2D penalized spline (continuous-by-continuous interaction)

Two examples (RWC, Section 13.1):

- Number of scallops caught off Long Island
  - Counts are made at specific coordinates.
- Incidence of AIDS in Italian MSM
  - Predictors are calendar year and age at diagnosis.

In one sense, these problems are identical; in another sense, they're not:

- AIDS: The predictors are on meaningful scales.
- Scallops: The coordinate axes have no inherent meaning; the analysis should be invariant to translations and rotations.

This difference could affect the choice of basis functions.

Basis #1: Tensor product basis (RWC Sec. 13.2)

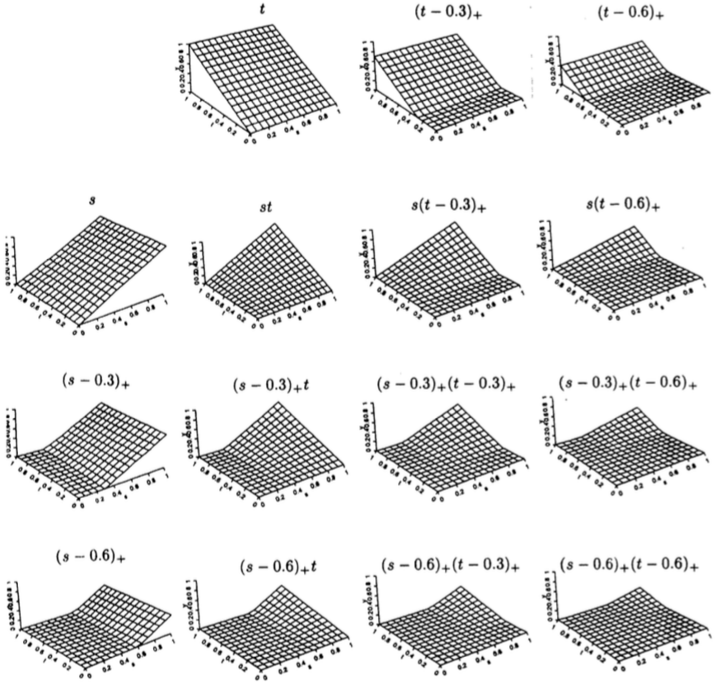
We want to model $y_i = f(s_i, t_i) + \epsilon_i$ for s and t continuous.

The natural extension of the truncated-lines basis in 1-D is:

$$
\begin{aligned}
y_i = \quad & \beta_0 && + \beta_s s_i + \sum_{k=1}^{K^s} u_k^s (s_i - \kappa_k^s)_+ \\
& && + \beta_t t_i + \sum_{k=1}^{K^t} u_k^t (t_i - \kappa_k^t)_+ \\
& + \beta_{st} s_i t_i && + \sum_{k=1}^{K^s} v_k^s s_i (t_i - \kappa_k^t)_+ \\
& && + \sum_{k=1}^{K^t} v_k^t t_i (s_i - \kappa_k^s)_+ \\
& && + \sum_{k=1}^{K^s} \sum_{k'=1}^{K^t} v_{kk'}^{st} (s_i - \kappa_k^s)_+ (t_i - \kappa_k^t)_+ \\
& + \epsilon_i,
\end{aligned}
$$

The first block is the main effects, the second block is the interactions.

The next slide shows the basis functions (2 knots in each of s and t).



t
$(t-0.3)_+$
$(t-0.6)_+$
s
st
$s(t-0.3)_+$
$s(t-0.6)_+$
$(s-0.3)_+$
$(s-0.3)_+t$
$(s-0.3)_+(t-0.3)_+$
$(s-0.3)_+(t-0.6)_+$
$(s-0.6)_+$
$(s-0.6)_+t$
$(s-0.6)_+(t-0.3)_+$
$(s-0.6)_+(t-0.6)_+$

Basis #2: Radial basis functions (RWC Sec. 13.2)

We saw a special case of these when we talked about penalized splines.

A more general form depends on a function $C(\| (s_i, t_i) - (\kappa_j^s, \kappa_k^t) \|)$

where: $C(\bullet)$ is a function from $\mathcal{R}^+ \to \mathcal{R}$, and
$\| \bullet \|$ is a distance measure.

The value of this basis function for observation i and knots (j, k) depends only on the distance $\| \bullet \|$ from (s_i, t_i) to (κ_j^s, κ_k^t).

With a radial basis built using such a C, the fit is invariant to axis translations and rotations.

General radial smoothing (RWC Sec. 13.4)

We'll develop this in 1 dimension; generalizing to > 1D is trivial.

This involves a lot of *ad hoc* tinkering, starting with a particular model and using a few kluges to achieve desirable properties.

I am OK with that, though some would not be.

This path's *ad hockery* suggests that many other paths could be fruitful.

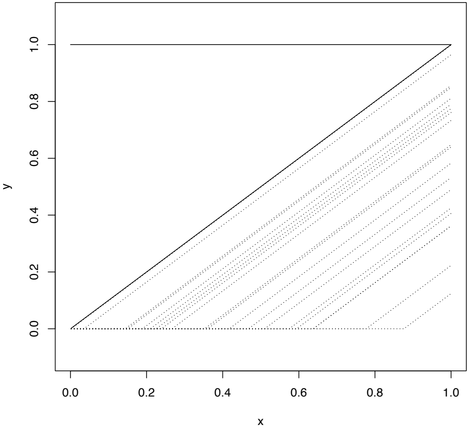
Radial bases – start with a simple case

Full-rank truncated-line basis for a 1D spline fit of $y_i = f(x_i) + \text{error}$.

$\mathbf{X}$ has rows $[1 \; x_i]$.

For now, consider a full-rank basis in which each unique value of x is a knot: $\mathbf{Z}$ has entry $(x_i - x_j)_+$ in row i and column j.

The columns of $\mathbf{X}$ and $\mathbf{Z}$ for 20 x_i, iid draws from a $U[0,1]$:



Step 1: Transform to a radial basis

Given λ^2, the fit is $\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} + \mathbf{Z}\hat{\mathbf{u}}$, where $(\hat{\boldsymbol{\beta}}, \hat{\mathbf{u}})$ solves

$$\text{argmin}_{\boldsymbol{\beta},\mathbf{u}} \left\{ (\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u}) + \lambda^2 \mathbf{u}'\mathbf{u} \right\}.$$

We'd like to have a radial basis in $\mathbf{Zu}$, while leaving $\mathbf{X}$ unchanged.

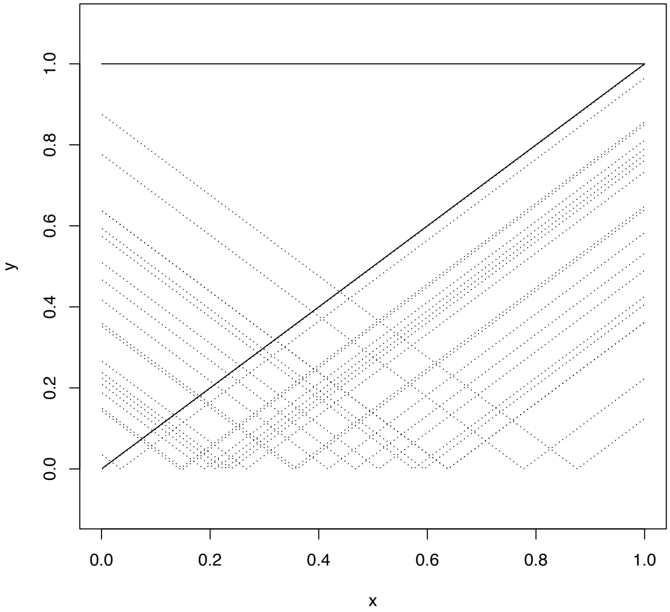
If each x_i is unique, then $\exists\ \mathbf{L}\ (n+2) \times (n+2) \ni$

$$[\mathbf{X}|\mathbf{Z}_R] = [\mathbf{X}|\mathbf{Z}]\,\mathbf{L},$$

where the columns of $\mathbf{Z}_R$ form a radial basis and $\mathbf{Z}_R$ is symmetric.

The next slide shows the resulting radial basis.

Step 1: Transform to a radial basis (continued)



Step 2: A kluge to make the penalty radially symmetric

With this new basis, the fitted values are

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}}_R + \mathbf{Z}_R\hat{\mathbf{u}}_R, \qquad \text{where } (\hat{\boldsymbol{\beta}}_R, \hat{\mathbf{u}}_R) \text{ solves}$$

$$\text{argmin}_{\boldsymbol{\beta}_R, \mathbf{u}_R} \left\{ (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}_R + \mathbf{Z}_R\mathbf{u}_R)'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}_R + \mathbf{Z}_R\mathbf{u}_R) + \lambda^2 \mathbf{u}_R'\mathbf{L}'\mathbf{D}\mathbf{L}\mathbf{u}_R \right\},$$

where $\mathbf{D}$ is diagonal with diagonal elements $(0, 0, \mathbf{1}_n')'$.

Problem: The penalty $\lambda^2 \mathbf{u}_R'\mathbf{L}'\mathbf{D}\mathbf{L}\mathbf{u}_R$ isn't radially symmetric, and it doesn't generalize readily to higher dimensions.

First kluge: Change the penalty to $\lambda^2 \mathbf{u}\mathbf{Z}_R\mathbf{u}$.

The spline fit is now the solution of

$$\text{argmin}_{\boldsymbol{\beta}, \mathbf{u}} \left\{ (\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}_R\mathbf{u})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}_R\mathbf{u}) + \lambda^2 \mathbf{u}\mathbf{Z}_R\mathbf{u} \right\}.$$

This is a member of the thin-plate spline family of smoothers.

Step 2 (continued): Why we'll need another kluge

The spline fit is now the solution of

$$\text{argmin}_{\boldsymbol{\beta},\mathbf{u}} \left\{ (\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}_R\mathbf{u})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}_R\mathbf{u}) + \lambda^2\mathbf{u}\mathbf{Z}_R\mathbf{u} \right\}.$$

Problem: $\mathbf{Z}_R$ can't be $\mathbf{G}^{-1}$ because it's not necessarily p.s.d.

Example:
Let x_i take values $1, 2, \ldots, 100$; $\mathbf{Z}_R$ has 99 negative eigenvalues.

Step 3: Second kluge + reduce the rank of the spline

Define $\mathbf{Z}_C = [C(|x_i - \kappa_k|)]$, where $C(\bullet) : \mathcal{R}^+ \to \mathcal{R}$ as before.

$\mathbf{Z}_C$ has rows $i = 1, \ldots, n$ and columns $k = 1, \ldots, K$, for knots κ_k.

Setting $C(r) = r$ and $K = n$ gives the radial basis we've used until now.

Now we fit a penalized spline by fitting this MLM:

$$\begin{aligned}
\mathbf{y} &= \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}_C\mathbf{u} + \boldsymbol{\epsilon}, \qquad \mathbf{R} = \sigma_e^2\mathbf{I}_n \\
\mathbf{G} &= \sigma_s^2(\Omega_K^{-0.5})(\Omega_K^{-0.5})' \\
\text{where } \Omega_K &= [C(|\kappa_k - \kappa_{k'}|)] \text{ for } k, k' = 1, \ldots, K.
\end{aligned}$$

$$\Omega_K^{-0.5} = \mathbf{U}\ \text{diag}(\mathbf{d}^{-0.5})\ \mathbf{V}', \text{ where } \Omega_K = \mathbf{UDV}' \text{ is the SVD.}$$

This is a legal mixed linear model.

Step 4: Make this look like earlier penalized splines

To make this model look like earlier penalized splines,

re-parameterize the random effect.

Replace $\mathbf{Z}_C$ with $\mathbf{Z} = \mathbf{Z}_C \Omega_K^{-0.5}$, giving

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\epsilon}, \qquad \mathbf{R} = \sigma_e^2 \mathbf{I}_n, \qquad \mathbf{G} = \sigma_s^2 \mathbf{I}_K,$$

recalling that $\mathbf{Z}_C = [C(|x_i - \kappa_k|)]$ is $n \times K$ and

$\Omega_K = [C(|\kappa_k - \kappa_{k'}|)]$ is $K \times K$.

Step 5: Extend to > 1 dimension

Up to this point, the development has been for 1-D splines.

The extension to $p > 1$ dimensions is now trivial:

In the definition of $\mathbf{Z}_C$, replace $|x_i - \kappa_k|$ with $\|x_i - \kappa_k\|$, and

Ω_K, replace $|\kappa_k - \kappa_{k'}|$ with $\|\kappa_k - \kappa_{k'}\|$,

where $\| \bullet \|$ is a p-D distance and x_i and κ_k are p-D.

This gives a penalized spline with a basis and penalty that are invariant to translations or rotations in the coordinate system.

You have to choose $C(\bullet)$ and knots

Thin-plate splines have a polynomial in each row of $\mathbf{X}$ and

$C(\mathbf{r}) = \| \mathbf{r} \|^{2m-d}$ for odd d,

$C(\mathbf{r}) = \| \mathbf{r} \|^{2m-d} \log \| \mathbf{r} \|$ for even d,

where d is the dimension of $\mathbf{r}$, $m >$ degree of polynomial.

A Matérn covariance function implies a C; the simplest are

$C(\mathbf{r}) = \exp(- \| \mathbf{r} \| / \rho)$, $\nu = 0.5$ in Matérn class

$C(\mathbf{r}) = \exp(- \| \mathbf{r} \| / \rho)(1 + \| \mathbf{r} \| / \rho)$, $\nu = 1.5$

where ρ is a scale parameter.

Note: $C(\mathbf{r})$ is increasing (decreasing) in $\| \mathbf{r} \|$ for the
thin-plate (Matérn-based) splines.

Knots: Use a space-filling algorithm.

That's nice . . . I suppose

I have no intuition at all for this construction. I have no idea

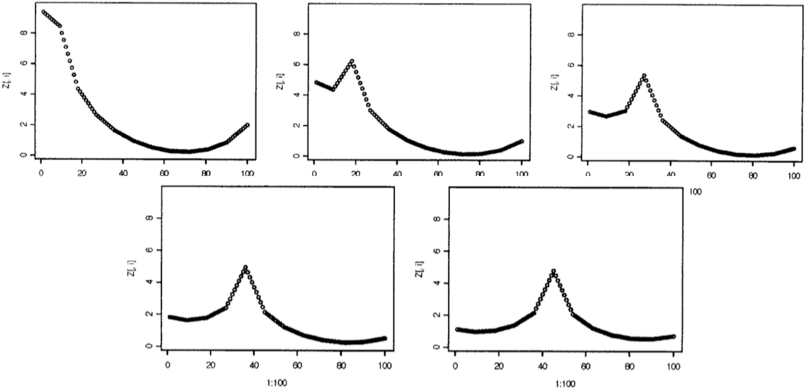
- what the columns of $\mathbf{Z} = \mathbf{Z}_C \Omega_K^{-0.5}$ look like;
- how much they depend on C, e.g., whether C is increasing or decreasing in $\mathbf{r}$; or
- within the Matérn class, how they depend on ν or ρ.

To get some intuition, you can spend a lot of time with a user-hostile literature, or you can look at some examples.

Dumbest, simplest case: 1-D, $C(r) = |r|$

For:
- observations at $\{x_i\} = \{1, 2, \ldots, 100\}$ - 50.5
- knots at $\{100/11, 200/11, \ldots, 1000/11\}$ - 50.5

Here are the first five columns of **Z** (the last 5 are symmetric):



$C(r)$ from a Matérn covariance gives **Z** with similar-looking columns.

Dumbest, simplest 2-D case

For: • $C(r) = \| \mathbf{r} \|$, Euclidean distance.

• observations at $\{x_i\} = \{1, 2, \ldots, 100\}$ - 50.5,
$\{y_i\} = \{1, 2, \ldots, 10\}$ - 5.5

• knots at (20*(1:4), 1+3*(1:2)) - (50.5,5.5)

Here are four columns of **Z** (the other 4 are symmetric):