A review of spline function selection procedures in R

Matthias Schmid

Department of Medical Biometry, Informatics and Epidemiology
University of Bonn

joint work with Aris Perperoglou
on behalf of TG2 of the STRATOS Initiative

September 1, 2016

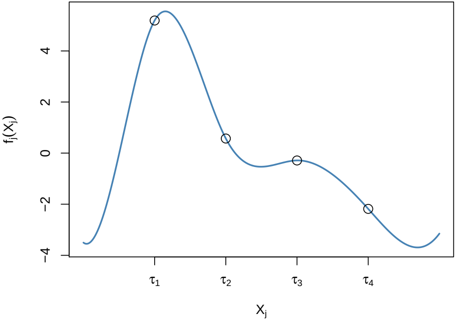
The Subject

- Fit a statistical model of the form $g(Y|X) = \beta_0 + f(X)$
    - p explanatory variables $X = (X_1, \ldots, X_p)$
    - f unknown, allowed to be nonlinear but should be interpretable
- Common specification: $f(X_1, \ldots, X_p) = f_1(X_1) + \ldots + f_p(X_p)$

  $\rightarrow$ Generalized additive models (GAMs)
- Splines are the most popular method to estimate $f_1, \ldots, f_p$
    - 668 articles in Biometrics, 790 articles in Statistics in Medicine, 173 articles in Biometrical Journal
    - GAM books by Hastie/Tibshirani and Wood are hugely popular ($> 12,000$ and $> 5,000$ citations, respectively)

Definition of Splines

- Set of piecewise polynomials, each of degree d
  - Joined together at a set of knots $\tau_1, \dots, \tau_K$
  - Continuous in value + sufficiently smooth at the knots



Spline Estimation

- Splines are typically represented by a set of (non-unique) basis functions $B_1, \ldots, B_{K+d+1}$
  - $\rightarrow$ $f_j(X_j) = \sum_{k=1}^{K+d+1} \beta_k B_k(X_j)$
  - $\rightarrow$ Linearization of the estimation problem
- Many types & subtypes, e.g.,
  - Natural splines: Required to be linear in τ_1 and τ_K
  - Penalized splines: Minimization of a critierion of the form

$$\text{Sum of residuals} + \lambda \cdot J_m$$

    with smoothing parameter λ and “wiggliness” parameter m
  - Example: $J_m = \int \left(\partial^m f_j / \partial x_j^m\right)^2 dx_j$

Spline Estimation (2)

- Types & subtypes (cont.):
  - Smoothing splines: Natural cubic splines with knots at the data values $x_{1j}, \dots, x_{nj}$
  - P-splines:
    - Use a B-spline basis with equidistant knots
    - Approximate integrated squared derivative penalty by an m-th order difference penalty
  - Thin plate regression splines: Penalized splines expressed in terms of radial basis functions
  - Additional restrictions: Monotonic splines, cyclic splines, ...

The Problem

- Spline modeling involves the selection of a comparatively large number of parameters
    - Number & placement of knots
    - Choice of basis & restrictions
    - Choice of smoothing parameter / optimization procedure
    - Choice of penalty order
- Mathematical properties are well understood, but . . .

The Problem (2)

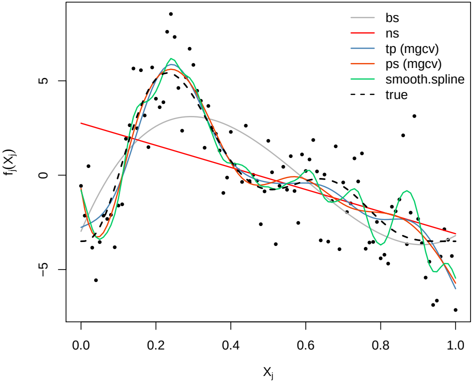
- Little guidance for applied statisticians
- Even for level 2 users it is hard to choose between competing approaches

  Broadly speaking the default penalized thin plate regression splines tend to give the best MSE performance, but they are slower to set up than the other bases. The knot based penalized cubic regression splines (with derivative based penalties) usually come next in MSE performance, with the P-splines doing just a little worse. However the P-splines are useful in non-standard situations.

  (Excerpt from `smooth.terms` help file in R package **mgcv**)

The Problem (3)

- Little guidance – so can we rely on software?
- Default results of some spline procedures in R:



Why R?

- Guidance on software plays a key role
- Why R?
  - Most widely spread software amongst people who do research in biomedical statistics
  - Also popular in the field of data science
  - Most widely used software in statistics courses

⇒ Start STRATOS work on splines with a project on spline implementations in R

First Steps

- Identify popular/relevant R packages and functions
  - Basic spline packages & functions
  - Regression packages
- Understand spline implementations
- Analyze documentation
- Compare approaches
- Provide first recommendations

Current Situation

- By July 2016, CRAN package repository featured 8,670 add-on packages
- 530 packages related to splines and/or spline modeling
- Even more packages were available on GITHUB, R-Forge etc. that we do not look into

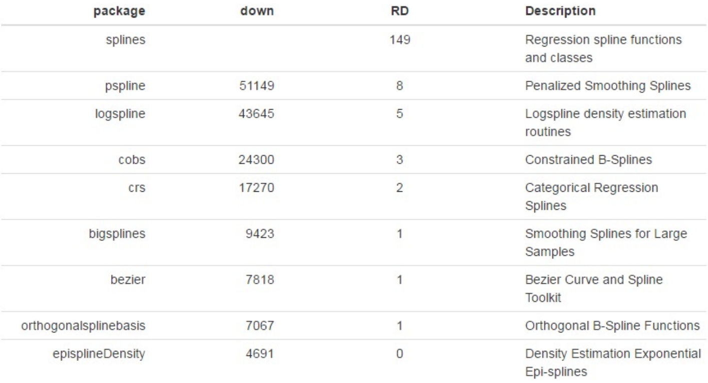
⇒ Restrict to packages that are both

(a) popular, and

(b) relevant for “standard” regression modeling

Basic Spline Packages

- Download numbers extracted from RStudio logs



package	down	RD	Description
splines		149	Regression spline functions and classes
pspline	51149	8	Penalized Smoothing Splines
logspline	43645	5	Logspline density estimation routines
cobs	24300	3	Constrained B-Splines
crs	17270	2	Categorical Regression Splines
bigsplines	9423	1	Smoothing Splines for Large Samples
bezier	7818	1	Bezier Curve and Spline Toolkit
orthogonalsplinebasis	7067	1	Orthogonal B-Spline Functions
episplineDensity	4691	0	Density Estimation Exponential Epi-splines

Preliminary Findings

- Lack of harmonization
  - Differences in terminology
  - Sometimes even within the same package
- Lack of documentation
  - R help files are often not detailed enough to fully understand how the implemented methods work

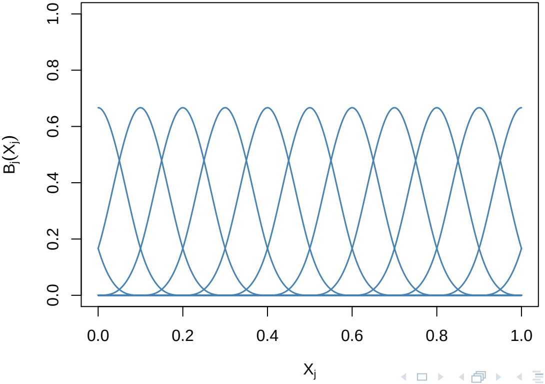
Analysis of the **splines** Package

- Consider the evaluation of cubic B-spline basis functions
- Most important functions: `splineDesign`, `bs`
- Both `splineDesign` and `bs` have an argument named `knots`
- Define knot sequence $0.1, 0.2, \ldots, 0.9$ for the above example

```
> knots <- seq(0.1, 0.9, 0.1)
> splineDesign(x, knots = knots)
Error in splineDesign(x, knots = knots) :
  die 'x' Daten müssen im Bereich 0.4 bis 0.6 liegen,
  außer es ist outer.ok = TRUE gesetzt
>
> dim(bs(x, knots = knots))
[1] 101  12
```

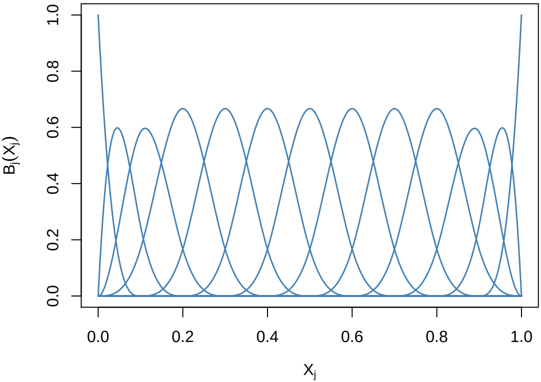
Analysis of the **splines** Package (2)

- B-spline basis requires definition of additional knots
- “Classical” B-spline plot contained in many textbooks:



Analysis of the **splines** Package (3)

- Basis functions produced by `bs`:



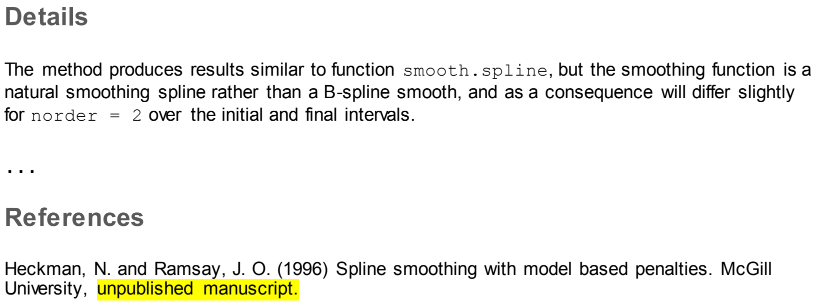
Smoothing Splines

- Evaluation of the smoothing spline basis
  - Apply `ns` function to a vector of length 10
  - Apply `smooth.spline` function

```
> ns(x, knots = x, intercept = TRUE)
Error in qr.default(t(const)) :
  NA/NaN/Inf in foreign function call (arg 1)
> N <- ns(x, knots = sort(x)[2:9], intercept = TRUE)
> dim(N)
[1] 10 10
>
> S <- smooth.spline(x, y)
> length(S$fit$coef)
[1] 12
```

Smoothing Splines (2)

- Apply `smooth.Pspline` function in R package **pspline**



Details

The method produces results similar to function `smooth.spline`, but the smoothing function is a natural smoothing spline rather than a B-spline smooth, and as a consequence will differ slightly for `norder = 2` over the initial and final intervals.

...

References

Heckman, N. and Ramsay, J. O. (1996) Spline smoothing with model based penalties. McGill University, unpublished manuscript.

Smoothing Splines (3)

- R code for `smooth.Pspline` function

```
> smooth.Pspline
function (x, y, w = rep(1, length(x)), norder = 2, df = norder +
    2, spar = 0, method = 1)
{
    my.call <- match.call()
    ...
    result <- .Fortran("pspline", as.integer(n), as.integer(nvar),
        as.integer(norder), as.double(x), as.double(w), as.double(y),
        as.double(yhat), as.double(lev), as.double(gcv), as.double(cv),
        as.double(df), as.double(spar), as.double(dfmax), as.double(work),
        as.integer(method), as.integer(irerun), as.integer(ier),
        PACKAGE = "pspline")
    ...
}
<environment: namespace:pspline>
```

Regression

- General features of popular regression packages

package	downloaded	vignette	book	website	data sets
quantreg	1,461,837	x	x		8
mgcv	1,055,188	x	x		2
survival	761,623	x	x		35
VGAM	230,784	x	x	x	50
gam	127,386		x		3
gamlss	60,868		(x)	x	29

Regression (2)

- Smooth terms in popular regression packages
  - **quantreg**
    - Allows for `bs`, `ns` etc. in model formula
    - `qss` – smoothing with "total variation" roughness penalty
  - **mgcv** implements (among many others)
    - Thin plate regression splines (`tp`, default)
    - Penalized natural cubic splines with cardinal spline basis `cr`
    - P-splines `ps`, bases on `splineDesign`
  - **survival**
    - Allows for `bs`, `ns` etc. in model formula
    - P-splines (`pspline`, same as `ps` in **mgcv**)

Regression (3)

- Smooth terms in popular regression packages (cont.)
  - **VGAM**
    - Allows for `bs`, `ns` etc. in model formula
    - P-splines (`ps`, similar to **mgcv**)
    - Smoothing splines (`s`, similar to `smooth.spline`)
  - **gam**
    - Functions `s`, `gam.s` – similar to `smooth.spline`
  - **gamlss**
    - Various P-spline and smoothing spline methods

Regression (4)

- Model selection in popular regression packages
  - **quantreg**: information criteria (but no `step` function)
  - **mgcv**: information criteria (but no `step` function), null space penalization / ridge penalty
  - **survival**: information criteria (but no `step` function, works with `stepAIC` from **MASS**
  - **VGAM:** information criteria (but no `step` function)
  - **gam:** information criteria, `step.gam` function
  - **gamlss:** information criteria, `stepGAIC` and related functions, null space penalty

Conclusion, Next Steps and Future Work

- Many regression packages rely on basic routines
- Variable selection based on information criteria or penalties
- Details of spline routines in regression packages are often not contained in help files + may be difficult to retrieve from literature
- Notable exception: **mgcv**
- Next steps:
  - Investigate routines for smoothing parameter optimization (GCV, UBRE, REML, etc.)
  - Investigate decomposition of splines (near + nonlinear parts)
- Future work:
  - Analyze monotonic splines, cyclic splines, multivariate splines
  - Extend analysis to other regression-type methods involving splines (**mboost**, **fda**)