

A toolbox of smooth effects and penalties

Matteo Fasiolo

matteo.fasiolo@bristol.ac.uk

Material available at:

https://github.com/mfasiolo/GAM_Workshop_Dortmund_25

These slides cover:

- 1 More on smoothing penalties
- 2 A toolbox of smooth effects
- 3 Practical modelling with mgcViz

Recall the GAM model structure:

$$y|\mathbf{x} \sim \text{Distr}\{y|\mu(\mathbf{x}), \theta_1, \dots, \theta_p\}$$

where $\mu(\mathbf{x}) = \mathbb{E}(y|\mathbf{x}) = g^{-1}\{\sum_{j=1}^m f_j(\mathbf{x})\}$.

The f_j 's can be

- parametric e.g. $f_j(\mathbf{x}) = \beta_1 x_j + \beta_2 x_j^2$
- random effects
- spline-based smooths such as

$$f_j(x_j) = \sum_{i=1}^r \beta_{ji} b_{ji}(x_j)$$

where β_{ji} are coefficients and $b_{ji}(x_j)$ are known spline basis functions.

NB: we call $\sum_{j=1}^m f_j(\mathbf{x})$ **linear predictor** because it is linear in β .

The β vector estimated by Maximum a Posterior (MAP)

$$\hat{\beta} = \underset{\beta}{\operatorname{argmax}} \log p(\beta|\mathbf{y}, \boldsymbol{\lambda}) = \underset{\beta}{\operatorname{argmax}} \left\{ \underbrace{\log p(\mathbf{y}|\beta)}_{\text{goodness of fit}} + \underbrace{\log p(\beta|\boldsymbol{\lambda})}_{\text{prior penalising complexity}} \right\}$$

In `mgcv` we use a Gaussian prior with log-density

$$\log p(\beta|\boldsymbol{\lambda}) \propto -\beta^T \mathbf{S}^\lambda \beta,$$

where

$$\mathbf{S}^\lambda = \sum_j \lambda_j \mathbf{S}_j.$$

and the \mathbf{S}_j are positive semi-definite matrices.

NOTE: the penalty is linear w.r.t. $\boldsymbol{\lambda}$.

But penalty $\lambda \boldsymbol{\beta}^\top \mathbf{S} \boldsymbol{\beta}$ is not very interpretable.

How do we relate it to something like $\lambda \int f''(x)^2 dx$?

Consider a spline-based effect

$$f(x) = \sum_k b_k(x) \beta_k = \mathbf{b}(x)^\top \boldsymbol{\beta}.$$

We have that

$$\begin{aligned} \int f''(x)^2 dx &= \int \{\mathbf{b}''(x)^\top \boldsymbol{\beta}\}^2 dx \\ &= \int (\mathbf{b}''(x)^\top \boldsymbol{\beta}) \mathbf{b}''(x)^\top \boldsymbol{\beta} dx \\ &= \boldsymbol{\beta}^\top \left\{ \int \mathbf{b}''(x) \mathbf{b}''(x)^\top dx \right\} \boldsymbol{\beta} \\ &= \boldsymbol{\beta}^\top \mathbf{S} \boldsymbol{\beta}. \end{aligned}$$

So, interpretable penalties can be “translated” to $\lambda \beta^T \mathbf{S} \beta$.

Note that $\lambda \int f''(x)^2 dx$ is just one type of penalty.

The type of penalty and type of bases function are often inter-related.

E.g., cubic regression splines (c.r.s.) are related to the optimal solution to

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

So, if you want $f(x)$ to be smooth in terms of $f''(x)$, use c.r.s..

Conversely, if you use c.r.s. it makes sense to penalise $f''(x)$.

But there are cases where you can “mix-and-match” bases and penalties.

These slides cover:

- 1 More on smoothing penalties
- 2 A toolbox of smooth effects
- 3 Practical modelling with mgcViz

`mgcv` offers a wide variety of smooths (see `?smooth.terms`).

Univariate types:

- `s(x, bs = "bs")` B-splines regression spline
- `s(x, bs = "ad")` adaptive smooth
- `s(x) = s(x, bs = "tp")` thin-plate-splines

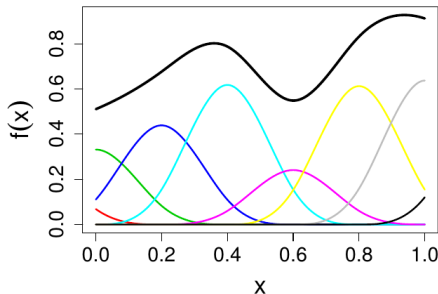
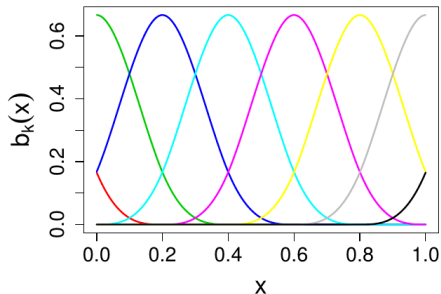
Multivariate type:

- `s(x1, x2) = s(x1, x2, bs = "tp")` thin-plate-splines
- `te(x1, x2)` tensor-product-smooth

They can depend on factors:

- `s(Subject, bs = "re")`
- `s(x, by = Subject)`
- `s(x, Subject, bs = "fs")`

B-splines: `s(x, bs = "bs", m = c(4, 2))`

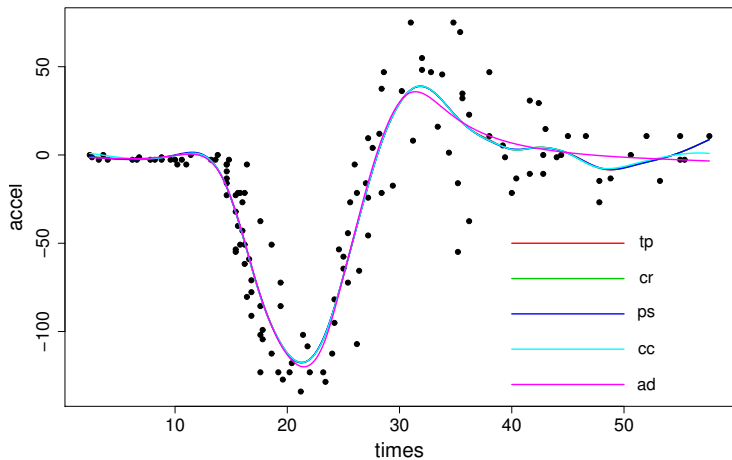


`m[1]` and `m[2]` are orders of the spline basis and penalty.

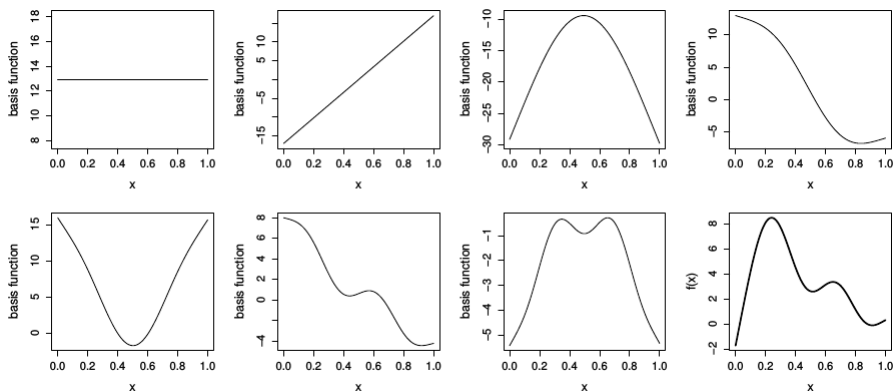
P-splines are B-splines with penalty such as $\sum_k (\beta_{k+1} - \beta_k)^2$.

Order of basis and penalty can be different, e.g.:

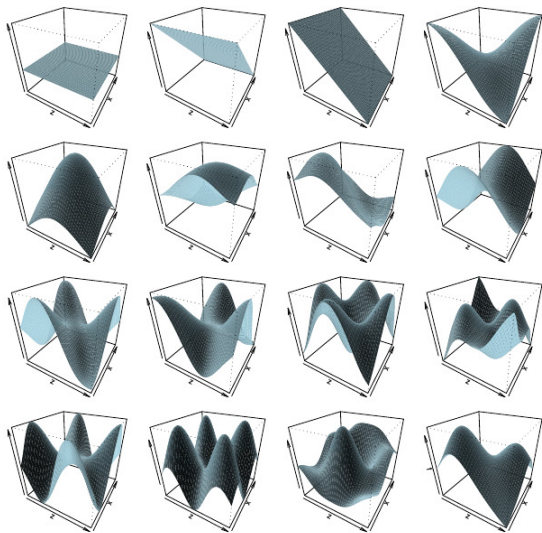
`s(x, bs = "ps", m = c(3, 1))`

Adaptive P-splines: $s(x, \text{bs} = \text{"ad"})$ 

The wiggleness or smoothness of $f(x)$ depends on x .

Thin plate regression splines (TPRS): $s(x)$ 

Rank 7 TPRS basis. Image from Wood (2006).



Rank 17 2D TPRS basis. Courtesy of Simon Wood.

$s(x_1, x_2), s(x_1, x_2, x_3), \dots$

Based on thin plate regression splines basis.

Related to optimal solution to:

$$\sum_i \{y_i - f(x_i, z_i)\}^2 + \lambda \int f_{xx}^2 + 2f_{xz}^2 + f_{zz}^2 dx dz$$

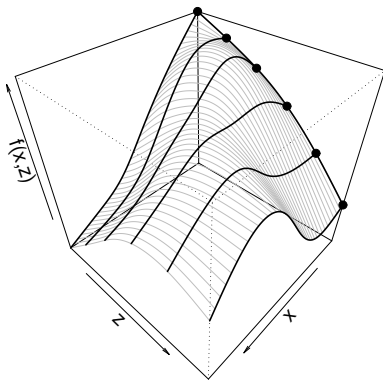
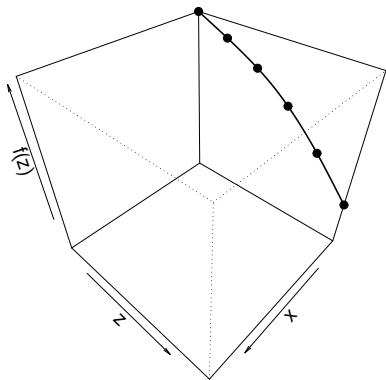
A single smoothing parameter λ .

Isotropic: same smoothness along x_1, x_2, \dots

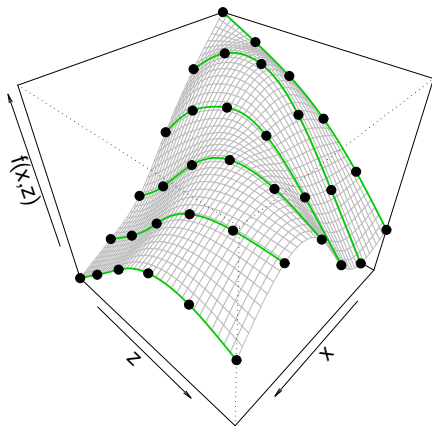
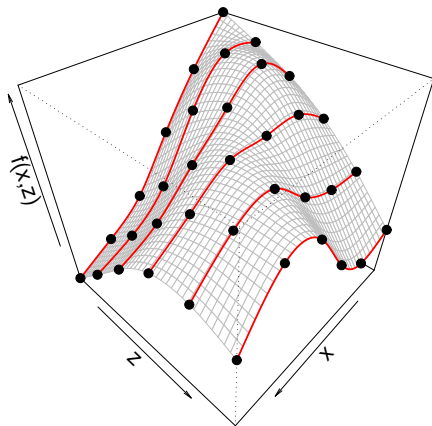
Isotropic effect of x_1, x_2 are in same unit (e.g. Km).

If different units better use tensor product smooths $\text{te}(x_1, x_2)$.

Construction: make a spline $f_z(z)$ a function of x by letting its coefficients vary smoothly with x



- x-penalty: average wiggleness of red curves
- z-penalty: average wiggleness of green curves



Can use (almost) any kind of marginal:

- `te(x1, x2, x3)` product of 3 cubic regression splines bases
- `te(x1, x2, bs = c("cc", "cr"), k = c(10, 6))`
- `te(L0, LA, t, d=c(2,1), k=c(20,10), bs=c("tp","cc"))`

Basis of `te` contains functions of the form $f(x_1)$ and $f(x_2)$.

To fit $f(x_1) + f(x_2) + f(x_1, x_2)$ separately use:

$$y \sim \text{ti}(x_1) + \text{ti}(x_2) + \text{ti}(x_1, x_2)$$

Random effects

Suppose we have data on bone mineral density (y) as a function of x .

We have m subjects and n data pairs per subject

- subj 1: $\{y_{11}, x_{11}\}, \dots, \{y_{n1}, x_{n1}\}$
- subj j : $\{y_{1j}, x_{1j}\}, \dots, \{y_{nj}, x_{nj}\}$
- subj m : $\{y_{1m}, x_{1m}\}, \dots, \{y_{nm}, x_{nm}\}$

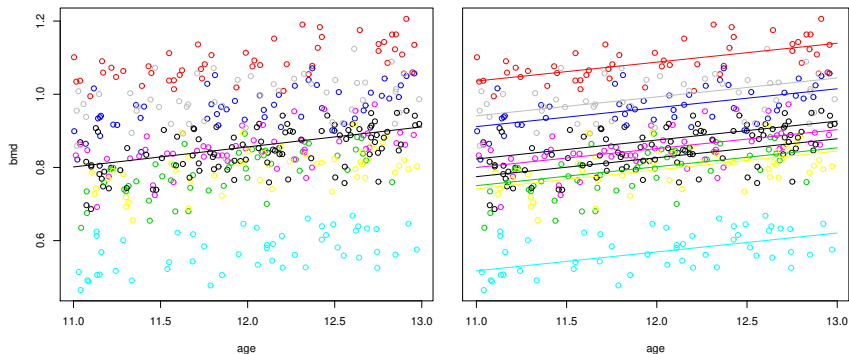
Standard linear model ignores individual differences

$$\mathbb{E}(y|x_{ij}) = \mu(x_{ij}) = \alpha + \beta x_{ij}.$$

We can include random intercept per subject

$$\mu(x_{ij}) = \alpha + \beta x_{ij} + a_j,$$

with penalty $\lambda_a \sum_j a_j^2 = \lambda_a \mathbf{a}^T \mathbf{a}$ via `s(subject, bs = "re")`



We can also include random slopes

$$\mu(x_{ij}) = \alpha + (\beta + b_j)x_{ij} + a_j,$$

with penalty $\lambda_b \mathbf{b}^T \mathbf{b}$ via `s(x, subject, bs = "re")`.

By-factor smooths

Approach (1) is `s(x, by = subject)`, which means

- $\mu(x) = f_1(x) + \dots$ if subject = 1
- $\mu(x) = f_2(x) + \dots$ if subject = 2
- ...

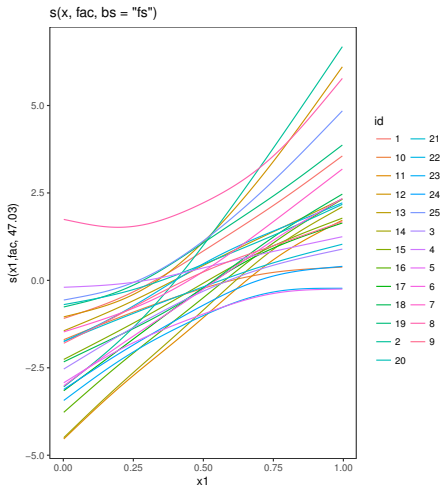
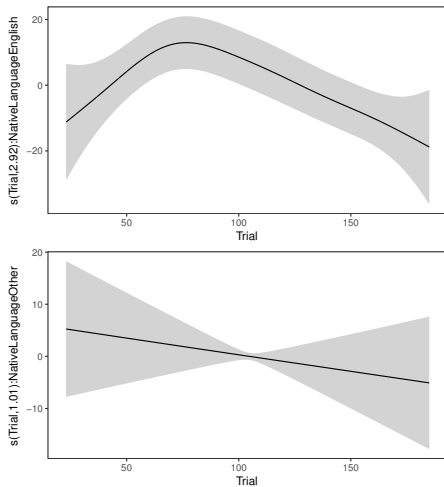
Approach (2) is `s(x, subject, bs = "fs")`, which means

- $\mu(x) = b_1 + f_1(x) + \dots$ if subject = 1
- $\mu(x) = b_2 + f_2(x) + \dots$ if subject = 2
- ...

where b_1, b_2, \dots are random intercepts.

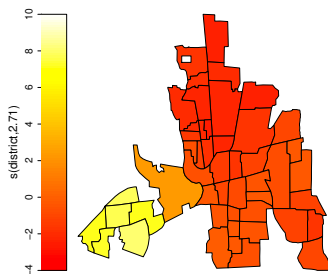
In (1) each f_j has its own smoothing parameter.

In (2) all f_j 's have the same smoothing parameter.



Markov random field effects

Sometimes data come allocated to irregular partitions of space.



- Markov random fields are a popular way of smoothing such data.
- The smooth has a coefficient, β_i , for each region.
- N_i is the set of indices of the neighbours of region i , then penalty is

$$\lambda \sum_i \left(\sum_{j \in N_i} (\beta_i - \beta_j) \right)^2.$$

Some extra smooths available in `mgcv`:

- smooth on sphere
- soap film smooths
- functional smooths (see `?linear.functional.terms`)

These are “standard”, i.e. linear w.r.t. β : $f(\mathbf{x}) = \sum_k b_k(\mathbf{x})\beta_k$.

`scam` package provides shape-constrained effects.

These are non-linear w.r.t. β .

gamFactory provides nested smooth effect

$$f(\mathbf{x}) = s(\tilde{s}(\mathbf{x})),$$

where

- \tilde{s} is a scalar-valued transformation
- s is a spline-based smooth

Simple example is single index effect

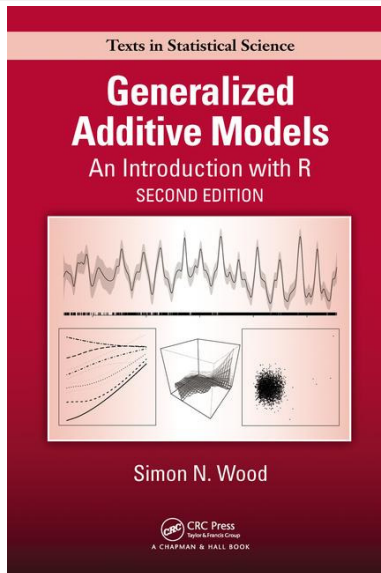
$$f(\mathbf{x}) = s(\boldsymbol{\alpha}^T \mathbf{x}).$$

The effect is non linear w.r.t. $\boldsymbol{\alpha}$.

If interested try

```
library(devtools)
install_github("mfasiolo/gamFactory")
```

Further reading



These slides cover:

- 1 More on smoothing penalties
- 2 A toolbox of smooth effects
- 3 Practical modelling with mgcViz

Now we'll look at `R_demos/2_effects_and_mgcViz.html`

References I

- Brezger, A., T. Kneib, and S. Lang (2003). Bayesx: Analysing bayesian structured additive regression models. Technical report, Discussion paper//Sonderforschungsbereich 386 der Ludwig-Maximilians
- Bürkner, P. C. et al. (2017). brms: An r package for bayesian multilevel models using stan. *Journal of Statistical Software* 80(1), 1–28.
- Eilers, P. H. and B. D. Marx (1996). Flexible smoothing with b-splines and penalties. *Statistical science* 11(2), 89–121.
- Fasiolo, M., R. Nedellec, Y. Goude, and S. N. Wood (2020). Scalable visualization methods for modern generalized additive models. *Journal of computational and Graphical Statistics* 29(1), 78–86.
- Hothorn, T., P. Bühlmann, T. Kneib, M. Schmid, and B. Hofner (2010). Model-based boosting 2.0. *The Journal of Machine Learning Research* 11, 2109–2113.
- Pya, N. and S. N. Wood (2015). Shape constrained additive models. *Statistics and computing* 25, 543–559.
- Rigby, R. A. and D. M. Stasinopoulos (2005). Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 54(3), 507–554.

References II

- Rue, H., S. Martino, and N. Chopin (2009). Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations. *Journal of the royal statistical society: Series b (statistical methodology)* 71(2), 319–392.
- Wood, S. (2006). *Generalized additive models: an introduction with R*. CRC press.