apts.ac.uk
Academy for PhD Training in Statistics

Flexible Regression

Session 3 - GAMs

Notes:https://warwick.ac.uk/fac/sci/statistics/apts/students/resources/

Slides available at:
https://www.stats.gla.ac.uk/~claire/APTS_FR_session_3.pdf

Claire Miller & Tereza Neocleous

Session 1 - nonparametric regression summary

$$Y_i = f(x_i) + \varepsilon_i, \quad \varepsilon_i \sim \mathrm{N}(0, \sigma^2)$$

- Estimate $f()$ using a regression framework: $\hat{\mathbf{y}} = \mathbf{B}\hat{\boldsymbol{\beta}}$;
- Regression splines fit: $\sum_{i=1}^{n}(y_i - f(x_i))^2$

$$\hat{\boldsymbol{\beta}} = (\mathbf{B}^\top \mathbf{B})^{-1}\mathbf{B}^\top \mathbf{y}$$

- Penalised regression splines fit: $\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda||\mathbf{D}\boldsymbol{\beta}||^2$

$$\hat{\boldsymbol{\beta}} = (\mathbf{B}^\top \mathbf{B} + \lambda \mathbf{D}^\top \mathbf{D})^{-1}\mathbf{B}^\top \mathbf{y}$$

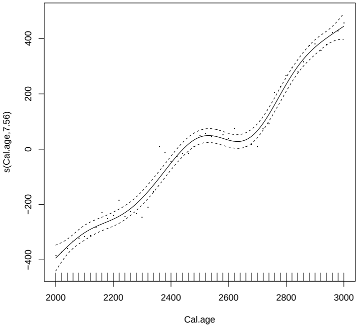
Session 1 - nonparametric regression summary

$$Y_i = f(x_i) + \varepsilon_i$$

- Estimate $f()$ using a regression framework: $\hat{\mathbf{y}} = \mathbf{B}\hat{\boldsymbol{\beta}}$
- Regression splines fit: $\hat{\boldsymbol{\beta}} = (\mathbf{B}^\top \mathbf{B})^{-1}\mathbf{B}^\top \mathbf{y}$
  - Level of smoothing determined by number of basis functions (number of knots and degree (3))
- Penalised regression splines fit: $\hat{\boldsymbol{\beta}} = (\mathbf{B}^\top \mathbf{B} + \lambda \mathbf{D}^\top \mathbf{D})^{-1}\mathbf{B}^\top \mathbf{y}$
  - Level of smoothing determined by using 'too many' basis functions (number of knots and degree (3)) and smoothing through λ.

Session 1 - nonparametric regression

```
library(mgcv)
model <- gam(Rc.age~s(Cal.age), data=radiocarbon)
model
plot(model, residuals=TRUE)
```



What's in this session?

- How much to smooth?
- How to select smoothing parameters?
- Nonparametric regression in higher dimensions
- (Generalised) Additive Models

4.1 How much to smooth?

Fitted values can be expressed as:

$$\hat{\mathbf{y}} = \hat{f} = \mathbf{S}\mathbf{y}$$

Define: **degrees of freedom for model**:

$$\mathrm{df}_{\mathrm{mod}} = \mathrm{tr}\left\{\mathbf{S}\right\}.$$

4.1 How much to smooth?

Regression spline

$$\mathbf{S} = \mathbf{B}(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{B}^\top$$

Penalised regression splines

$$\mathbf{S}_\lambda = \mathbf{B}(\mathbf{B}^\top\mathbf{B} + \lambda\mathbf{D}^\top\mathbf{D})^{-1}\mathbf{B}^\top$$

Define **effective degrees of freedom**:

$$\mathrm{edf}_{\mathrm{mod}(\lambda)} = \mathrm{tr}(\mathbf{S}_\lambda),$$

4.1 How much to smooth?

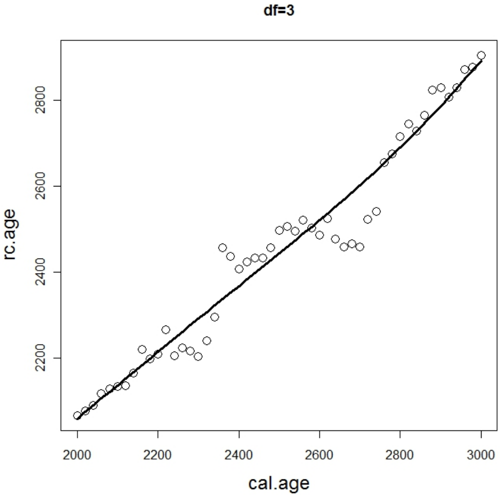


Figure: Radiocarbon data with fit from local linear regression with four different degrees of freedom

4.1 How much to smooth?

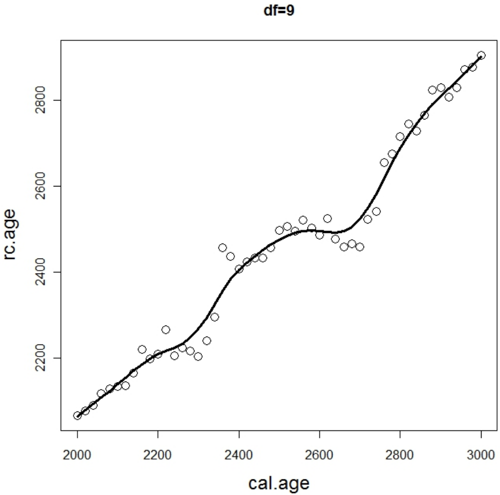


Figure: Radiocarbon data with fit from local linear regression with four different degrees of freedom

4.1 How much to smooth?

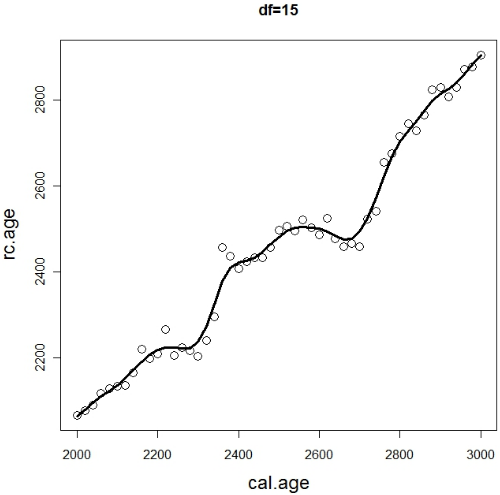


Figure: Radiocarbon data with fit from local linear regression with four different degrees of freedom

4.1 How much to smooth?

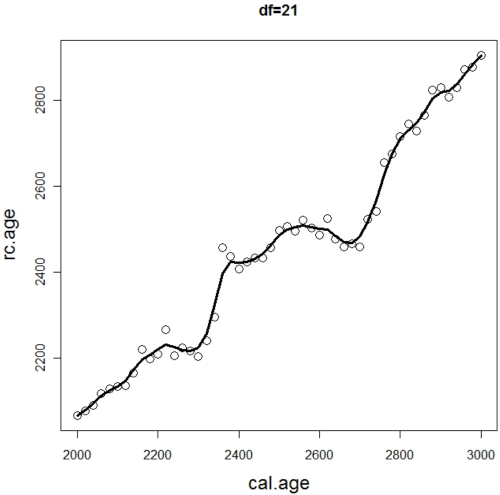


Figure: Radiocarbon data with fit from local linear regression with four different degrees of freedom

4.1 How much to smooth?

Error variance

$$\mathrm{RSS} = \sum \{y_i - \hat{f}(x_i)\}^2.$$

$$\hat{\sigma}^2 = \mathrm{RSS}/\mathrm{df}_{\mathrm{err}}.$$

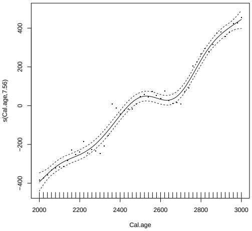
$\mathrm{df}_{\mathrm{err}} = n - \mathrm{tr}(\mathbf{S})$ if $\mathbf{S}^\top = \mathbf{S}$ and $\mathbf{S}^2 = \mathbf{S}$

4.1 How much to smooth?

Standard errors

$$\text{Var}\left\{\hat{f}\right\} = \text{Var}\{\mathbf{Sy}\} = \mathbf{SS}^{\mathsf{T}}\sigma^2$$

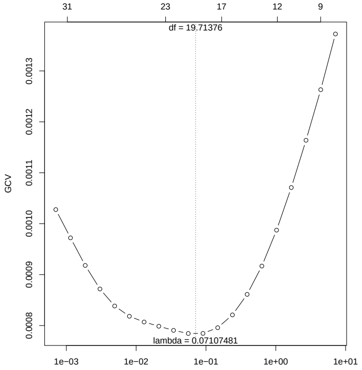
and so, by plugging in $\sqrt{\mathbf{SS}^{\mathsf{T}}\hat{\sigma}^2}_{ii}$ the standard errors at each evaluation point are obtained.



4.2 Automatic methods for smoothing

- We can use the criteria (AIC, AICc, BIC, GCV, CV, . . .) to automatically select smoothing parameters.
- General tendencies:
  - AIC and cross-validation tend to overfit.
  - BIC tends to underfit.
- For penalised regression spline models a mixed-model approach or a Bayesian approach for estimating / averaging over the smoothing parameter (to follow....).

Selecting λ by GCV – Radiocarbon dating



$\lambda = 0.07$ selected as the smoothing parameter in a penalised regression fit.

4.2.1 Random effects interpretation

- We can interpret the penalised regression spline model (2.2) as a random effects model

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda||\mathbf{D}\boldsymbol{\beta}||^2$$

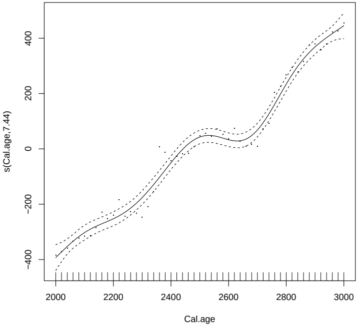
$$||\mathbf{y} - \mathbf{B}\boldsymbol{\beta}||^2 + \lambda||\mathbf{D}\boldsymbol{\beta}||^2$$

- We need to "split" $\boldsymbol{\beta}$ into an unpenalised fixed effect and a penalised random effect.

- Benefit: We can use mixed-model (REML) to estimate $\lambda = \frac{\sigma^2}{\tau^2}$.

4.2.1 Random effects interpretation

```
library(mgcv)
model <- gam(Rc.age~s(Cal.age), method="REML")
```



Comparison of automatic smoothing methods

Method	GCV	REML	ML
edf	7.56	7.44	7.42

4.2.2 Bayesian point-of-view

- Alternatively treat as a fully Bayesian model with priors on σ^2 and τ^2:

$$\begin{aligned}
\mathbf{D}\boldsymbol{\beta}|\tau^2 &\sim \mathrm{N}(\mathbf{0}, \tau^2\mathbf{I}) \\
\mathbf{y}|\boldsymbol{\beta}, \sigma^2 &\sim \mathrm{N}(\mathbf{B}\boldsymbol{\beta}, \sigma^2\mathbf{I}) \\
\sigma^2 &\sim \mathrm{IG}(a_{\sigma^2}, b_{\sigma^2}) \\
\tau^2 &\sim \mathrm{IG}(a_{\tau^2}, b_{\tau^2})
\end{aligned}$$

- Inference can be done by a Gibbs sampler (`BayesX`)

4.3 Nonparametric regression in higher dimensions

We want to develop a spline basis for a model of the form

$$\mathbb{E}(Y_i) = f(x_{i1}, x_{i2}),$$

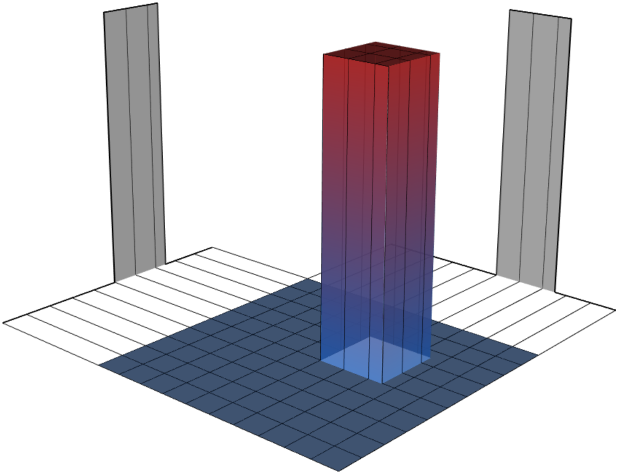
4.3.2 Tensor-product splines

- We will use the following strategy.
  - Place a basis on each dimension separately. $\rightsquigarrow$ Two bases $(B_1^{(1)}(x_{11}), \dots, B_{l_1+r-1}^{(1)}(x_{n1}))$ and $(B_1^{(2)}(x_{12}), \dots, B_{l_2+r-1}^{(1)}(x_{n2}))$
  - Define bivariate-basis functions as

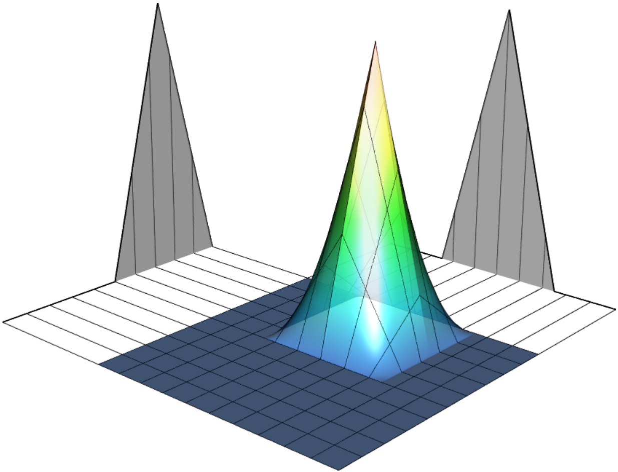
$$B_{jk}(x_1, x_2) = B_j^{(1)}(x_1) \cdot B_k^{(2)}(x_2)$$

for $j \in 1, \dots, l_1 + r - 1$ and $k \in 1, \dots, l_2 + r - 1$.

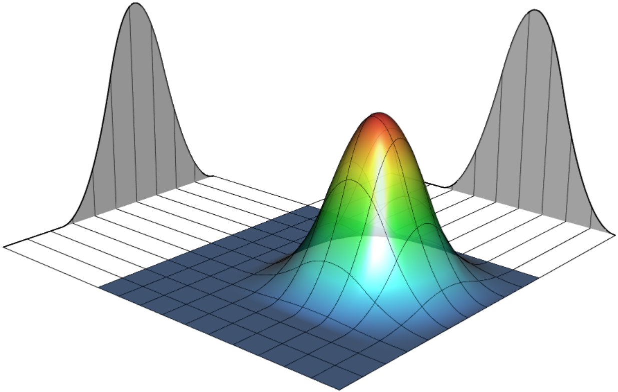
4.3.2 Tensor-product splines: basis degree 0



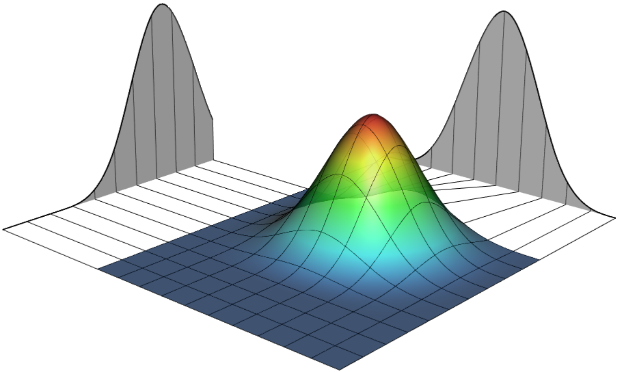
4.3.2 Tensor-product splines: basis degree 1



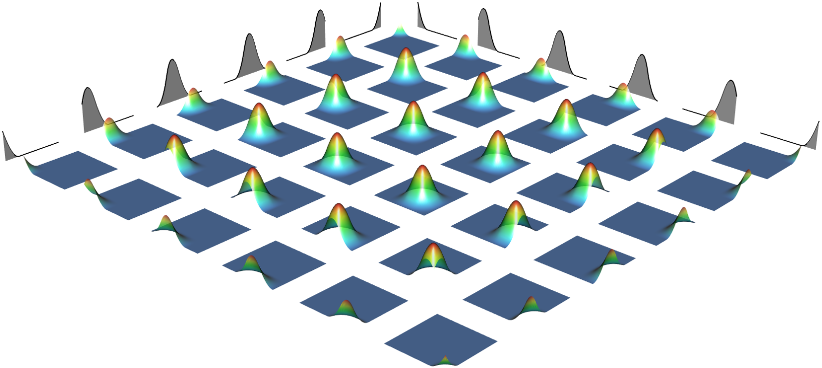
4.3.2 Tensor-product splines: basis degree 2



4.3.2 Tensor-product splines: basis degree 3



4.3.2 Tensor-product splines: entire basis



6 basis functions for each dimension

$\rightsquigarrow 36 = 6^2$ basis functions for the bivariate surface

4.3.2 Tensor-product splines: model fitting

- We will now use the basis expansion

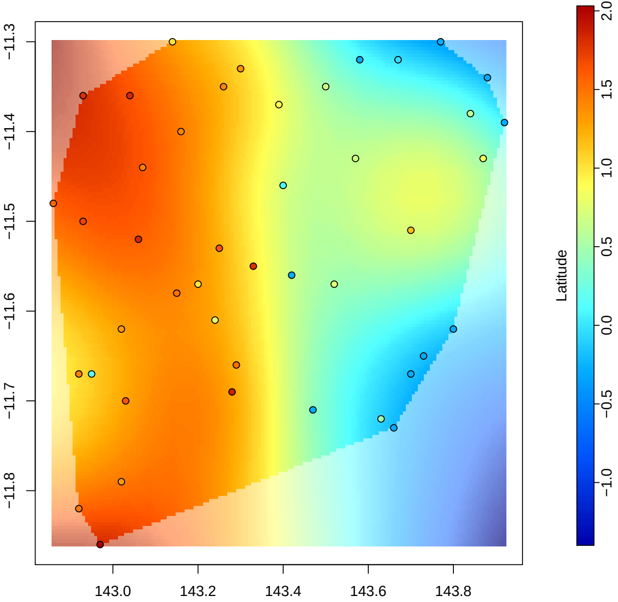
$$f(x_{i1}, x_{i2}) = \sum_{j=1}^{l_1+r-1} \sum_{k=1}^{l_2+r-1} \beta_{jk} B_{jk}(x_1, x_2)$$

- This corresponds to the design matrix

$$\mathbf{B} = \begin{pmatrix} B_{11}(x_{11}, x_{12}) & \dots & B_{1,l_2+r-1}(x_{11}, x_{12}) & B_{21}(x_{11}, x_{12}) & \dots & B_{l_1+r-1,l+2+r-1}(x_{11}, x_{12}) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ B_{11}(x_{n1}, x_{n2}) & \dots & B_{1,l_2+r-1}(x_{n1}, x_{n2}) & B_{21}(x_{n1}, x_{n2}) & \dots & B_{l_1+r-1,l+2+r-1}(x_{n1}, x_{n2}) \end{pmatrix}$$

- We apply univariate penalties to the “rows” and “columns” of the bivariate basis.

4.3.2 Tensor-product splines: Great Barrier Reef



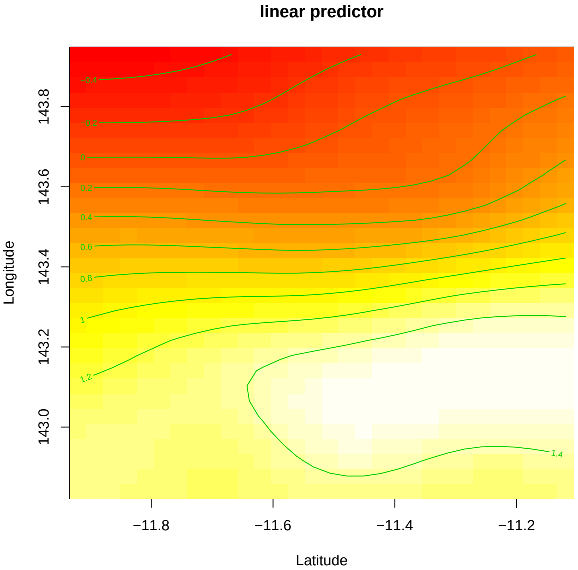
4.3.2 Thin-plate splines - an alternative

Advantage: only one smoothing parameter is estimated (isotrophic smoothness assumption).

Thin-plate splines are the default in `mgcv`'s function `gam`.

```
model <- gam(Score1~s(Latitude, Longitude), data=trawl)
vis.gam(model, plot.type="contour")
```

4.3.2 Thin-plate splines: Great Barrier Reef



4.3.2 Thin plate splines

In fact, we need to minimise the objective function

$$\sum_{i=1}^{n} \left(y_i - f(x_{i1}, x_{i2})\right)^2 + \lambda \boldsymbol{\beta}' \mathbf{R} \boldsymbol{\beta}$$

subject to the constraints that
$\sum_{i=1}^{n} \beta_{2+i} = \sum_{i=1}^{n} x_{i1}\beta_{2+i} = \sum_{i=1}^{n} x_{i2}\beta_{2+i} = 0$, where

$$\mathbf{R} = \begin{pmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & K\left((x_{11}, x_{12}), (x_{11}, x_{12})\right) & \cdots & K\left((x_{11}, x_{12}), (x_{n1}, x_{n2})\right) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & K\left((x_{n1}, x_{n2}), (x_{11}, x_{12})\right) & \cdots & K\left((x_{n1}, x_{n2}), (x_{n1}, x_{n2})\right) \end{pmatrix}$$

4.4 Additive models

$$Y_i = \beta_0 + f_1(x_{1i}) + \ldots + f_p(x_{pi}) + \varepsilon_i, \qquad i = 1, \ldots, n,$$

where the f_i are functions whose shapes are unrestricted, apart from an assumption of smoothness.

We can have:

- More than one covariate;
- Smooth functions can be univariate, bivariate,......;
- Computational challenges can arise for higher dimensions.

Consider the case of only **two covariates**,

$$Y_i = \beta_0 + f_1(x_{1i}) + f_2(x_{2i}) + \varepsilon_i, \qquad i = 1, \ldots, n.$$

4.4 Additive models

A rearrangement of this as:

$$y_i - \beta_0 - f_2(x_{2i}) = f_1(x_{1i}) + \varepsilon_i$$

suggests that an estimate of component f_1 can then be obtained by smoothing the residuals of the data after fitting $\hat{f}_2$,

$$\hat{f}_1 = S_1(\mathbf{y} - \bar{\mathbf{y}} - \hat{f}_2)$$

and that, similarly, subsequent estimates of f_2 can be obtained.

$\rightsquigarrow$ the **backfitting algorithm**.

4.4 Additive models

If a **spline basis** is used, then the backfitting algorithm is not required as we have a form of linear model with a penalty term

$$Y_i = \mathbf{B}\boldsymbol{\beta} + \varepsilon_i$$

The model is fitted by choosing the vector of weights $\boldsymbol{\beta}$ to minimise

$$(\mathbf{y} - \mathbf{B}\boldsymbol{\beta})^\mathsf{T}(\mathbf{y} - \mathbf{B}\boldsymbol{\beta}) + \boldsymbol{\beta}^\mathsf{T} P \boldsymbol{\beta},$$

where the penalty matrix P is of block-diagonal form, constructed from the penalties from the individual model components, with the jth component $\lambda_j \mathbf{D}_j^\mathsf{T} \mathbf{D}_j$, where $\mathbf{D}_j$ is a differencing matrix.

4.4 Additive models

This leads to the direct solution

$$\hat{\beta} = \left(\mathbf{B}^\mathsf{T}\mathbf{B} + P\right)^{-1}\mathbf{B}^\mathsf{T}\mathbf{y}.$$

Constraint for identifiability:

$$\sum_{i=1}^{n} f_j(x_{ij}) = 0$$

for each component j.

All of the fitting methods above can be extended for more than 2 covariates (section 4.5).

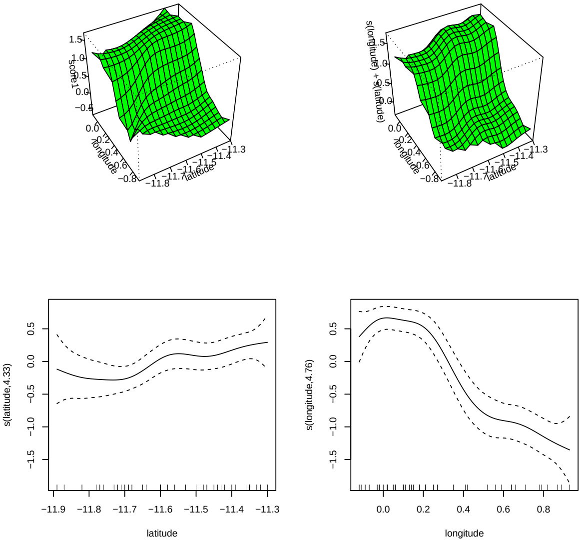
4.4 Additive models - example

Two models fitted to the Reef data:

$$Y_i = f(\mathrm{lat}_i, \mathrm{long}_i) + \varepsilon_i$$

$$Y_i = \beta_0 + f(\mathrm{lat}_i) + f(\mathrm{long}_i) + \varepsilon_i$$

4.4 Additive models - example



4.6 Fitting GAMs

As illustrated previously, one way to fit (Generalised) Additive Models is to use the `mgcv` library in R.

`gam(y~s(x)+s(z)+s(t))`

- `bam`
- `plot(model)`
- many options for different smoothers including cyclic, `bs='cc'`
- multiple family items for non-normal response distributions e.g. `ziP` - zero-inflated poisson
- the default basis functions can be altered, `s(x, k=15)`
- basis dimension and diagnostics can be assessed, `gam.check()`

4.7 Inference - comparing additive models

One approach - approximate F-test:

$$F = \frac{(\mathrm{RSS}_2 - \mathrm{RSS}_1)/(\mathrm{df}_2 - \mathrm{df}_1)}{\mathrm{RSS}_1/\mathrm{df}_1},$$

RSS: $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$, $\mathrm{df} =$ degrees of freedom for error

No general expression for the distribution of this test statistic is available.

Approximate guidance can be given by referring F to an F distribution $((\mathrm{df}_2 - \mathrm{df}_1), \mathrm{df}_1)$.

4.8 Example - Mackerel eggs

A multi-country survey of mackerel eggs in the Eastern Atlantic:

$$\log(\mathrm{density}_i) = \beta_0 + f_1(\mathrm{depth}) + f_2(\mathrm{temp}) + f_{34}(\mathrm{lat}_i, \mathrm{long}_i) + \varepsilon_i,$$

$$\varepsilon_i \sim N(0, \sigma^2)$$

```
model1  <- gam(log(Density) ~ s(log(mack.depth))
                              + s(Temperature)
                              + s(mack.lat, mack.long))
```

4.8 Example - Mackerel eggs

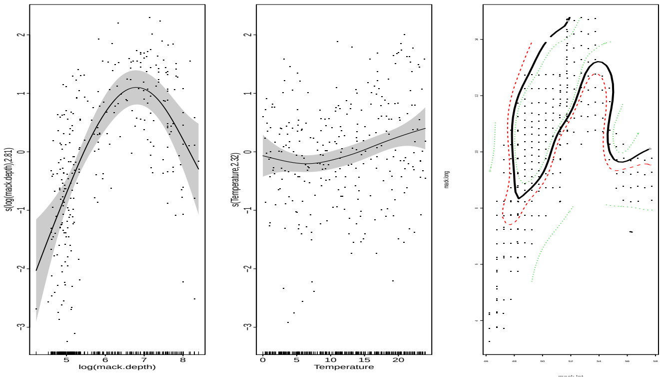


Figure: Depth (left), Temperature (middle), and spatial location (right - longitude (y-axis), latitude (x-axis))

4.8 Example - Mackerel eggs

```
Approximate significance of smooth terms:
                            edf Ref.df      F  p-value
s(log(mack.depth))        2.815  3.538 18.055 9.55e-12
s(Temperature)            2.316  2.904  3.872   0.0147
s(mack.lat,mack.long)    20.197 24.788  5.060 1.03e-12
```

4.8.2 Correlation in GAMs

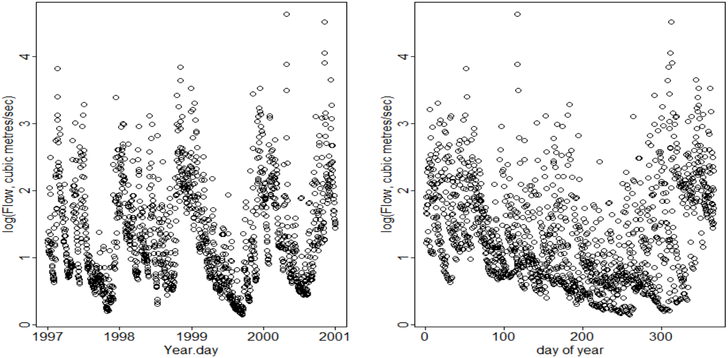
The random effects framework introduced earlier can also be used in order to incorporate, and account for, correlation in GAMs.

(Example 4.5)

- Daily river flow data were collected for a Scottish river between 1997 and 2001.
- It was of interest to investigate the long-term trend and any cyclical patterns in the data.

4.8.2 Correlation in GAMs

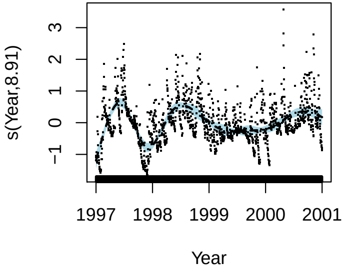
Flow data:



4.8.2 Correlation in GAMs

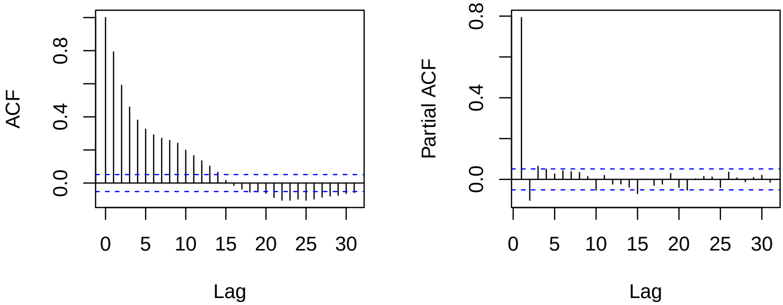
$$\log(\text{flow}_i) = \beta_0 + s(\text{Year}_i) + s(\text{Day of Year}_i) + \varepsilon_i$$

$$\varepsilon_i \sim N(0, \sigma^2)$$



4.8.2 Correlation in GAMs

ACF/PACF of residuals:



4.8.2 Correlation in GAMs

Incoporating correlated errors:

Take, $\varepsilon \sim N(0, V\sigma^2)$ for a correlation matrix V.

Therefore, here we will fit:

$$\varepsilon_i = \phi\varepsilon_{i-1} + \epsilon_i,$$

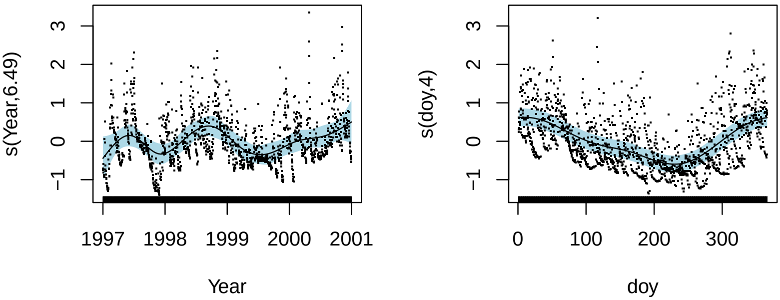
with $\epsilon_i \sim N(0, \sigma^2)$.

Fitting in `R`:

```
gamm(log(Flow)~s(Year,bs="cr")+s(doy, bs="cc"), correlation=corAR1(form=~1))
```

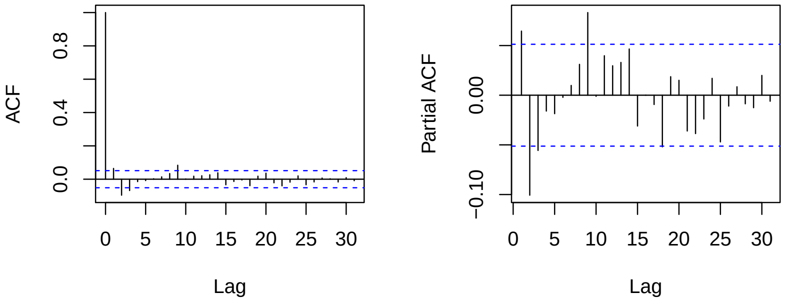
4.8.2 Correlation in GAMs

Fitted models after incoporating correlated errors:



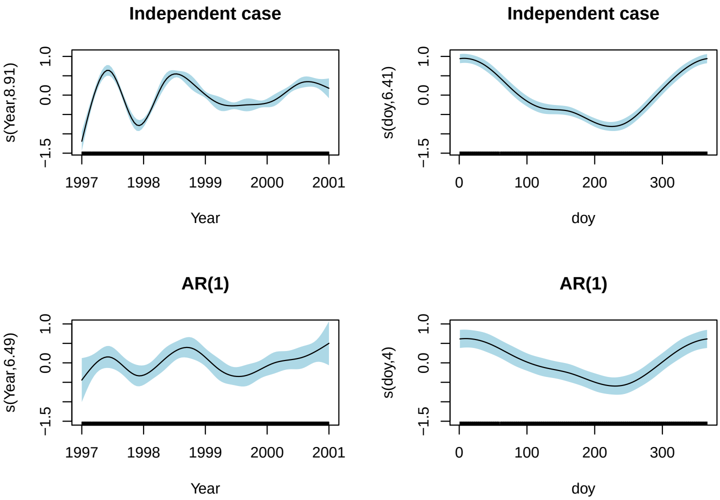
4.8.2 Correlation in GAMs

ACF/PACF of residuals after incorporating correlated errors:



4.8.2 Correlation in GAMs

Fitted models:



4.8.3 Bayesian additive models

A fully **Bayesian approach** can be used extending the ideas in section 4.2.2, including priors for the unknown hyperparameter λ.

The `R2BayesX` package can be used to experiment with this approach.

Reef data example, Fig 4.20:

```
model2 <- bayesx(Score1 ~ sx(Longitude) + sx(Latitude))
```

Summary

What have we covered?

- How much to smooth?
- How to select smoothing parameters?
  - random effect and fully Bayesian implementations
- Nonparametric regression in higher dimensions
- (Generalised) Additive Models