K \boldsymbol{\beta}_1$$

$$X_0^\top \boldsymbol{\beta}_1 = 0$$

where $\boldsymbol{\beta}_0 = [\beta_0, \beta_1]^\top$, $\boldsymbol{\beta}_1 = [\beta_{11}, \ldots, \beta_{1n}]^\top$, $X_0 = [1, x_i]_{1 \leq \text{ß} \leq n}$ and $X_1 = K = \left[|x_i - x_j|^3\right]_{1 \leq i,j \leq n}$

Cubic approximation

Computational saving can be obtained by specifying a knot sequence $k_1, \ldots, k_K$ and using $K = \left[|k_i - k_j|^3\right]_{1 \le i,j \le K}$ and $X = \left[|x - k_i|^3\right]_{1 \le i \le K}$