Smooth Terms

Simon N. Wood

University of Bath

Overview

- A major part of additive smooth model specification is the choice of component smooth functions.
- For smooths of one variable there is a range of basis-penalty smoothers to choose from, but differences in performance are typically small.
- For smooths of several variables there is an important choice to make:
  1. Thin plate spline like smooths are isotropic, and offer rotational invariance.
  2. Tensor product smooths are invariant to independent linear rescaling of covariates.

Penalized regression splines

- Represent each smooth using a low rank spline like basis. . .

$$f_j(x) = \sum_k \beta_{jk} b_{jk}(x)$$

  where $b_{jk}(x)$ are known basis functions and the β_{jk}s are coefficients to be estimated.
- Associate a wiggliness penalty with each smooth. e.g.

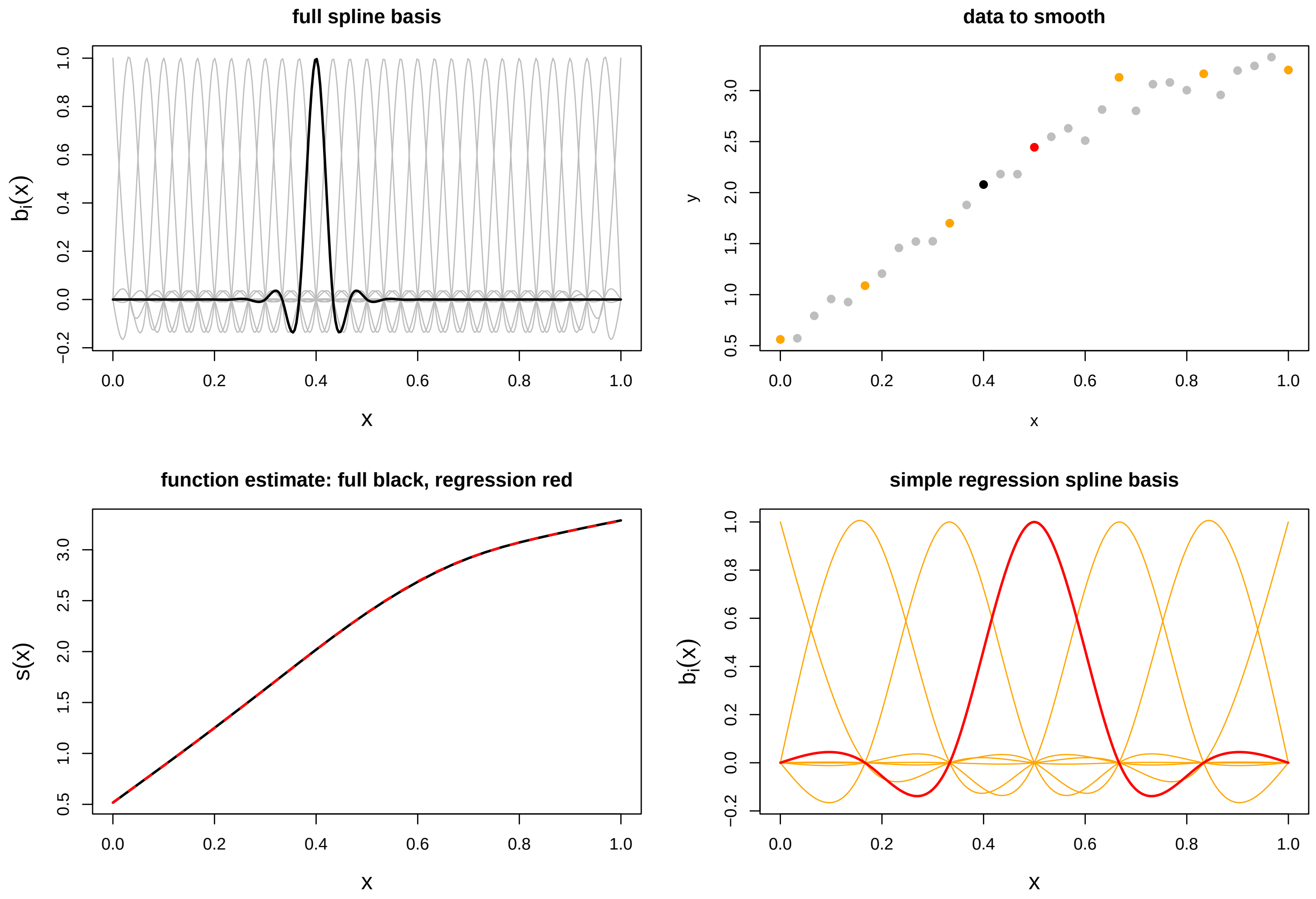
$$\int f_j''(x)^2 dx = \boldsymbol{\beta}_j^{\mathsf{T}} \mathbf{S}_j \boldsymbol{\beta}_j$$

  $\mathbf{S}_j$ is a matrix of known coefficients derived from $b_{jk}(x)$.

Simple regression splines

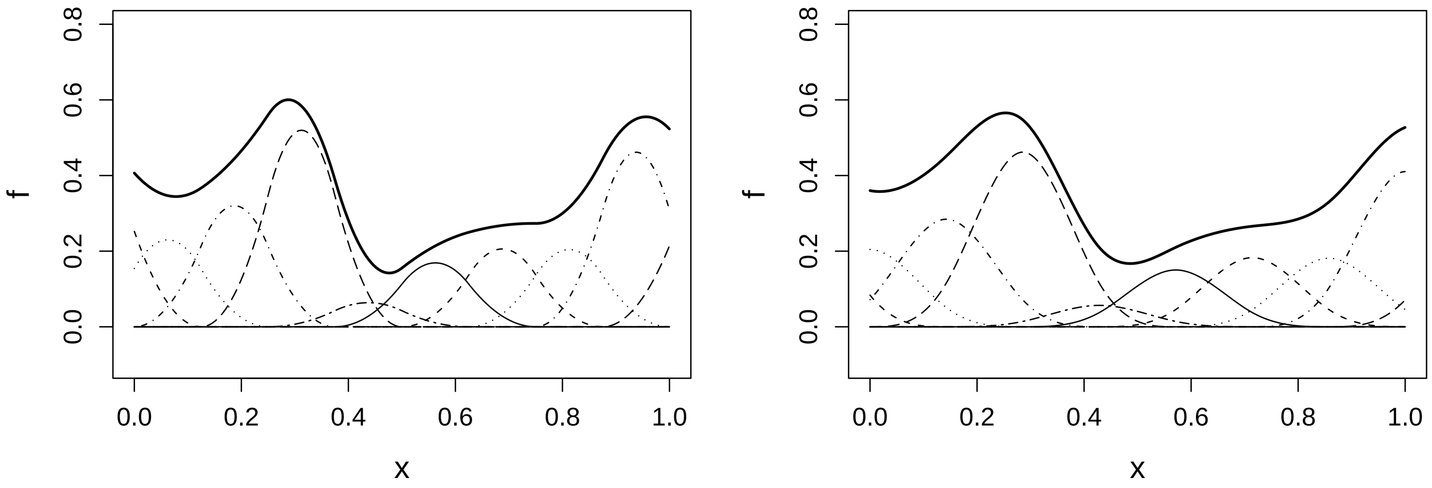
- Get a spline basis and penalty for a ‘representative’ subset of the real data.
- Use this basis to model the real data.
- This is very cheap, but somewhat ad hoc.
- Easily extended to produce ‘cyclic’ smooth functions.

Simple regression splines



P-splines (Eilers & Marx)

- Use B-spline basis functions.
- Apply simple difference penalties directly to the coefficients multiplying the basis functions.
- Flexible and easy to implement.



Thin plate splines

- Thin plate splines are functions minimizing (isotropic) objectives like

$$\sum (y_i - f(x_i))^2 \quad + \quad \lambda \int f_{xx}^2 dx$$

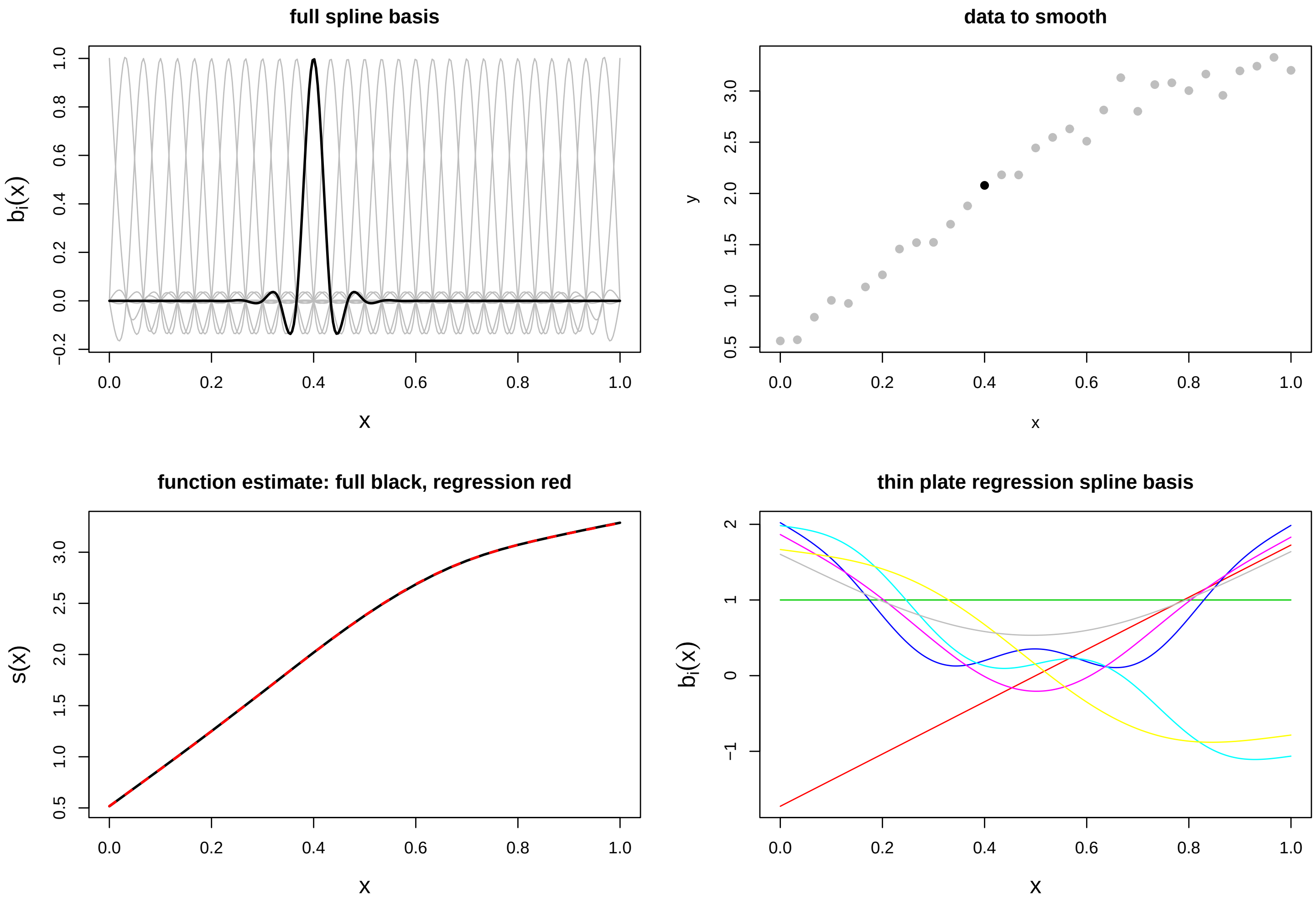
$$\sum (y_i - f(x_i, z_i))^2 \quad + \quad \lambda \int f_{xx}^2 + 2f_{xz}^2 + f_{zz}^2 dx dz$$

- Solution is of form $\hat{f}(\cdot) = \sum_{i=1}^n \delta_i \eta_i(\cdot) + \sum_{j=1}^M \alpha_j \phi_j(\cdot)$ where $\boldsymbol{\alpha}$ and $\boldsymbol{\delta}$ minimize $\|\mathbf{y} - \mathbf{E}\boldsymbol{\delta} - \mathbf{T}\boldsymbol{\alpha}\|^2 + \lambda \boldsymbol{\delta}^\mathsf{T}\mathbf{E}\boldsymbol{\delta}$, $\mathbf{T}^\mathsf{T}\boldsymbol{\delta} = \mathbf{0}$; η_i and ϕ_j are known and give $\mathbf{E}$ and $\mathbf{T}$.

Thin plate regression splines

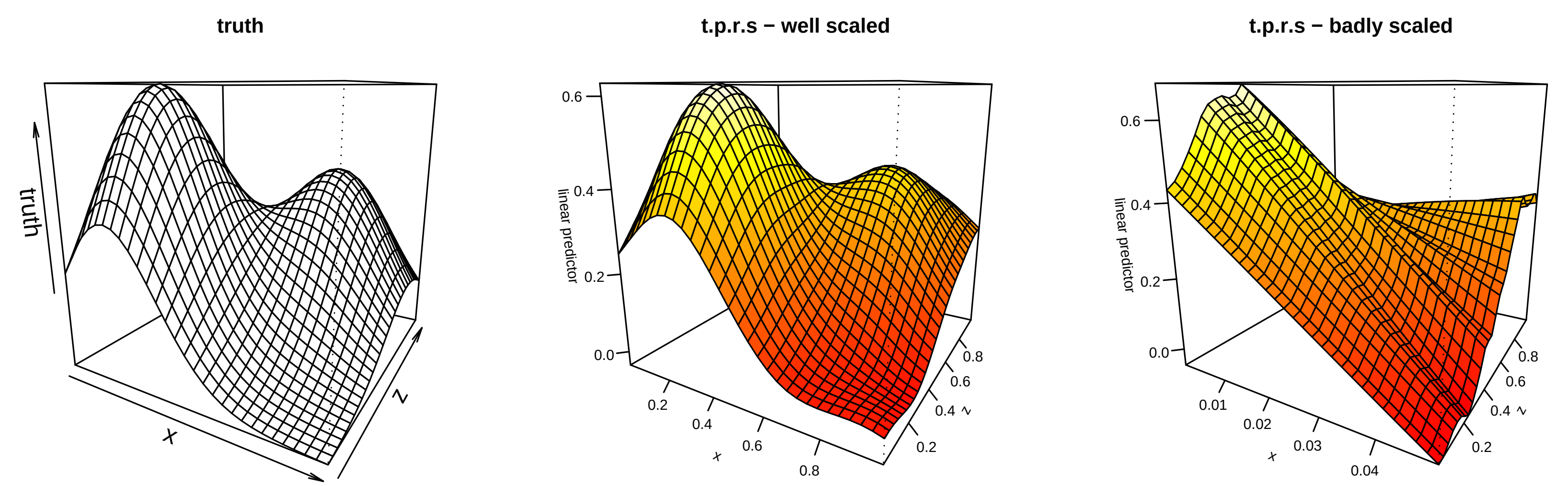
- Thin plate splines are computationally expensive [$O(n^3)$].
- By replacing $\mathbf{E}$ by its rank k truncated eigen-decomposition we can get an 'optimal' (Wood, 2003, JRSSB) rank k approximation to a thin plate spline that is much cheaper to work with.
- Lanczos iteration gives this truncated decomposition relatively cheaply.
- General method for efficient isotropic smoothing. No need to choose 'knots'!

TPRS 1D example



TPRS limitations

- If covariates are not naturally on the same scale, then isotropic smoothness may be inappropriate.
- Thin plate (regression) splines are sensitive to arbitrary independent linear rescaling of covariates.



Tensor product smooths

- Idea: combine bases and penalties for representing smooth functions $f_x(x)$ and $f_z(z)$ to create a basis and penalties for $f(x,z)$.
- Suppose $f_x(x) = \sum_i \alpha_i a_i(x)$ and $f_z(z) = \sum_j \beta_j b_j(z)$.
- Can let f_x vary smoothly with z by letting its parameters, α_i, vary smoothly with z, using the basis for f_z. i.e. let $\alpha_i(z) = \sum_j \beta_{ji} b_j(z)$ so that

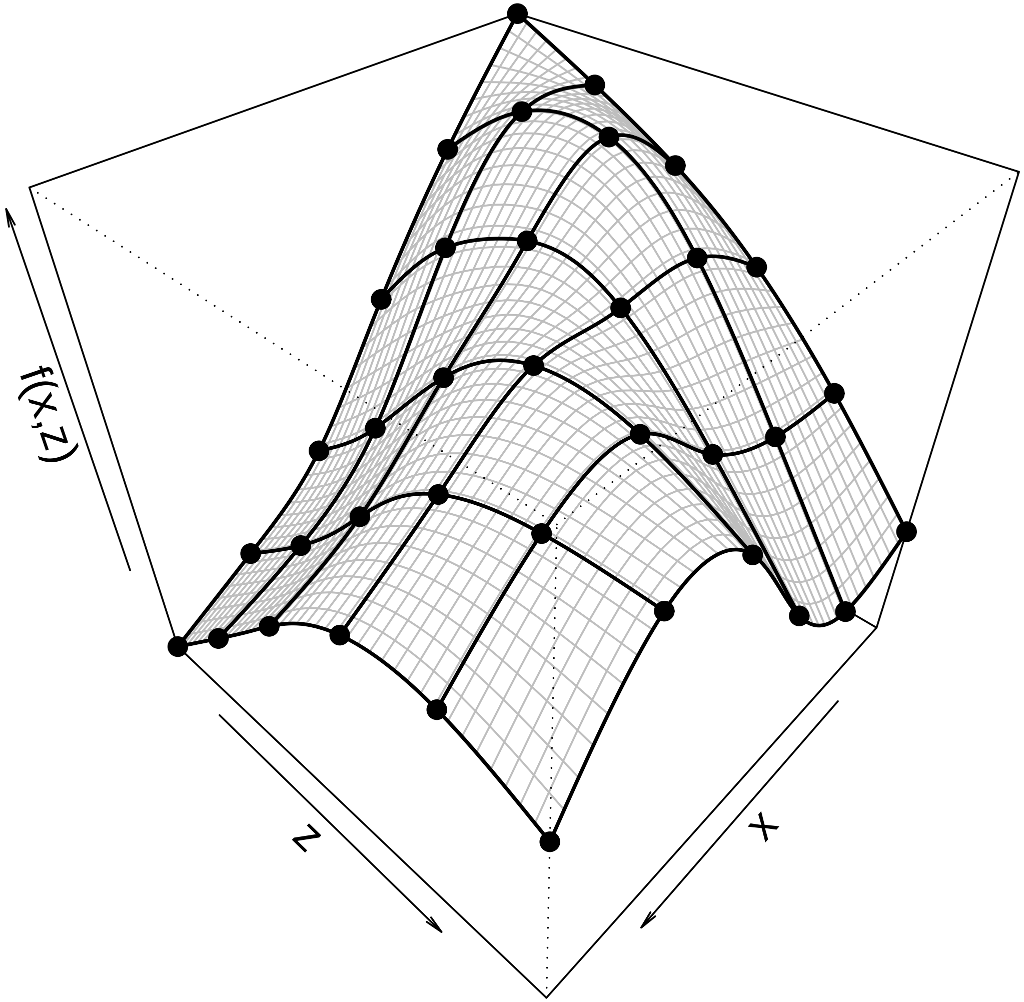
$$f(x,z) = \sum_i \sum_j \beta_{ij} a_i(x) b_j(z)$$

- Construction generalizes and is symmetric.

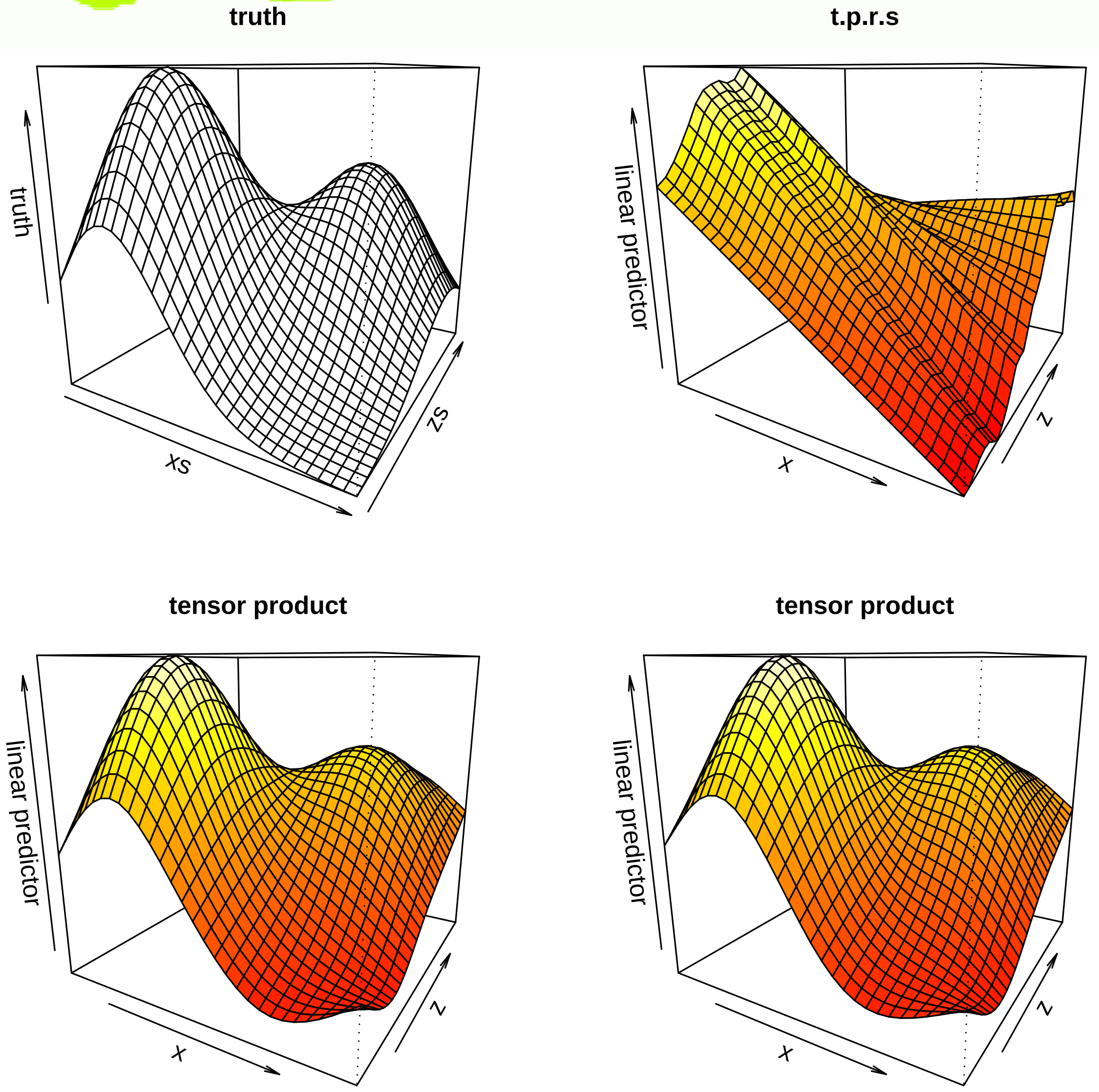
Tensor product penalties

- Can achieve scale invariance by measuring wiggliness w.r.t. each axis separately.
- Use wiggliness measure for marginal smooth f_x to measure average wiggliness of a set of curves in the x direction over $f(x,z)$. Do same for wiggliness in z using wiggliness measure for f_z.
- End up with one penalty per dimension: each quadratic in the parameters of the smooth, and generated automatically from the marginal penalties.

Tensor smooth illustration



Scale invariance!



Smooths and `gam`

Terms like `s(x)` include 1D and TPRS smooths in a `gam`.
In full `s(x,k=10,fx=FALSE,bs="tp",m=2,by=NA)`

- The first arguments specify the covariates (just `x` here).
- `k` is the basis dimension (auto-initializes if left out).
- `fx` should be set to `TRUE` to leave the smooth *unpenalized*.
- `bs` specifies the smoothing basis, e.g. `"cr"` (cubic regression), `"cc"` (cyclic) or `"tp"` (default TPRS). You can add smooth types - see `?p.spline`.
- `m` specifies the penalty order for a TPRS.
- `by` specifies a covariate by which whole smooth should be multiplied — implements variable coefficient models (geographic regression) and conditioning on factors.

Tensor smooths and `gam`

- Tensor product smooths are generated by `te` terms in the `gam` formula.
- Tensor products of any smooth available to `s` terms can be produced (including user defined smooths).
- Some examples are helpful. . .
  - `te(x,z)`; a tensor product smooth of `x` and `z`, using default `"cr"` marginal bases each of default dimension `k=5`.
  - `te(x,z,v,d=c(2,1),k=c(30,6),bs=c("tp","cc"))` a tensor product of a 2 covariate `"tp"` spline of basis dimension `k=30` and a cyclic spline with `k=6`.

Choosing k

- The basis dimension, k, for a term, is not critical provided (i) it is not so small as to force the model to over-simplify and (ii) it is not so large that computation becomes unbearably slow.
- It is the smoothing parameters that control the actual DoF for a term.
- k can be checked informally by extracting deviance residuals from a fit, and then fitting just the suspect smooth to those residuals, with k increased substantially (using penalized least squares and GCV). If the original fit missed substantial pattern, you will pick it up.