UNIVERSITÀ DEGLI STUDI DI TRIESTE

Deams
Dipartimento di
Scienze Economiche, Aziendali,
Matematiche e Statistiche "Bruno de Finetti"

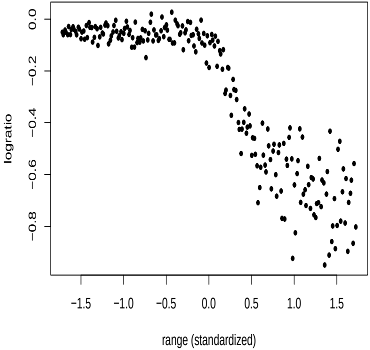
Non parametric statistics

Introduction

Francesco Pauli

A.A. 2021/2022

Motivating example: lidar data



LIDAR = light detection and ranging

- Is a technique to detect chemical compounds in the atmosphere
- x: distance traveled before reflection
- y: log of the ratio of received light between two laser sources

- We want to estimate

$$f(x) = E(Y|X = x)$$

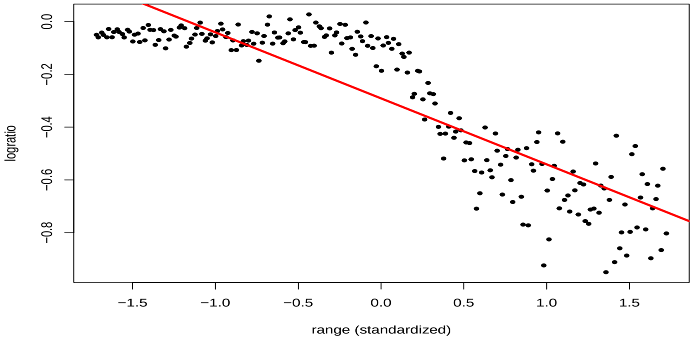
- Well known example of non linear relationship where polynomial regression does non work very well.

Estimation of f

The linear model

$$y_i = f(x_i) + \varepsilon_i = \beta_1 + \beta_2 x_i + \varepsilon_i$$

where $\varepsilon_i \sim IID(\mathcal{N}(0, \sigma^2))$ clearly does not work

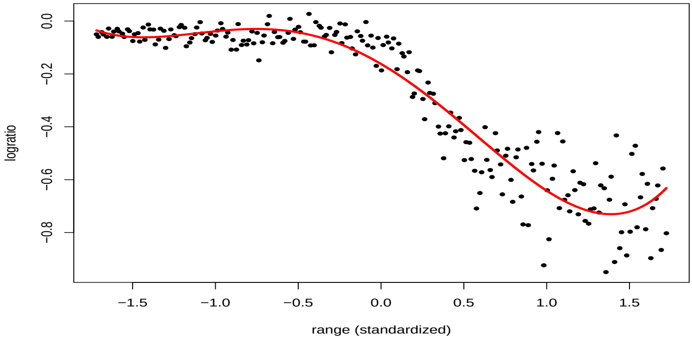


Naive solution 1: polynomial model

A first idea is using a polynomial model, effective if the degree is high enough

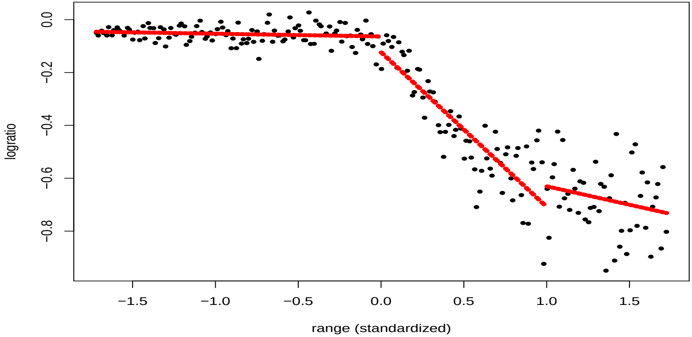
$$f(x;\beta) = \beta_1 + \beta_2 x + \beta_3 x^2 + \beta_4 x^3 + \beta_5 x^4$$

however, it has various problems.



Naive solution 2: piecewise linear model

An alternative might be using a piecewise linear model.



Indice

Spline

Penalized likelihood

Why splines

Other basis

More covariates

Generalized models (non gaussian data)

Linear spline basis: a two-knots example

A more sophisticated but still (piecewise) linear fit is obtained by

$$y_i = \beta_1 + \beta_2 x_i + \beta_3 (x_i - \kappa_1)_+ + \beta_4 (x_i - \kappa_2)_+ + \varepsilon_i$$

where

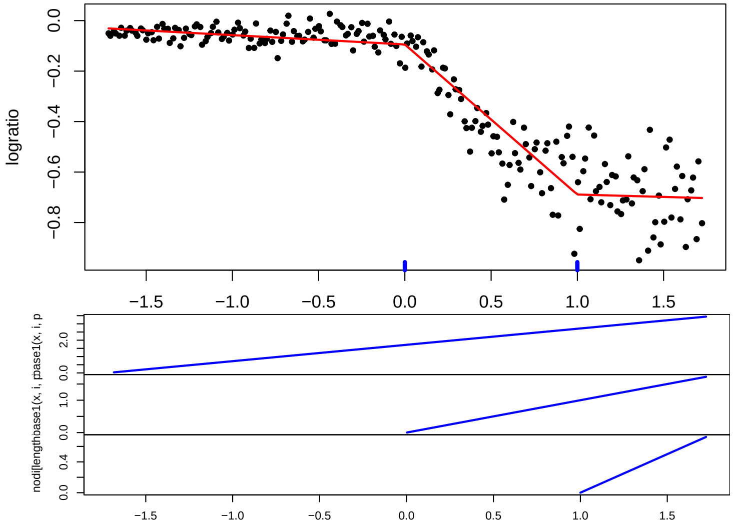
- $\varepsilon_i \sim IID(\mathcal{N}(0, \sigma^2))$,
- κ_1 and κ_2, the knots, are fixed numbers within the range of x
- β_i are estimated using maximum likelihood, that is,

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - f(x_i; \boldsymbol{\beta}))^2$$

  where

$$f(x_i; \boldsymbol{\beta}) = \beta_1 + \beta_2 x_i + \beta_3 (x_i - \kappa_1)_+ + \beta_4 (x_i - \kappa_2)_+$$

Linear spline basis: a two-knots example



Linear spline basis: general specification

More generally, we fix K knots

$$\kappa_1, \ldots, \kappa_K$$

and estimate the model (notation changed!)

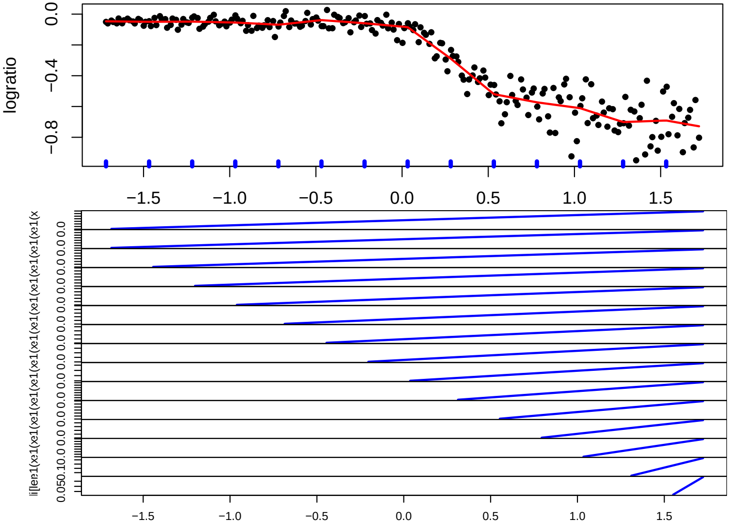
$$y_i = \beta_1 + \beta_2 x_i + \sum_{k=1}^{K} b_k (x_i - \kappa_k)_+ + \varepsilon_i$$

The spline function is represented as

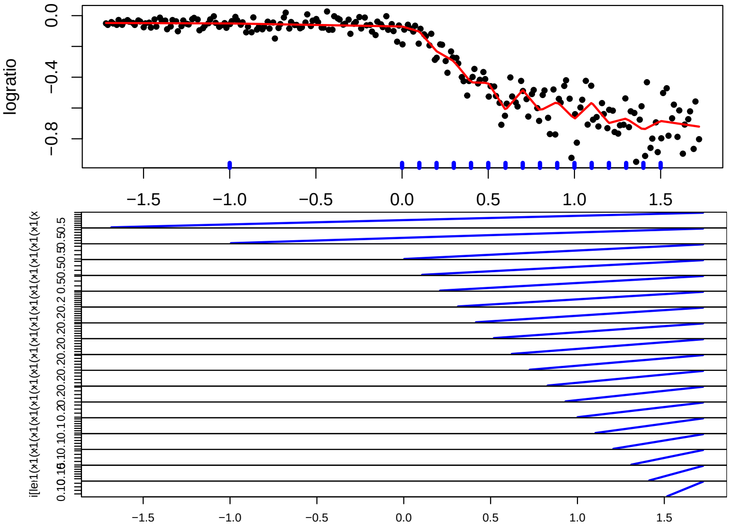
$$f(x) = \beta_1 + \beta_2 x + \sum_{k=1}^{K} b_k (x - \kappa_k)_+$$

and is smoother the fewer knots are used.

Linear spline basis: general specification



Linear spline basis: general specification



Truncated power basis

An obvious extension of the linear basis is the truncated power basis

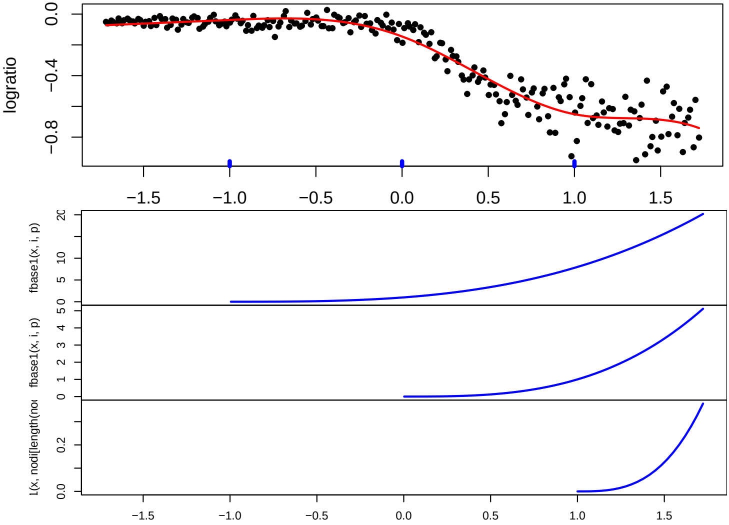
$$y_i = \beta_1 + \beta_2 x_i + \ldots + \beta_{p+1} x^p + \sum_{k=1}^{K} b_k (x_i - \kappa_k)_+^p + \varepsilon_i$$

then the spline function is

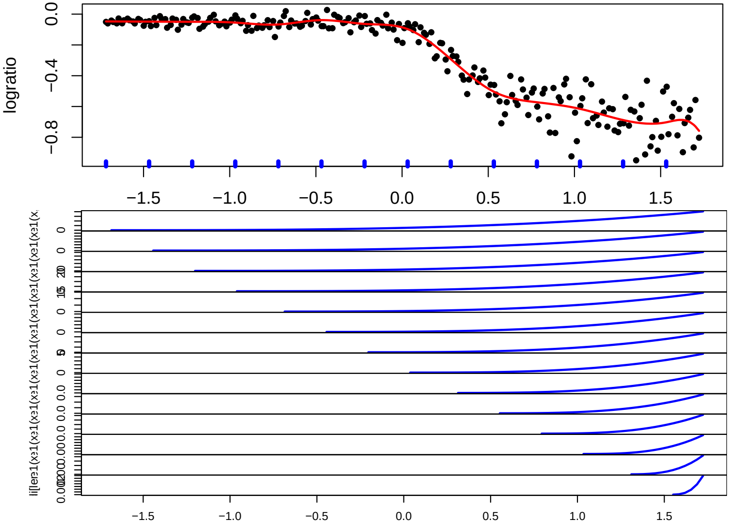
$$f(x) = \beta_1 + \beta_2 x + \ldots + \beta_{p+1} x^p + \sum_{k=1}^{K} b_k (x - \kappa_k)_+^p$$

- A spline with degree p has $p-1$ continuous derivatives,
- $p = 3$ is sufficient for most purposes (unless we want smooth derivatives).

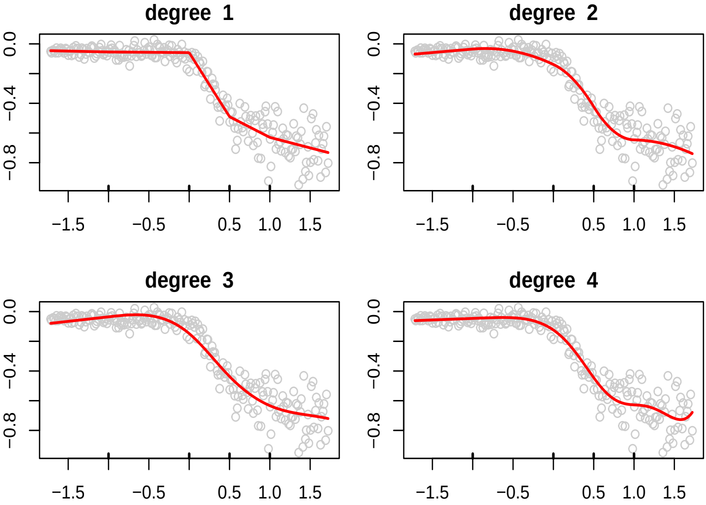
Truncated power basis



Truncated power basis



TPB: different degrees



Smoothness of the spline

Based on what we have seen, the spline as a function is more or less flexible (less or more smooth) depending on the number of knots and the degree

- A higher degree implies more smoothness (however one generally does not go further than $p = 3$).
- For a fixed degree, smoothness is inversely related to the number of knots:
  - no knots means that a linear (polynomial) regression is performed;
  - knots equal to unique observations leads to a spline that interpolates observations exactly;
- (Also, note that the position of the knots determines in which subranges the spline is more/less smooth.)

On the other hand, the more the knots (degree) the more the parameters to be estimated, so more flexibility would lead to higher variance of estimates.

Smoothness of the spline: bias variance trade off

Assuming that the degree is fixed (as is usually done), the choice of the number of knots is analogous to the choice of the bandwidth in kernel regression in that it implies a bias variance trade off

- more knots ↔ less smoothing ↔ less bias but more variance;
- less knots ↔ more smoothing ↔ more bias but less variance.

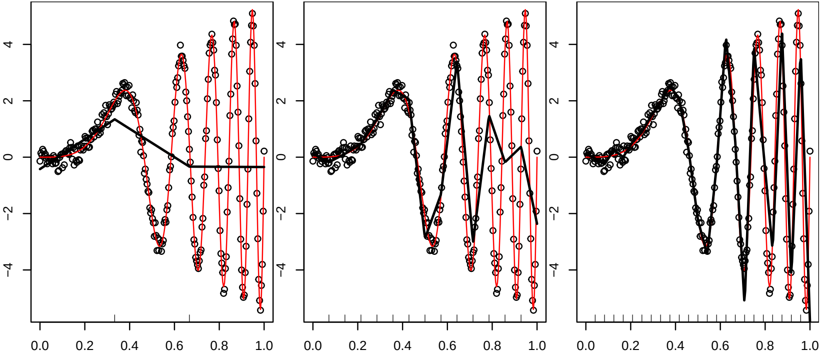
As for the choice of the bandwidth in kernel regression, choice of the smoothness of the spline is crucial.

In principle, we may seek for the optimal level of smoothing by choosing the number of knots by estimating the mean square errors implied by different choices.

This means we must minimize the MSE with respect to the knots number and position, which is a difficult task (although doable, this a legitimate strategy which is actually pursued).

Number of knots and bias

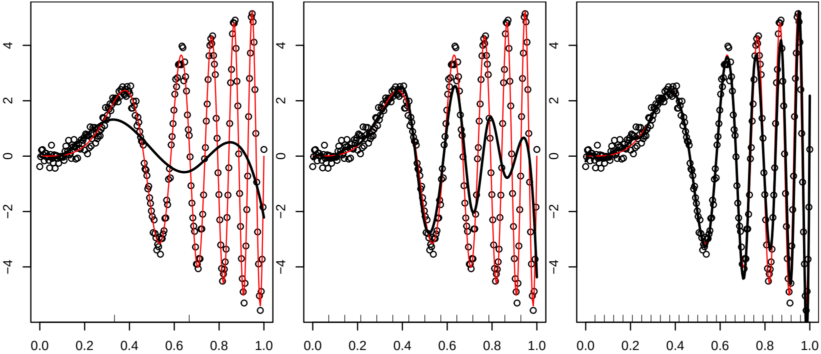
Bias is greater with curvature.



Knots represented on the x axis.

Number of knots and bias

With higher degree splines curvature is less of a problem, however if the function is complicated few knots lead to a great bias.



Knots represented on the x axis.

Smoothness of the spline: fixed knots

A different strategy may be used by first fixing the knots and then

> impose some restriction on the coefficients such that by changing the restriction we change the level of smoothing.

or

> rather than estimating the coefficient as the minimum of the sum of squares, add a penalization to it which favours smoother functions over wigglier ones.

In this way, the smoothness can be tuned by a number (rather than a vector of unknown length), which is easier to work with.

The second alternative leads to the idea of penalized sum of squares. (Note that it is equivalent to the first for some choices of constraint/penalization.)

Indice

Spline

Penalized likelihood

Why splines

Other basis

More covariates

Generalized models (non gaussian data)

Smoothness and penalization

We consider penalized splines: instead of minimizing

$$\sum_{i=1}^{n}(y_i - f(x_i; \boldsymbol{\beta}, \boldsymbol{b}))^2$$

we minimize

$$\sum_{i=1}^{n}(y_i - f(x_i, \boldsymbol{\beta}, \boldsymbol{b}))^2 + \lambda S(f(x, \boldsymbol{\beta}, \boldsymbol{b}))$$

where

- $S(f(x, \boldsymbol{\beta}, \boldsymbol{b}))$ is a measure of the smoothness of f, increasing as the wiggliness of f increases
- $\lambda > 0$ is a fixed constant (for now).

Penalization: ridge type

Different penalizations may be considered, a convenient one for the above basis is a ridge-type one

$$S(f(x)) = \sum_{i=1}^{K} b_i^2 = \boldsymbol{b}^T \boldsymbol{b}$$

It can be shown that using such a penalization is the same as imposing a constraint

$$\sum_{i=1}^{K} b_i^2 < C$$

for some value of C.

Penalization: second derivative

An alternative penalization, more appropriate if the knots are unequally spaced, and theoretically appealing is

$$S(f(x)) = \int (f^{(q)}(t))^2 dt$$

$q \leq$ spline degree. (Typically, $q = 2$ for a cubic spline.)
Let $\mathrm{B}() = (B_1(), \ldots, B_K())$ be the basis functions, so that

$$\hat{f}(x) = \boldsymbol{b}^T \mathrm{B}(x)$$

then the above penalization is equal to

$$\boldsymbol{b}^T D \boldsymbol{b}$$

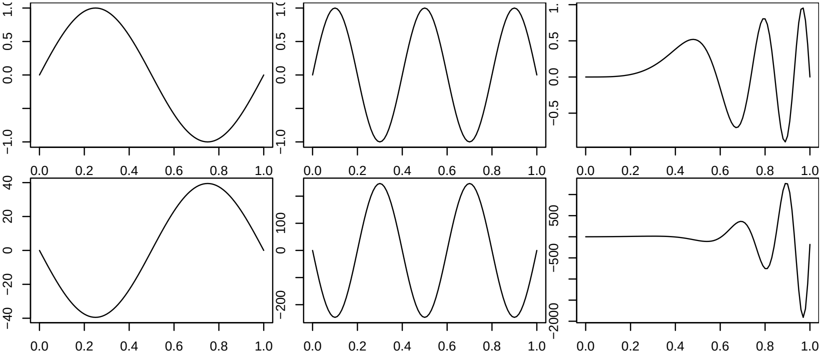
where

$$D = \int_a^b \mathrm{B}^{(q)}(x)[\mathrm{B}^{(q)}(x)]^T dx$$

(Note that approximations are sometimes used.)

Functions and their second derivative

The second derivative, which is typically used for splines of third degree, is easily interpreted since it is a measure of the curvature (although not mathematically precise)



TPB in matrix form

Let us consider the truncated power basis

$$f(x) = \beta_1 + \beta_2 x + \ldots + \beta_{p+1} x^p + \sum_{k=1}^{K} b_k (x - \kappa_k)_+^p$$

and let

$$\boldsymbol{\theta} = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_{p+1} \\ b_1 \\ \vdots \\ b_K \end{bmatrix}, \quad X = \begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & (x_1 - \kappa_1)_+^3 & \ldots & (x_1 - \kappa_K)_+^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_i & x_i^2 & x_i^3 & (x_i - \kappa_1)_+^3 & \ldots & (x_i - \kappa_K)_+^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & x_n^3 & (x_n - \kappa_1)_+^3 & \ldots & (x_n - \kappa_K)_+^3 \end{bmatrix}$$

so that $f(\mathrm{x}) = X\boldsymbol{\theta}$ and the model can be written as

$$\mathrm{y} = X\boldsymbol{\theta} + \varepsilon$$

Penalization and TPB

Let the penalization be the ridge type

$$S(f(x,\boldsymbol{\theta})) = \boldsymbol{\theta}^T D\boldsymbol{\theta}$$

where $D = \text{diag}(0_{p+1}, 1_K)$,

The minimizer of

$$\sum_{i=1}^{n}(y_i - f(x_i,\boldsymbol{\theta}))^2 - \lambda\boldsymbol{\theta}^T D\boldsymbol{\theta}$$

is then

$$\hat{\theta} = (X^T X + \lambda D)^{-1} X^T y$$

So that the spline smoother written as a linear smoother is

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

How many parameters?

The model has $K + p + 1$ parameters (and the variance)

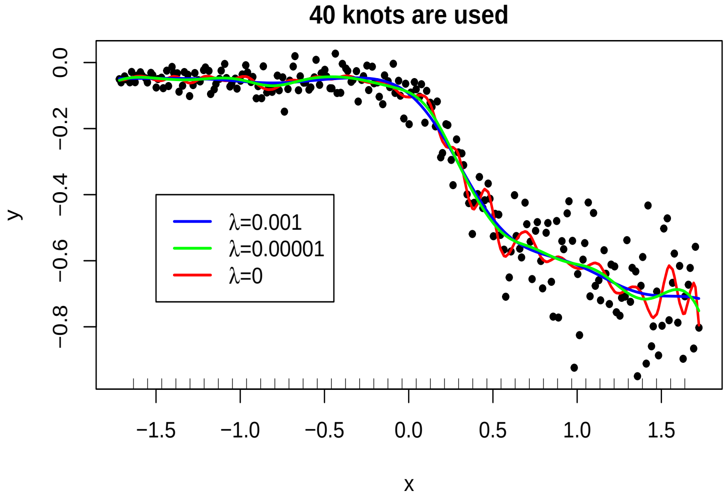
However, they are not free because of the penalization.

The effective number of parameters is given by the trace of the smoothing matrix.

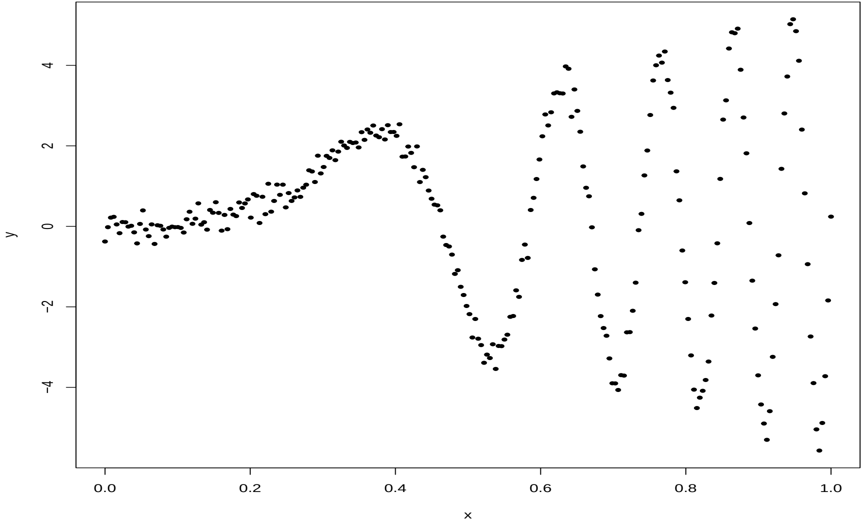
$$\text{trace}(X(X^T X + \lambda D)^{-1} X^T)$$

(Compare with the LM where the number of parameters is the trace of the projection matrix.)

Penalized splines



Unpenalized and penalized splines



In the end, how many knots?

Knots can be fixed in advance, but how are they chosen?

- The idea is that, using the penalization, the choice of the knots does not really matter as long as
  - they are not too few (general recommendation: 20 to 40 knots),
  - there are observations between knots (as a rule of thumb, 4-5 observations minimum)
- One can fix a knot for each observation (smoothing splines), but this can be computationally not convenient and is generally not needed.
- Common strategies for knots choice are
  - Using quantiles of the edf of x
  - equispaced knots.

From now on, the knots $\kappa_1, \ldots, \kappa_K$ are fixed.

Note that we can check if the knots are enough by estimating with more knots and comparing the results.

Smoothing splines v. low rank

In order to understand what we might 'loose' (or not) by using regression splines (few knots) instead of smoothing splines (all knots) consider two smoothers for 20 points

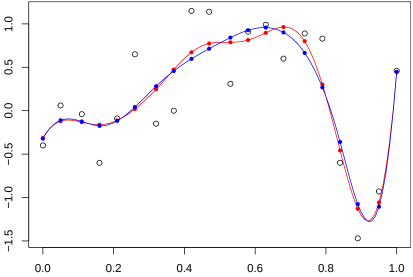
- a smoothing spline (one knot for each x_i) → L_S
- a regression spline with 6 knots (radial basis) → L_R

the two are tuned so that the edf are similar, in particular

$$\text{tr}(L_S) = 8.4174605$$

$$\text{tr}(L_R) = 8.1595772$$

We see they give similar results.



- red: smoothing
- blue: regression

Smoothing splines v. low rank

In order to understand what we might 'loose' (or not) by using regression splines (few knots) instead of smoothing splines (all knots) consider two smoothers for 20 points

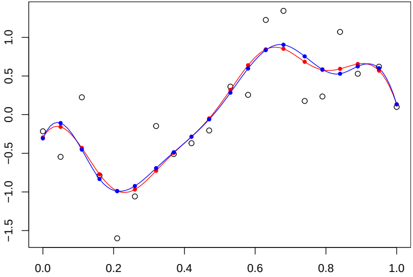
- a smoothing spline (one knot for each x_i) → L_S
- a regression spline with 6 knots (radial basis) → L_R

the two are tuned so that the edf are similar, in particular

$$\mathrm{tr}(L_S) = 8.4174605$$

$$\mathrm{tr}(L_R) = 8.1595772$$

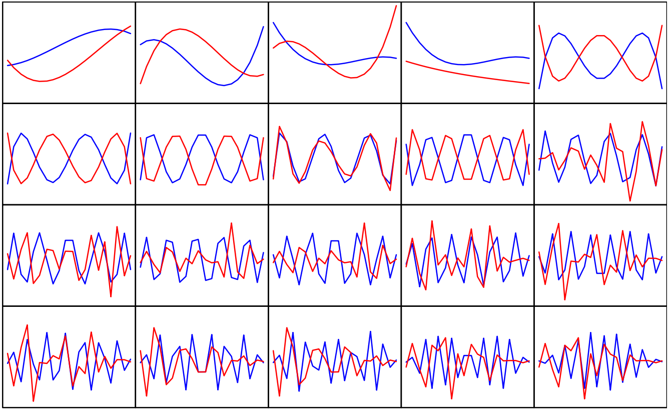
We see they give similar results.



- red: smoothing
- blue: regression

Smoothing splines v. low rank: decomposing L

To appreciate the difference more precisely consider an eigenvalue decomposition of the two matrices L_S and L_R.



Smoothing splines v. low rank: decomposing L

To appreciate the difference more precisely consider an eigenvalue decomposition of the two matrices L_S and L_R.

Note that if λ_1, λ_n and $\mathrm{v}_1, \ldots, \mathrm{v}_n$ are the eigenvalues and eigenvectors of L, then

$$L\mathrm{v}_i = \lambda_i \mathrm{v}_i \quad \forall i$$

being $\mathrm{v}_1, \ldots, \mathrm{v}_n$ a basis for R^n, the respose y can be written as

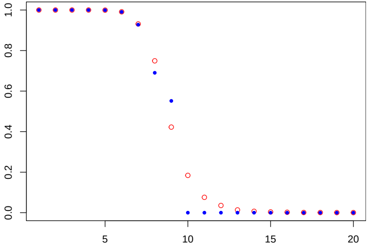
$$\mathrm{y} = \sum_{i=1}^{n} \alpha_i \mathrm{v}_i$$

hence the smoothed $\hat{\mathrm{y}}$ is

$$\hat{\mathrm{y}} = L\mathrm{y} = L\sum_{i=1}^{n} \alpha_i \mathrm{v}_i = \sum_{i=1}^{n} \alpha_i L\mathrm{v}_i = \sum_{i=1}^{n} \alpha_i \lambda_i \mathrm{v}_i$$

Smoothing splines v. low rank: decomposing L

To appreciate the difference more precisely consider an eigenvalue decomposition of the two matrices L_S and L_R.



The eigenvalues of both the smoothing spline (red) and the regression spline (blue) show that only the first 9 eigenvectors count (in both), the following are exactly 0 for the regression spline but are very low for the smoothing spline, thus they lead to similar smooths.

Smoothing splines v. low rank: decomposing L

To appreciate the difference more precisely consider an eigenvalue decomposition of the two matrices L_S and L_R.

Note that the rank of L (the number of non null eigenvalues) is referred to as **rank of the smoother**:

- the smoothing spline, with (approximately) as many basis functions as the data points), is a full rank smoother;
- the regression spline, with considerably less than n basis functions, are a low rank smoother.

(see 3.12 RWC)

Practical: fitting splines without penalization

```
## Create a sample
x=seq(0,1,length=250) #sort(runif(150,0,1)))
m=m=sin(2*pi*x^3) #(0.5+5*x)*sin(10*pi*x^3)
y=m+rnorm(length(x),0,0.4)
##
## compute the covariate matrix
base=function(x,nodi,p=1){
  X=cbind(outer(x,0:p,FUN=function(x,y) x^y),
          outer(x,nodi,FUN=function(x,y) ifelse(x-y>0,(x-y)^p,0)))
}
##
## fix knots and degree
nodi=seq(0.1,0.9,by=0.1)
p=3
X=base(x,nodi,p)
##
## If no penalization is required the fit is obtained through
fit=lm(y~X-1)
yt=X %*% fit$coef
##
## or directly
yt1 = X %*% solve(t(X) %*% X) %*% t(X) %*% y
##
## we can plot the results as
plot(x,y)
rug(nodi)
lines(x,m,col="red")
lines(x,yt,lwd=2)
lines(x,yt1,lwd=2,col="green")
```

Indice

Spline

Penalized likelihood

Choice of λ

Why splines

Other basis

More covariates

Generalized models (non gaussian data)

Choice of λ

The role of λ is analogous to that of the bandwidth in kernel regression and loess.

The same strategy: cross validation, can be used to choose λ.

Note that the spline smoother is a linear smoother, so the results which simplify the formula for computing CV score avoiding repeating estimation of model are valid.

Also, the GCV formula can be used.

Mean summed square error

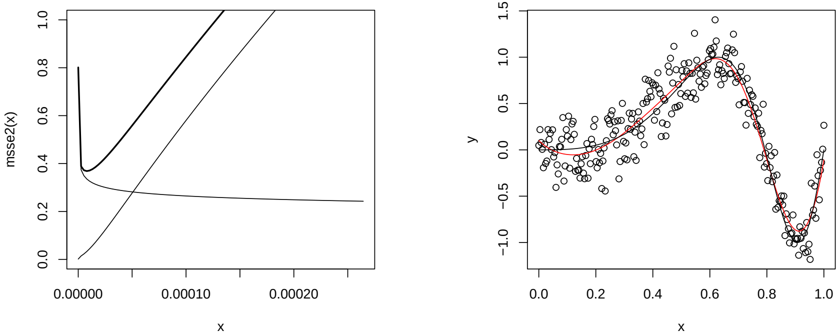
Let

$$
\begin{aligned}
\text{MSSE}(\hat{f}()) &= E\left(\sum_{i=1}^{n}(\hat{f}(x_i) - f(x_i))^2\right) \\
&= \sum_{i=1}^{n}[(E(\hat{f}(x_i)) - f(x_i))^2 + V(\hat{f}(x_i))] \\
&= (E(L\boldsymbol{y}) - \boldsymbol{f})^T(E(L\boldsymbol{y}) - \boldsymbol{f}) + \sum_{i=1}^{n} V(L\boldsymbol{y})_{ii} \\
&= \boldsymbol{f}^T(L-I)^T(L-I)\boldsymbol{f} + \text{trace}[V(L\boldsymbol{y})] \\
&= \boldsymbol{f}^T(L-I)^T(L-I)\boldsymbol{f} + \text{trace}[LV(\boldsymbol{y})L^T] \\
&= \boldsymbol{f}^T(L-I)^T(L-I)\boldsymbol{f} + \sigma_\varepsilon^2\text{trace}[LL^T]
\end{aligned}
$$

In this decomposition, the first part represents the squared bias, the second part is the variance.

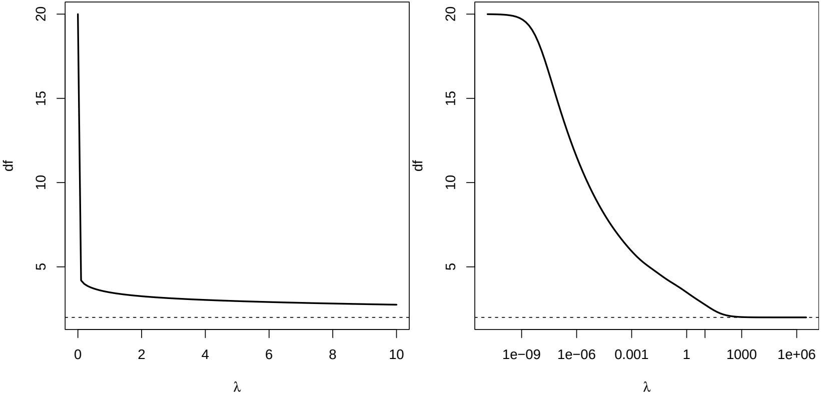
Mean summed square error

Example of MSSE decomposed into bias squared and variance



Degrees of freedom and penalization

The coefficient λ determines the smoothness of the estimated function, it is relevant to look at the relationship between λ and the degrees of freedom of the fit: $\text{df} = \text{tr}(L)$.



Fitting a spline: gam (mgcv)

There are many packages in R to estimate a spline, one of the most powerful and versatile is the package gam by Wood.

The functions provided have many options to tune the estimate, we will look at many of them, first, however, we use it in tis simplest form.

```
fit=gam(y~s(x))
fit.s=summary(fit)
plot(fit)
```

This performs a fit with GCV choice of the smoothing parameter and default choice of knots.

Indice

Spline

Penalized likelihood

Why splines

Other basis

More covariates

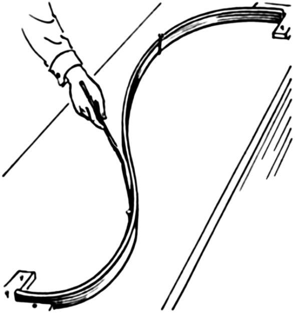
Generalized models (non gaussian data)

Spline

A spline of order k with knots $\kappa_1, \ldots, \kappa_k$ is a continuous piecewise polynomial with continuous derivatives up to order $k-1$.

The term spline is adopted from the name of a flexible strip of metal commonly used by drafters to assist in drawing curved lines.

A **cubic spline** is a spline of order 3 (continuous up to the second derivative).



A spline, or the more modern term flexible curve, consists of a long strip fixed in position at a number of points that relaxes to form and hold a smooth curve passing through those points for the purpose of transferring that curve to another material. (Wikipedia)

Natural cubic splines are best interpolant

Let

- (x_i, y_i), $i = 1, \ldots, n$: assume $x_i < x_{i+1}$
- $g(x)$ be the natural cubic spline interpolating these points (natural means $g''(x_1) = g''(x_n) = 0$)

then $g()$ is the smoothest, in the sense of minimizing

$$J(f) = \int_{x_1}^{x_n} f''(x)^2 dx$$

among the functions f that interpolate the points, are absolutely continuous and have first derivative continuous.

In other words, the natural cubic spline is the smoothest function interpolating the points.

Proof

Let $f()$ interpolate (x_i, y_i) and let $h = f - g$

$$\begin{aligned}
\int_{x_1}^{x_n} f''(x)^2 dx &= \int_{x_1}^{x_n} (g''(x) + h''(x))^2 dx \\
&= \int_{x_1}^{x_n} g''(x)^2 dx + \int_{x_1}^{x_n} g''(x)h''(x) dx + \int_{x_1}^{x_n} h''(x)^2 dx
\end{aligned}$$

Proof

Let $f()$ interpolate (x_i, y_i) and let $h = f - g$

$$\int_{x_1}^{x_n} f''(x)^2 dx = \int_{x_1}^{x_n} g''(x)^2 dx + \int_{x_1}^{x_n} g''(x)h''(x)dx + \int_{x_1}^{x_n} h''(x)^2 dx$$

We also have, integrating by parts

$$\begin{aligned}\int_{x_1}^{x_n} g''(x)h''(x)dx &= g''(x_n)h'(x_n) - g''(x_1)h'(x_1) - \int_{x_1}^{x_n} g'''(x)h'(x)dx \\ &= -\int_{x_1}^{x_n} g'''(x)h'(x)dx \\ &= -\sum_{i=1}^{n-1} g'''(x_i^+) \int_{x_1}^{x_n} h'(x)dx \\ &= -\sum_{i=1}^{n-1} g'''(x_i^+)(h(x_{i+1}) - h(x_i)) = 0\end{aligned}$$

Proof

Let $f()$ interpolate (x_i, y_i) and let $h = f - g$

$$\int_{x_1}^{x_n} f''(x)^2 dx \;=\; \int_{x_1}^{x_n} g''(x)^2 dx + \int_{x_1}^{x_n} g''(x)h''(x)dx + \int_{x_1}^{x_n} h''(x)^2 dx$$

We also have, integrating by parts

$$\int_{x_1}^{x_n} g''(x)h''(x)dx = 0$$

Hence

$$\int_{x_1}^{x_n} f''(x)^2 dx = \int_{x_1}^{x_n} g''(x)^2 dx + \int_{x_1}^{x_n} h''(x)^2 dx \geq \int_{x_1}^{x_n} g''(x)^2 dx$$

where equality holds iff $h''(x) = 0$ for $x_1 < x < x_n$ so only if $f = g$.

Consequence

The above property also means that if we minimize

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int f''(x)^2 dx$$

among all functions f that are continuous with continuous first derivative on $[x_1, \ldots .x_n]$, then the minimum is a natural cubic spline.

Proof: suppose that f^* minimizes the above expression and is not a natural cubic spline, then take the natural cubic spline which interpolates $(x_i, f^*(x_i))$, this realizes the same sum of squares but a lower penalization.

Variance estimation

The variance σ^2 of the error may be estimated, by analogy with LM, as

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^n (y_i - \hat{f}(x_i))^2}{n - \text{df}} = \frac{RSS}{n - \text{df}}$$

We already defined the degrees of freedom of the spline as $\text{tr}(L)$. However, note that

$$\begin{aligned} E(RSS) &= E((\mathrm{y} - \hat{\mathrm{y}})^T(\mathrm{y} - \hat{\mathrm{y}})) \\ &= E(\mathrm{y}^T(L-I)^T(L-I)\mathrm{y}) \\ &= \mathrm{f}^T(L-I)^T(L-I)\mathrm{f} + \sigma^2 \text{tr}((L-I)^T(L-I)) \\ &= \mathrm{f}^T(L-I)^T(L-I)\mathrm{f} + \sigma^2(\text{tr}(LL^T) - 2\text{tr}(L) + n) \end{aligned}$$

thus, assuming the bias is negligible, an unbiased estimator for σ^2 is

$$\tilde{\sigma}^2 = \frac{RSS}{n - 2\text{tr}(L) + \text{tr}(LL^T)}$$

Variance estimation

The variance σ^2 of the error may be estimated, by analogy with LM, as

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^n (y_i - \hat{f}(x_i))^2}{n - \mathrm{df}} = \frac{RSS}{n - \mathrm{df}}$$

We already defined the degrees of freedom of the spline as $\mathrm{tr}(L)$. However, note that

$$\begin{aligned} E(RSS) &= E((\mathrm{y} - \hat{\mathrm{y}})^T(\mathrm{y} - \hat{\mathrm{y}})) \\ &= \mathrm{f}^T(L - I)^T(L - I)\mathrm{f} + \sigma^2(\mathrm{tr}(LL^T) - 2\mathrm{tr}(L) + n) \end{aligned}$$

thus, assuming the bias is negligible, an unbiased estimator for σ^2 is

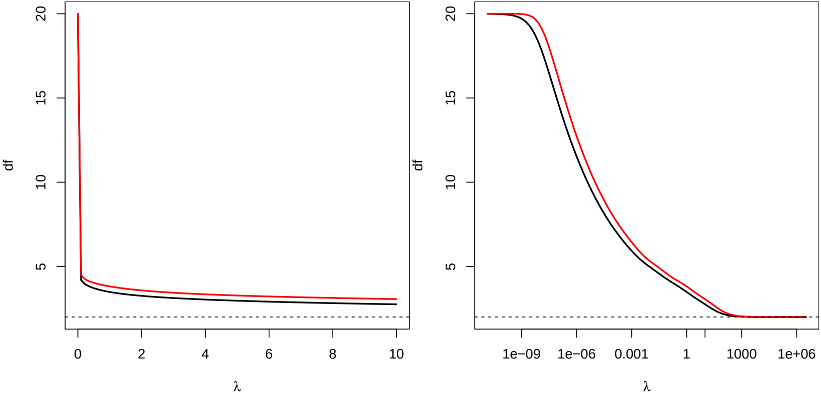
$$\tilde{\sigma}^2 = \frac{RSS}{n - 2\mathrm{tr}(L) + \mathrm{tr}(LL^T)}$$

Note that

- $n - 2\mathrm{tr}(L) + \mathrm{tr}(LL^T)$ are the residual degrees of freedom
- $2\mathrm{tr}(L) - \mathrm{tr}(LL^T)$ is an alternative measure of the dof of the spline

Degrees of freedom: the two versions

The two measures of the degrees of freedom of the smoother are different especially in the mid-range of λ.



Why are the two equal for no smoothing and infinite smoothing?

Indice

Spline

Penalized likelihood

Why splines

Other basis

More covariates

Generalized models (non gaussian data)

Alternative basis

We have introduced the truncated power basis, which is the easier to work with from a theoretical point of view but is quite bad as far as computational properties.

The main problem is that the design matrix X has strongly correlated columns, thus leading to numerical instability (example later in discussing Bayesian estimates).

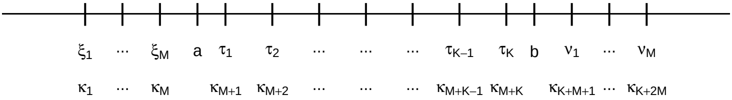
A number of alternative basis exist, namely

- B-splines
- P-splines
- radial basis
- ...

Note that, in principle, a change of basis does not lead to a change of the fit.

B-spline: basis construction

- let $\tau_1 < \ldots < \tau_K$ be the internal nodes;
- let $[a, b]$ be the range on which we are interested ($a < \tau_1$, $b > \tau_K$);
- fix, arbitrarily, $\xi_1 \leq \ldots \xi_M \leq a$ and $\nu_M \geq \ldots \geq \nu_1 \geq b$ (one can set $\xi_i = a$ and $\nu_i = b$);
- we then have a sequence $\kappa_1, \ldots, \kappa_{K+2M}$.


ξ_1 ... ξ_M a τ_1 τ_2 ... ... τ_{K-1} τ_K b ν_1 ... ν_M
κ_1 ... κ_M κ_{M+1} κ_{M+2} ... ... κ_{M+K-1} κ_{M+K} κ_{K+M+1} ... κ_{K+2M}

B-spline: basis construction (continua)

Let then $B_{i,m}$ denote the i-th basis function of order $m < M$, $i = 1, \ldots, K + 2M - m$, this is defined recursively by

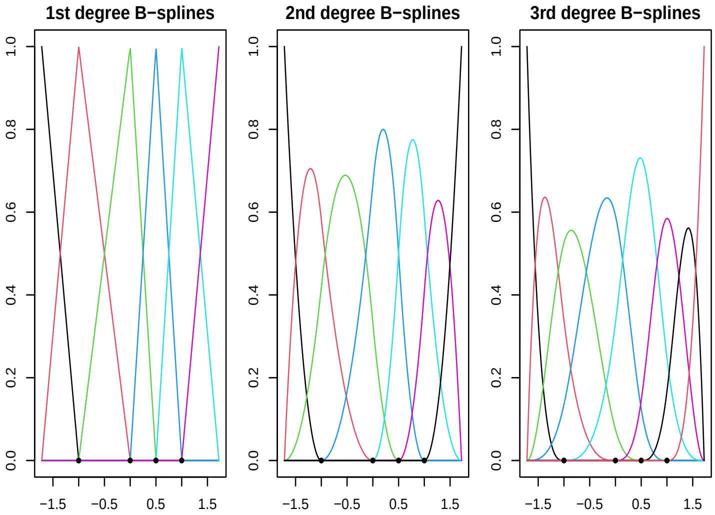
$$B_{i,1}(x) = \begin{cases} 1 & \text{if } \kappa_i \leq x < \kappa_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1, \ldots, K + 2M - 1$ and

$$B_{i,m}(x) = \frac{x - \kappa_i}{\kappa_{i+m-1} - \kappa_i} B_{i,m-1}(x) + \frac{\kappa_{i+m} - x}{\kappa_{i+m} - \kappa_{i+1}} B_{i+1,m-1}(x)$$

$i = 1, \ldots, K + 2M - m$.
For $M = 4$ one obtains $K + 4$ cubic splines.

B-splines


1st degree B−splines
2nd degree B−splines
3rd degree B−splines
0.0
0.2
0.4
0.6
0.8
1.0
−1.5
−0.5
0.5
1.5

B-spline: penalty matrix

Setting $M = 4$, the penalty matrix is defined as

$$\Omega_{ij} = \int_a^b B_i''(x) B_j''(x) dx$$

Wand and Ormerod (2009) obtained formulas for calculating Ω in practice

$$\Omega = (\tilde{B}'')^T \mathrm{diag}(\mathrm{w}) \tilde{B}''$$

where

$$[\tilde{B}'']_{ij} = \tilde{B}_j(\tilde{x}_i) \in \mathcal{M}_{3(K+7)\times(K+4)}$$

$$\tilde{x} = \left(\kappa_1, \frac{\kappa_1 + \kappa_2}{2}, \kappa_2, \ldots, \kappa_{K+7}, \frac{\kappa_{K+7} + \kappa_{K+8}}{2}, \kappa_{k+8}\right)$$

$$\mathrm{w} = \left(\frac{1}{6}(\Delta\kappa)_1, \frac{4}{6}(\Delta\kappa)_1, \frac{1}{6}(\Delta\kappa)_1, \ldots, \frac{1}{6}(\Delta\kappa)_{K+7}, \frac{4}{6}(\Delta\kappa)_{K+7}, \frac{1}{6}(\Delta\kappa)_{K+7}\right)$$

where $(\Delta\kappa)_h = \kappa_{h+1} - \kappa_h$.

P-splines

P-splines are a low rank smoother using

- a *B*-splines basis
- a difference penalty (see below)

Usually they are defined on equally spaced knots (which makes the difference penalty more sensible).

Penalization: difference penalty

The **difference penalty** (joint with B-spline basis, see above, constitutes the P-splines) given by

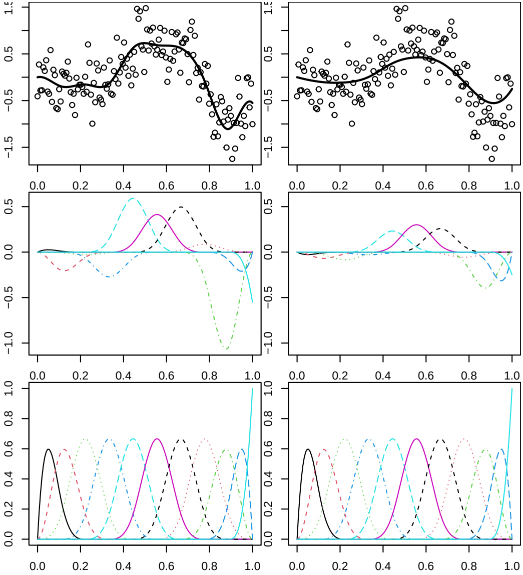
$$\sum_{i=1}^{K-1}(b_{i+1}-b_i)^2$$

that is

$$\boldsymbol{b}^T \begin{bmatrix} 1 & -1 & 0 & \cdot & \cdot & \\ -1 & 2 & -1 & \cdot & \cdot & \\ 0 & -1 & 2 & \cdot & \cdot & \\ & \cdot & \cdot & \cdot & \cdot & \cdot \\ & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \boldsymbol{b}$$

Note that the difference penalty is sensible if the knots are equally spaced.

Effect of difference penalty



Two spline estimates with a B-spline bases, left one has

$$\sum_{i=1}^{K-1}(b_{i+1}-b_i)^2 = 8.29$$

while for the right one

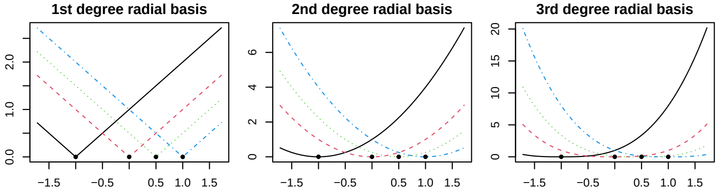
$$\sum_{i=1}^{K-1}(b_{i+1}-b_i)^2 = 0.83$$

(Middle panel: basis functions multiplied by the respective coefficients, that is, the final curve is the sum of these.)

Radial basis

A radial basis of order m with nots $\kappa_1, \ldots, \kappa_K$ is defined as

$$1, x, \ldots, x^m, B_k(x) = |x - \kappa_k|^m$$



The penalty matrix having elements

$$[D]_{ij} = |\kappa_i - \kappa_j|^3$$

Fitting with gam, options for s()

In `gam(y~s(x)` the function `s()` can take arguments, some of them

x is the covariate of the smooth (can have any name!): some types of smooth can have several covariates (e.g. `tp`).

bs is the type of basis-penalty smoother.

k is the basis dimension for the smooth (before imposing any identifiability constraints).

id used to allow different smooths to be forced to use the same basis and smoothing parameter.

sp allows the smoothing parameter to be supplied.

fx if TRUE then the term is unpenalized.

by allows specification of interactions of the smooth with a factor or metric variable.

m specifies the penalty order for some bases.

Built-in bases

Values for the bs argument in `s()`: `s(x,bs="...")`

- "cr" a penalized cubic regression spline ("cc" for cyclic version).
- "ps" Eilers and Marx style P-splines ("cp" for cyclic).
- "ad" adaptive smoothers based on "ps".
- "tp" Optimal low rank approximation to thin plate spline, any dimension and permissable penalty order is possible.

Indice

Spline

Penalized likelihood

Why splines

Other basis

More covariates

Generalized models (non gaussian data)

More covariates

Suppose now that observations involve more covariates

$$x_i, z_i, u_{i1}, \dots, u_{iq}$$

Different models may be considered

- parametric and non parametric component

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f(x_i) + \varepsilon_i$$

- parametric and multiple non parametric component

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f_x(x_i) + f_z(z_i) + \varepsilon_i$$

- parametric and non parametric multivariate component

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f(x_i, z_i) + \varepsilon_i$$

Parametric and non parametric component

Given a representation for the spline f

$$f(x_i) = b_0 + b_1 x_i + \sum_{j=1}^{K} B_j(x_i) b_{1+j}$$

with penalty matrix S, the model

$$y_i = \boldsymbol{\beta}^T \mathrm{u} + f(x_i) + \varepsilon_i$$

is estimated by minimizing the objective function

$$\sum_{i=1}^{n} (y_i - \boldsymbol{\beta}^T \mathrm{u}_i - f(x_i))^2 + \lambda \mathrm{b}^T S \mathrm{b}$$

in matrix form

$$||\boldsymbol{y} - H\boldsymbol{\theta}||^2 + \lambda \boldsymbol{\theta}^T S' \boldsymbol{\theta}$$

where

$$H = [U\ X], \quad \boldsymbol{\theta}^T = (\boldsymbol{\beta}, \mathrm{b}), \quad S' = ?$$

Parametric and multiple non parametric component

Consider a representation for the two splines f_x, f_z

$$f_x(x_i) = b_0 + b_1 x_i + \sum_{j=1}^{K_B} B_j(x_i) b_{1+j}$$

$$f_z(z_i) = d_1 z_i + \sum_{j=1}^{K_D} D_j(z_i) d_{1+j}$$

with penalty matrices S_B, S_D.
Note that the representation for f_z does not involve the intercept to guarantee identifiability of the model

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f_x(x_i) + f_z(z_i) + \varepsilon_i$$

Parametric and multiple non parametric component

The model

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f_x(x_i) + f_z(z_i) + \varepsilon_i$$

is estimated by minimizing the objective function

$$\sum_{i=1}^{n}(y_i - \boldsymbol{\beta}^T \mathrm{u}_i - f_x(x_i) - f_z(z_i))^2 + \lambda_x \mathrm{b}^T S_x \mathrm{b} + \lambda_z \mathrm{d}^T S_z \mathrm{d}$$

in matrix form

$$||\boldsymbol{y} - H\boldsymbol{\theta}||^2 + \lambda_x \mathrm{b}^T S_x \mathrm{b} + \lambda_z \mathrm{d}^T S_z \mathrm{d}$$

where

$$H = [U\ X\ Z], \quad \boldsymbol{\theta}^T = (\boldsymbol{\beta}, \mathrm{b}, \mathrm{d})$$

Non parametric multivariate component

Model

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f(x_i, z_i) + \varepsilon_i$$

requires that we define a bivariate spline.

- in principle this works the same way
- **curse of dimensionality**
  - computational burden can increase exponentially with dimension
  - mean square error, if the sample has size n and the dimension is d then typically

    $$\mathrm{MSE} \approx \frac{c}{n^{4/(4+d)}}$$

    that is, the sample size required to keep the MSE at a specified level δ grows exponentially with d

    $$n \approx (c/\delta)^{d/4}$$

More covariates: syntax in `gam`

Suppose now that observations involve more covariates

$$x_i, z_i, u_{i1}, \ldots, u_{iq} \rightarrow \texttt{x, z, u1, ..., uq}$$

Different models may be considered

- parametric and non parametric component

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f(x_i) + \varepsilon_i \rightarrow \texttt{y\~{}u1+...+uq+s(x)}$$

- parametric and multiple non parametric component

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f_x(x_i) + f_z(z_i) + \varepsilon_i \rightarrow \texttt{y\~{}u1+...+uq+s(x)+s(z)}$$

- parametric and non parametric multivariate component

$$y_i = \boldsymbol{\beta}^T \mathrm{u}_i + f(x_i, z_i) + \varepsilon_i \rightarrow \texttt{y\~{}u1+...+uq+s(x,z)}$$

Interaction with a smooth component: syntax in gam

Suppose that observations involve

- a continuous covariate x
- a qualitative variable (factor) v

we may consider a model with a different smooth function for each value of v

$$y_i = f_{v_i}(x_i) + \varepsilon_i$$

which in gam is specified as

```
y~s(x,by=v)
```

Indice

Truncated power basis

The truncated power representation (of order 1) for a univariate spline (note that there are $K + 2$ parameters)

$$f(x) = \beta_1 + \beta_2 x + \sum_{k=1}^{K} b_k (x - \kappa_k)_+$$

has the natural extension (with $4 + K^{(x)} + K^{(z)} + K^{(x)}K^{(z)}$ parameters)

$$\begin{aligned} f(x,z) = \quad & \beta_1^{(x)} + \beta_2^{(x)} x + \sum_{k=1}^{K^{(x)}} b_k^{(x)} (x - \kappa_k^{(x)})_+ + \\ & \beta_2^{(z)} z + \sum_{k=1}^{K^{(z)}} b_k^{(z)} (z - \kappa_k^{(z)})_+ + \\ & \beta_2^{(xz)} xz + \sum_{k=1}^{K^{(x)}} \sum_{k=1}^{K^{(z)}} b_k^{(xz)} (x - \kappa_k^{(x)})_+ (z - \kappa_k^{(z)})_+ \end{aligned}$$

Radial basis

Given the knots $\{\kappa_k^{(x)}; k = 1, \dots, K^{(x)}\}$ and $\{\kappa_k^{(z)}; k = 1, \dots, K^{(z)}\}$ a radial basis function has the form

$$C\left(\left\|\begin{bmatrix} x \\ z \end{bmatrix} - \begin{bmatrix} \kappa_k^{(x)} \\ \kappa_{k'}^{(z)} \end{bmatrix}\right\|\right)$$

for $k = 1, \dots, K^{(x)}$ and $k' = 1, \dots, K^{(z)}$ and for some function $C : \mathbb{R} \to \mathbb{R}^+$.

The radial basis has the property of being rotationally invariant.

Thin plate splines

Consider observations (y_i, x_i), $\mathrm{x}_i \in \mathbb{R}^d$ and the model

$$y_i = f(\mathrm{x}_i) + \varepsilon_i$$

thin plate splines are defined as the function f that minimizes

$$\|\mathrm{y} - \mathrm{f}\| + \lambda J_{md}(f)$$

where

$$J_{md}(f) = \int \dots \int_{\mathbb{R}^d} \sum_{v_1+\dots+v_d=m} \frac{m!}{v_1! \dots v_d!} \left(\frac{\partial^m f}{\partial x_1^{v_1} \dots \partial x_d^{v_d}} \right)^2 dx_1 \dots dx_d$$

Thin plate splines

Consider observations (y_i, x_i), $\mathrm{x}_i \in \mathbb{R}^d$ and the model

$$y_i = f(\mathrm{x}_i) + \varepsilon_i$$

thin plate splines are defined as the function f that minimizes

$$\|\mathrm{y} - \mathrm{f}\| + \lambda J_{md}(f)$$

where

$$J_{md}(f) = \int \dots \int_{\mathbb{R}^d} \sum_{v_1 + \dots + v_d = m} \frac{m!}{v_1! \dots v_d!} \left(\frac{\partial^m f}{\partial x_1^{v_1} \dots \partial x_d^{v_d}} \right)^2 dx_1 \dots dx_d$$

In the $d = 2$ case

$$J_{22} = \int \int_{\mathbb{R}^2} \left(\frac{\partial^2 f}{\partial x_1^2} \right)^2 + \left(\frac{\partial^2 f}{\partial x_1 \partial x_2} \right)^2 + \left(\frac{\partial^2 f}{\partial x_2^2} \right)^2 dx_1 dx_2$$

Thin plate splines solution

It can be shown that the minimizer is of the form

$$f(\mathrm{x}) = \sum_{i=1}^{n} \delta_i \eta_{md}(\|\mathrm{x} - \mathrm{x}_i\|) + \sum_{j=1}^{M} \alpha_j \phi_j(\mathrm{x})$$

where

- the ϕ_j are l.i. and span the space where J_{md} is zero
- $T^T \boldsymbol{\delta} = 0$ where $T_{ij} = \phi_j(\mathrm{x}_i)$
- $\eta_{md}(r) = \begin{cases} \frac{(-1)^{m+1+d/2}}{2^{2m-1}\pi^{d/2}(m-1)!(m-d/2)!} r^{2m-d} \log(r) & d \text{ even} \\ \frac{\Gamma(d/2-m)}{2^{2m}\pi^{d/2}(m-1)!} r^{2m-d} & d \text{ odd} \end{cases}$

and $\boldsymbol{\alpha}, \boldsymbol{\delta}$ are the solution of

$$\min \|\mathrm{y} - E\boldsymbol{\delta} - T\boldsymbol{\alpha}\|^2 + \lambda \boldsymbol{\delta}^T E \boldsymbol{\delta} \quad \text{s.t.} T^T \boldsymbol{\delta} = 0$$

where $E_{ij} = \eta_{md}(\|\mathrm{x}_i - \mathrm{x}_j\|)$.

Thin plate regression splines

The above basis has dimension n, low rank thin plate regression splines can be obtained in two ways

- making an eigen decomposition of E and keeping the first eigenvalues (note that an approximate decomposition must be used to keep tprs computationally convenient)

Thin plate regression splines

The above basis has dimension n, low rank thin plate regression splines can be obtained in two ways

- knot based approximations: we let

$$f(\mathrm{x}) = \sum_{i=1}^{K} \delta_i \eta_{md}(\|\mathrm{x} - \boldsymbol{\kappa}_i\|) + \sum_{j=1}^{M} \alpha_j \phi_j(\mathrm{x})$$

and minimize

$$\|\mathrm{y} - X\boldsymbol{\beta}\| + \lambda \boldsymbol{\beta}^T S \boldsymbol{\beta} \quad \text{s.t.} C\boldsymbol{\beta} = 0$$

where

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\delta} \\ \boldsymbol{\alpha} \end{bmatrix} \quad X_{ij} = \begin{cases} \eta_{md}(\|\mathrm{x} - \boldsymbol{\kappa}_j\|) & j \leq K \\ \phi_{j-k}(x_i) & j > K \end{cases}$$

$$C_{ij} = \begin{cases} \phi_i(\boldsymbol{\kappa}_j) & j \leq K \\ 0 & j > K \end{cases} \quad S_{ij} = \begin{cases} \eta_{md}(\|\mathrm{x} - \boldsymbol{\kappa}_j\|) & j \leq K \\ 0 & j > K \end{cases}$$

Properties of thin plate regression splines

Properties of thin plate regression splines

- avoid the problem of knot placement if the first option is used
- cheap to compute
- defined for any dimension
- rotationally invariant (isotropy)
- approximate thin plate splines (which are optimal in the sense given above)

Note that the penalization is such that wiggliness in any direction is weighted the same.

This may be a desirable property whenever the covariates have the same weight, for instance when they are geographic coordinates.

If this is not the case then standardization is usually performed.

Tensor product basis

The tensor product basis is a way to build a multivariate basis from univariate ones, suppose that

$$f_x(x) = \sum_{i=1}^{K^{(x)}} \beta_i^{(x)} B_i^{(x)}(x)$$

and analogous for $f_z(z)$ and $f_v(v)$.

A smooth function of (x, z) can be obtained by **letting f_x to vary smoothly with** z, that is, we let the coefficients $\beta^{(x)}$ vary smoothly with z by defining

$$\beta_i^{(x)}(z) = \sum_{j=1}^{K^{(z)}} \beta_{ij}^{(xz)} B_j^{(z)}(z)$$

so that

$$f_{xz}(x, z) = \sum_{i=1}^{K^{(x)}} \sum_{j=1}^{K^{(z)}} \beta_{ij}^{(xz)} B_j^{(z)}(z) B_i^{(x)}(x)$$

Tensor product basis

The tensor product basis is a way to build a multivariate basis from univariate ones, suppose that

$$f_x(x) = \sum_{i=1}^{K^{(x)}} \beta_i^{(x)} B_i^{(x)}(x)$$

and analogous for $f_z(z)$ and $f_v(v)$.
Proceeding the same way one obtains

$$f_{xzv}(x, z, v) = \sum_{i=1}^{K^{(x)}} \sum_{j=1}^{K^{(z)}} \sum_{k=1}^{K^{(v)}} \beta_{ijk}^{(xzv)} B_k^{(v)}(v) B_j^{(z)}(z) B_i^{(x)}(x)$$

where if X_x, X_y, X_z are the matrices containing the basis functions evaluated at observations for each of the univariate spline, then the matrix corresponding to the tensor product basis is their Kronecker product

$$X = X_x \otimes X_y \otimes X_v$$

Penalty for tensor product basis

If $J(f_\bullet)$ represent the univariate penalty, then it is natural to choose as a penalty for f_{xzv}

$$J(f) = \lambda_x \int J_x(f_{x|zv})dzdv + \lambda_z \int J_z(f_{z|xv})dxdv + \lambda_v \int J_v(f_{v|xz})dxdz$$

for example with the usual square of second derivative penalty one would get

$$J(f) = \int \lambda_x \left(\frac{\partial^2 f}{\partial x^2}\right)^2 + \lambda_z \left(\frac{\partial^2 f}{\partial z^2}\right)^2 + \lambda_v \left(\frac{\partial^2 f}{\partial v^2}\right)^2 dxdzdv$$

It can be shown that this can be written as a quadratic form (sometimes approximation may be used since this involve numerical integrations)

Multivariate spline in `gam`

A multivariate spline is specified as either an

- isotropic spline `s(x,z)` appropriate if the variables are on the same scale
- tensor product `te(x,z,bs=c("cr","cc"),d=c(2,1),k=(20,5))`, so that the smooth has a different penalty for each marginal basis, which can be specified

Indice

Spline

Penalized likelihood

Why splines

Other basis

More covariates

Generalized models (non gaussian data)

Generalized additive models

The generalization to non gaussian data works pretty similarly as the extension of lm to glm.

$$\begin{array}{ccc}
\text{LM} & \rightarrow & \text{GLM} \\
Y_i \sim \mathcal{N}(\mu_i, \sigma^2) & & Y_i \sim \text{Expon}(\theta_i, \phi_i) \\
E(Y_i) = \eta_i & & g(E(Y_i)) = \eta_i \\
\eta_i = \mu_i = \mathrm{x}_i\boldsymbol{\beta} & & \eta_i = \mathrm{x}_i\boldsymbol{\beta}
\end{array}$$

$$\begin{array}{ccc}
\text{AM} & \rightarrow & \text{GAM} \\
Y_i \sim \mathcal{N}(\mu_i, \sigma^2) & & Y_i \sim \text{Expon}(\theta_i, \phi_i) \\
E(Y_i) = \eta_i & & g(E(Y_i)) = \eta_i \\
\eta_i = \mu_i = f(x_i) & & \eta_i = f(x_i)
\end{array}$$

where

$$\ell(\beta, \mathrm{b}, \phi) = \sum_{i=1}^{n} \log(p(y_i; \theta_i)) = \sum_{i=1}^{n} (y_i\theta_i - r_i(\theta_i)))/\phi + c(\phi; y_i)$$

Generalized additive models

The generalization to non gaussian data works pretty similarly as the extension of lm to glm. Given a representation for the spline

$$f(x) = \beta_1 + \beta_2 x + \sum_{j=1}^{K} \beta_{1+j} B_j(x)$$

the penalized least squares criterion

$$\sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda S(f(x))$$

is substituted by the penalized likelihood

$$\ell(\beta, \mathrm{b}, \phi) - \lambda S(f(x))$$

where

$$\ell(\beta, \mathrm{b}, \phi) = \sum_{i=1}^{n} \log(p(y_i; \theta_i)) = \sum_{i=1}^{n} (y_i \theta_i - r_i(\theta_i)))/\phi + c(\phi; y_i)$$

P-IRLS

In order to fit a GLM one uses the IRLS algorithm, whose k-th step is

1 compute pseudodata

$$z_i^{[k]} = g'(\mu_i^{[k]})(y_i - \mu_i^{[k]}) + \eta_i^{[k]}$$

and the diagonal weighting matrix

$$W_{ii}^{[k]} = \frac{1}{V(\mu_i^{[k]})g'(\mu_i^{[k]})^2}$$

2 set

$$\boldsymbol{\beta}^{[k+1]} \leftarrow \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\| \sqrt{W^{[k]}}(\mathrm{z}^{[k]} - X\boldsymbol{\beta}) \right\|^2$$

In an AM the objective function in the second step is replaced by

$$\boldsymbol{\beta}^{[k+1]} \leftarrow \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\| \sqrt{W^{[k]}}(\mathrm{z}^{[k]} - X\boldsymbol{\beta}) \right\|^2 + \lambda \boldsymbol{\beta}^T S \boldsymbol{\beta}$$

Non gaussian models in gam

We specify it as `gam(y x,family=...)`, possible values for family are

- gaussian (default) is useful for real valued response data.
- Gamma is useful for strictly positive real valued data.
- poisson is useful when the response is count data of some sort.
- binomial
- inverse.gaussian is for strictly positive real response variables: useful for various time to event data.
- quasi does not define a full distribution, but allows inference when only the mean variance relationship can be well approximated. quasipoisson and quasibinomial are special cases. Not useable with likelihood based smoothness selection.
- Tweedie
- negbin is useful for overdispersed count data, but computation is slow.

Inference

We can employ a Bayesian approach, assume that the penalization can be written as

$$S(f(x)) = \theta^T S\theta$$

for some matrix S where $\theta = (\beta, b)$, then the likelihood is

$$\ell(\theta, \phi) - \lambda\theta^T S\theta$$

which is proportional to a Bayesian posterior assuming the following prior

$$\pi(\theta) \propto \exp\left(-\frac{1}{2}\lambda\theta^T S\theta\right)$$

then approximately (for large samples) the posterior is

$$\beta|y \sim \mathcal{N}\left(\hat{\beta}, (X^T WX + \lambda S)^{-1}\phi\right)$$

which can be used to obtain credibility intervals with good frequentist properties (see Wood)

Indice

Spline

Penalized likelihood

Why splines

Other basis

More covariates

Generalized models (non gaussian data)

Choice of λ with non Gaussian data

Smoothness selection,

The following approaches can be used

- minimize prediction error
  - to use GCV for unknown scale parameter and
  - Mallows' Cp/UBRE/AIC for known scale
- treat smooths as random effect in a Bayesian model and estimate λ through marginal likelihood
  - REML estimation, including of unknown scale
  - ML estimation, including of unknown scale
- estimate a full Bayesian model

Smoothness selection,

The following approaches can be used → `gam(method=...)`

- minimize prediction error → `method=GCV.Cp`
  - to use GCV for unknown scale parameter and
  - Mallows' Cp/UBRE/AIC for known scale
- treat smooths as random effect in a Bayesian model and estimate λ through marginal likelihood
  - REML estimation, including of unknown scale → `method=REML`
  - ML estimation, including of unknown scale → `method=ML`
- estimate a full Bayesian model

Prediction error with non Gaussian data

With Gaussian data, choice of λ has been based on estimation of the error

$$\sum_{i=1}^{n}(\hat{f}(x_i) - y_i)^2$$

performed by CV which leads to the GCV criterion because of the linearity of the splines as smoothers.

The same strategy can in principle be used with non Gaussian data, **but** the smoother is not a linear smoother for the y_i anymore, hence GCV is not available and also theoretical derivations do not apply.

GCV score for GAM

The GAM fitting objective can be written in terms of the deviance

$$D(\boldsymbol{\beta}) = 2(\ell(\boldsymbol{\beta}_{\text{max}}) - \ell(\boldsymbol{\beta}))$$

as

$$D(\boldsymbol{\beta}) + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

whose quadratic approximation is, for a fixed λ,

$$\left\| \sqrt{W}(\text{z} - X\boldsymbol{\beta}) \right\|^2 + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

GCV score for GAM

The GAM fitting objective can be written as

$$D(\boldsymbol{\beta}) + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

whose quadratic approximation is, for a fixed λ,

$$\left\| \sqrt{W}(\mathrm{z} - X\boldsymbol{\beta}) \right\|^2 + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

from which one could compute the GCV score (valid locally)

$$\frac{n \left\| \sqrt{W}(\mathrm{z} - X\boldsymbol{\beta}) \right\|^2}{n - \mathrm{tr}(L)}$$

GCV score for GAM

The GAM fitting objective can be written as

$$D(\boldsymbol{\beta}) + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

whose quadratic approximation is, for a fixed λ,

$$\left\| \sqrt{W}(\mathrm{z} - X\boldsymbol{\beta}) \right\|^2 + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

from which one could compute the GCV score (valid locally) and then the globally applicable GCV score

$$\frac{n \left\| \sqrt{W}(\mathrm{z} - X\boldsymbol{\beta}) \right\|^2}{n - \mathrm{tr}(L)} \quad \rightarrow \quad \frac{nD(\hat{\beta})}{n - \mathrm{tr}(L)}$$

UBRE

Recall that the idea of CV arises from estimation of the predictive risk, $E((m(x)-\hat{m}(x))^2)$, that is

$$E\left(\|\boldsymbol{\mu}-L\mathrm{y}\|^2\right)=\frac{1}{n}E\left(\|\mathrm{y}-L\mathrm{y}\|^2\right)-\sigma^2+2\mathrm{tr}(L)\frac{\sigma^2}{n}$$

when the scale parameter σ^2 is known this can be done by minimizing the UBRE (Unbiased Risk Estimator)

$$\frac{1}{n}\|\mathrm{y}-L\mathrm{y}\|^2-\sigma^2+2\mathrm{tr}(L)\frac{\sigma^2}{n}$$

which is equal to Mallow's C_p.

The GCV arises as an alternative in the Gaussian case since typically σ^2 must be estimated and if this is the case then the above criterion is not suitable.

UBRE is appropriate for GAM where the scale parameter is known.

UBRE computation

Based again on

$$D(\boldsymbol{\beta}) + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

and the quadratic approximation

$$\left\| \sqrt{W}(\mathrm{z} - X\boldsymbol{\beta}) \right\|^2 + \sum_{j=1}^{d} \lambda_j \boldsymbol{\beta}^T S_j \boldsymbol{\beta}$$

we obtain the UBRE criterion

$$\frac{1}{n} \left\| \sqrt{W}(\mathrm{z} - X\boldsymbol{\beta}) \right\|^2 - \sigma^2 + \frac{2\sigma^2}{n} \mathrm{tr}(L)$$

$$\frac{1}{n} D(\hat{\beta}) - \sigma^2 + \frac{2\sigma^2}{n} \mathrm{tr}(L)$$