Advanced Bayesian Methods: Theory and Applications in R

Penalized Spline Smoothing

Nikolaus Umlauf
https://nikum.org/abm.html

Linear Models

- The work horse of statistical modelling and analysis is the linear model where

$$y_i = \mathbf{x}_i^\top \boldsymbol{\beta} = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_k x_{ik} + \varepsilon_i, \quad \varepsilon_i \overset{i.i.d.}{\sim} N(0, \sigma^2).$$

- The parameters β_j can be related to the expected change in the response associated with differences in x_j.
  $\Rightarrow$ Parameters have a specific meaning and purpose.
- Statistical inference is facilitated by the distributional assumptions on the error terms.
- However, in many practical situations the linear model is not flexible enough and/or assumptions may be questionable.

Nonlinear Effects

- Common practice if the linearity of the effect of x_j is questionable: Include low-order polynomials, e.g. replace $x_j\beta_j$ by

$$x_j\beta_j + x_j^2\beta_{j+1} + x_j^3\beta_{j+2}.$$

- Imposes strong assumptions on the form of the effect and is not very flexible.
- Ideally, the form of an effect should be left unspecified and should be determined by the data (under mild, qualitative assumptions).
- Additive model:

$$y_i = \beta_0 + f_1(x_{i1}) + \ldots + f_k(x_{ik}) + \varepsilon_i, \quad \varepsilon_i \overset{i.i.d.}{\sim} N(0, \sigma^2).$$

- We will use penalized splines to represent the effects $f_j(x_{ij})$.

Clustered Data

- For longitudinal data $(y_{it}, \mathbf{x}_{it})$, $i = 1 \ldots, n$, $t = 1, \ldots, T$, a classical model of the form

$$y_{it} = \mathbf{x}_{it}^\top \boldsymbol{\beta} + \varepsilon_{it}$$

  may be questionable for a number of reasons:

  - Unobserved heterogeneity due to individual-specific, unobserved confounders that have not been included in the model,
  - Dependence between observations on one individual, or
  - Individual-specific regression coefficients.

Clustered Data

- Similarly applies to other grouping structures (families, geographical regions, school classes, . . .)
- Random effects models are then often considered, e.g. random intercepts

$$y_{it} = \gamma_{0i} + \mathbf{x}_{it}^{\top}\boldsymbol{\beta} + \varepsilon_{it}$$

  with $\gamma_{i0} \overset{i.i.d.}{\sim} N(0, \tau^2)$.

- More complex models can also have individual-specific random slopes or other additional structures.

Spatial Dependence

- For spatial regression data $(y(s), (s))$, one may similarly question whether linear models take unobserved spatial heterogeneity and/or dependence into account.

- Include spatially correlated random effects, leading to

$$y(s) = \gamma(s) + \mathbf{x}(s)^\top \boldsymbol{\beta} + \varepsilon(s)$$

  with $\gamma(s)$ being an appropriately specified spatial stochastic process.

Bayesian Additive Regression

- Bayesian additive regression provides a unifying framework for dealing with the challenges discussed so far.
- The model also supports other effect types, e.g., varying coefficients or interaction surfaces.
- The models can be conveniently represented in a hierarchical fashion that enables us to benefit from the flexibility of Bayesian inference.
- Tomorrow, we will discuss Bayesian distributional regression that allows us to overcome the normality assumption for the error terms.

Example

Car insurance data from two insurance companies in Belgium:

- Sample of approximately 160.000 policyholders.
- Aims: Separate risk analyses for claim size and claim frequency to predict risk premium from covariates.
- Variables of primary interest: Claim size $\texttt{amount}_i$ or claim frequency $\texttt{nclaims}_i$ of policyholders.

Example

Variable	Description
`agec`	Vehicle's age.
`ageph`	Policyholder's age.
`power`	Vehicle's horsepower.
`bm`	Bonus-malus score.
`region`, `NAME_4`	District in Belgium.
`lon`, `lat`	Longitude/Latitude coordinates of districts.
`fleet`	Vehicle belongs to a fleet (`"yes"`, `"no"`).
`sex`	Gender of the policy holder (`"male"`, `"female"`).
`coverage`	Possible other guarantees subscribed, 1 = TPL only, 2 = limited material damage or theft in addition to TPL, 3 = comprehensive coverage in addition to TPL.

Example

- Generalised linear models:
  - Gaussian model for log-costs $\log(\texttt{amount})$:

$$\log(\texttt{amount}) \sim N(\mathbf{x}^\top \boldsymbol{\beta}, \sigma^2).$$

  - Poisson model for frequencies $\texttt{nclaims}_i$:

$$\texttt{nclaims} \sim Po(\exp(\mathbf{x}^\top \boldsymbol{\beta})).$$

  - Linear predictors formed as a linear combination of (possibly transformed) covariates:

$$\eta = \mathbf{x}^\top \boldsymbol{\beta} = \beta_0 + x_1\beta_1 + \ldots + x_p\beta_p.$$

Example

- Subject-matter knowledge:
  - Young and old drivers have a higher claims expenditure. This hints at a quadratic instead of a linear age effect, but the precise form is unknown.

    $\Rightarrow$ Replace the parametric effect with a nonparametric effect $f(\texttt{ageph})$.
  - Male and female drivers have a different claims expenditure. This hints at an interaction between age and gender, but the effect should be allowed to vary with age.

    $\Rightarrow$ Instead of a parametric model of the form $\beta_1\texttt{ageph} + \beta_2\texttt{sex} + \beta_3 ageph \cdot \texttt{sex}$ consider a model of the form $f_1(\texttt{ageph}) + f_2(\texttt{ageph}) \cdot \texttt{sex}$.

Example

- Drivers in rural areas cause less accidents with a higher average claim amount while drivers in urban areas cause more but smaller claims. The effect may change smoothly between rural and urban areas such that modeling based on a rural vs. urban dummy is too simplistic.

  $\Rightarrow$ Include a spatial function $f_{spat}(\texttt{NAME\_4})$ based on the region `NAME_4` a driver is living in.

Example

- Model specifications:
  - Gaussian model for log-costs $\log(\texttt{amount})$:

    $$\log(\texttt{amount}) \sim N(\eta, \sigma^2)$$

    with

    $$\eta = f_1(\texttt{agec}) + f_2(\texttt{ageph}) + f_3(\texttt{bm}) + f_4(\texttt{power}) + f_{spat}(\texttt{NAME\_4}) + \mathbf{x}^\top \boldsymbol{\beta}.$$

  - Poisson model for frequencies $\texttt{nclaims}_i$:

    $$\texttt{nclaims} \sim Po(\exp(\eta))$$

    with

    $$\eta = f_1(\texttt{agec}) + f_2(\texttt{ageph}) \cdot \texttt{sex} + f_3(\texttt{bm}) + f_4(\texttt{power}) + f_{spat}(\texttt{NAME\_4}) + \mathbf{x}^\top \boldsymbol{\beta}.$$

Scatterplot Smoothing

- Start from scatterplot smoothing

$$y_i = f(z_i) + \varepsilon_i, \quad \varepsilon_i \stackrel{\text{i.i.d.}}{\sim} \mathrm{N}(0, \sigma^2)$$

  where $f(z)$ should be inferred based on observations (z_i, y_i), $i = 1, \ldots, n$, for a continuous covariate z and response y.
- Common approach: Approximate $f(z)$ by a low-order polynomial

$$f(z_i) = \gamma_0 + \gamma_1 z_i + \ldots + \gamma_l z_i^l$$

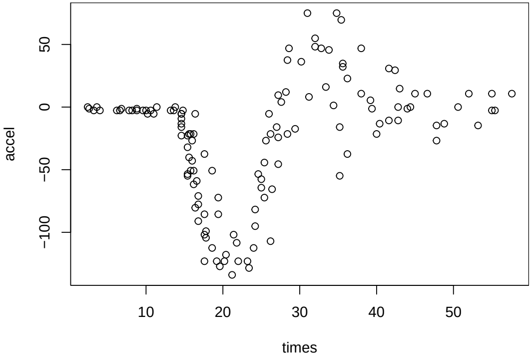
  since any smooth function $f(\cdot)$ can be approximated arbitrarily accurately if the degree l is chosen large enough.

Scatterplot Smoothing

- In statistics, the problem of estimating the coefficients $\gamma_0, \ldots, \gamma_l$ limits the applicability of high polynomial degrees:

Scatterplot Smoothing

```
R> data("mcycle", package = "MASS")
R> par(mar = c(4, 4, 0, 0))
R> plot(mcycle)
```



Polynom Splines

- If a linear fit is too simple we could use a polynomial model

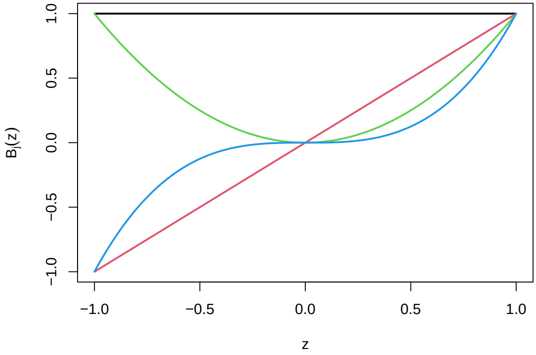
$$\texttt{accel}_i = \gamma_0 + \gamma_1 \texttt{times}_i + \ldots + \gamma_l \texttt{times}_i^l.$$

- The parameters can be estimated by ordinary least squares.
- Note that we write γ instead of β to better distinguish between simple linear and nonlinear effects here.
- The design matrix has the following form

$$\mathbf{Z} = \begin{pmatrix} 1 & \texttt{times}_1 & \ldots & \texttt{times}_1^l \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \texttt{times}_n & \ldots & \texttt{times}_n^l \end{pmatrix}$$

Polynom Splines

- The columns of Z are also called basis functions $B_j(z)$, $j = 0, \ldots, l$. In this case a polynomial basis.
- With sorted z they have a nice visual representation.



Polynom Splines

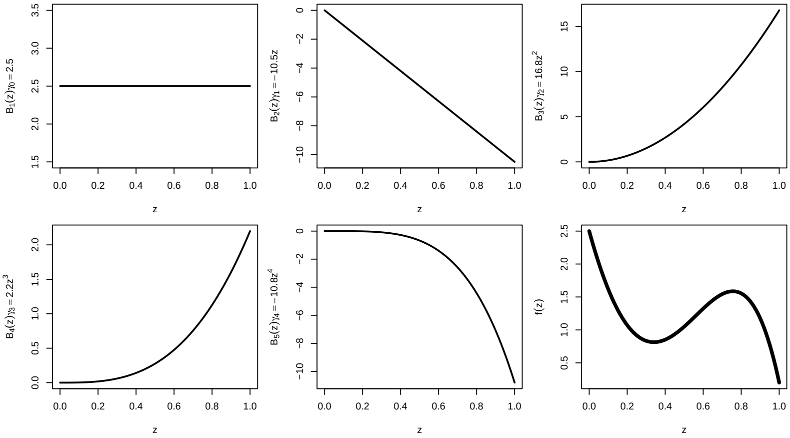
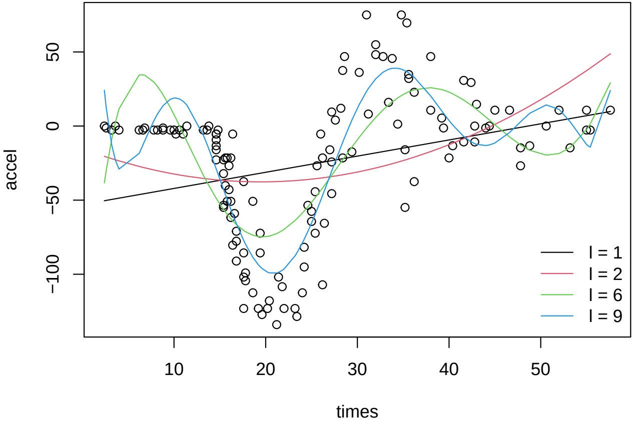


Illustration of how $f(z)$ is represented in terms of $B_j(z)\gamma_j$.

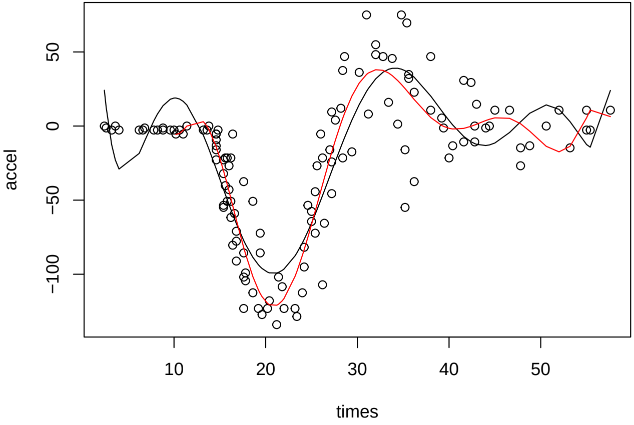
Polynom Splines

Effect of increasing the degree *l* of the polynomial.



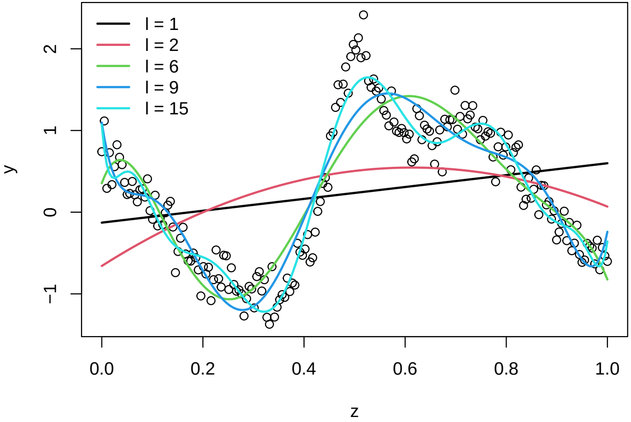
Polynom Splines

Polynomial boundary effects.



Polynom Splines

Effect of increasing the degree *l* of the polynomial.



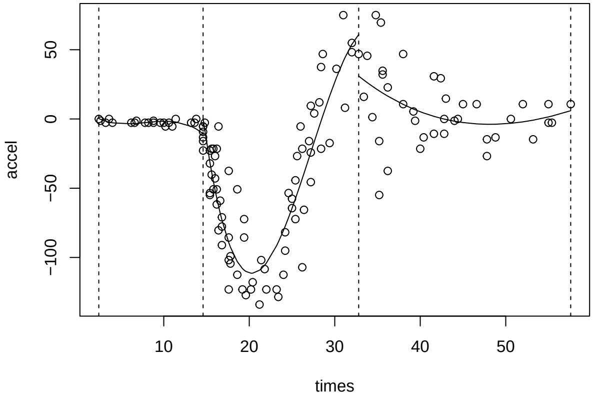
Polynom Splines

Problems:

- High degree needed for decent curve fit.
- Higher degree polynomials are numerically unstable.
- Basis funtions are global.
- Unexpected wiggles.
- Round-off problems with $\hat{\gamma} = (\mathbf{Z}^\top \mathbf{Z})^{-1} \mathbf{Z}^\top \mathbf{y}$.
- Partial remedy: center and normalize z.
- Better use orthogonal polynomials instead, see also function `poly()`.

Polynom Splines

Divide the range of z in equidistant intervals with boundaries $\boldsymbol{\kappa}$ (knots) and fit polynomial models within each section.



Polynom Splines

- The fitted functions don't form a nice overall smooth function, see the jumps at the boundaries.
- We need additional requirements to construct a smooth functional form.

A function f is called polynomial spline of degree $l \geq 0$ with knots $min(z) = \kappa_1 < \ldots < \kappa_m = max(z)$ (the interval boundaries), if it satisfies

1. $f(z)$ is $(l-1)$ times continuously differentiable,
2. $f(z)$ is a polynom of degree l in each interval $[\kappa_j, \kappa_{j+1})$.

Every spline may be represented by a linear combination of basis functions, i.e.

$$f(z_i) = \gamma_1 \cdot B_1(z_i) + \gamma_2 \cdot B_2(z_i) + \ldots + \gamma_{l+m-1} \cdot B_{l+m-1}(z_i).$$

Polynom Splines

Polynom splines with truncated powers

- Regression model

$$y_i = \gamma_1 + \gamma_2 z_i + \ldots + \gamma_{l+1} z_i^l + \sum_{j=2}^{m-1} \gamma_{l+j} (z_i - \kappa_j)_+^l + \varepsilon_i.$$

  where

$$(z_i - \kappa_j)_+^l = \begin{cases} (z_i - \kappa_j)^l & z_i \geq \kappa_j \\ 0 & \text{else.} \end{cases}$$

- Corresponding basis functions

$$B_1(z_i) = 1, \quad B_2(z_i) = z_i, \quad \ldots \quad , \quad B_{l+1}(z_i) = z_i^l,$$
$$B_{l+2}(z_i) = (z_i - \kappa_2)_+^l, \quad \ldots \quad , \quad B_k(z_i) = (z_i - \kappa_{m-1})_+^l.$$

Polynom Splines

- Model using basis function representation

$$y_i = f(z_i) + \varepsilon_i = \sum_{j=1}^{k} \gamma_j B_j(z_i) + \varepsilon_i.$$

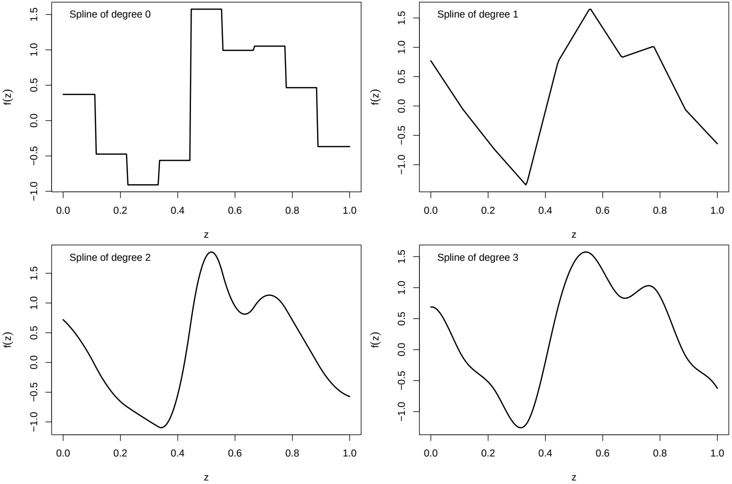
- The corresponding design matrix is

$$\mathbf{Z} = \begin{pmatrix} B_1(z_1) & \dots & B_k(z_1) \\ \vdots & & \vdots \\ B_1(z_n) & \dots & B_k(z_n) \end{pmatrix} = \begin{pmatrix} 1 & z_1 & \dots & z_1^l & (z_1 - \kappa_2)_+^l & \dots & (z_1 - \kappa_{m-1})_+^l \\ \vdots & & & & & & \vdots \\ 1 & z_n & \dots & z_n^l & (z_n - \kappa_2)_+^l & \dots & (z_n - \kappa_{m-1})_+^l \end{pmatrix},$$

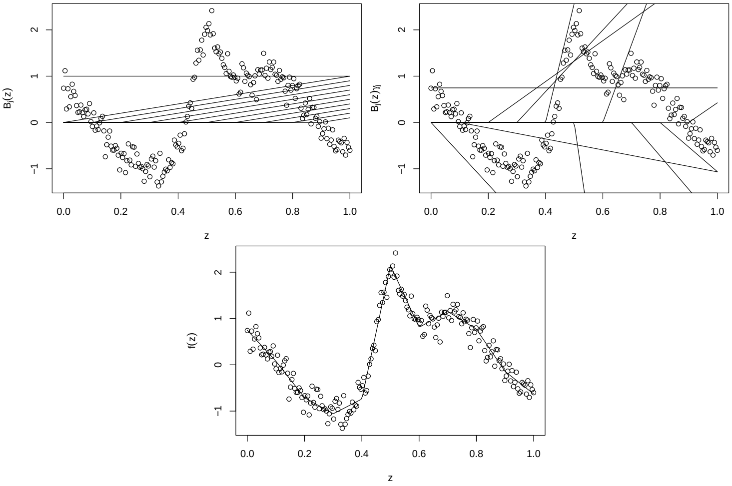
- In matrix notation

$$\mathbf{y} = \mathbf{Z}\gamma + \varepsilon \text{ and } \hat{\gamma} = (\mathbf{Z}^\top \mathbf{Z})^{-1} \mathbf{Z}^\top \mathbf{y}.$$

Polynom Splines



Polynom Splines



Penalized Regression

- Although most of the automatic knot selection procedures have exhibited good performance, they are usually quite complicated and computational intensive.
- We therefore seek a simpler method for flexible spline-based regression.
- As mentioned before, the roughness of a fit is due to there being too many knots in the model.
- Another way to overcome this problem is to retain all of the knots but to constrain their influence.
- The hope is that this will result in a less variable fit.

Penalized Regression

Penalized Regression with TP-Splines:

- Consider the truncated polynomial model

$$f(z_i) = \gamma_1 + \gamma_2 z_i + \ldots + \gamma_{l+1} z_i^l + \sum_{j=2}^{m-1} \gamma_{l+j}(z_i - \kappa_j)_+^l.$$

- The wiggliness of the fit is mainly the result of too large variability of the coefficients of the truncated bases.
- Constraints on the γ_{l+j} that might rectify this situation are
  1. $max|\gamma_{l+j}| < C$,
  2. $\sum |\gamma_{l+j}| < C$, and
  3. $\sum \gamma_{l+j}^2 < C$.
- Each of these will lead to a smoother fit, however, the third constraint is much easier to implement.

Penalized Regression

- Define the $((m-2)+l) \times ((m-2)+l)$ matrix

$$\mathbf{K} = \begin{pmatrix} \mathbf{0}_{l \times l} & \mathbf{0}_{l \times (m-2)} \\ \mathbf{0}_{(m-2) \times l} & \mathbf{I}_{(m-2) \times (m-2)} \end{pmatrix},$$

  then our minimization problem can be written as

$$min||\mathbf{y} - \mathbf{Z}\boldsymbol{\gamma}||^2 \text{ subject to } \boldsymbol{\gamma}^\top \mathbf{K} \boldsymbol{\gamma} < C.$$

- Using a Lagrange multiplier argument, it can be shown that this is equivalent to choosing γ to minimize

$$||\mathbf{y} - \mathbf{Z}\boldsymbol{\gamma}||^2 + \lambda \boldsymbol{\gamma}^\top \mathbf{K} \boldsymbol{\gamma}$$

  for some $\lambda \geq 0$.

Penalized Regression

- The solution is then given by

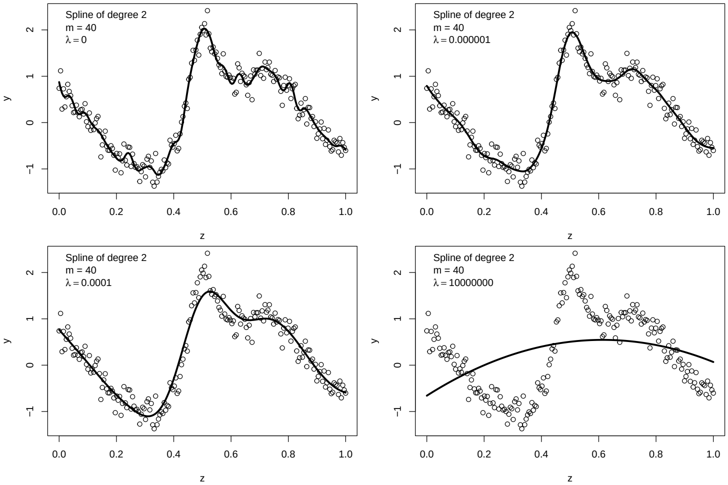
$$\hat{\gamma} = \left(\mathbf{Z}^\top \mathbf{Z} + \lambda \mathbf{K}\right)^{-1} \mathbf{Z}^\top \mathbf{y}.$$

- The fitted values for a penalized spline regression are

$$\hat{\mathbf{y}} = \mathbf{Z}\left(\mathbf{Z}^\top \mathbf{Z} + \lambda \mathbf{K}\right)^{-1} \mathbf{Z}^\top \mathbf{y} = \mathbf{S}_\lambda \mathbf{y},$$

  where $\mathbf{S}_\lambda$ is called smoother matrix.

Penalized Regression



P-Splines

- Truncated power bases can sometimes lead to numerical instability when there is a large number of knots and the smoothing parameter is small.
- Therefore, in practical use it is advisable to work with equivalent bases with more stable numerical properties.
- The most common choice is the B-spline basis.
- B-spline basis can represent cubic splines (and also higher or lower orders).
- The advantage of B-splines is that they are strictly local – each basis function is only non-zero over $l + 1$ adjacent knots.

P-Splines

- To define a B-spline with k basis functions we need to set up $m + l + 1$ knots

$$\kappa_1 < \kappa_2 < \ldots < \kappa_{m+l+1},$$

  where the interval over which the spline is to be evaluated is $[\kappa_{l+1}, \kappa_k]$, i.e., the first and the last l knot locations are essentially arbitrary.
- Every basis function overlaps with $2l$ neighboring basis functions and is positive over $l + 2$ neighboring knots.
- The B-spline is $l - 1$ times continuously differentiable.
- A lth order B-spline is then represented by

$$f(z_i) = \sum_{j=1}^{k} B_j^l(z_i)\gamma_j.$$

P-Splines

- The B-spline basis functions are most conveniently defined recursively as follows:
$$B_j^l(z_i) = \frac{z_i - \kappa_j}{\kappa_{j+l} - \kappa_j} B_j^{l-1}(z_i) + \frac{\kappa_{j+l+1} - z_i}{\kappa_{j+l+1} - \kappa_{j+1}} B_{j+1}^{l-1}(z_i),$$
- where
$$B_j^0(z_i) = \begin{cases} 1 & \kappa_j \leq z_i < \kappa_{j+1}, \\ 0 & \text{else.} \end{cases}$$
- A common choice is a cubic spline basis with $l = 3$.

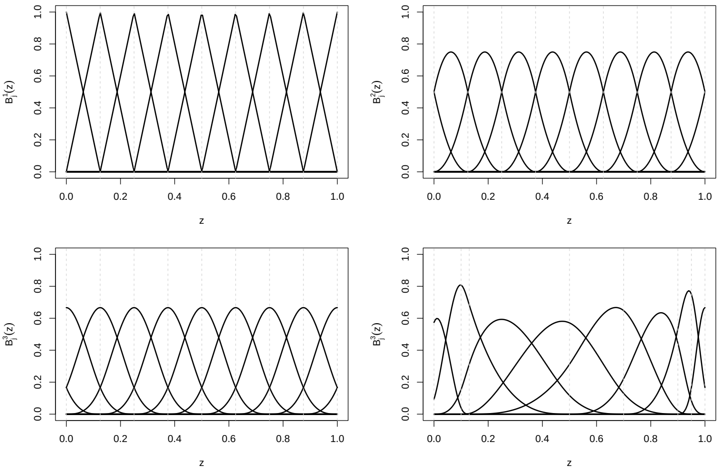
P-Splines

```
R> ## Evaluate a B-spline design matrix
R> ## first, define the B-spline basis function
R> ## recursively.
R> bsbasis <- function(z, knots, j, degree) {
+    if(degree == 0)
+      B <- 1 * (knots[j] <= z & z < knots[j + 1])
+    if(degree > 0) {
+      b1 <- (z - knots[j]) / (knots[j + degree] - knots[j])
+      b2 <- (knots[j + degree + 1] - z) /
+        (knots[j + degree + 1] - knots[j + 1])
+      B <- b1 * bsbasis(z, knots, j, degree - 1) +
+        b2 * bsbasis(z, knots, j + 1, degree - 1)
+    }
+    B[is.na(B)] <- 0
+    return(B)
+  }
```

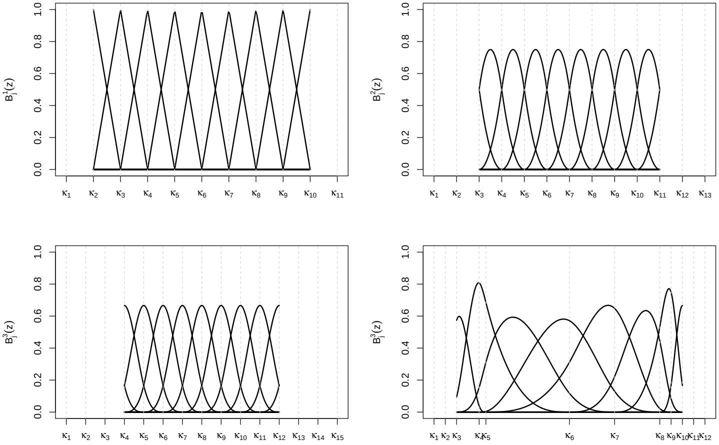
P-Splines

```
R> ## Now, compute the design matrix for all knots.
R> bsDesign <- function(z, degree = 3, knots = NULL) {
+     ## Compute knots.
+     if(is.null(knots))
+       knots <- 40
+     if(length(knots) < 2) {
+       step <- (max(z) - min(z)) / (knots - 1)
+       knots <- seq(min(z) - degree * step,
+         max(z) + degree * step, by = step)
+     }
+
+     ## Evaluate each basis function
+     ## and return the full design matrix B.
+     B <- NULL
+     for(j in 1:(length(knots) - degree - 1))
+       B <- cbind(B, bsbasis(z, knots, j, degree))
+     return(B)
+   }
```

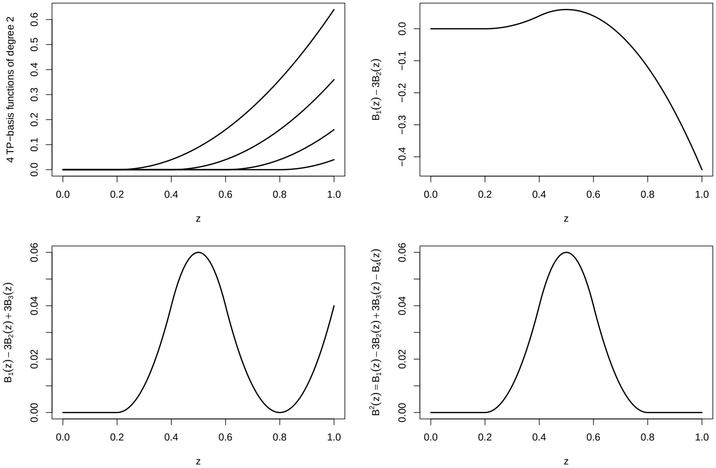
P-Splines



P-Splines


$B_j^1(z)$
$B_j^2(z)$
$B_j^3(z)$
$B_j^3(z)$
0.0 0.2 0.4 0.6 0.8 1.0
κ_1 κ_2 κ_3 κ_4 κ_5 κ_6 κ_7 κ_8 κ_9 κ_{10} κ_{11}
κ_1 κ_2 κ_3 κ_4 κ_5 κ_6 κ_7 κ_8 κ_9 κ_{10} κ_{11} κ_{12} κ_{13}
κ_1 κ_2 κ_3 κ_4 κ_5 κ_6 κ_7 κ_8 κ_9 κ_{10} κ_{11} κ_{12} κ_{13} κ_{14} κ_{15}
κ_1 κ_2 κ_3 κ_4 κ_5 κ_6 κ_7 κ_8 κ_9 κ_{10} κ_{11} κ_{12}

P-Splines


4 TP−basis functions of degree 2
$B_1(z) - 3B_2(z)$
$B_1(z) - 3B_2(z) + 3B_3(z)$
$B^2(z) = B_1(z) - 3B_2(z) + 3B_3(z) - B_4(z)$
z
0.0
0.2
0.4
0.6
0.8
1.0

P-Splines

- With B-spline basis functions a penalty on the regression coefficients is not obvious, since we do not divide in a parametric and non-parametric part.
- Since we want an overall smooth function we could use the following penalty

$$\lambda \int (f''(z))^2 dz.$$

- For B-splines we can construct simpler equivalent penalty terms

$$||\mathbf{y} - \mathbf{Z}\boldsymbol{\gamma}||^2 + \lambda \boldsymbol{\gamma}^\top \mathbf{K} \boldsymbol{\gamma}$$

with

$$\lambda \boldsymbol{\gamma}^\top \mathbf{K} \boldsymbol{\gamma} = \sum_{j=k+1}^{k} (\Delta^d \gamma_j)^2.$$

P-Splines

- Δ^d is the dth order difference which is defined recursively

$$\begin{aligned}
\Delta^1\gamma_j &= \gamma_j - \gamma_{j-1} \\
\Delta^2\gamma_j &= \Delta^1\Delta^1\gamma_j = \Delta^1\gamma_j - \Delta^1\gamma_{j-1} = \gamma_j - 2\gamma_{j-1} + \gamma_{j-2} \\
&\vdots \\
\Delta^d\gamma_j &= \Delta^{d-1}\gamma_j - \Delta^{d-1}\gamma_{j-1}.
\end{aligned}$$

- The first order difference matrix is then given by

$$\mathbf{D}_1 = \begin{pmatrix} -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & -1 & 1 \end{pmatrix} \text{ with } \mathbf{D}_1\gamma = \begin{pmatrix} \gamma_2 - \gamma_1 \\ \vdots \\ \gamma_k - \gamma_{k-1} \end{pmatrix}.$$

P-Splines

- The difference matrices can be computed recursively with

$$\mathbf{D}_d = \mathbf{D}_1 \mathbf{D}_{d-1}.$$

- Now, the resulting penalty matrix $\mathbf{K}$ is

$$\mathbf{K} = \mathbf{D}_k^\top \mathbf{D}_k.$$

```
R> ## Penalty matrix based on difference matrices.
R> penalty <- function(order = 2, k = 10) {
+    D <- diag(k)
+    for(i in 1:order)
+      D <- diff(D)
+    K <- crossprod(D, D)
+    return(K)
+  }
```

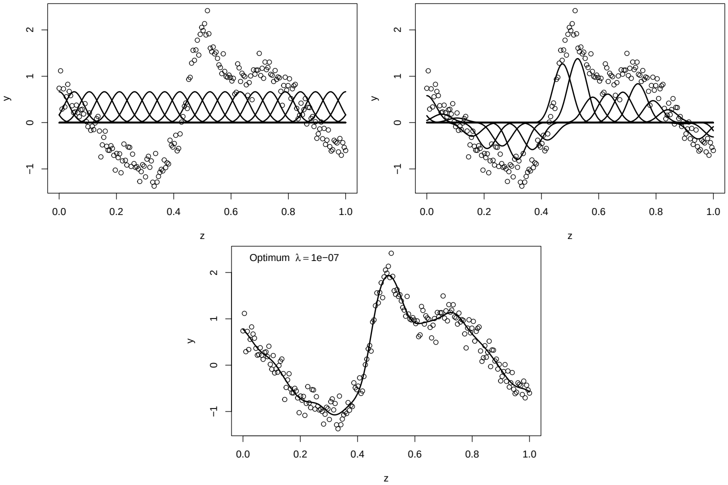
P-Splines

```
R> penalty(order = 1)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
 [1,]    1   -1    0    0    0    0    0    0    0     0
 [2,]   -1    2   -1    0    0    0    0    0    0     0
 [3,]    0   -1    2   -1    0    0    0    0    0     0
 [4,]    0    0   -1    2   -1    0    0    0    0     0
 [5,]    0    0    0   -1    2   -1    0    0    0     0
 [6,]    0    0    0    0   -1    2   -1    0    0     0
 [7,]    0    0    0    0    0   -1    2   -1    0     0
 [8,]    0    0    0    0    0    0   -1    2   -1     0
 [9,]    0    0    0    0    0    0    0   -1    2    -1
[10,]    0    0    0    0    0    0    0    0   -1     1
```

P-Splines

```
R> penalty(order = 2)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
 [1,]    1   -2    1    0    0    0    0    0    0     0
 [2,]   -2    5   -4    1    0    0    0    0    0     0
 [3,]    1   -4    6   -4    1    0    0    0    0     0
 [4,]    0    1   -4    6   -4    1    0    0    0     0
 [5,]    0    0    1   -4    6   -4    1    0    0     0
 [6,]    0    0    0    1   -4    6   -4    1    0     0
 [7,]    0    0    0    0    1   -4    6   -4    1     0
 [8,]    0    0    0    0    0    1   -4    6   -4     1
 [9,]    0    0    0    0    0    0    1   -4    5    -2
[10,]    0    0    0    0    0    0    0    1   -2     1
```

P-Splines



Bayesian P-splines

Penalized splines can also be derived in a Bayesian framework

In particular, this allows us to employ Bayesian approaches for the estimation of P-splines including the smoothing parameter.

Lets start with the observation model

$$y_i = \sum_{j=1}^{d} \gamma_j B_j(z_i) + \varepsilon_i \qquad \varepsilon_i \sim N(0, \sigma^2),$$

with B-spline basis functions B_j.

Instead of imposing a penalty, we will now develop an appropriate prior assumption for γ that enforces a smooth function estimation.

Bayesian P-splines

Priors of regression coefficients

- The stochastic analogue for the difference penalty are *random walks* of order k (RWk).
- A random walk of first order (RW1) is defined by

$$\gamma_j = \gamma_{j-1} + u_j, \quad u_j \sim N(0, \tau^2), \quad j = 2, \ldots, d,$$

  or equivalently

$$\gamma_j - \gamma_{j-1} = u_j, \quad u_j \sim N(0, \tau^2), \quad j = 2, \ldots, d,$$

  so that a connection to the first order difference penalty is recognizable.
- We have to make further assumptions for the prior of the starting value γ_1 and a noninformative prior distribution, $p(\gamma_1) \propto const$ will be our standard option.

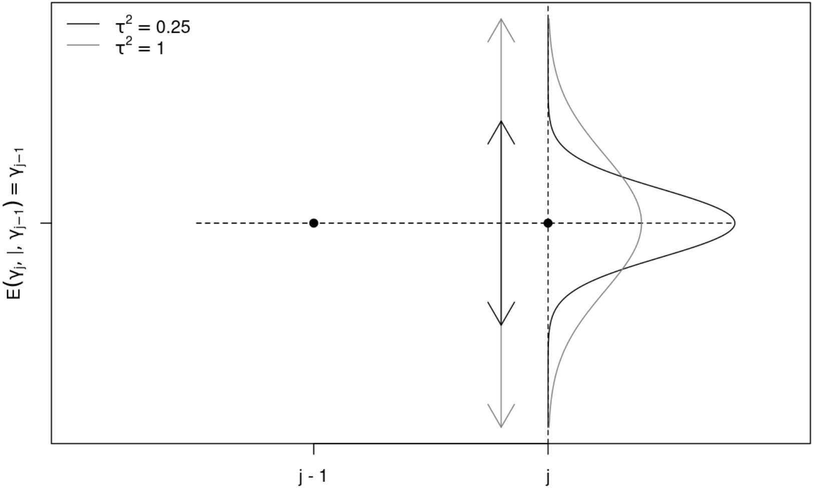
Bayesian P-splines

- When considering the conditional distributions defined by a RW1, we have

$$\gamma_j | \gamma_{j-1}, \ldots, \gamma_1 \sim N(\gamma_{j-1}, \tau^2).$$

- The RW1 has a special dependence structure such that the conditional distribution of γ_j given all previous values is only dependent on the value lagged by one, i.e. γ_{j-1}.
- Therefore, the RW1 has the (first order) *Markov property*.
- According to this formulation, the conditional expectation of γ_j is simply the lagged value γ_{j-1} such that we obtain a constant trend for the expected value.

Bayesian P-splines



Bayesian P-splines

- The larger the variance, the larger the possible deviation from the conditional expectation.
- A constant value of all B-spline coefficients leads to a constant estimate for the function $f(z)$. This corresponds to the case that the variance of the RW1 is (almost) zero, since only very little deviation between γ_j and γ_{j-1} is allowed in this situation resulting in a (near) constant trend for the sequence $\gamma_1, \dots, \gamma_d$.
- In contrast, when having a large variance τ^2, neighboring coefficients are able to deviate from each other, which in turn leads to a rough estimated function.
- It follows that we can interpret the variance parameter τ^2 as related to an inverse smoothing parameter.

Bayesian P-splines

- The joint multivariate prior distribution for γ is then given by

$$
\begin{aligned}
p(\gamma|\tau^2) &= \prod_{j=1}^{d} p(\gamma_j|\gamma_{j-1},\ldots,\gamma_1) = p(\gamma_1)\prod_{j=2}^{d} p(\gamma_j|\gamma_{j-1}) \\
&\propto \prod_{j=2}^{d} \frac{1}{\sqrt{2\pi\tau^2}} \exp\left(-\frac{1}{2\tau^2}(\gamma_j-\gamma_{j-1})^2\right) \\
&= \frac{1}{(2\pi\tau^2)^{(d-1)/2}} \exp\left(-\frac{1}{2\tau^2}\sum_{j=2}^{d}(\gamma_j-\gamma_{j-1})^2\right) \\
&= \frac{1}{(2\pi\tau^2)^{(d-1)/2}} \exp\left(-\frac{1}{2\tau^2}\boldsymbol{\gamma}^\top \mathbf{K}_1 \boldsymbol{\gamma}\right).
\end{aligned}
$$