Splines, Knots and Penalties

The Craft of Smoothing

Brian Marx
Paul Eilers

9th School on Regression Models, Sao Paulo, 2005

See www.stat.lsu.edu/bmarx for S-PLUS/ R code and demos

Preface

These are the hand-outs of the slides we will present in our course "Splines, Knots and Penalties. The Craft of Smoothing" at the 9th School on Regression Models, State of Sao Paulo, Brazil during February 2005. We describe in detail the basics and use of P-splines, a combination of regression on a B-spline basis and difference penalties (on the B-spline coefficients).
Our approach is practical. We see smoothing as an everyday tool for data analysis and statistics. We emphasize the use of modern software and we provide functions for R/S-Plus and Matlab.
The notes contain six sessions, time permitting we will cover sessions 1 through 5, only highlighting session 6 :

- Session 1 will present the idea of bases for regression. It will show why global bases, like power functions or orthogonal polynomials are ineffective and why local bases (gaussian bell- shaped curves or B-splines) are attractive.

- In Session 2 penalties will be introduced, as a tool to give complete and easy control over smoothness. The combination of B-splines and difference penalties will be studied for smoothing, interpolation and extrapolation.

- In the first two sessions the data are assumed to be normally distributed around a smooth curve. In session 3 we extend P-splines to non-normal data, like counts or a binomial response. The penalized regression framework makes it straightforward to transplant most ideas from generalized linear models to P-spline smoothing. Important applications are density estimation and variance smoothing.

- Any smoothing method has to balance fidelity to the data and smoothness. An optimal balance can be found by cross-validation or AIC. This subject is studied in Session 4, as well as the computation of error bands of an estimated curve. We also show how optimal smoothing performs on simulated data, to get confidence that it makes the right choices.

- In the first four section we only consider one-dimensional smoothing. When there are multiple explanatory variables, we can use generalized additive models, varying-coefficient models, or combinations of them. Tensor products of B-splines and multi-dimensional difference penalties make an excellent tool for smoothing in two (or more) dimensions.

- Session 6 places P-splines in perspective. It presents Bayesian and mixed model interpretations of P-splines. This session ends with a comparison of the strengths and weaknesses of P-splines and other popular smoothers. It also compares "our" P-splines to a competing approach that uses truncated power functions and ridge penalties. We also consider complications that can occur with correlated noise.

This is the fourth time we will present this course. We have learned a lot from the first times, and changed the material accordingly, but there will probably be some rough edges remaining. We hope that you will not hesitate to confront us with anything that is not clear or that you consider missing or superfluous.
We very much hope that you will find this course useful, and interesting, and enjoyable.

Paul Eilers (p.eilers@lumc.nl)
Brian Marx (bmarx@lsu.edu)

Session 1

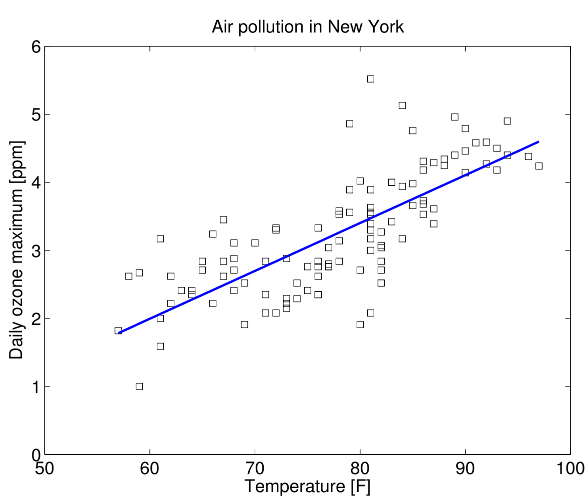
Basics of Bases

Session 1

Basics of Bases

Linear regression line

- Scatterplot: pairs (x_i, y_i), $i = 1 \ldots m$
- Assumption: straight line fits data well
- Equation: $\mu_i = \alpha_0 + \alpha_1 x_i$



How to fit the line

- Least squares: minimize

$$S = \sum_{i=1}^{m} (y_i - \alpha_0 - \alpha_1 x_i)^2 = \sum_{i=1}^{m} (y_i - \mu_i)^2$$

- Matrix notation:

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_m \end{bmatrix} \quad \alpha = \begin{bmatrix} \alpha_0 \\ \alpha_1 \end{bmatrix} \quad \mu = X\alpha$$

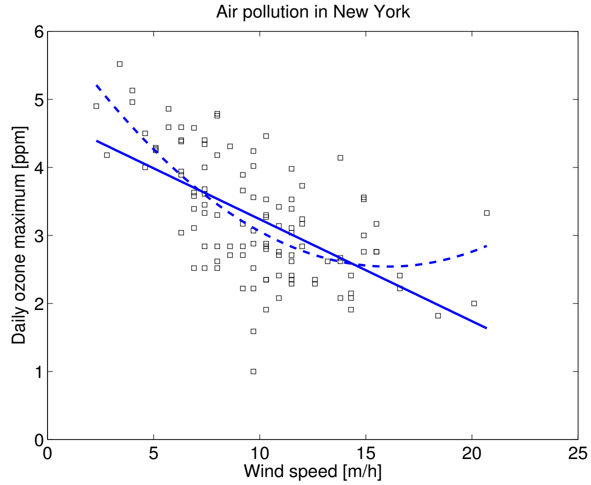
- Minimize $|y - X\alpha|^2 \Rightarrow X'X\hat{\alpha} = X'y \Rightarrow \hat{\alpha} = (X'X)^{-1}X'y$

How to do it in S+/R

```
# Construct matrix with columns 1 and x
X <- outer(x, 0:1, "^")

# Do regression
fit <- lsfit(X, y, intercept = F)
alpha <- fit$coef

# Compute fitted values
mu <- X %*% alpha
```

Curved relationships

- Linear fit not always OK
- Judged by eye, or after studying residuals



Fitting curved relationships

- Linear fit too simple? Add higher powers of x:

$$\mu_i = \alpha_0 + \alpha_1 x_i + \alpha_2 x_i^2 + \alpha_3 x^3 + \ldots$$

- More columns in matrix X

$$X = \begin{bmatrix} 1 & x_1 & x_1^2 & x_1^3 & \ldots & x_1^n \\ 1 & x_2 & x_2^2 & x_2^3 & \ldots & x_2^n \\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ 1 & x_m & x_m^2 & x_m^3 & \ldots & x_m^n \end{bmatrix}$$

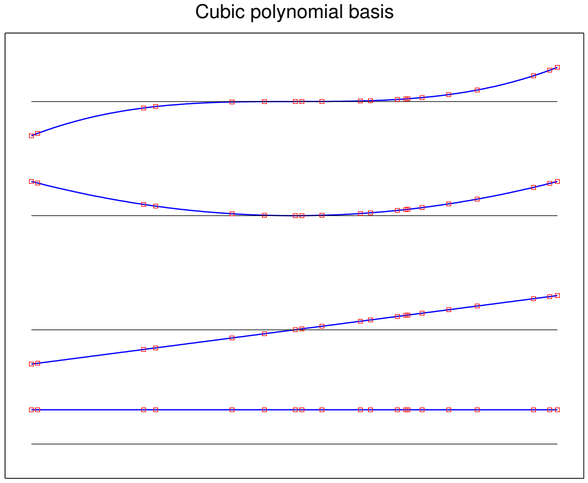
- Same regression equations: $X'X\alpha = X'y \Rightarrow \hat{\alpha} = (X'X)^{-1}X'y$

Higher degree fit in S+/R

```
# Construct matrix with columns of powers of x
n <- 3
X <- outer(x, 0:n, "^")

# Do regression
fit <- lsfit(X, y, intercept = F)
alpha <- fit$coefficients

# Compute fitted values
mu <- X %*% alpha
```

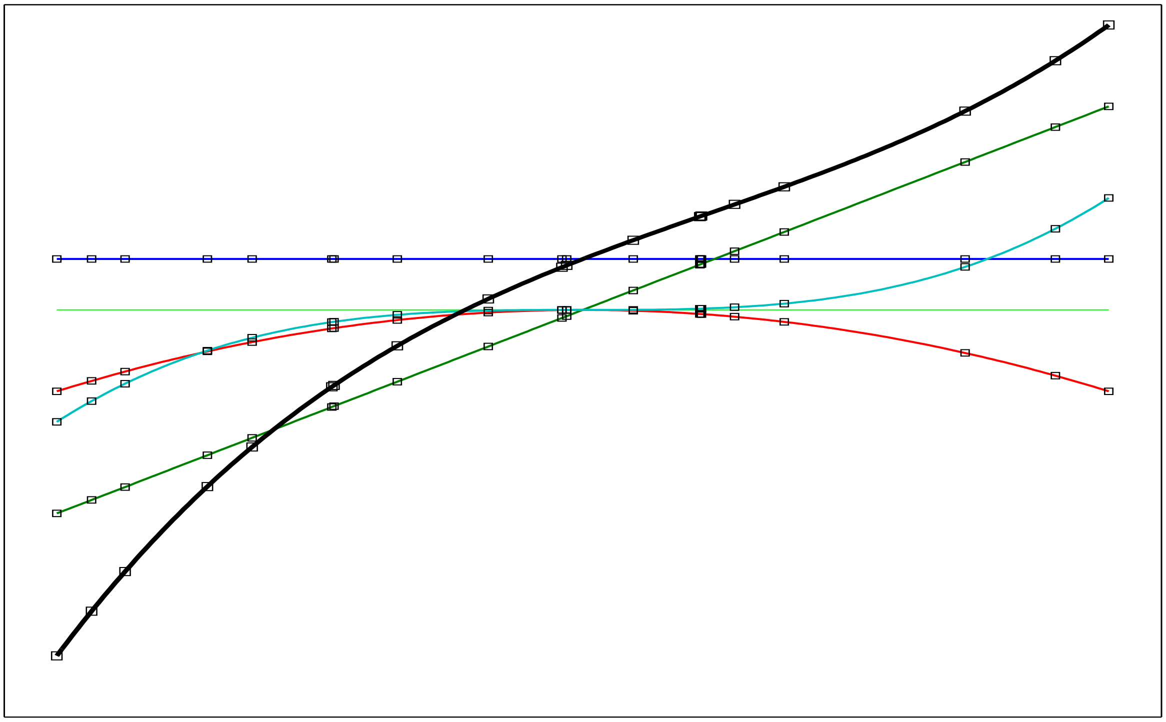

Basis functions

- Regression model $\mu = X\alpha$
- Columns of X: basis functions. Polynomial basis
- With sorted x nice visual representation



Basis functions scaled and added

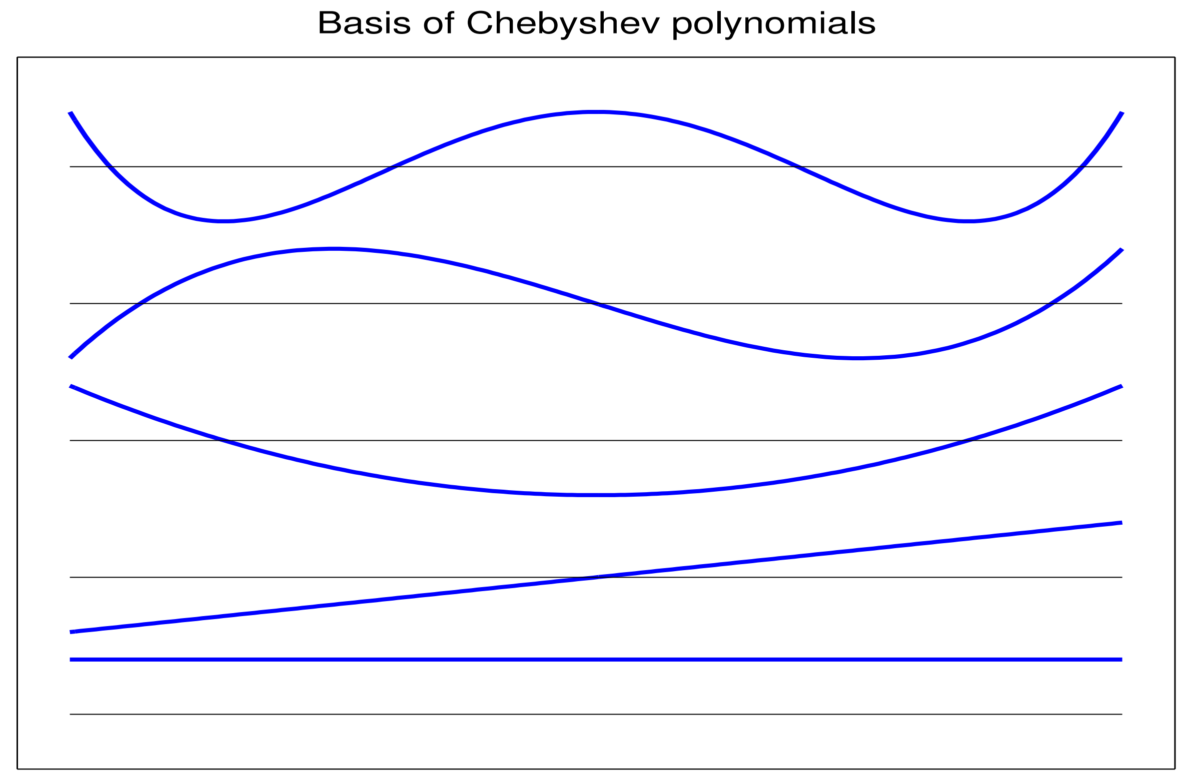
Weighted sum of cubic polynomial basis



Numerical aspects

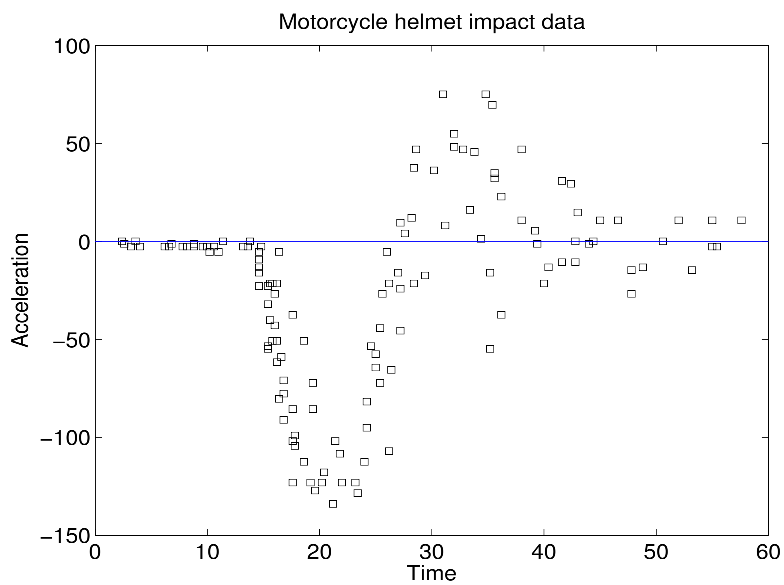
- Higher degree polynomials numerically unstable
- Round-off problems with $\hat{\alpha} = (X'X)^{-1}X'y$
- Partial remedy: center and normalize x
- Better: use orthogonal polynomials
- Eliminate powers up to $p - 1$ from p-th basis function
- Chebyshev nice and easy: $C(x; p) = \cos[(p - 1)\arccos(x)]$

Chebyshev basis



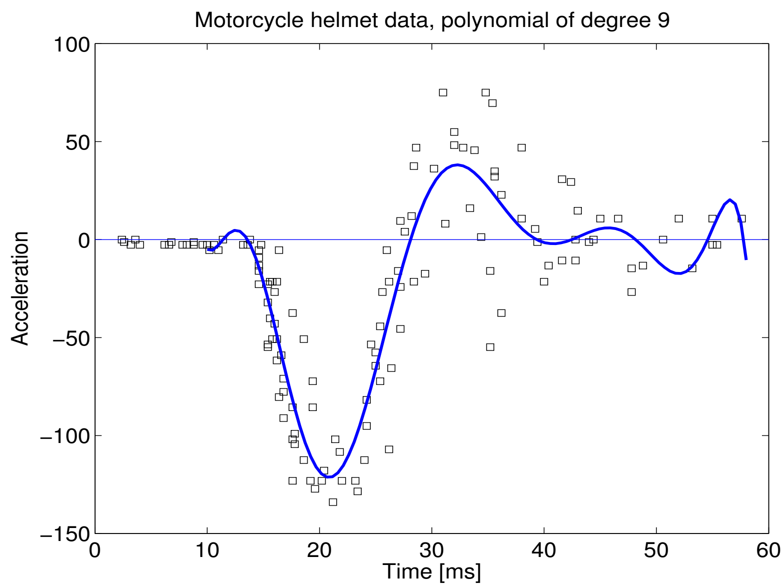
The motorcycle data

- Simulated crash experiment
- Acceleration of motorcycle helmets measured



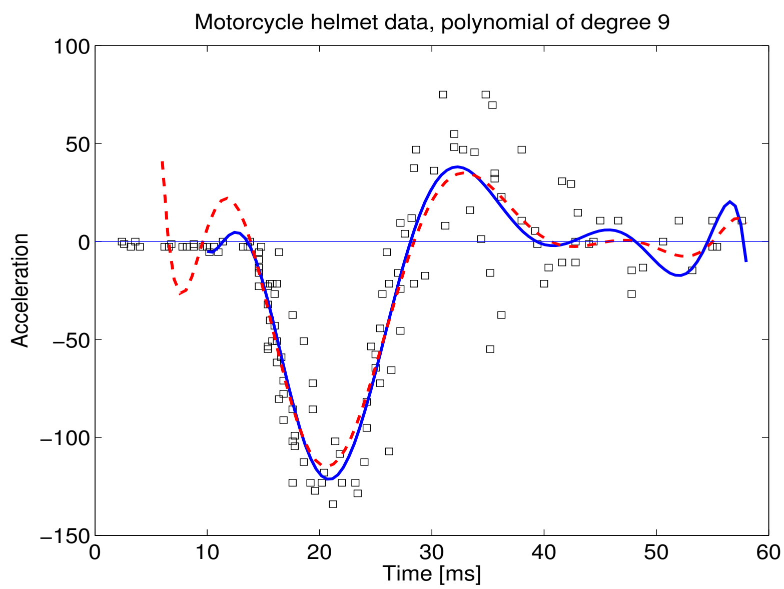
More motorcycle pictures

- High degree needed for decent curve fit
- Bad numerical condition (use orthogonal polynomials)



Sensitivity to data changes

- Longer left part (near zero)
- Notice wiggles

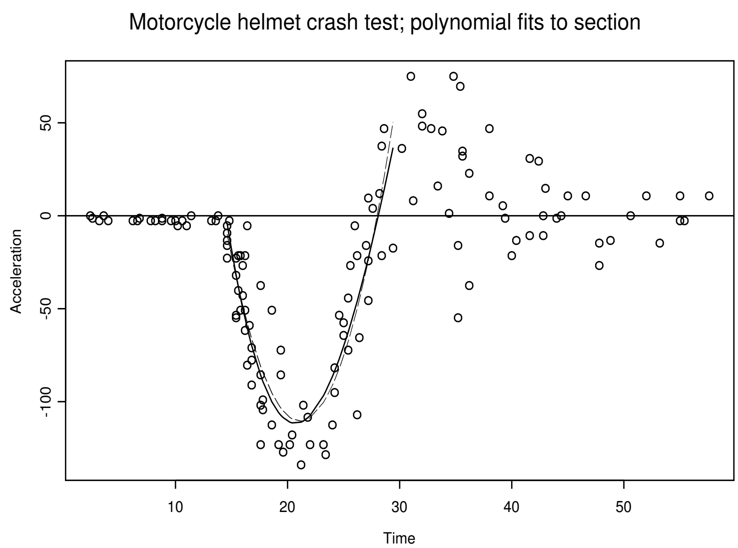


The trouble with polynomials

- High degree (10 or more) may be needed
- Basis functions (powers of x) are global
- Moving one end (vertically) moves other end too
- Good fit at one end spoils things at other end
- Unexpected wiggles
- The higher the degree the more sensitive
- Global polynomials are a dead end

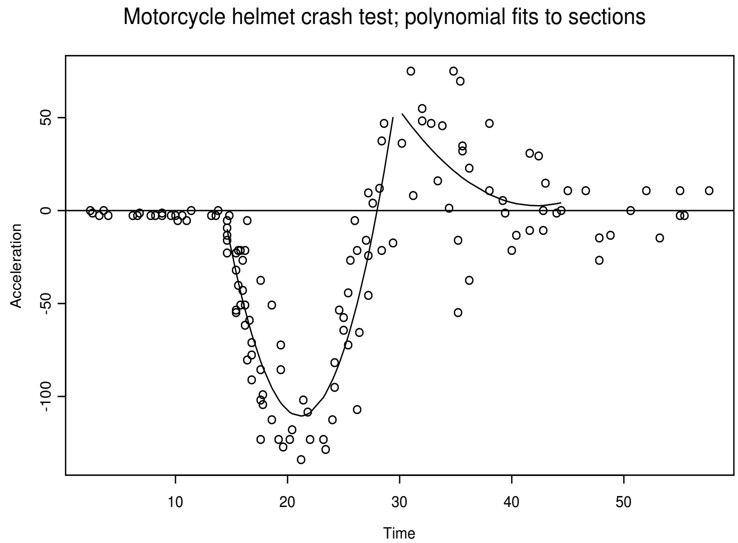
Working with sections

- Fit only small sections with low degree polynomial
- Width of sections?



Neighboring sections

- No nice connection of sections
- Jumps at boundaries



An alternative: local basis functions

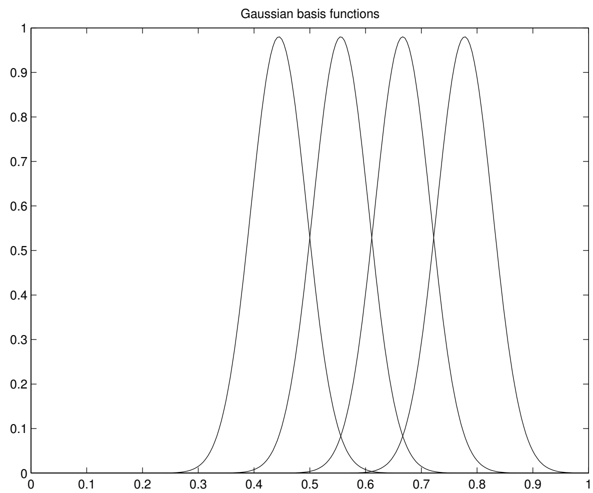
- Get rid of global basis functions
- Local basis functions: non-zero on limited domain
- There they can change freely without harm elsewhere
- Simple example: Gaussian curve, mean τ, SD σ

$$g(x|\tau,\sigma) = \exp[\frac{-(x-\tau)^2}{2\sigma^2}]$$

- Essentially 0 for $|x-\tau| > 3\sigma$
- (No division by $\sigma\sqrt{2\pi}$: peak of g always 1)

Gaussian basis

- A set (basis) of Gaussian functions
- All the same σ, but different τs
- Spacing of τs: 2σ



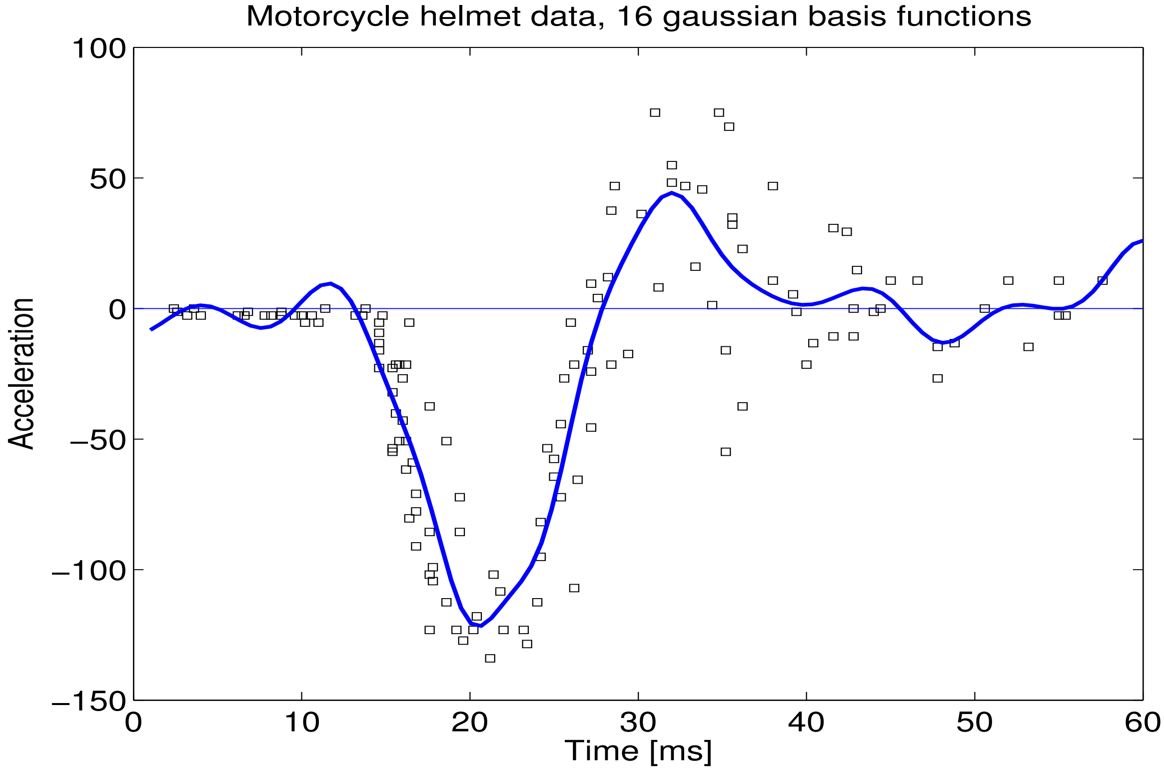
Fitting a curve with Gaussian basis functions

- Basis functions are columns in matrix G
- One row for each x, one column for each τ

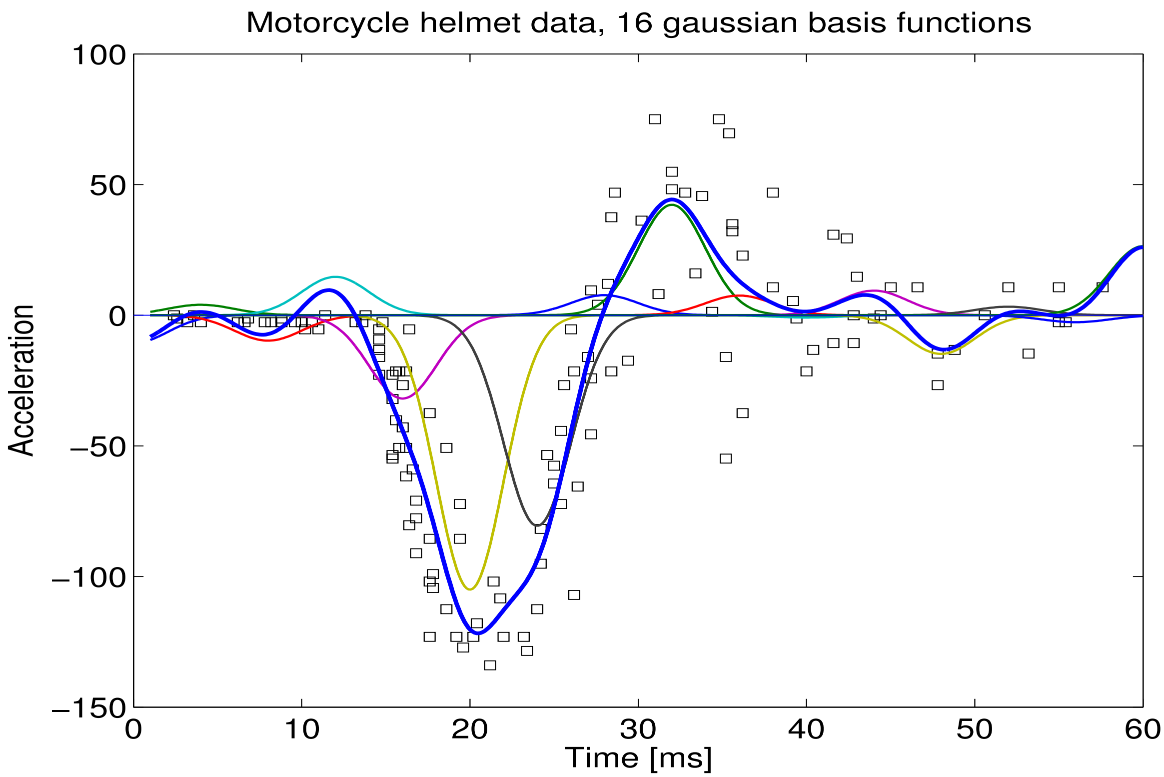
$$g_{ij} = g(x_i|\tau_j, \sigma) = \exp[\frac{-(x - \tau_j)^2}{2\sigma^2}]$$

- Model $E(y) = \mu = G\alpha$
- Linear regression: minimize $S = |y - G\alpha|^2$
- Normal equations $G'G\hat{\alpha} = G'y$
- Explicit solution $\hat{\alpha} = (G'G)^{-1}G'y$

Motorcycle fit with a Gaussian basis



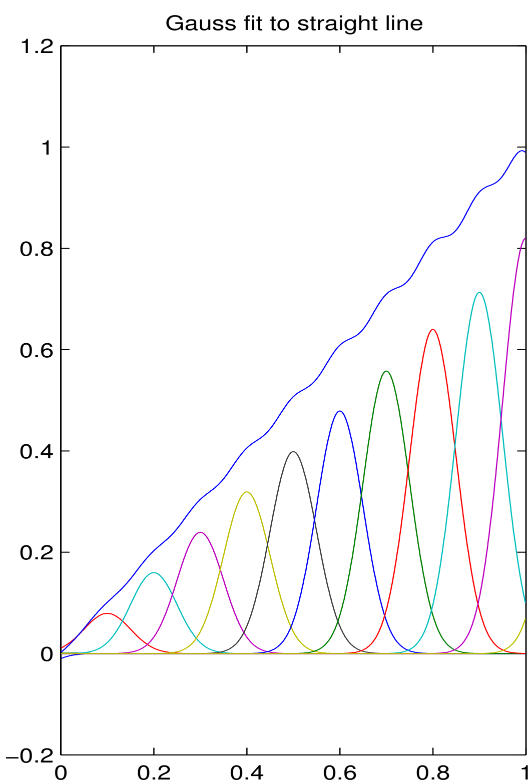
Components of the Gaussian fit

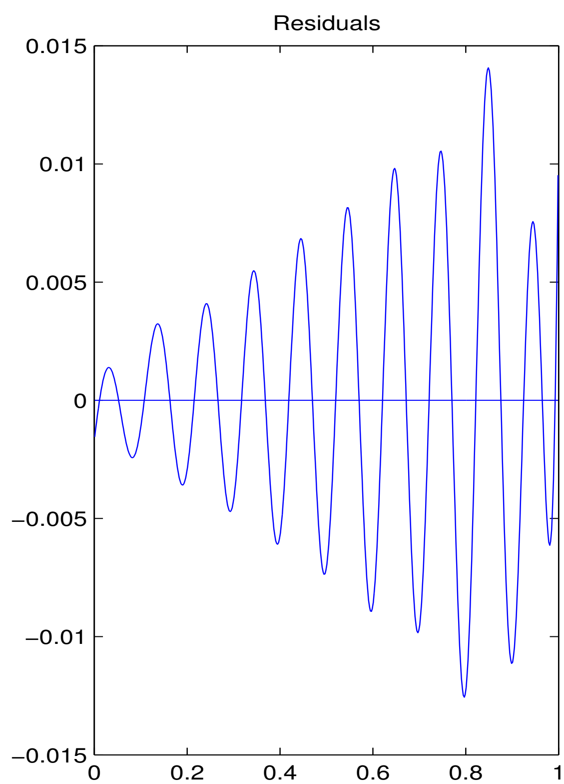


Properties of the Gaussian basis

- Gaussian basis functions are quite practical
- Easy to compute
- Easy to explain
- Disadvantage 1: not really local
- Disadvantage 2: no exact fit to line (polynomial)
- Alternative: B-splines

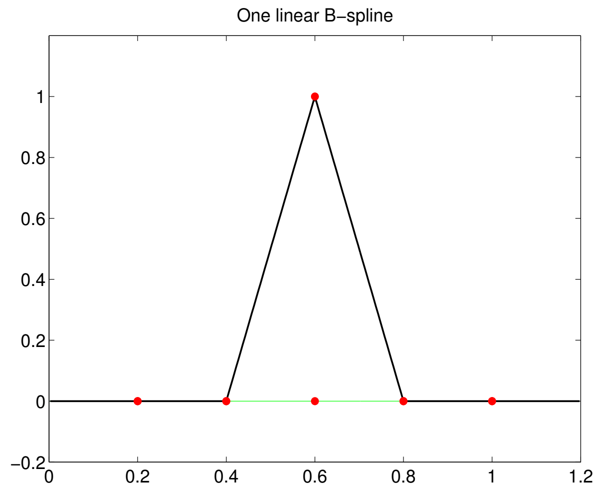
The Gaussian ripple





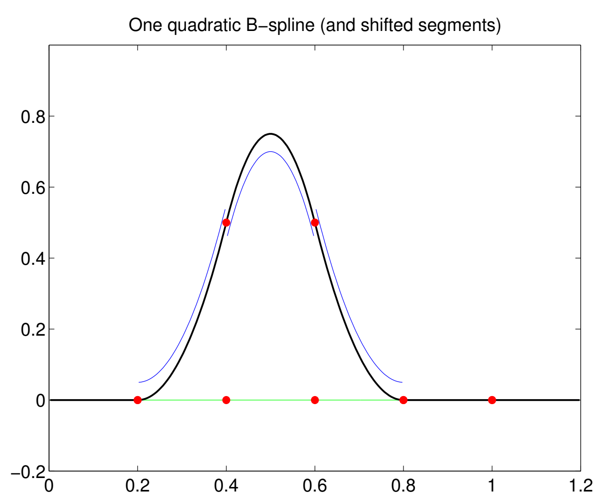
One linear B-spline

- Two pieces, each a straight line, rest zero
- Nicely connected at knots (t_1 to t_3) same value
- Slope jumps at knots



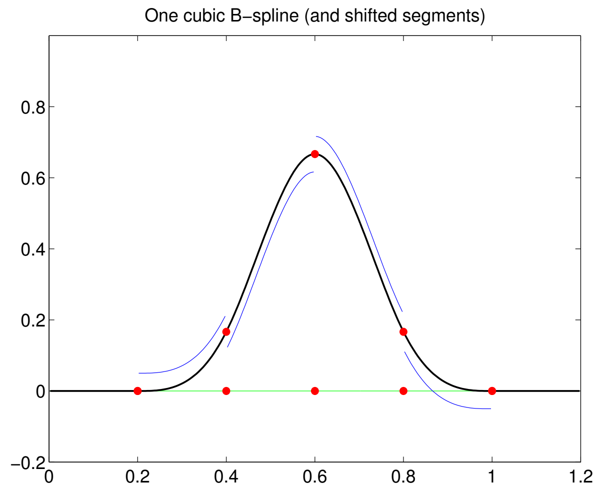
One quadratic B-spline

- Three pieces, each a quadratic segment, rest zero
- Nicely connected at knots (t_1 to t_4): same values and slopes
- Shape similar to Gaussian

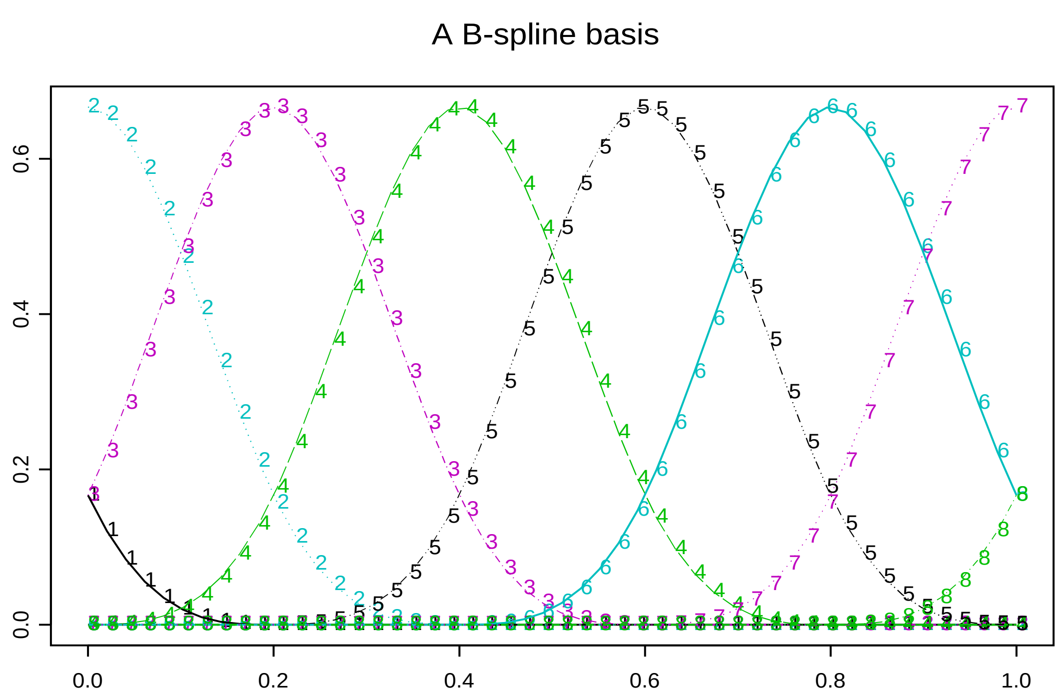


One cubic B-spline

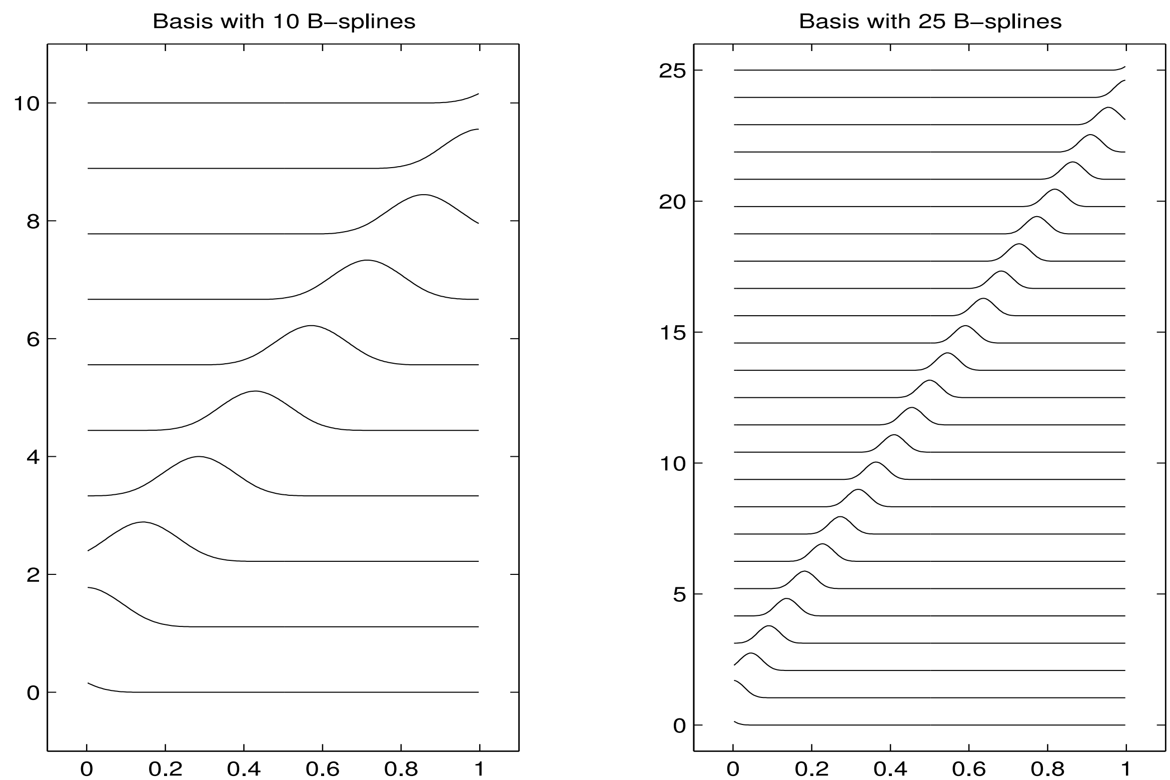
- Four pieces, each a cubic segment, rest zero
- At knots (t_1 to t_5): same values, first & second derivatives
- Shape more similar to Gaussian



A set of cubic B-splines



B-splines in perspective



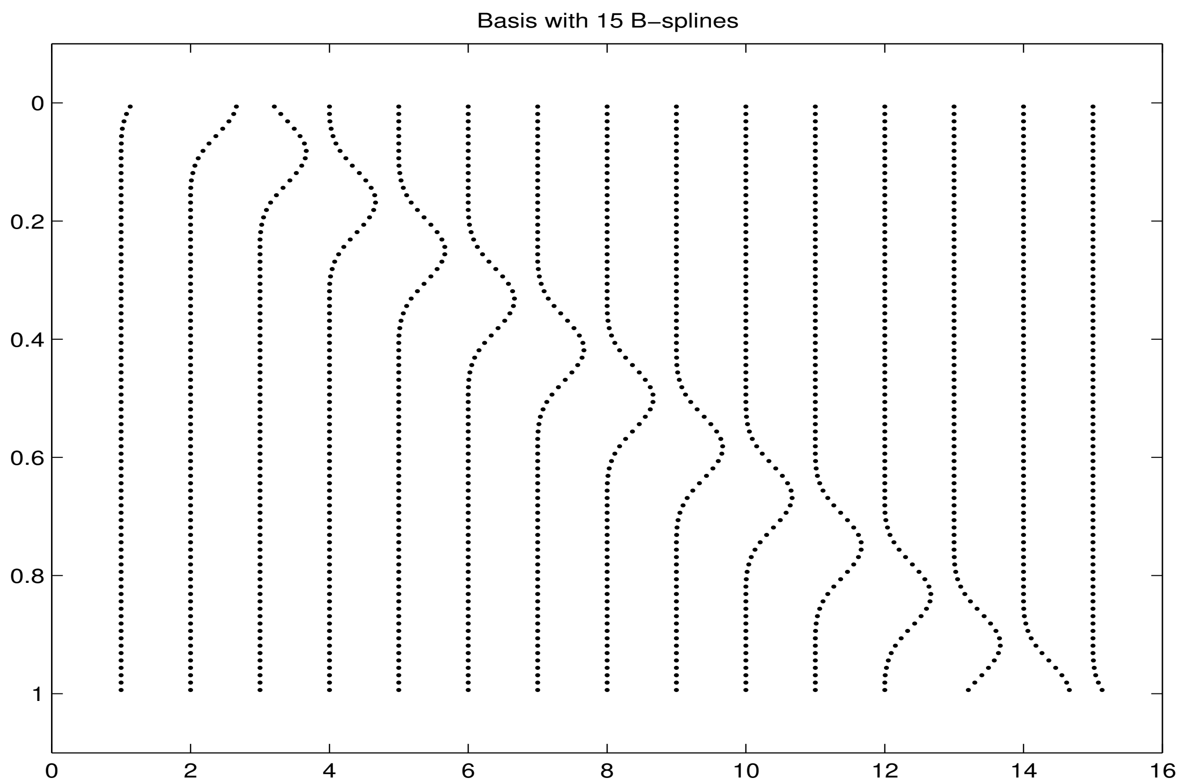
B-spline basis

- Basis matrix B
- Columns are B-splines

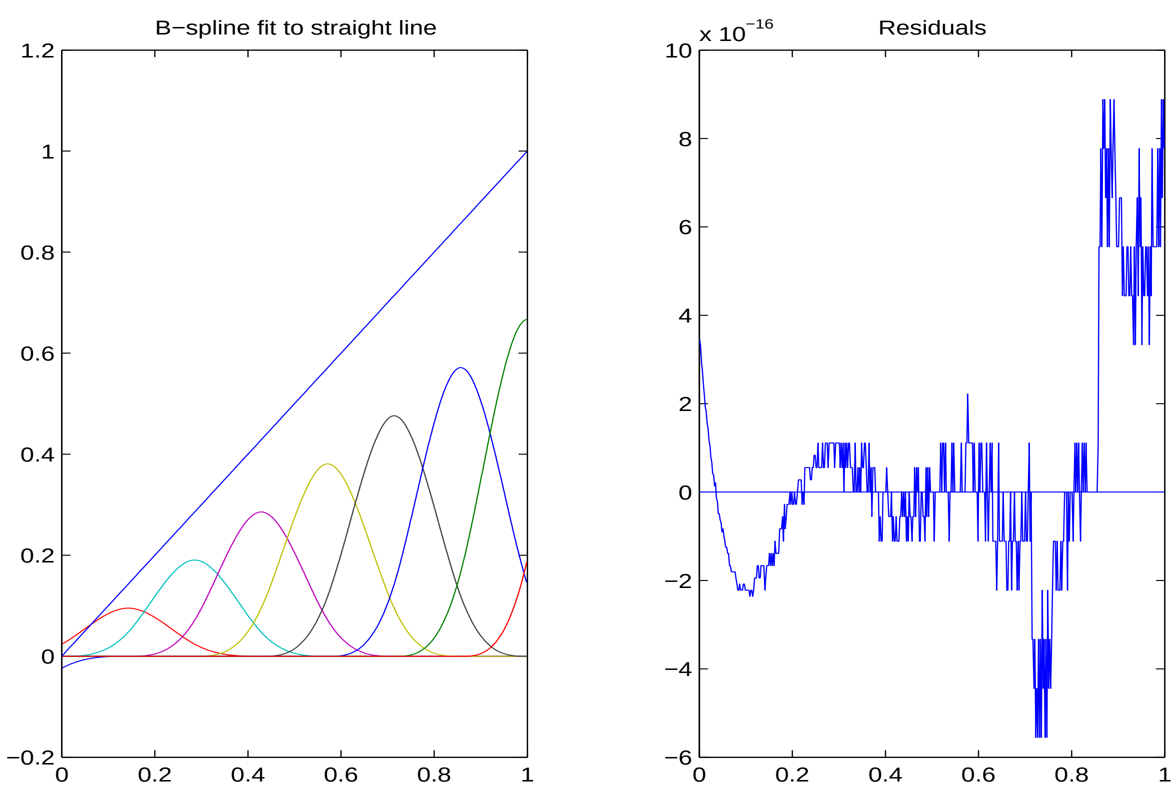
$$\begin{bmatrix} B_1(x_1) & B_2(x_1) & B_3(x_1) & \dots & B_n(x_1) \\ B_1(x_2) & B_2(x_2) & B_3(x_2) & \dots & B_n(x_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ B_1(x_m) & B_2(x_m) & B_3(x_m) & \dots & B_n(x_m) \end{bmatrix}$$

- In each row only a few non-zero elements (degree plus one)
- Only a few basis functions contribute to $\mu_i = \sum b_{ij}\alpha_j = B'_{i\bullet}\alpha$

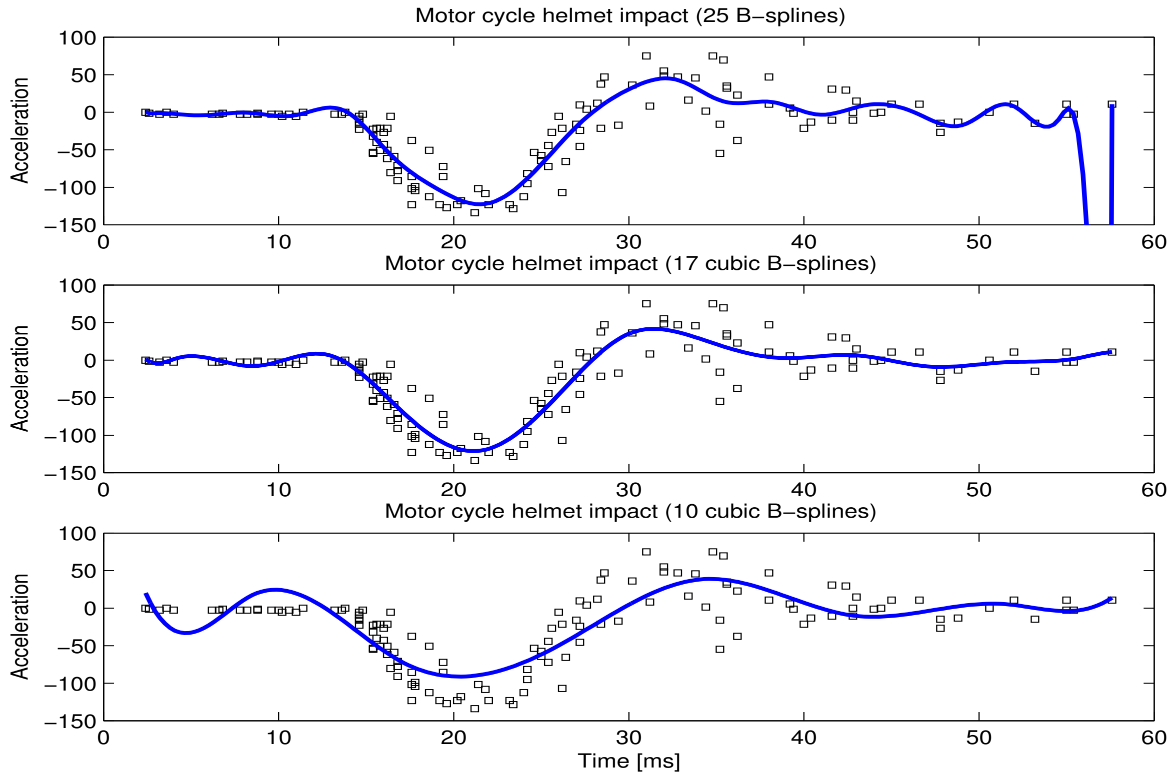
B-splines in rotated perspective



B-splines have no ripple



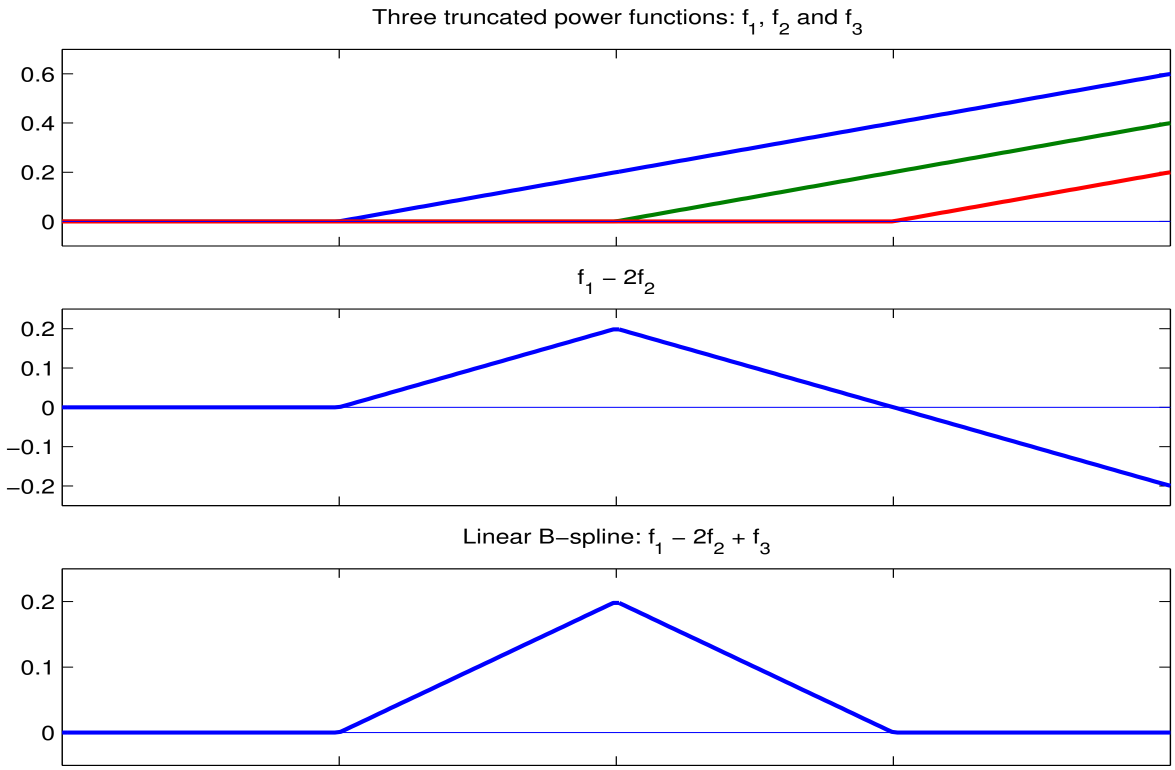
B-splines in action



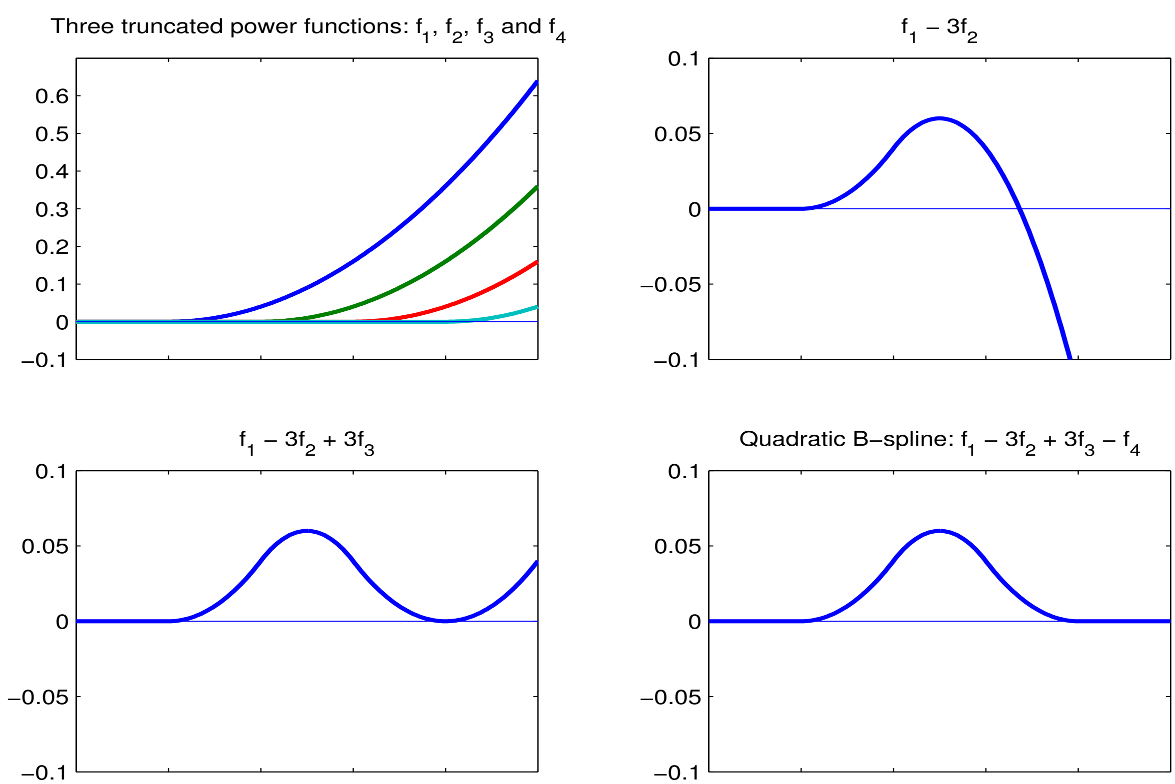
How to compute B-splines

- Work from first principles
- Compute parameters of the polynomial segments
- Nine (3 times 3) coefficients, 8 constraints, height arbitrary
- Easier: recursive formula De Boor
- Yet easier: differences of truncated power functions (TPF)
- TPF: $f(x|t,p) = (x-t)_+^p = (x-t)^p I(x>t)$
- Power function when $x > t$, otherwise 0
- Avoids bad numerical condition of TPF (De Boor)

B-splines and truncated power functions 1



B-splines and truncated power functions 2



B-spline summary

- B-splines are local functions, look like Gaussian
- B-splines are columns of basis matrix B
- Scaling and summing gives fitted values: $\mu = B\alpha$
- The knots determine the B-spline basis
- Polynomial pieces make up B-splines, join at knots
- General patterns of knots are possible
- We consider only equal spacing
- Number of knots determines width and number of B-splines

Wrap-up

- Polynomials essentially useless for complicated curve fit
- Local bases are better
- Gaussian bell curves: to get the idea
- B-splines are better
- B-splines are differences of truncated power functions
- But not ideal: problems with sparse data
- Next session: penalties

Session 2

The Power of Penalties

Session 2

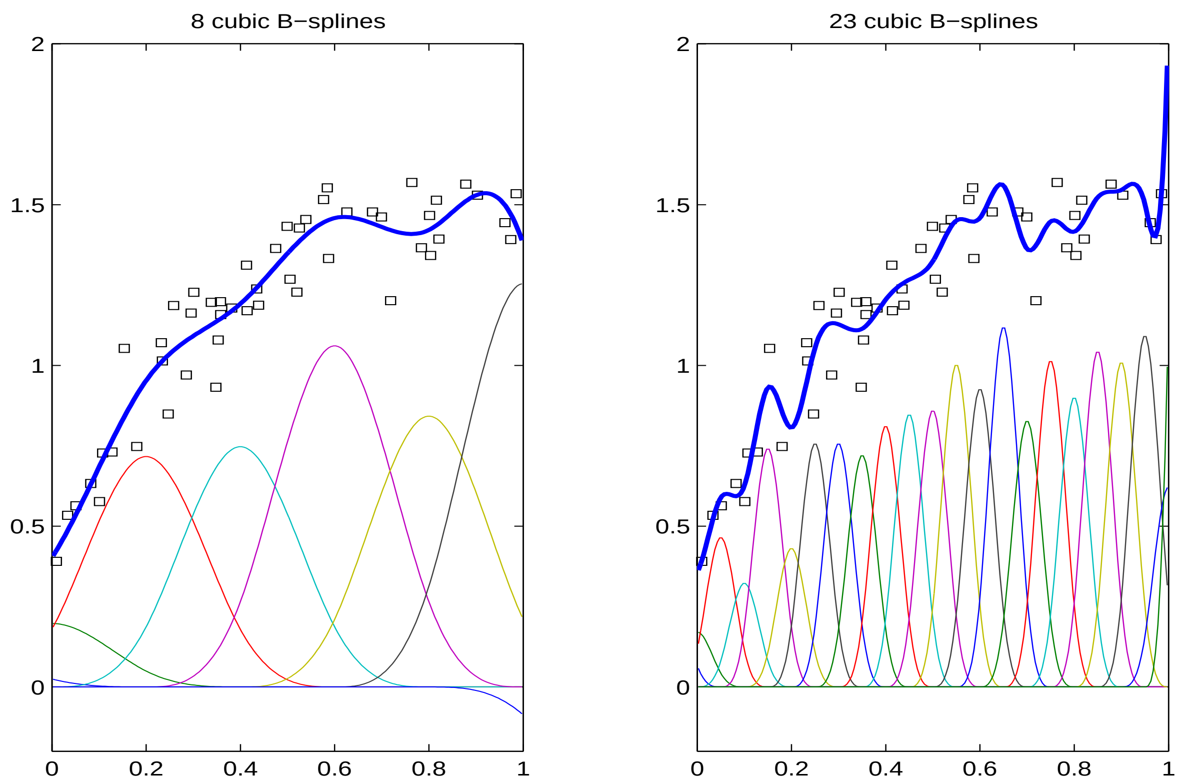
The Power of the Penalty

The Craft of Smoothing 2

B-spline basis revisited

- Regression on local basis functions
- Each basis function is a B-spline
- In a column of a matrix B
- Weighted sum gives fit $\mu = B\alpha$
- More B-splines: more detail in μ possible

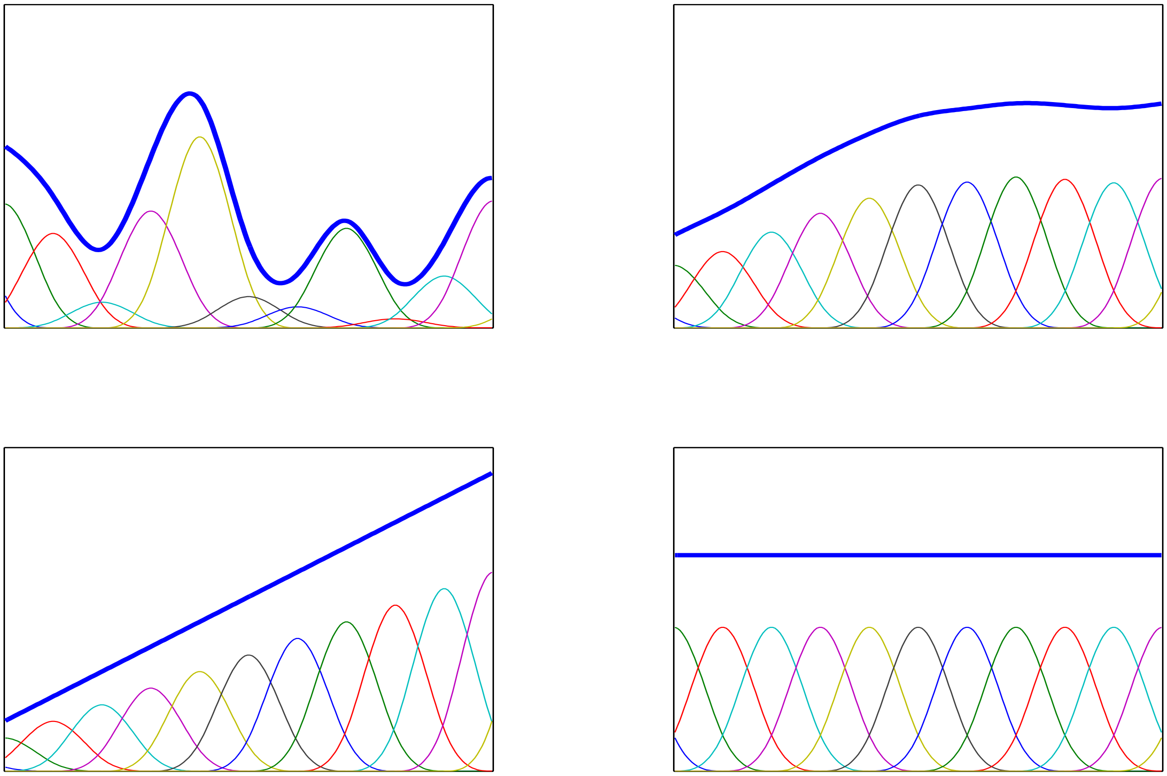
Illustrating the number of B-splines



Smoothness and size of Basis

- More B-splines in basis: more detail is possible
- But it is not necessary!
- Perfectly smooth curves $\mu = B\alpha$ are possible
- It all depends on α
- P-spline idea: control smoothness of α
- Introduce a penalty on roughness of α
- While using a “rich” B-spline basis

Smoothness with many B-splines



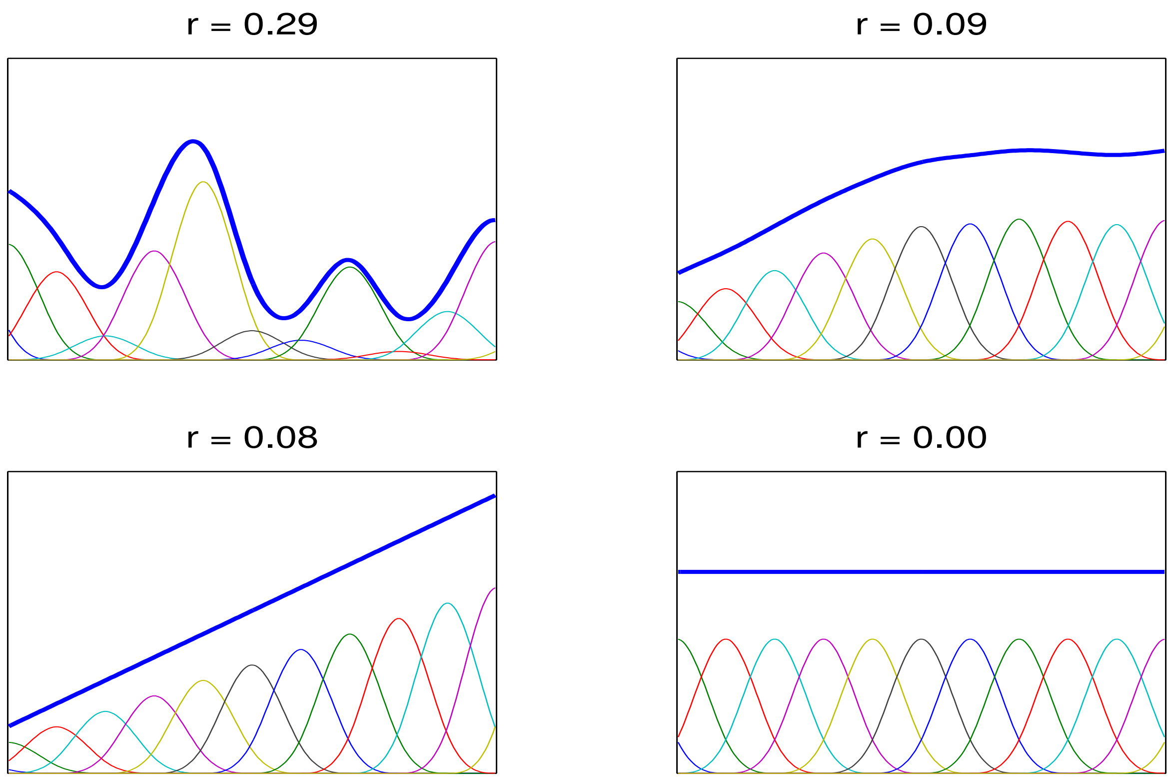
How to measure roughness

- The coefficients determine roughness
- High roughness: α erratic
- Little roughness: smoothly varying α
- Simple numerical measure:

$$R = \sum_{j=2}^{n} (\alpha_j - \alpha_{j-1})^2$$

- Or RMS “change to neighbor”: $r = \sqrt{R/(n-1)}$

Roughness number illustrated



Differences and matrices

- We are interested in $\Delta \alpha_j = \alpha_j - \alpha_{j-1}$
- Special matrix makes life easy:

$$\Delta \alpha = D\alpha; \quad D = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

- D has $n-1$ rows, n columns

Roughness with a matrix

- Roughness measure R:

$$R = |\Delta\alpha|^2 = |D\alpha|^2 = \alpha' D' D\alpha$$

- Matrix D easily computed
- In S+/R: `D <- diff(diag(n))`
- So we have easy tool to express roughness

Penalized least squares

- We set a double goal:
  - good fit to the data: low $S = |y - B\alpha|^2$
  - smooth curve, i.e. low roughness: $R = |D\alpha|^2$
- Balance of the two in one function ($\lambda > 0$):

$$Q = S + \lambda R = |y - B\alpha|^2 + \lambda |D\alpha|^2$$

- Last term known as penalty
- User sets λ (for now, automatic choice later)
- Penalized least squares

Penalized least squares solution

- Minimize

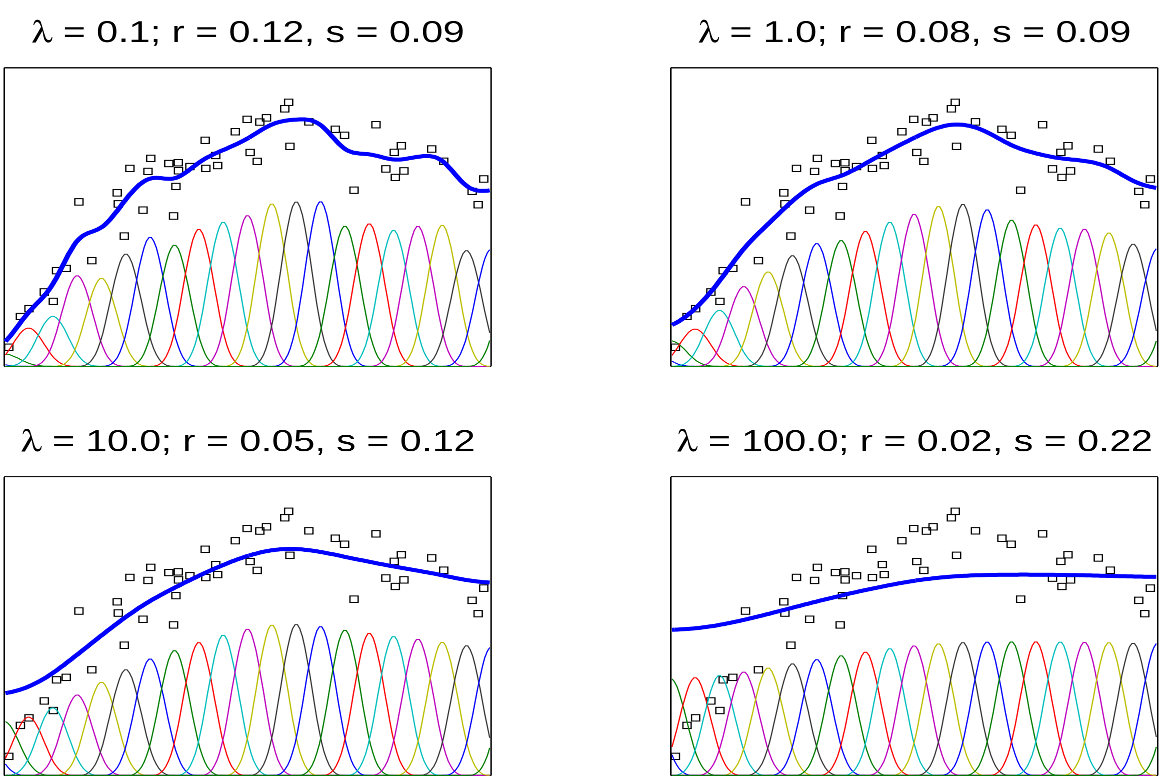
$$
\begin{aligned}
Q &= S + \lambda R = |y - B\alpha|^2 + \lambda|D\alpha|^2 \\
&= y'y - 2\alpha'B'y + \alpha'(B'B + \lambda D'D)\alpha
\end{aligned}
$$

- Set derivative to α zero; result:

$$(B'B + \lambda D'D)\alpha = B'y \Rightarrow \hat{\alpha} = (B'B + \lambda D'D)^{-1}B'y$$

- Small modification of $B'B\alpha = B'y$

The penalty in action



The bias problem

- Increased λ gives smoother curve
- But also it tends to horizontal line
- This bias may prevent enough smoothness
- Solution: use higher order differences

Second order differences

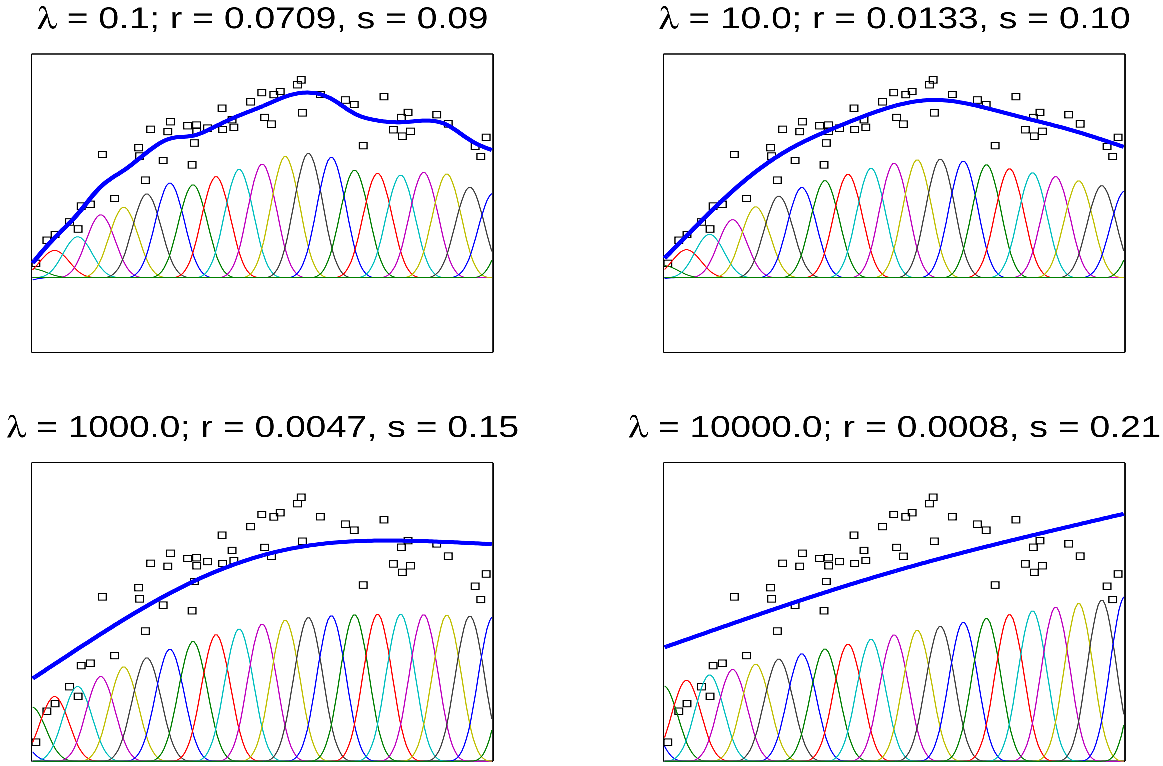
- First order: $\Delta\alpha_j = \alpha_j - \alpha_{j-1}$
- Second order: $\Delta(\Delta\alpha) = \Delta^2\alpha$

$$\Delta^2\alpha_j = (\alpha_j - \alpha_{j-1}) - (\alpha_{j-1} - \alpha_{j-2}) = \alpha_j - 2\alpha_{j-1} + \alpha_{j-2}$$

- Matrix in S+/R: `D <- diff(diag(n), diff = 2)`

$$D_2 = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \end{bmatrix}$$

Second order penalty in action



Higher order differences

- Third order

$$\Delta^3 \alpha_j = \alpha_j - 3\alpha_{j-1} + 3\alpha_{j-2} - \alpha_{j-3}$$

- Matrix

$$D_3 = \begin{bmatrix} -1 & 3 & -3 & 1 & 0 & 0 \\ 0 & -1 & 3 & -3 & 1 & 0 \\ 0 & 0 & -1 & 3 & -3 & 1 \end{bmatrix}$$

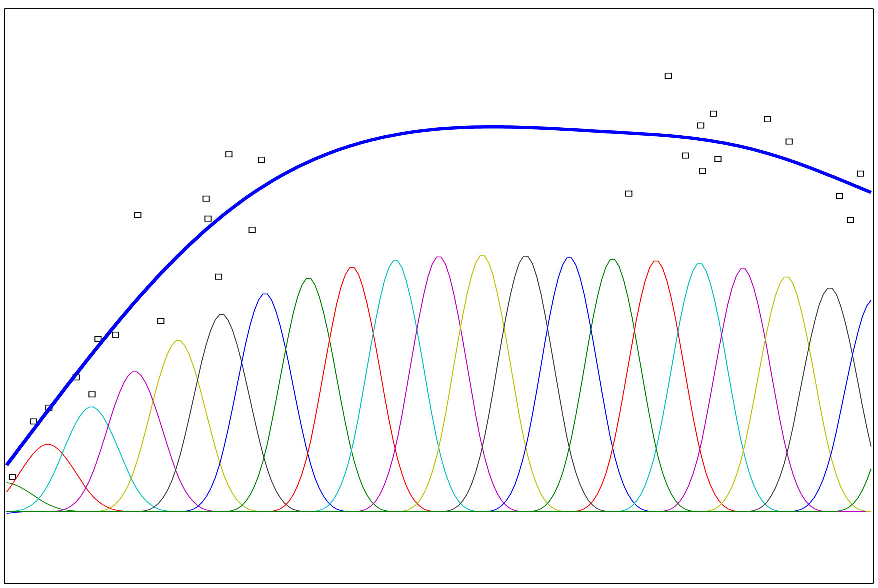
- In S+/R: `D <- diff(diag(n), diff = 3)`

Interpolation without a penalty

- Interpolation with B-splines
  - fit B-splines by regression: $\hat{y} = B\hat{\alpha}$
  - compute B-splines at new $\tilde{x}$: $B_j(\tilde{x})$, $j = 1 \ldots n$
  - $\mu(\tilde{x}) = \sum_j B_j(\tilde{x})\hat{\alpha}$
- Works fine if you can estimate α
- Regression may fail with large gaps in x
- Then some B-splines have no support
- Singular system of equations

Interpolation with a penalty

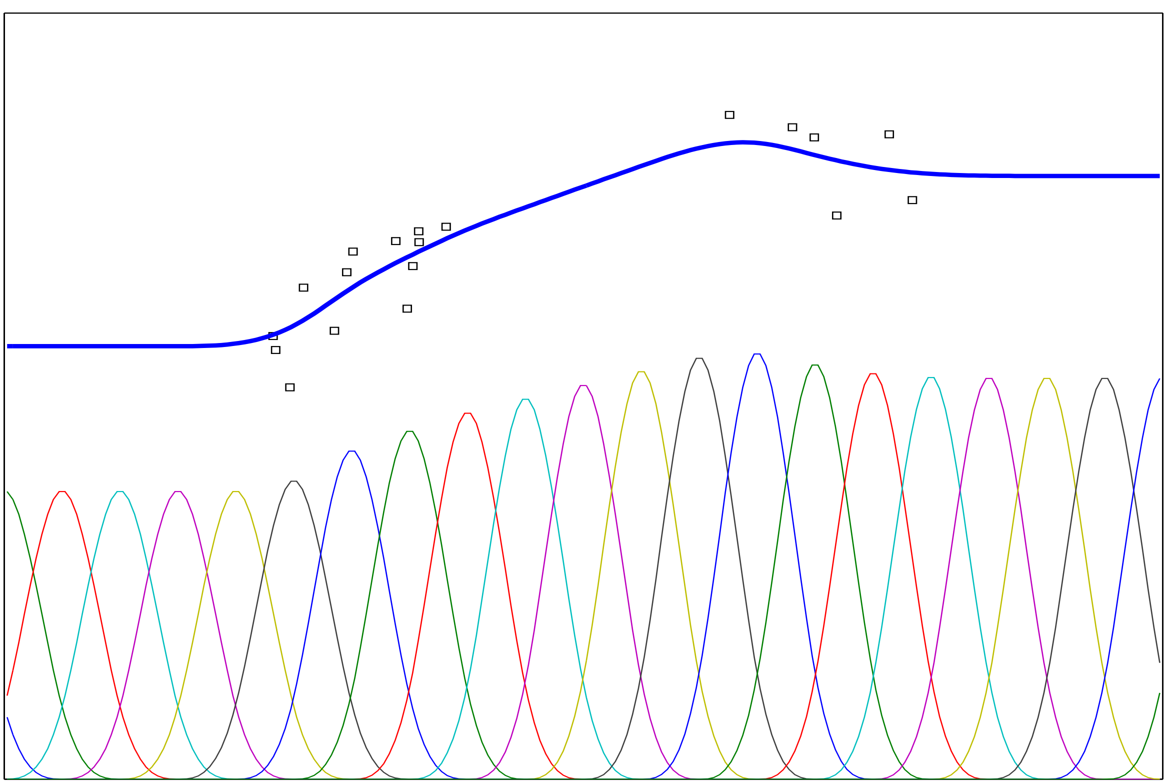
- Penalty let elements of α hold hands
- They bridge the gap(s) automatically!



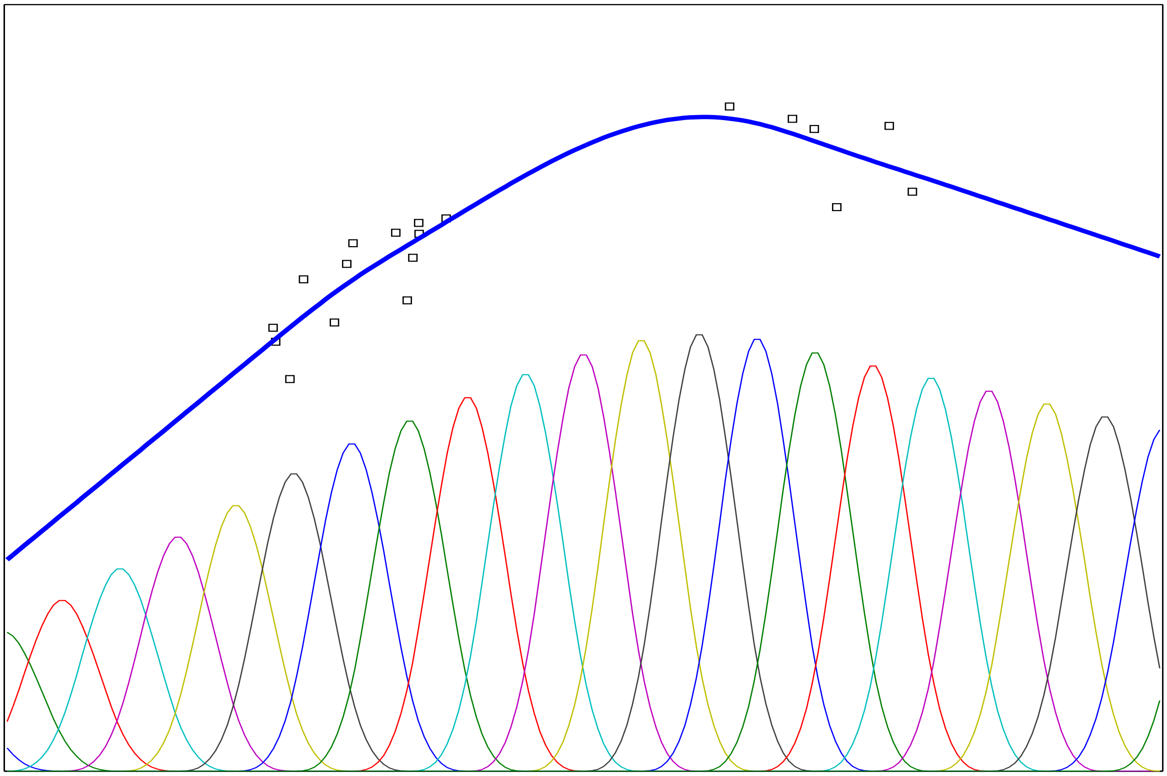
Extrapolation

- Extrapolation works the same
- Just choose the domain of x wide enough
- Take a generous number of (cubic) B-splines
- Penalty again bridges the gap
- Smooth fit and neat extrapolation automatically
- Interpolation (of coefficients): polynomial of degree $2d - 1$
- Extrapolation (of coefficients): polynomial of degree $d - 1$
- With differences of order d in penalty

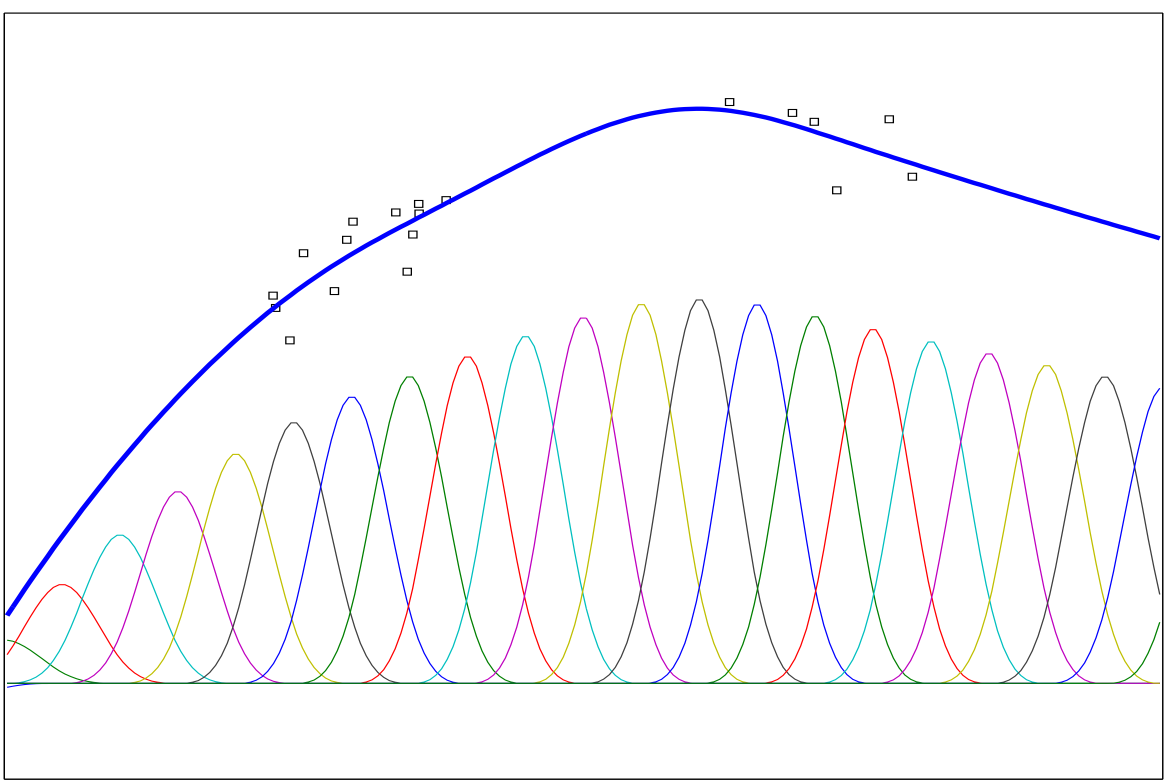
Inter- and extrapolation with $d = 1$



Inter- and extrapolation with $d = 2$



Inter- and extrapolation with $d = 3$



Limits of strong smoothing

- What happens when λ gets very large?
- Penalty term overwhelming in $|y - B\alpha|^2 + \lambda |D_d \alpha|^2$
- Hence $D_d\alpha$ essentially zero
- $D_d\alpha = 0$ if $\alpha_j = \sum_{k=0}^{d-1} \gamma_k \, j^k$
- B-splines property: if α polynomial in j, $B\alpha$ polynomial in x
- Thus fit $\hat{y}$ approaches polynomial in x of degree $d - 1$
- It is the least squares polynomial that minimizes $|y - B\alpha|^2$

Conservation of moments

- Vector v_k, with elements $v_{ik} = x_i^k$, integer k
- In-product $y'v_k$ called k-th moment of y
- Think of classical “method of moments”
- Moment property of P-splines fit ($\mu = B\alpha$)
- For $0 < k < d$ we always have $y'v_k = \mu' v_k$ (for any λ)
- “Conservation of moments”
- Usefulness will be become clear in density estimation

Wrap-up

- Basis with many B-splines allows detail
- But smoothness depends on coefficients α in $B\alpha$
- Difference penalty on α to tune smoothness
- P-splines allow flexible interpolation and extrapolation
- In the limit we get a polynomial
- Moments are conserved (up to order $d - 1$)
- Next session: generalized linear smoothing
- Counts, binary data, gamma distribution (variance)

Session 3

Generalized Linear Smoothing

Session 3

Generalized Linear Smoothing

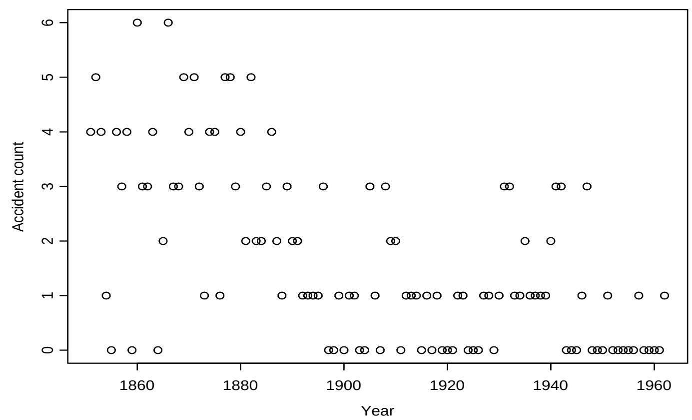
What you will get

- Applications of the generalized linear model [1]
- Specifically: Poisson and binomial P-spline smoothing
- Difference penalty:
- Penalized maximum likelihood estimation
- Goodness-of-fit, effective dimension of model
- (Twice) standard error bands for the mean smooth
- Mean and variance smoothing in scatterplots
- S-PLUS/ R code

[1] Nelder and Wedderburn (1972, *JRSS B*)

Poisson motivating example

- British coal mining data
- Response: accident counts (Poisson)
- Regressor: years 1851-1962 ($m = 112$)

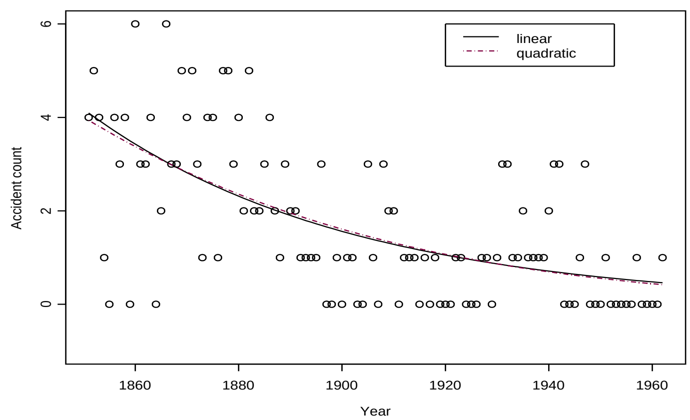


Generalized linear model: 3 components

1. Random component (Y):
   - Identify probability distribution of response Y: $E(Y) = \mu$
   - e.g. Poisson (counts) or binomial (0/1)
   - Independent observations in Y
2. Systematic component (η):
   - Regressors expressed as linear predictor
   - e.g. $\eta = B\alpha$ or $\eta = \alpha_0 + \alpha_1\text{Year}$
3. Monotone link function (g):
   - Links expectation to systematic component: e.g. log, sqrt, identity
   - e.g. $g(\mu) = \log(\mu) = \eta = B\alpha$

Poisson regression example: polynomial basis

- Natural link: $\ln(\mu) = X\alpha = \alpha_0 + \alpha_1 \text{Year}$
- Mean μ is expected value of a Poisson (mean=var, scale=1)
- Inverse link: $\mu = \exp(\alpha_0 + \alpha_1 \text{Year})$, always positive



The Normal model

- Transformed response?
  - Attempt to coerce normal theory methods
  - Objective: transform to normal/ constant variance
  - Transformation may not give both!
- In general, GLMs use maximum likelihood methods
  - Not restricted to normal responses
  - Choice of link separate from response
  - Link need not stabilize variance or produce normality
  - Efficient estimation: use mean/variance relationship
  - MLE $\hat{\alpha}$: approximately Normal, consistent...
- Normal, identity link just special GLM: OLS equivalent

Poisson maximum likelihood estimation (MLE)

- Likelihood function with independent data

$$L = \prod_{i=1}^{m} \frac{e^{-\mu_i}\mu_i^{y_i}}{y_i!}$$

- Maximize $\ln(L) = l$, e.g.

$$l = \sum_{i=1}^{m} \{y_i \ln(\mu_i) - \mu_i - \ln(y_i!)\}$$

- Make l function of α, substitute above with

$$\ln(\mu_i) = x_i'\alpha \qquad \mu_i = \exp(x_i'\alpha)$$

Method of scoring/ Newton-Raphson

- Differentiate $l(\alpha)$ and set to zero (score):

$$0 = \frac{dl}{d\alpha} = \frac{dl}{d\mu}\,\frac{d\mu}{d\eta}\,\frac{d\eta}{d\alpha}$$

- "Normal" equations: $X'(y - \mu) = 0$ (nonlinear in α)
- Linearize and shuffle: $0 = \frac{dl}{d\alpha}|_{\alpha_o} + \frac{d^2 l}{d\alpha d\alpha'}|_{\alpha_o}(\alpha - \alpha_o)$
- Expectation simplifies expression (specifically $\frac{d^2 l}{d\alpha d\alpha'}$ term)
- Solve the (iterative) expression for α

Just a few (gory) details!

$$
\begin{aligned}
0 &= \frac{dl}{d\alpha}|_{\alpha_o} + \frac{d^2 l}{d\alpha d\alpha'}|_{\alpha_o}(\alpha - \alpha_o) \\
&= s_o + H_o(\alpha - \alpha_o)
\end{aligned}
$$

$$
\begin{aligned}
\alpha &= \alpha_o - H_o^{-1} s_o \\
&= \alpha_o + (X'W_oX)^{-1} s_o \quad \text{under expectation} \\
&= (X'W_oX)^{-1} X'W_o\hat{z} \quad \text{shuffle into IWLS form}
\end{aligned}
$$

MLE leads to iterative weighted system/ solution

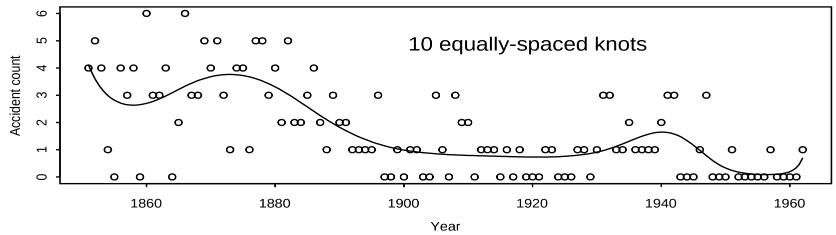
- IRWLS equations:

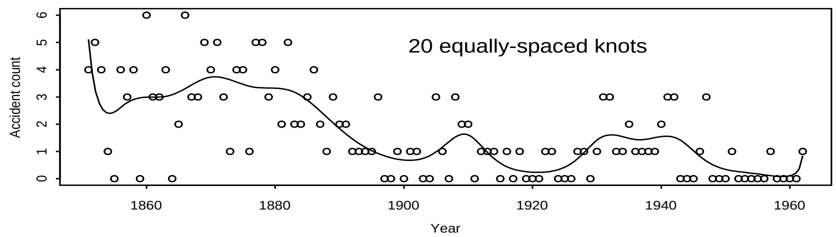
$$(X'\hat{W}X)\hat{\alpha} = X'\hat{W}\hat{z} \quad \Rightarrow \quad \hat{\alpha}_{t+1} = (X'\hat{W}_tX)^{-1}X'\hat{W}_t\hat{z}_t$$

- Starting values on working vector: $\hat{z}_0 = \ln(y + 0.5)$
- $\hat{W}$, $\hat{z}$ are simple functions of $\hat{\alpha}$
- e.g. Poisson (log link)
  - $\hat{W} = \text{diag}\{\hat{\mu}\} = \text{diag}\{\exp(X\hat{\alpha})\}$
  - $\hat{z} = \hat{W}^{-1}(y - \hat{\mu}) + X\hat{\alpha}$
- Other GLM responses/links change $\hat{W}$ and $\hat{z}$

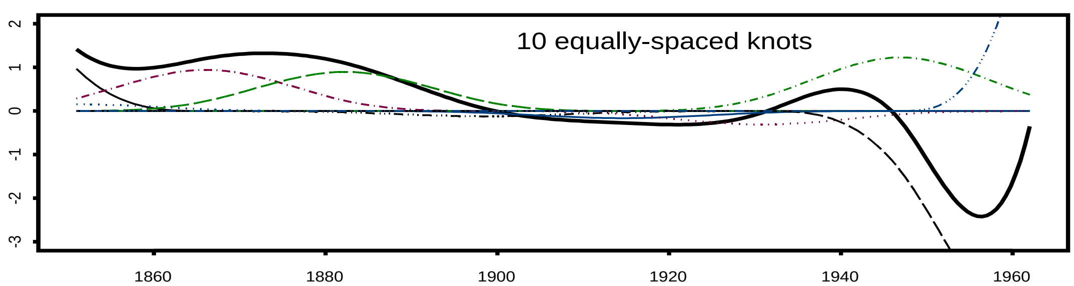
Generalized (cubic) B-spline smoothing

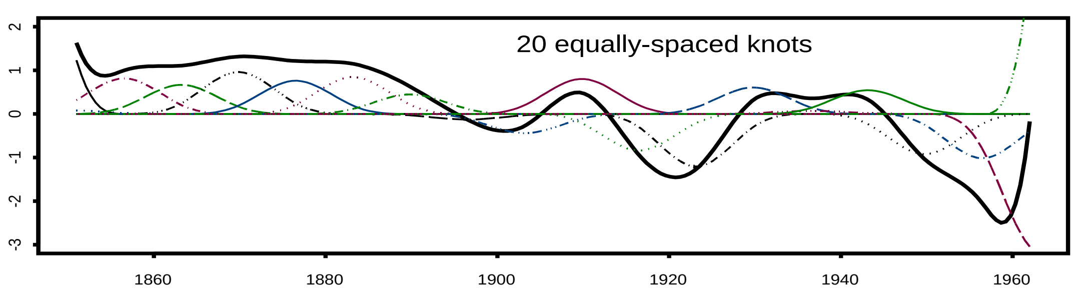
- Now $g(\mu) = \eta = B\alpha$





Skeletal view of $\hat{\eta} = B\hat{\alpha}$





Penalty: generalized P-spline smoothing

- Maximize $l,$ now subject to difference penalty on α

$$l^\star = l(\alpha;\ B,\ y) - \frac{1}{2}\lambda |D_d\alpha|^2$$

- Penalized system of equations:

$$B'(y-\mu) = \lambda D_d' D_d \alpha$$

- Still nonlinear in $\alpha;$ apply method of scoring
- Penalized iterative solution:

$$\hat{\alpha}_{t+1} = (B'\hat{W}_t B + \lambda D_d' D_d)^{-1} B' \hat{W}_t \hat{z}_t$$

Recall: the difference penalty

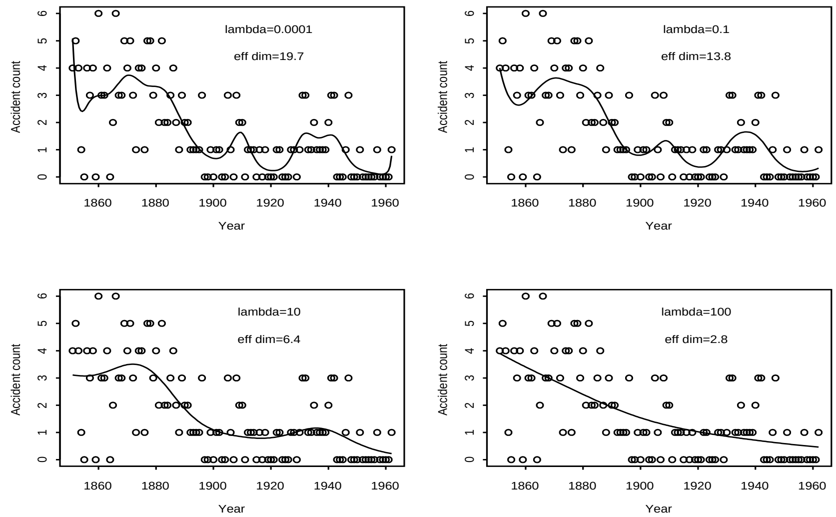
- $|D_d\alpha|^2 = \alpha' D_d' D_d \alpha$
- Regularization penalty: $\lambda \geq 0$, continuous control
- First/second order difference penalty ($d = 1,\ 2$)

$$D_1 = \begin{bmatrix} -1 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & & & & \vdots & & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}$$

$$D_2 = \begin{bmatrix} 1 & -2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & & & & \vdots & & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{bmatrix}$$

Poisson P-spline smoothing

- 20 segments, d=2, vary λ with log link: $\mu = \exp(B\alpha)$



S-PLUS/ R code

```
fit1 <- ppoisson(Year, Count, 20, 3, 2, 0.0001, plot=F)

plot(Year, Count, ylab="Accident count", xlab='Year' )

lines(fit1$xgrid, fit1$ygrid, lwd=2)

text(1920, 5.5, 'lambda=0.0001')

text(1920, 4.5, 'eff dim=19.7')
```

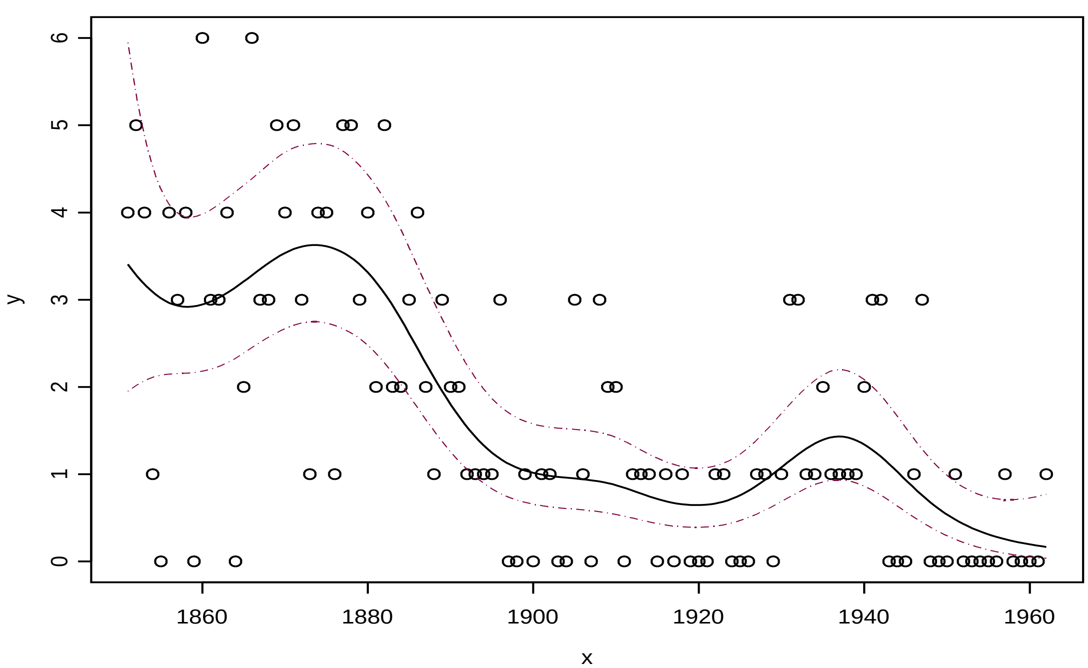
Twice standard error bands

- Sandwich estimator for

$$\begin{aligned}\mathrm{var}(\hat{\eta}) &= \mathrm{var}(B\hat{\alpha}) = \mathrm{var}(H\hat{z}) \\ &= H\,\overbrace{\mathrm{var}(\hat{z})}^{\hat{W}}\,H' \\ &\approx \underbrace{B(B'\hat{W}B + \lambda D_d' D_d)^{-1}B'}_{H}\,\hat{W}B(B'\hat{W}B + \lambda D_d' D_d)^{-1}B'\end{aligned}$$

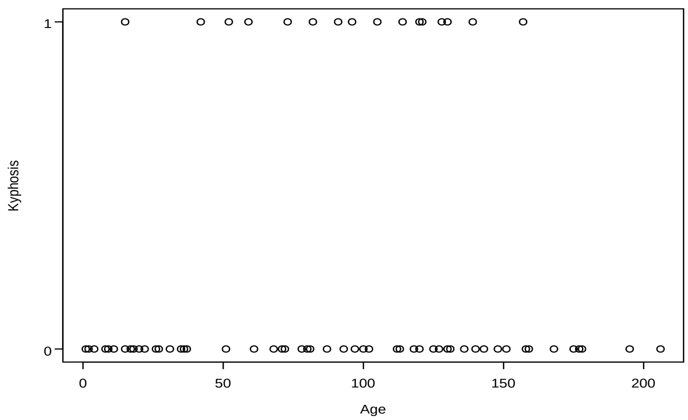
- $\mathrm{se}(\hat{\eta})$: sqrt of diagonal
- $\hat{\eta}$ approximately Normal: $(L,\ U):\ \ \hat{\eta} \pm 2\,\mathrm{se}(\hat{\eta})$
- CI for μ: $[g^{-1}(L),\ g^{-1}(U)]$

Coal mining smooth, twice s.e. bands



Binomial P-spline smoothing

- Kyphosis data ($m = 81$)
- Post-operative deformity: presence (1) or absence (0)
- Regressor: Age (months)

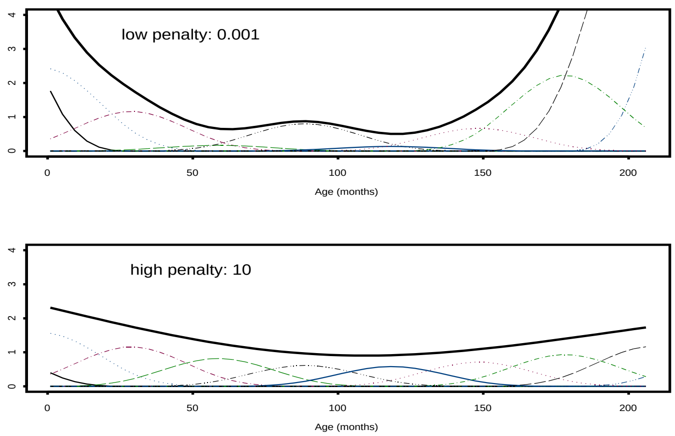


Modified GLM ingredients

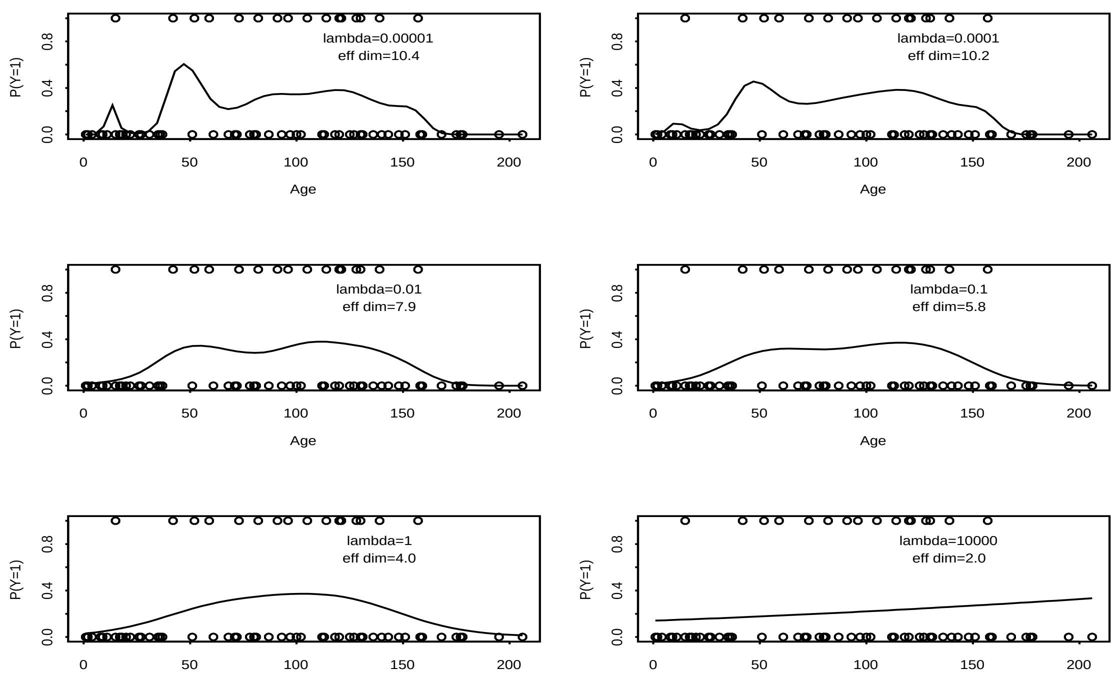
- $g(\mu) = \ln\frac{\pi}{1-\pi} = B\alpha = \eta$
- Response= binomial; link= logit
- Probability inside $(0, 1)$ interval, smooth in Age
- $\mu = \pi = \frac{\exp(B\alpha)}{1+\exp(B\alpha)}$
- $W = \mathrm{diag}\{\pi(1 - \pi)\}$
- $z = W^{-1}(y - \pi) + B\alpha$
- Apply iterative solution
- Efficient parameter estimation

Skeletal view of linear predictor

- B: cubic B-spline basis (10 segments), $d = 2$
- Kyphosis: skeletal view of $-B\hat{\alpha}$



P-spline smooth: 10 segments, $d = 2$, vary λ



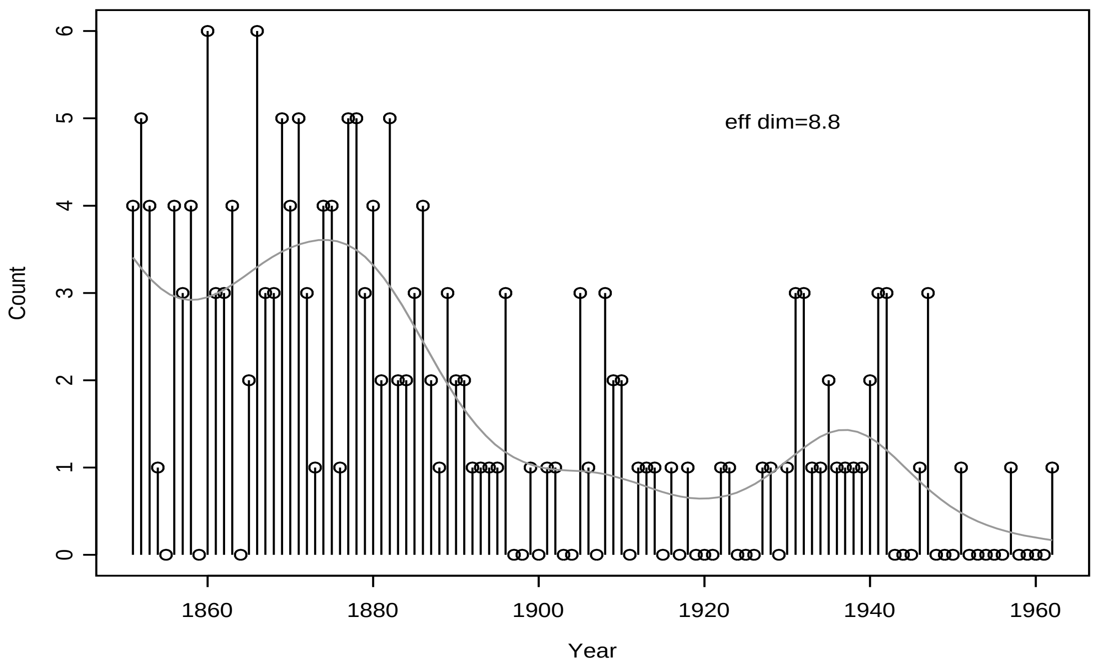
S-PLUS/ R code

```
pbinomial(Age, Kyphosis, 10, 3, 2, 0.0001, plot=T, se=T)

text(100, 0.9, 'lambda=0.0001')

text(100, 0.8, 'eff dim=9.8')
```


Density smoothing: an important exploratory tool

- Previously, Poisson time series: now step into densities
- Idea: P-spline smooth density overlayed on histogram
- Density estimation as Poisson regression
- $\log(\mu) = B\alpha$
- Regressor: midpoints of (narrowly) binned histograms
- Response: (Poisson) counts in bins
- Coal mining accident data naturally occurs in bins
- Often need to process data with `hist(x, breaks)`

Optimal density for coal mining accidents



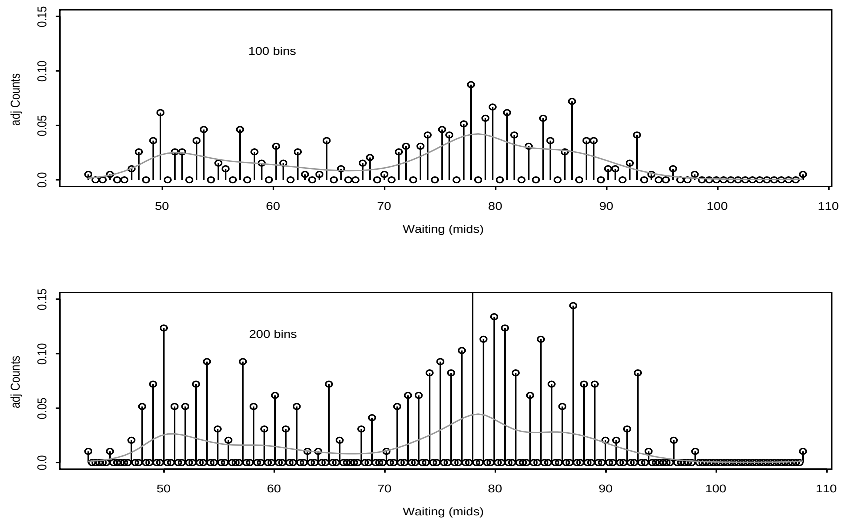
S-PLUS/ R code

- Cubic B-splines, `nseg=20`, `pord=2`, opt $\lambda = 3.2$

```
den.coal <- ppoisson(Year, Count, 20, 3, 2, lambda=3.2, plot=F)

plot(Year, Count)

lines(Year, Count, type='h')

lines(den.coal$xgrid, den.coal$ygrid, lwd=2)
```

Old faithful geyser data

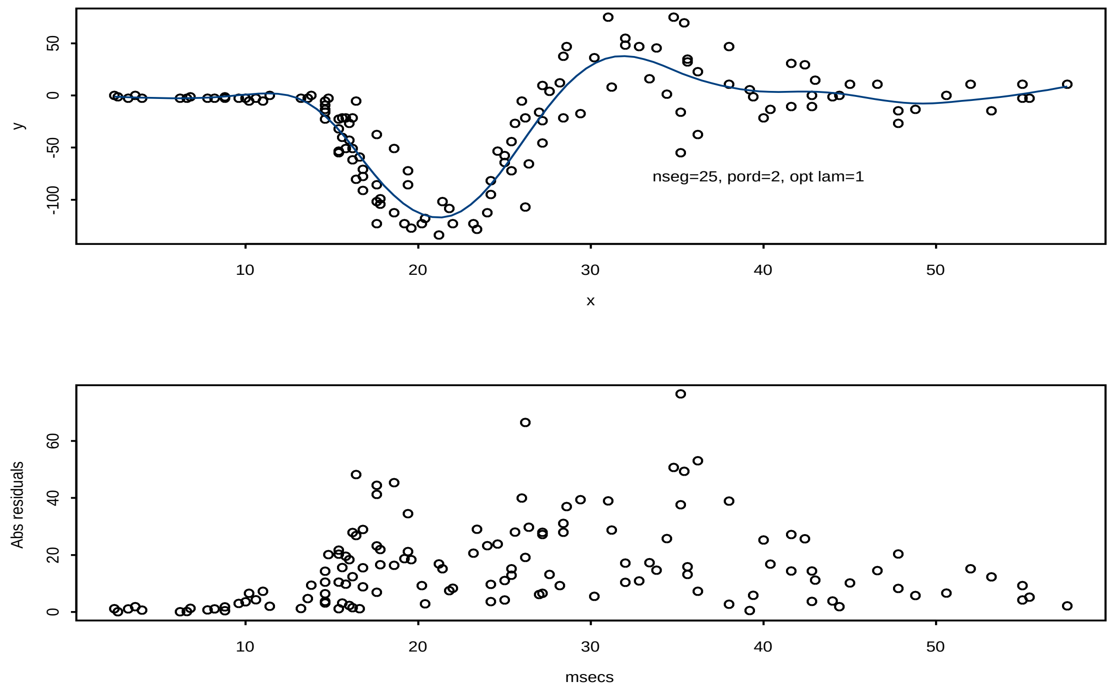
- Waiting time in minutes of eruptions
- Continous data between August 1-15, 1985 ($m = 299$)



Trend and spread

- We concentrated on trend modelling
- With normal and non-normal (counts, binomial) data
- What about the spread around the trend?
- Data (x_i, y_i), trend model $(x_i, \hat{y}_i)$
- Residuals $r_i = y_i - \hat{y}_i$
- How does variance of r change with x?

Motorcycle residuals: non-constant variance



Variance smoothing I

- Residuals r_i from smoothing (or regression model)
- Their variance shows a smooth trend
- Assume normal distribution $r_i \sim N(0,\ \sigma_i^2)$
- Combine P-splines and GLM idea for trend of σ^2
- Log link: $\ln(\sigma^2) = B\alpha = \eta$
- Likelihood:

$$L(\sigma; r) = \prod_{i=1}^{m} \frac{1}{\sigma_i \sqrt{2\pi}} \exp(\frac{-r_i^2}{2\sigma_i^2})$$

Variance smoothing II

- Penalized log-likelihood ($\sigma_i^2 = e^{\eta_i}$)

$$l(\sigma^2; r_i) = -\frac{1}{2}\sum_{i=1}^{m}(\eta_i + r_i^2 e^{-\eta_i}) - \frac{1}{2}\lambda|D\alpha|^2$$

- Likelihood equations, with $V = \mathrm{diag}(r^2)$:

$$B'(1 - Ve^{-\eta}) = \lambda D'D\alpha$$

- Linearization leads to iterative P-spline smoothing:

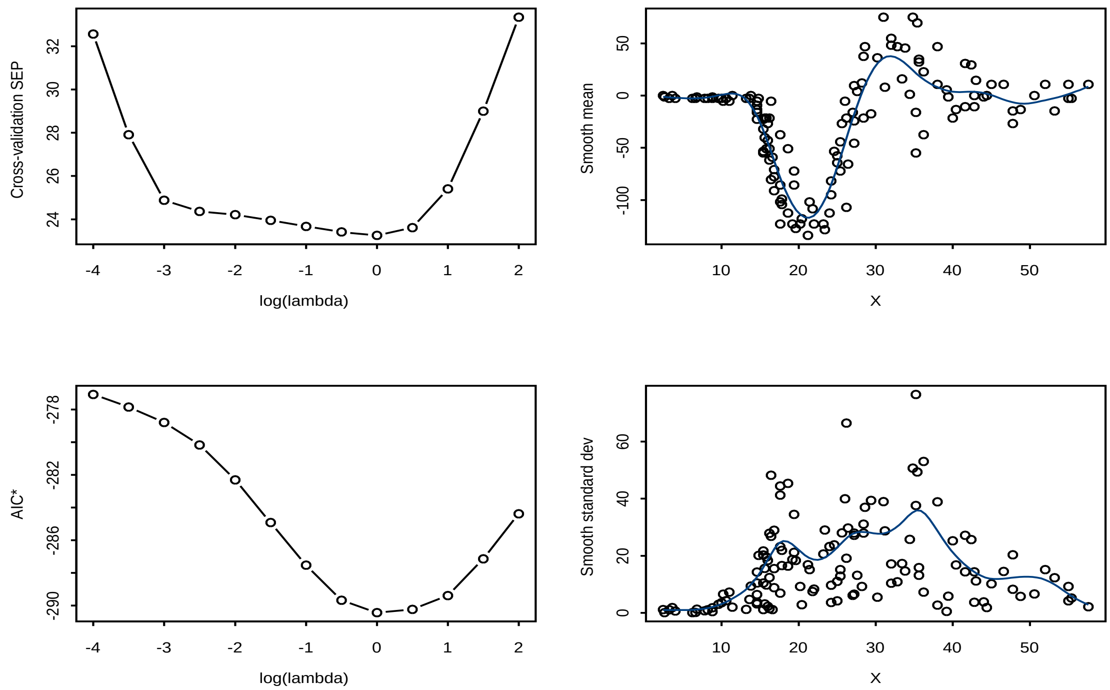
$$\tilde{\alpha} = (B'B + \lambda D'D)^{-1}B'\tilde{z}$$

$$z = 1 - Ve^{-\eta} + \eta$$

Extensions and details

- Combined approach possible
- Smooth variance gives optimal weights for trend estimation
- New trend, new residuals, new smooth variance, ...
- Needs further research: stability and speed still unclear
- Note
  - $\mathrm{dev}(r, \lambda) = \sum_{i=1}^{m}(\hat{\eta}_i + r_i^2 e^{-\hat{\eta}_i})$
  - eff dim = $\mathrm{trace}(B'B(B'B + \lambda D'D)^{-1})$

Motorcycle mean and variance



Wrap-up

- Poisson and binomial smoothing
- Penalized maximum likelihood estimation
- P-splines useful for trend estimation in scatterplots
- But also for smoothing the variance
- *Next,* P-spline recipe and optimal smoothing
- Density estimation: P-spline Poisson smoothing
- Processing count data with narrow histogram bins

Session 4

Optimal Smoothing in Action

Session 4

Optimal Smoothing in Action

The Craft of Smoothing 4

What you will get

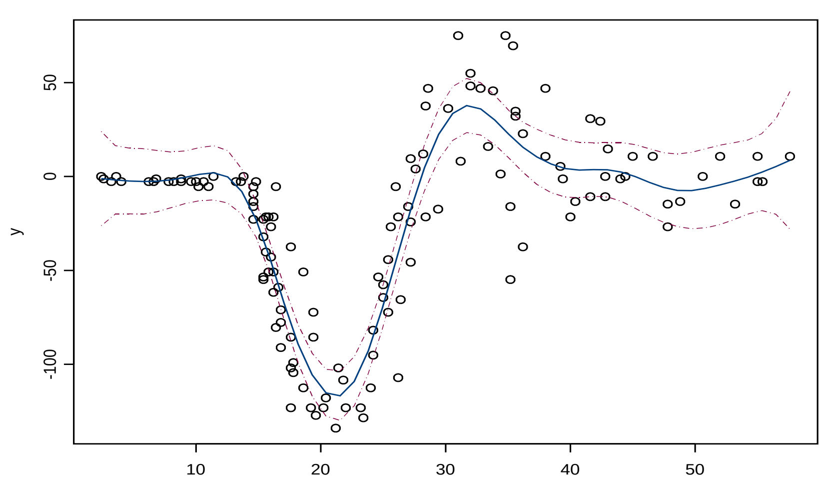
- Practical recipe for P-splines
- Cross-validation measures
- Standard error of prediction
- Optimal smoothing
- Effective dimension of P-spline model
- Limiting polynomials
- Standard error bands
- Density estimation using P-splines

Practical recipe for P-splines

- Choose "too many" equally-spaced (cubic) B-splines
- Default penalty order $d = 2$ or $d = 3$
- Measure performance with cross-validation (CV) or an information criterion (AIC, BIC)
- Vary λ's on a logarithmic grid search
- Find minimum of performance criterion
- Report $\hat{\alpha}$, the P-spline coefficients, which is compact form of the smooth

Pnormal function in S-PLUS/R

```
par(mfrow=c(1,1))
x<-Motimp$V1
y<-Motimp$V2 pnormal(x, y, nseg=25,
bdeg=3, pord=2, lam=1, plot=T, se=T)
```



General: cross-validation (CV)

- Data splitting (1 fit), for example:
  - $\frac{2}{3}$ model training
  - $\frac{1}{3}$ model testing/validation
- 10-fold CV (10 fits):
  - leave out 10%, fit on remaining 90%
  - test/validate on 10%
  - cycle for all 10 partitions
- Leave-one-out (m fits*):
  - same idea as 10-fold
  - just leaving out a single observation
  - test/validate one observation at at time

CV summary measure

- Standard error of prediction (CVSEP)
- Leave-one-out CVSEP: $\hat{y}_{-i}$ is predicted value at ith point using the trained model without the ith point

$$\text{CVSEP} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_{-i})^2}$$

Cross-validation

- Straightforward recipe is expensive
  - Fix λ, remove ith observation
  - Fit model $(-i)$
  - Predict at i location using model $(-i)$
  - Cycle through all i
  - Summarize with CVSEP
  - Repeat for λ along a grid search
  - Seek λ corresponding to minimum CVSEP

Penalized hat matrix

- Instead use hat matrix H (not a projection matrix)

$$\begin{aligned}\hat{y} &= B\hat{\alpha}_\lambda \\ &= \underbrace{B(B'B + \lambda D_d' D_d)^{-1} B'}_{H(\lambda)} \, y \\ &= H(\lambda)\, y\end{aligned}$$

- $H(\lambda)$ turns y into $\hat{y}$

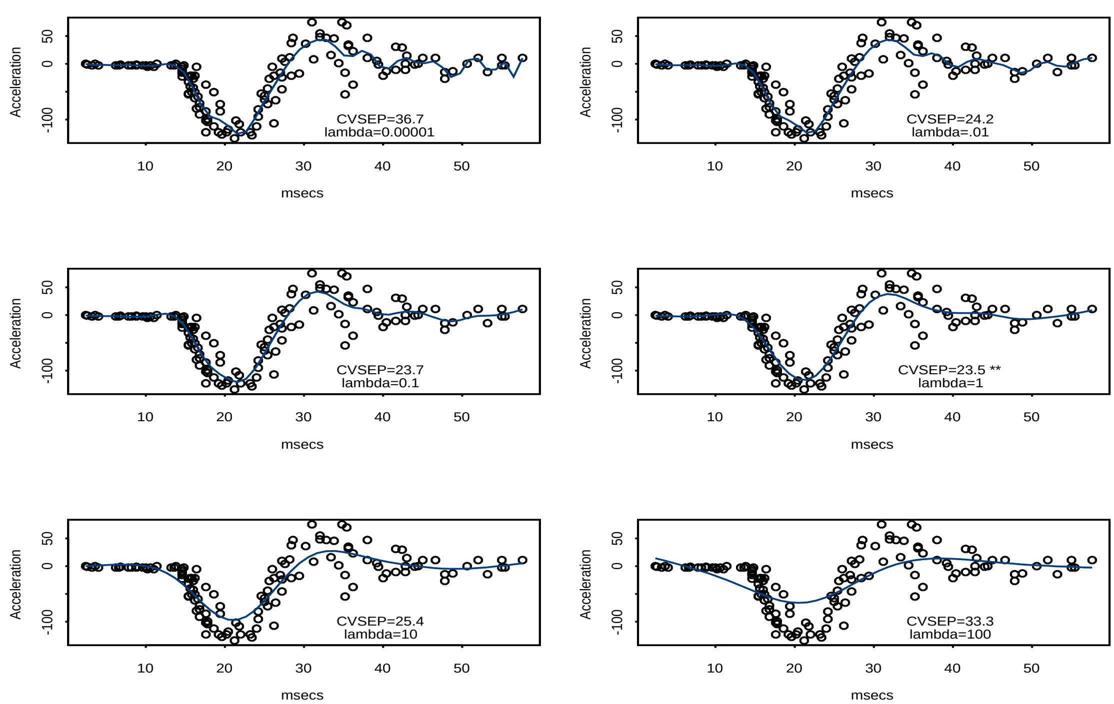
Computational relief for cross-validation

- Use $H(\lambda)$ to get at CVSEP with one fit
- It is well known that:

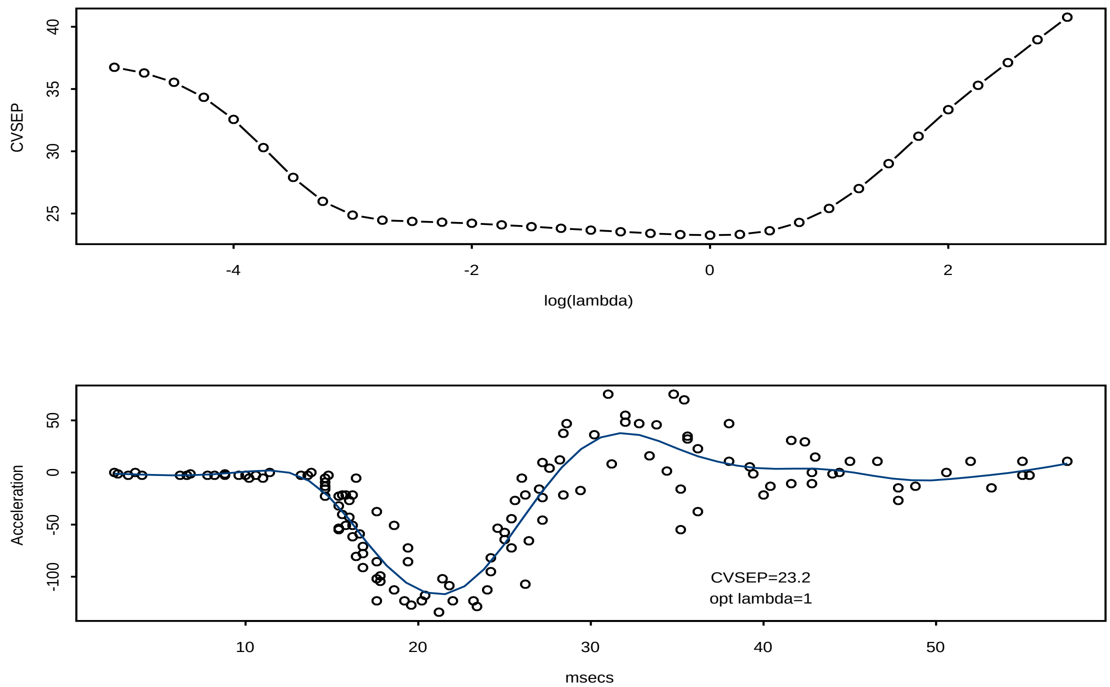
$$y_i - \hat{y}_{-i} = \frac{y_i - \hat{y}_i}{1 - h_{ii}}$$

- $h_{ii} = b_i'(B'B + \lambda D_d' D_d)^{-1} b_i$, where b_i' is ith row of B
- Only $\hat{y}$ and diag(H) needed; both can be computed quickly
- Cross-validation with little extra work
- Now minimize CVSEP as λ (not the knot number) varies

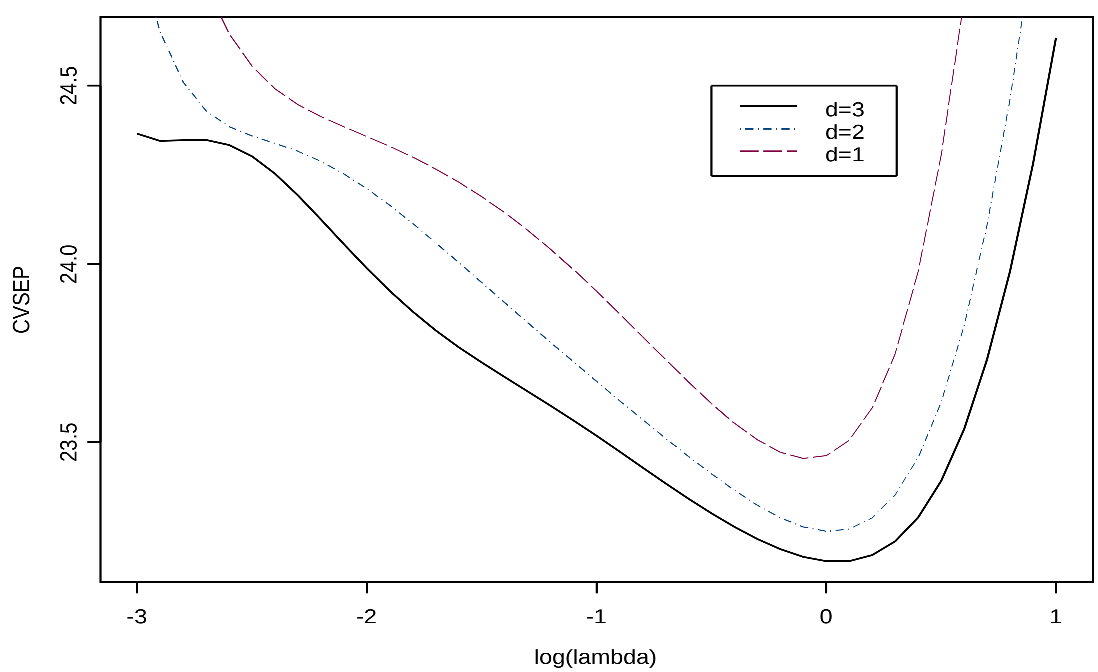
Motorcycle helmet: (nseg=25, pord=2, vary λ)



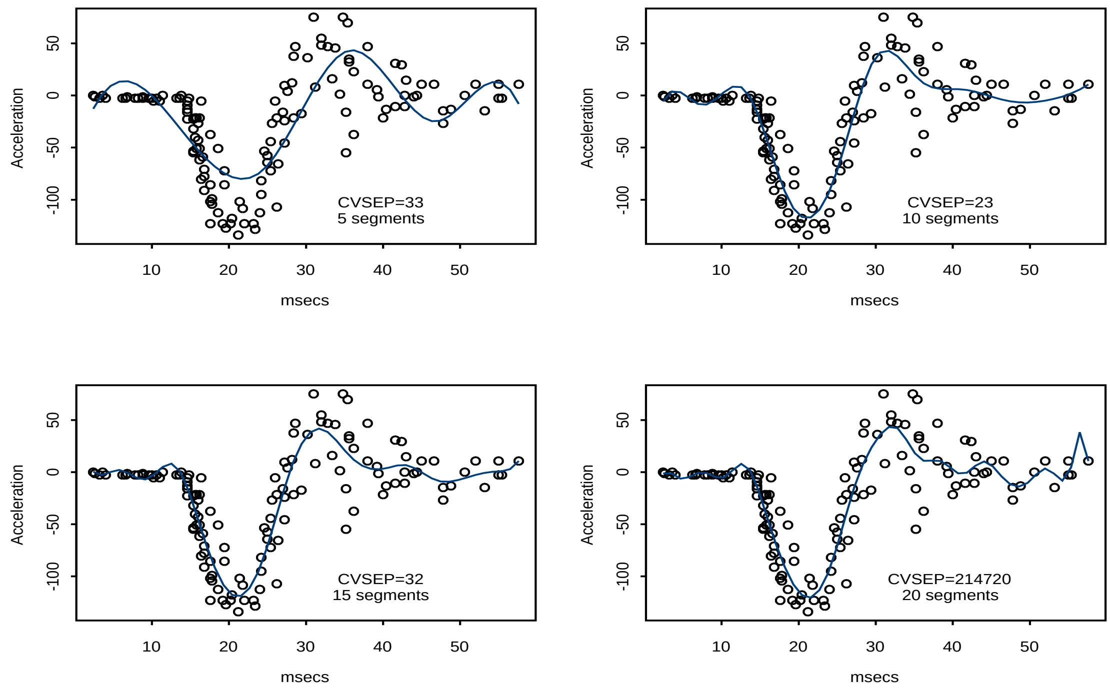
Optimal seeking function, based on CVSEP



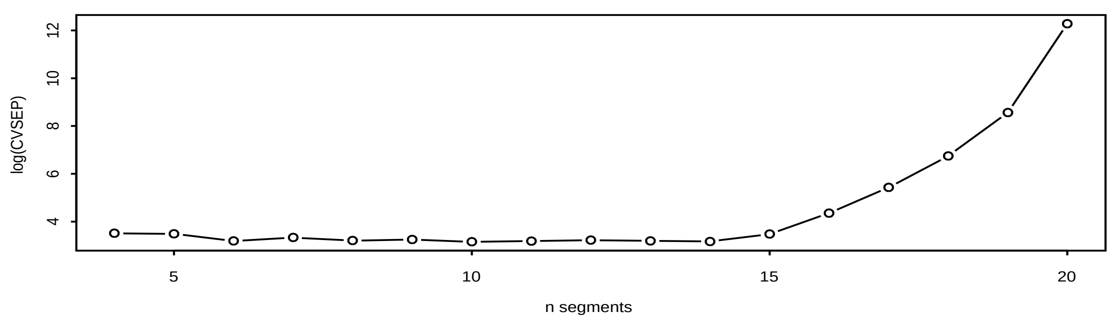
CVSEP vs. $\log(\lambda)$, by penalty order

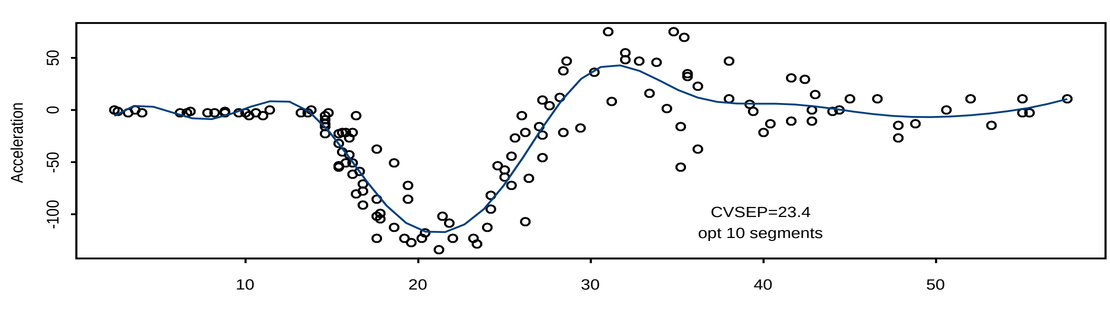


Aside: B-spline CV for motorcycle data



Aside: optimal B-spline fit using CVSEP





(Effective) dimension: linear regression

- Recall standard linear regression: $\hat{y} = X\hat{\alpha} = Hy$
- Property trace$(H) = p$, where $p =$ ncol(X)

$$H = X(X'X)^{-1}X'$$

- Thus the trace of hat matrix provides dimension
- For general smoothers, we have $\hat{y} = Sy$
- Trace(S) approximates dimension of fit
- Result can be used with P-splines

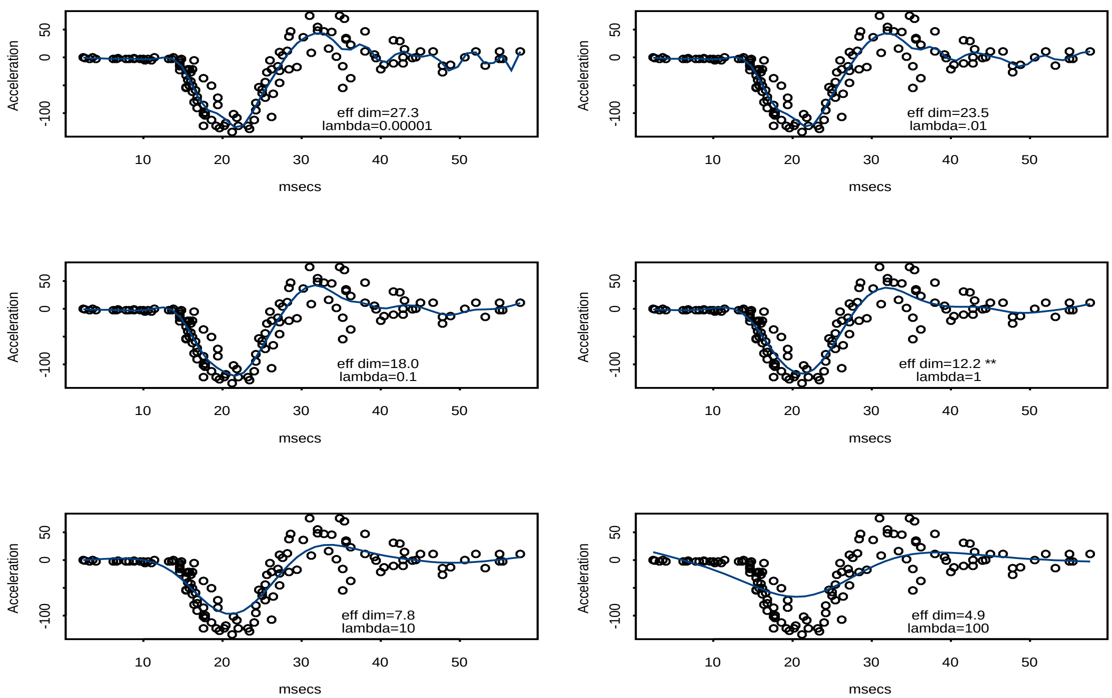
Effective dimension (ED): P-splines

- P-spline hat matrix: $H(\lambda) = B(B'B + \lambda D_d' D_d)^{-1}B'$
- Effective dimension: ED(λ)= trace$\{H(\lambda)\}$
- More efficient to compute (n vs. m):

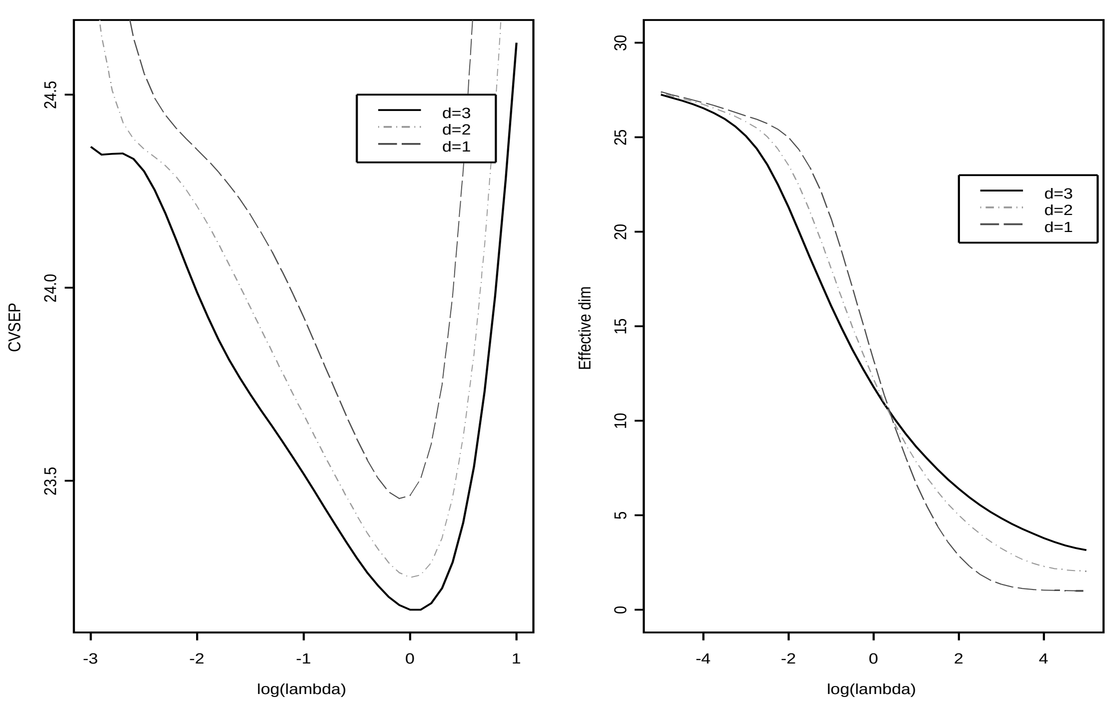
$$\text{trace}\{B'B(B'B + \lambda D_d' D_d)^{-1}\}$$

- Invariant to cyclical permutations, trace(ABC)=trace(CAB)
- One-to-one relationship between effective dimension and λ
- As λ gets large, then effective dimension goes to d

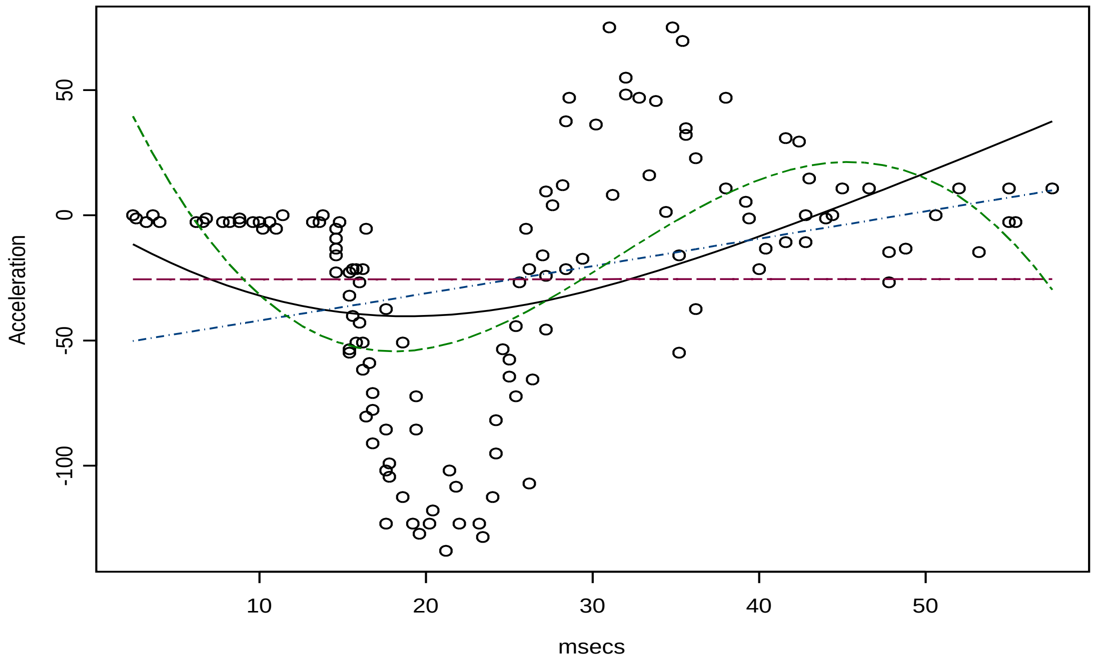
Effective dimension for motorcycle data



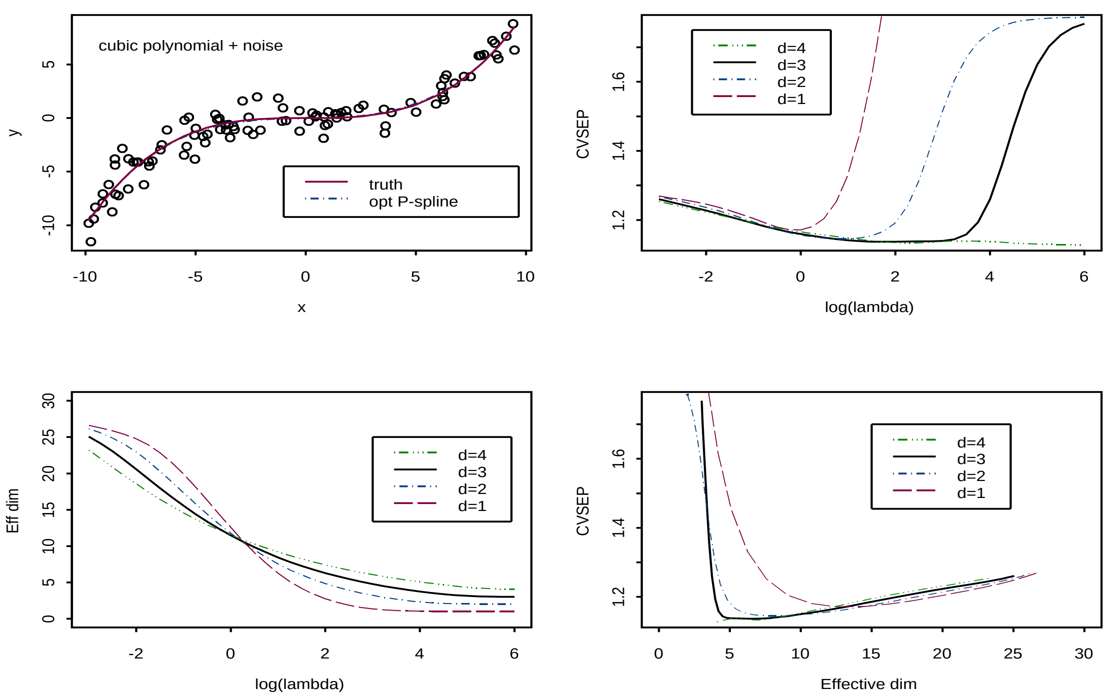
Motorcycle: ED vs. $\log(\lambda)$, by penalty order



Polynomial limit: $\lambda = 10^6$, pord=1,2,3,4



Confidence builder: cubic polynomial + noise



Insight: polynomial limit

- Suppose (y, x) truly linear
- Follows B-spline parameters: $\alpha_j = a + bj$

$$\begin{aligned} \Delta\alpha &= \alpha_{j+1} - \alpha_j \\ &= a + b(j+1) - (a + bj) = b \end{aligned}$$

- Thus $\Delta^2\alpha = b - b = 0$
- Recall: large λ weighs (dth order) penalty, not RSS
- Consequence: min Q as $\lambda \to \infty$,
  smooth goes toward polynomial $d - 1$

Standard errors

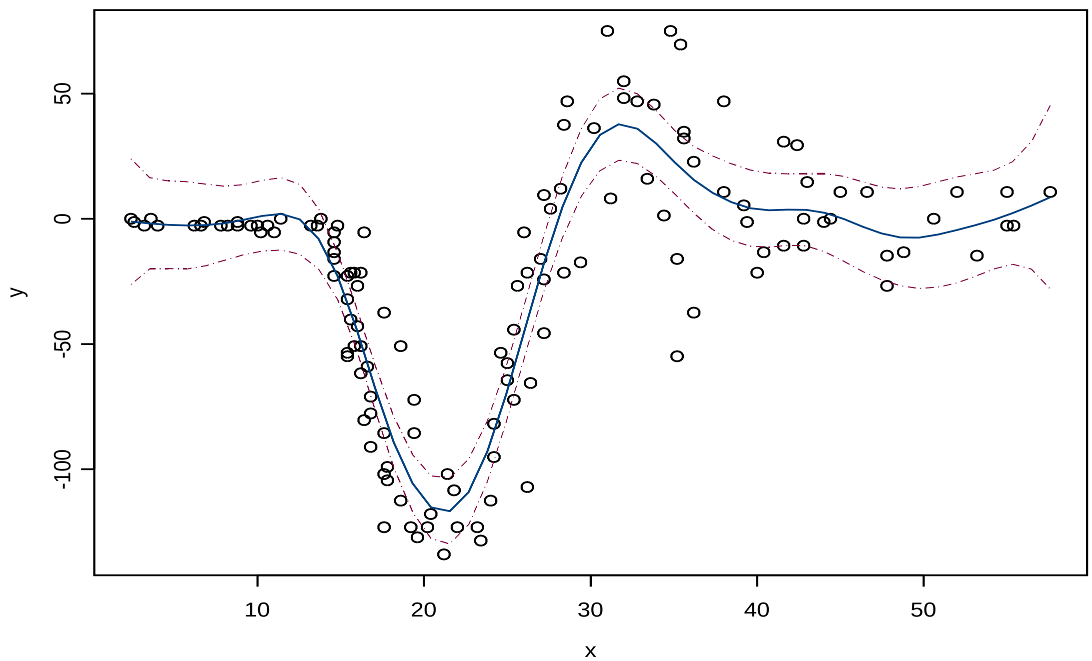
- Sandwich estimator

$$\begin{aligned} \mathrm{var}(\hat{y}) &= \mathrm{var}(Hy) \\ &= H\,\overbrace{\mathrm{var}(y)}^{\sigma^2 I}\,H' \\ &\approx \sigma^2\,\underbrace{B(B'B + \lambda D_d' D_d)^{-1}B'}_{H}\,B(B'B + \lambda D_d' D_d)^{-1}B' \end{aligned}$$

- Use sqrt of diagonal, $\hat{\alpha}$ approx. normal, $\hat{y} \pm 2\mathrm{se}(\hat{y})$
- Again, effective model dimension: $\mathrm{tr}(H)$
- Variance estimate

$$\hat{\sigma}^2 = \frac{|y - \hat{y}|^2}{m - \mathrm{tr}(H)}$$

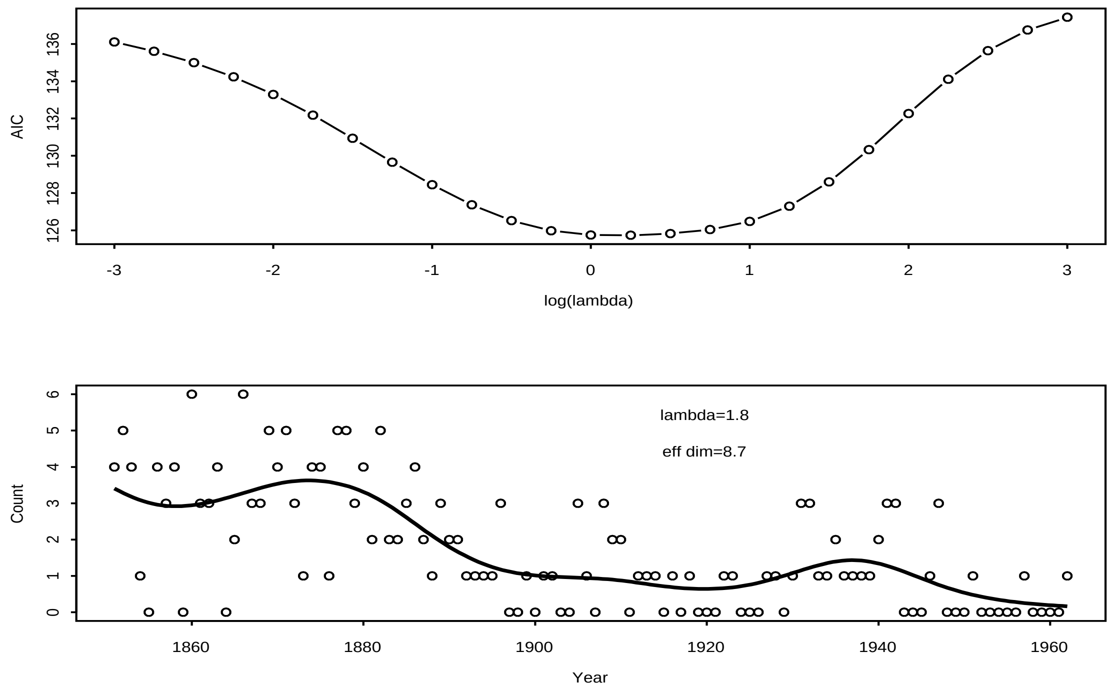
Optimal P-spline fit with twice se bands



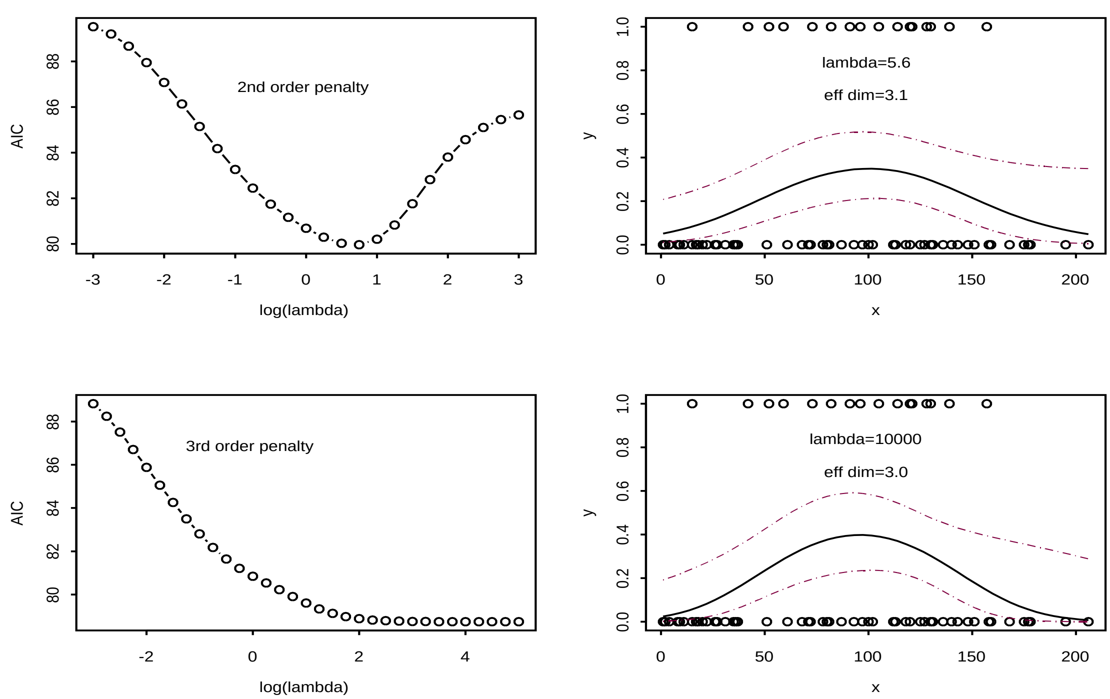
GLM optimal smoothing: choice of λ

- Leave-one-out CV is less direct and approximate
- Alternatively use cost-complexity information criterion
- AIC(λ) = deviance($y; \lambda$) + 2 effective dimension(λ)
- Compromise: fidelity + roughness
- Choose λ to minimize AIC
- Other information criteria exist

Coal mining optimal smooth, 20 segments



Kyphosis optimal smooth, twice s.e. bands



Details: deviance

- Log-likelihood difference (×2): current vs. "perfect" model
- Deviance = 0, when all $\hat{\mu} = y$
- Deviance:
  - Normal: $\frac{1}{\sigma^2} \sum_{i=1}^{m} (y_i - \hat{\mu}_i)^2$
  - Poisson: $2 \sum_{i=1}^{m} y_i \ln\left(\frac{y_i}{\mu_i}\right)$
  - Binomial: $2 \sum_{i=1}^{m} \left(y_i \ln\left[\frac{y_i}{\hat{\mu}_i}\right] + (1 - y_i) \ln\left[\frac{1-y_i}{1-\hat{\mu}_i}\right]\right)$

A closer look at Poisson deviance

- Likelihood: $L = \prod_{i=1}^{m} \frac{\exp(-\mu_i)\mu_i^{y_i}}{(y_i!)}$
- Log L

$$l_\mu = \sum_{i=1}^{m} (\mu_i - y_i \ln(\mu_i) - y_i!)$$

- "Perfect" $\mu = y$:

$$l_y = \sum_{i=1}^{m} (y_i - y_i \ln(y_i) - y_i!)$$

- Deviance: $2(l_\mu - l_y)$
- Function of α: $\ln(\mu) = B\alpha$ and $\mu = \exp(B\alpha)$

Details: effective dimension (ED)

- Approximate model dimension: trace$\{\hat{H}(\lambda)\}$

$$\hat{\eta} = B\hat{\alpha} = \underbrace{B(B'\hat{W}B + \lambda D_d'D_d)^{-1}B'\hat{W}}_{\hat{H}(\lambda)}\hat{z}$$

- Smooths current working vector $\hat{z}$ into a linear predictor $\hat{\eta}$
- Converged $\hat{W}$, $\hat{z}$
- Cyclical permutations okay for trace computations:

$$\mathrm{ED}(\lambda) = \mathrm{trace}\{B'\hat{W}B(B'\hat{W}B + \lambda D_d'D_d)^{-1}\}$$

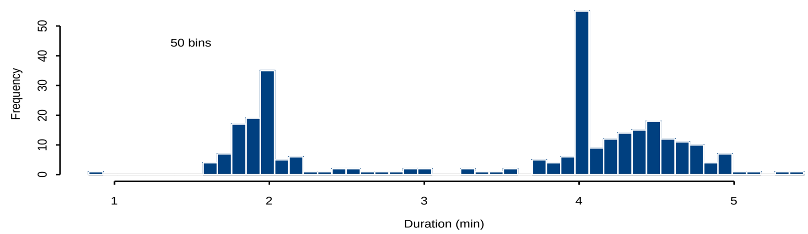
- Advantage: $n \times n$ rather than $m \times m$

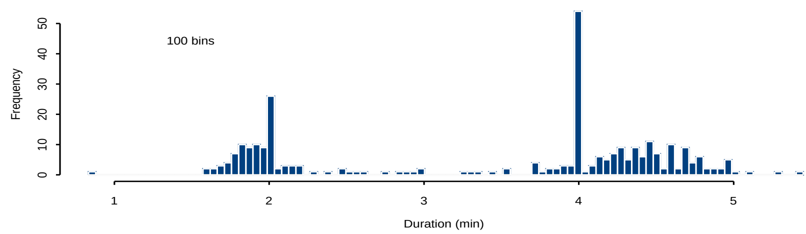
Density smoothing: an important exploratory tool

- Previously, Poisson time series: now step into densities
- Idea: P-spline smooth density overlayed on histogram
- Density estimation as Poisson regression
- $\log(\mu) = B\alpha$
- Regressor: midpoints of (narrowly) binned histograms
- Response: (Poisson) counts in bins
- Often need to process data with `hist(x, breaks)`

Old faithful geyser data

- Duration in minutes of eruptions (not waiting time)
- Continuous data between August 1-15, 1985 ($m = 299$)

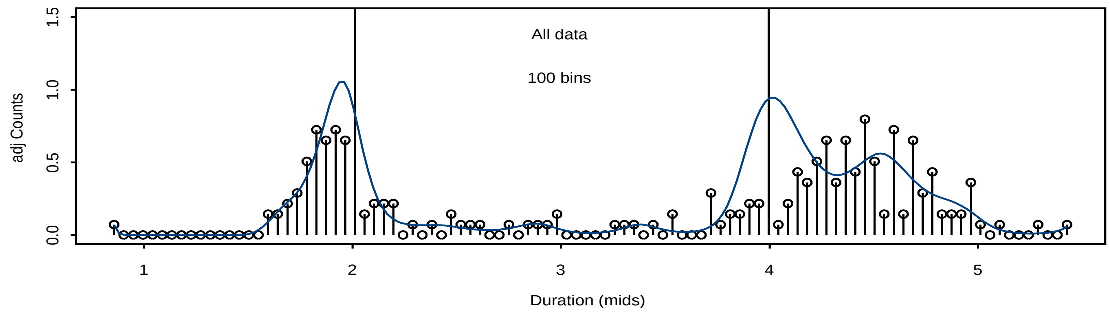


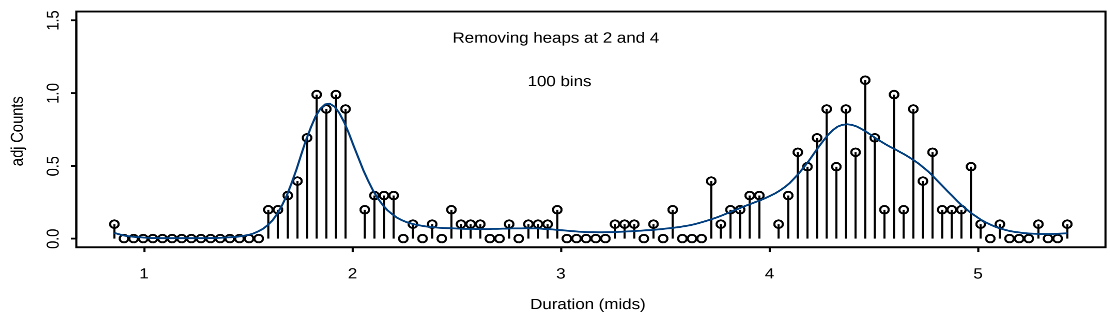


Digit preference at 2 and 4 minutes

- Heaps of data at 2 and 4 minutes
- Explanation: these recorded at dark
- Digit preference, rounded estimates
- Opportunity to interpolate with P-splines
- Leave out these bins (not set them to zero)

Geyser: density interpolation

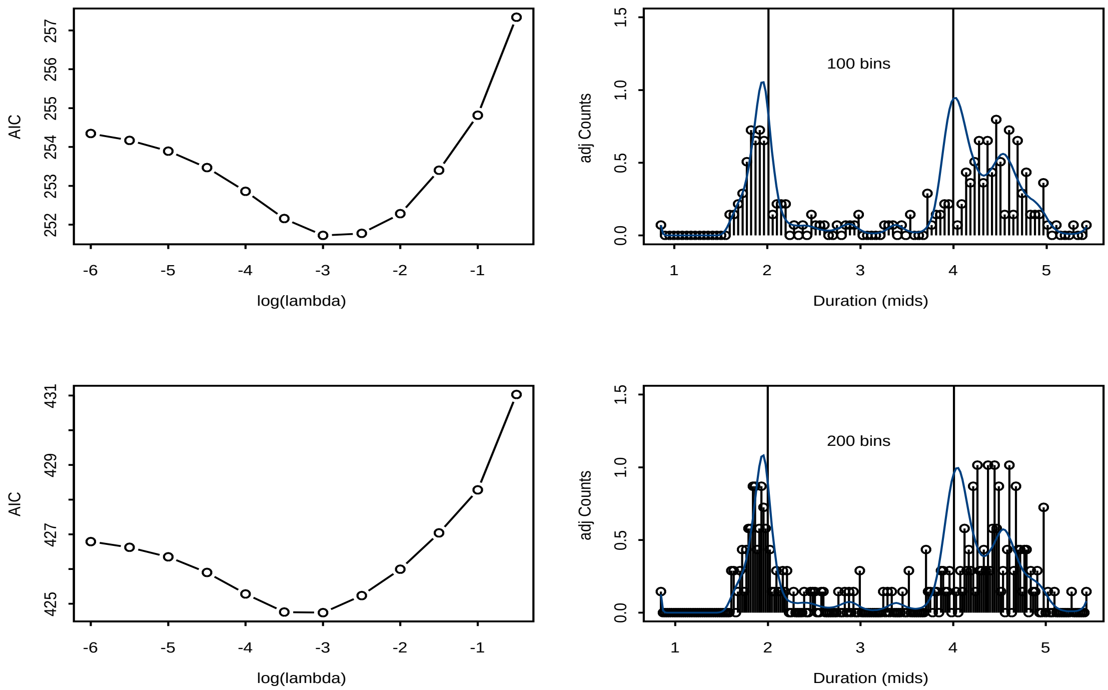




S-PLUS/ R code

```
Breaks <- seq(from=min(Duration), to=max(Duration), length=101)

ghist <- hist(Duration, breaks=Breaks, plot=F)

Counts <- ghist$counts

nBreaks <- length(Breaks)

mids <- Breaks[-nBreaks] + diff(Breaks)/2

pden <- ppoisson(x=mids, y=Counts, 20, 3, 2, 0.001, plot=F)

width <- mean(diff(Breaks)); adj <- width*sum(Counts)

plot(mids, Counts/adj, type='h')

lines(pden$xgrid, pden$ygrid/adj )
```

Optimal density for geyser data: 100 and 200 bins



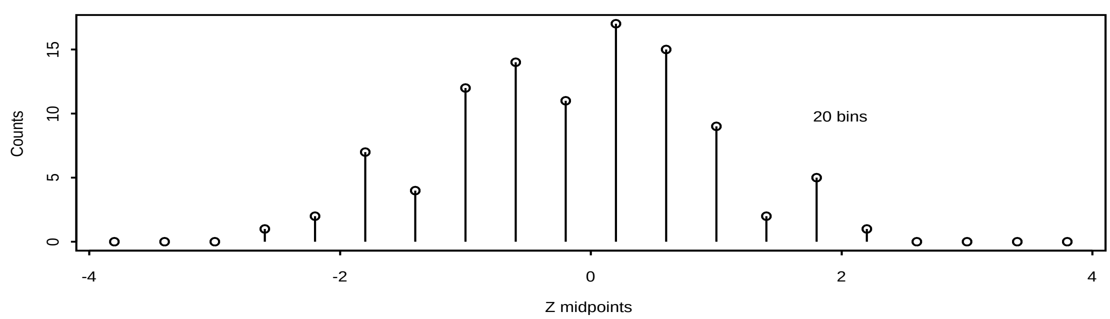
Beauty of P-spline densities

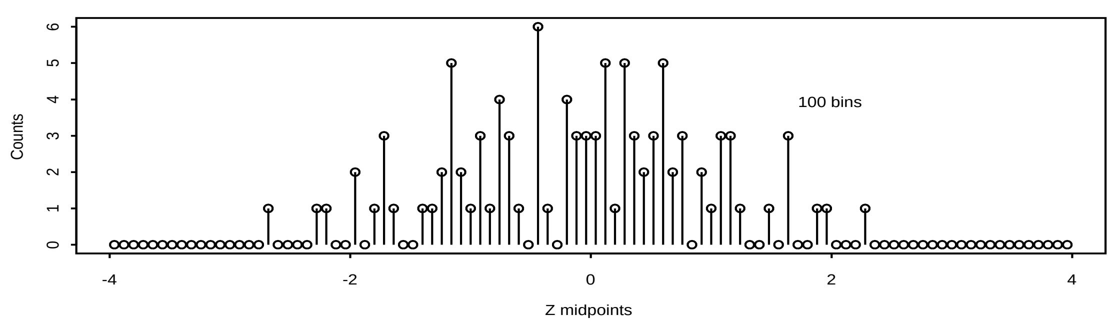
- Density constrained positive by inverse link: $\mu = \exp(B\alpha)$
- As λ gets large, then limit:
  - Normal (`pord=3`), exponential (`pord=2`)
- No boundary effects
- Specialized limits encouraged
- Mean and variance conserved from histogram to density
- Compact result: density compressed in $\hat{\alpha}$

Confidence builders

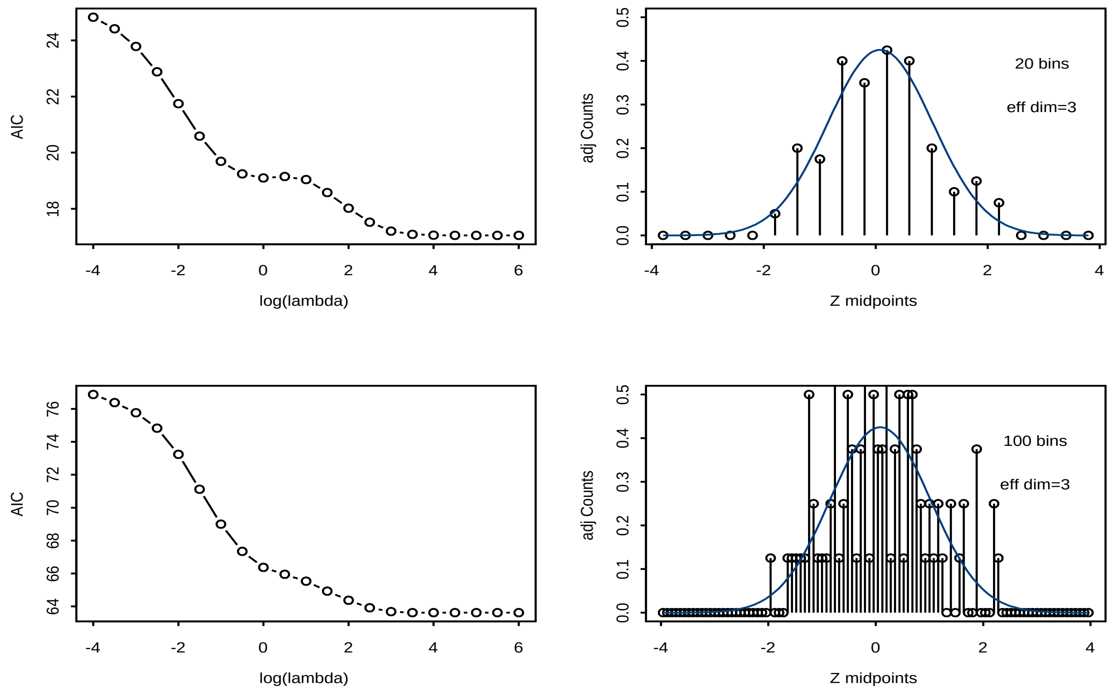
- Randomly generate $m = 100$ Normal(0, 1) variates
- Process data: histograms with 20 or 100 bins
- Also vary the limits of the histogram: e.g. $\pm 4,\ 5,\ 6$
- Use P-splines: `nseg=20` and `pord=3`
- Optimize density using AIC: expect large λ
- Now repeat exercise, random exponentials, `pord=2`

Histograms of Normals

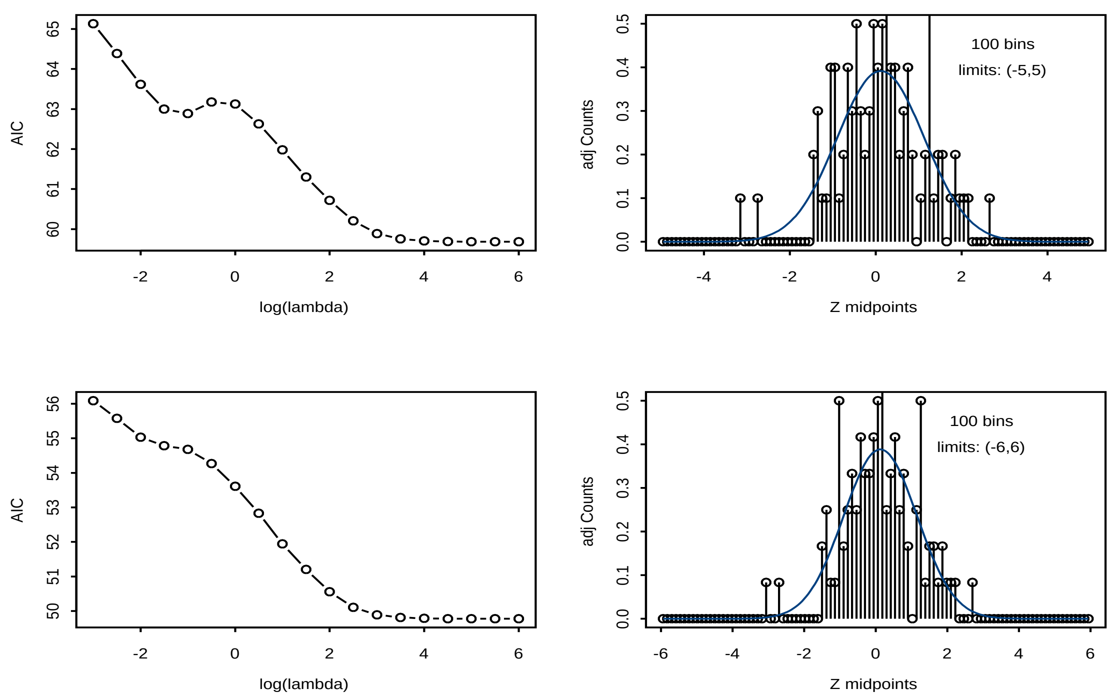




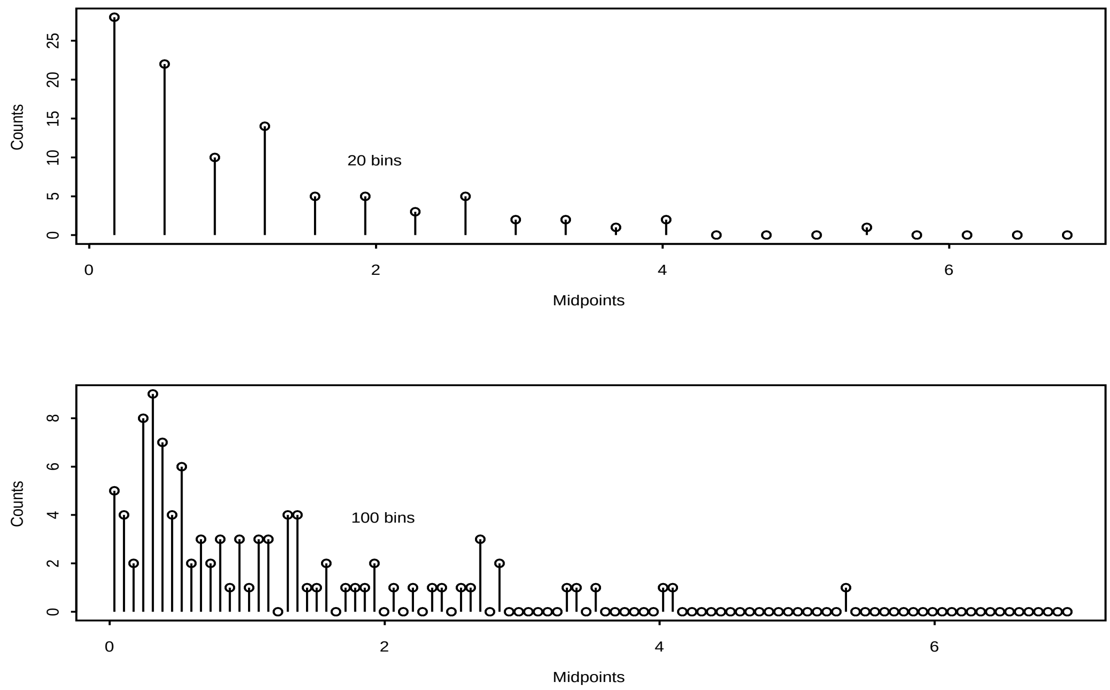
Optimal densities



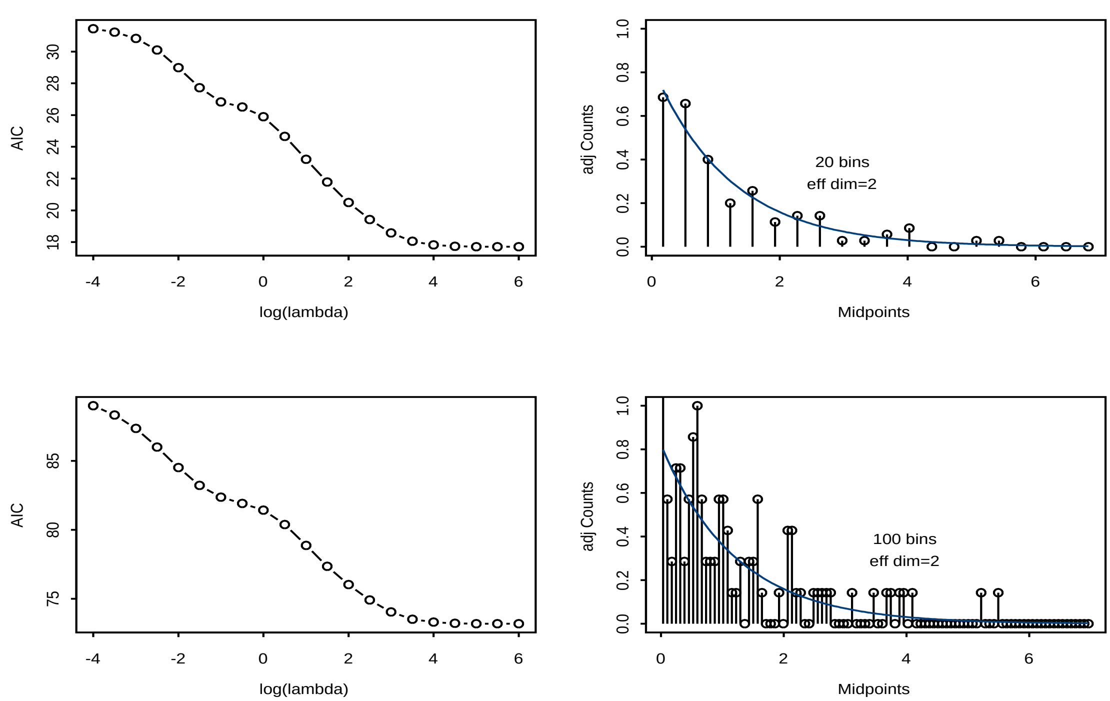
Optimal densities for different limits



Histograms of exponential data



Optimal densities



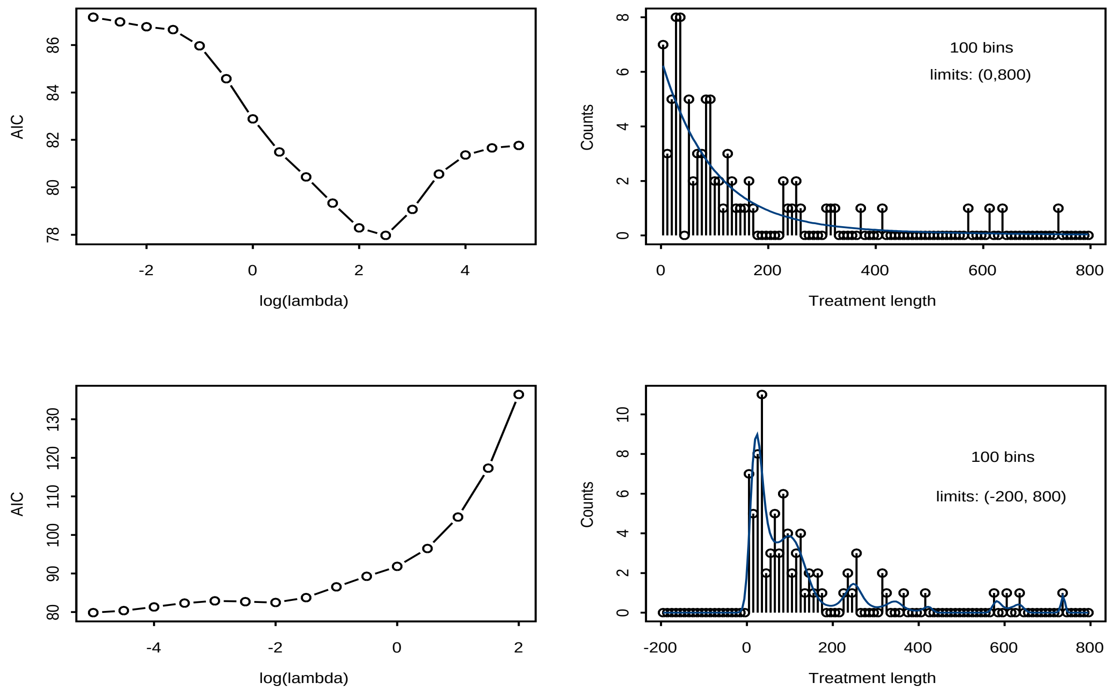
P-spline conservation of moments

- Histogram with m bins and c_i counts
- Fitted $\hat{\mu}_i$ at each midpoint x_i
- Penalty of order d
- For $d = 1$: $\sum_{i=1}^{m} c_i = \sum_{i=1}^{m} \hat{\mu}_i$ (proper density)
- For $d = 2$ also holds $\sum_{i=1}^{m} x_i c_i = \sum_{i=1}^{m} x_i \hat{\mu}_i$ (same mean)
- For $d = 3$ also holds $\sum_{i=1}^{m} x_i^2 c_i = \sum_{i=1}^{m} x_i^2 \hat{\mu}_i$ (same variance)
- True for any $\lambda \geq 0$

The influence of boundaries

- Example: observations that cannot be negative
- Left boundary of density at zero
- Choose domain of B-splines with left boundary at zero too
- If not: histogram bins below zero with no counts
- Smoother does his best to fit
- Result: AIC indicates light smoothing
- (Boundary problem familiar in kernel smoothing)

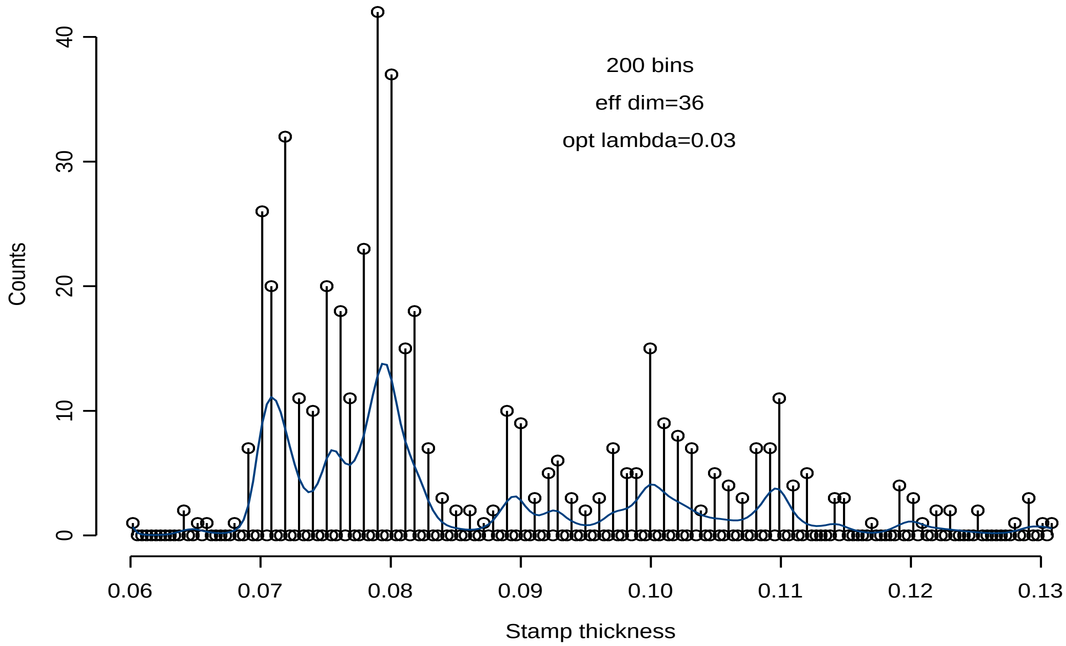
Specialized limits for suicide treatment study



Hidalgo stamp thickness: many modes

- Measure approximately 500 stamps with micrometer
- Tendency (human) to pick rounded measurements
- AIC indicates light smoothing as optimal
- Digit preference? Modes concentrate at multiples of 0.01

Optimal Hidalgo P-spline density



Wrap-up

- Practical recipe for P-splines, using CVSEP
- Effective dimensions,limiting polynomials, se bands
- P-splines very effective for density smoothing
- Limits, boundaries are respected, mean/ var conserved
- AIC indicates right amount of smoothing
- *Next*, extensions to generalized additive models
- P-spline varying coefficient models
- Some P-spline extensions into 2-D smoothing

Session 5

Multi-dimensional Modelling with P-splines

Session 5

Multidimensional Modelling with P-splines

The Craft of Smoothing 5

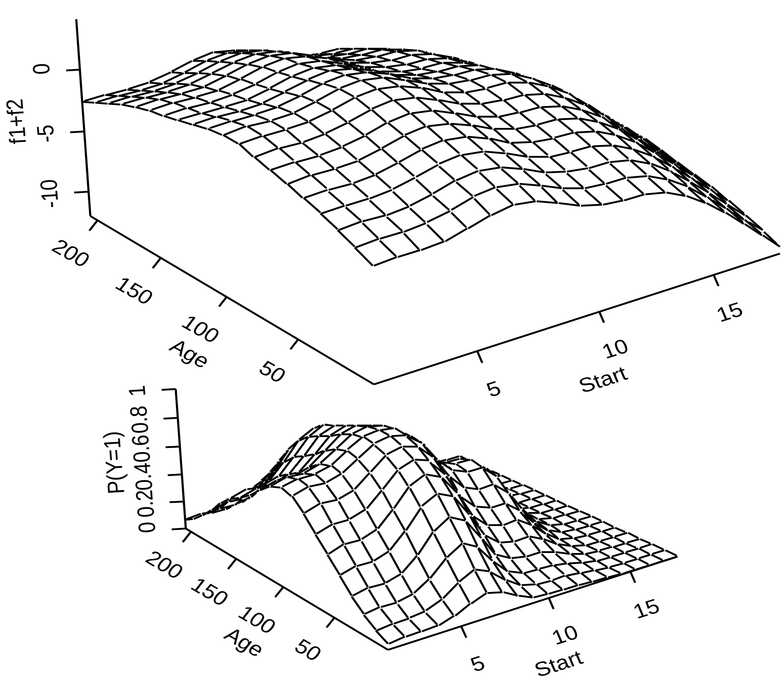
What you will get

- Extensions into generalized additive models
- Extensions into varying coefficient models
- 2-D smoothing using tensor product B-splines
- Double penalization: rows and columns of B-splines

Generalized additive models

- One-dimensional smooth model: $\eta = f(x)$
- Two-dimensional smooth model: $\eta = f(x_1, x_2)$
- General f: any interaction between x_1 and x_2 allowed
- Complex: two-dimensional smoothing
- Compromise: (generalized) additive model: $\eta = f_1(x_1) + f_2(x_2)$
- Both f_1 and f_2 smooth (Hastie and Tibshirani, 1990)
- Higher dimensions straightforward

Kyphosis: Additive predictor, P(Y=1)



Backfitting for GAM estimation

- Assume linear model: $E(y) = \mu = f_1(x_1) + f_2(x_2)$
- Assume: approximations $\tilde{f}_1$ and $\tilde{f}_2$ available
- Compute partial residuals $r_1 = y - \tilde{f}_2(x_2)$
- Smooth scatterplot of (x_1, r_1) to get better $\tilde{f}_1$
- Compute partial residuals $r_2 = y - \tilde{f}_1(x_1)$
- Smooth scatterplot of (x_2, r_2) to get better $\tilde{f}_2$
- Repeat to convergence

More on backfitting

- Start with $\tilde{f}_1 = 0$ and $\tilde{f}_2 = 0$
- Generalized residuals and weights for non-normal data:
- Any smoother can be used
- Convergence can be proved, but may take many iterations
- Convergence criteria should be strict

A recent problem with backfitting

- Backfitting (in S-PLUS) is de-facto GAM standard
- Heavily used in air pollution epidemiology
- Smooth term for time trend
- Parametric or smooth terms for temperature, fined dust, ...
- Used by several groups/studies in Europe, US, Canada
- Spring 2002, Johns Hopkins discovers incomplete convergence
- Upheaval in the press, all computations redone

PGAM: GAM with P-splines

- Use B-splines: $\eta = f_1(x_1) + f_2(x_2) = B_1\alpha_1 + B_2\alpha_2$
- Combine B_1 and B_2 to matrix, α_1 and α_2 to vector:

$$\eta = [B_1 : B_2] \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = B\alpha$$

- Difference penalties on α_1, α_2, in block-diagonal matrix
- Penalized GLM as before: no backfitting

P-GAM fitting

- Maximize:

$$l^* = l(\alpha; B, y) - \frac{1}{2}\lambda_1 |D_{d1}\alpha_1|^2 - \frac{1}{2}\lambda_2 |D_{d2}\alpha_2|^2$$

- Iterative solution:

$$\hat{\alpha}_{t+1} = (B'\hat{W}_t B + P)^{-1} B'\hat{W}_t \hat{z}_t^\star$$

where

$$P = \begin{bmatrix} \lambda_1 D'_{d1} D_{d1} & 0 \\ 0 & \lambda_2 D'_{d2} D_{d2} \end{bmatrix}$$

PGAM advantages

- No backfitting call needed, directly fit
- Fast computation
- Equations moderate size, Compact result (α^*)
- Explicit computation of hat matrix: fast CV, eff dim, AIC
- Easy standard errors
- No iterations, no convergence criteria to set

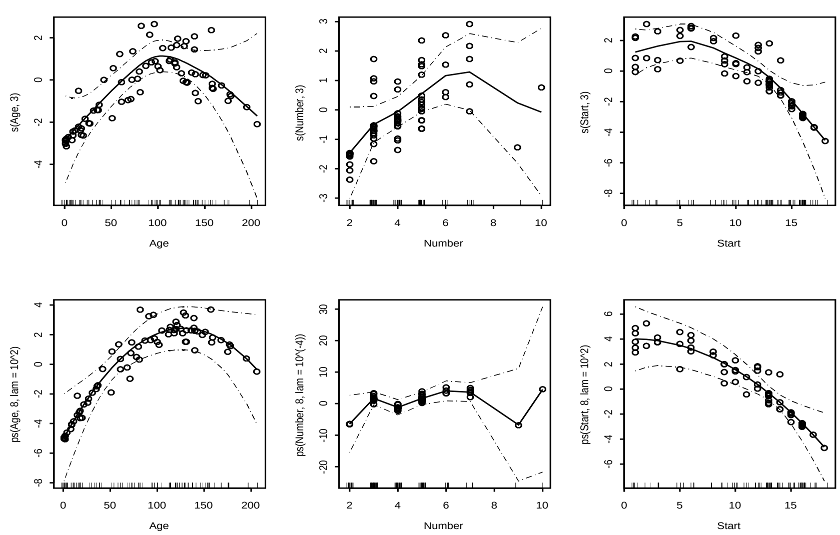
Features of P-spline GAMs

- *Eff dim* = trace($\hat{H}$) =trace($B(B'\hat{W}B + P)^{-1}B'\hat{W}$)
- AIC = deviance($y;\hat{\alpha}$) + 2 trace($\hat{H}$)
- Standard error of jth smooth

$$B_j(B'\hat{W}B + P)^{-1}B'\hat{W}B(B'\hat{W}B + P)^{-1}B_j'$$

- GLM diagnostics accessible
- Easy combination with additional linear regressors/factors
- Example: $[B_1 : B_2 : X]$ (no penalty on X coeffs)

Three regressor kyphosis GAM



- *Top*: smoothing splines, *df*=3 each
- *Bottom*: cubic P-splines, $d = 3$, *eff dim*=(3, 7.2, 3)

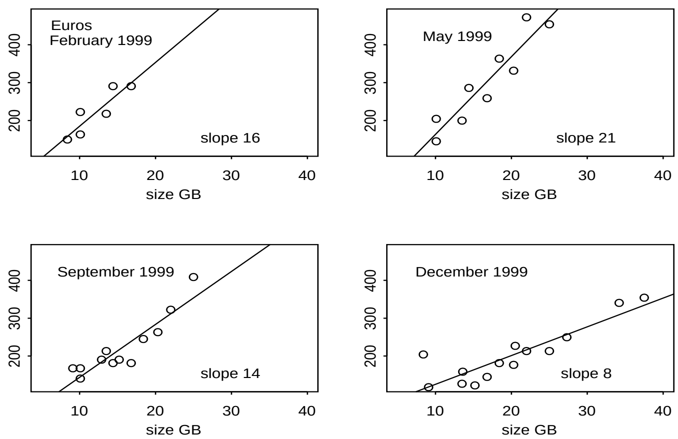
Kyphosis grid search: top 5 performers

$d = 2$				$d = 3$			
AIC	γ_1	γ_2	γ_3	AIC	γ_1	γ_2	γ_3
59.249	-1	-4	1	58.173	4	-4	4
59.273	-1	-4	0	58.173	3	-4	4
59.324	-1	-4	2	58.173	4	-4	3
59.334	-1	-4	3	58.174	3	-4	3
59.335	-1	-4	4	58.175	2	-4	4

- Age, Number, Start
- $\gamma = \log_{10}(\lambda)$: $-4,\ -3, \cdots,\ 3,\ 4$
- $9^3 \times 2$ models
- Polynomial limits

Varying coefficient models (VCM)

- Motivating example:
  price of IBM hard drives vs. size (GB)
- Monthly samples: Feb 1999 – Jan 2000



Combine months: one model

- VCMs allow coefficients to vary smoothly (interact) with another variable t (say, time or space)

$$\begin{aligned} \mu &= x(t)f(t) \\ &= \begin{pmatrix} x_1 & 0 & \cdots & 0 \\ 0 & x_2 & \cdots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \cdots & x_m \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_m \end{pmatrix} \\ &= \mathrm{diag}\{x\} \overbrace{B\alpha}^{f(t)} \\ &= U\alpha \end{aligned}$$

- $U = \mathrm{diag}\{x\}B$,   B-spline index is t

Estimation for VCMs

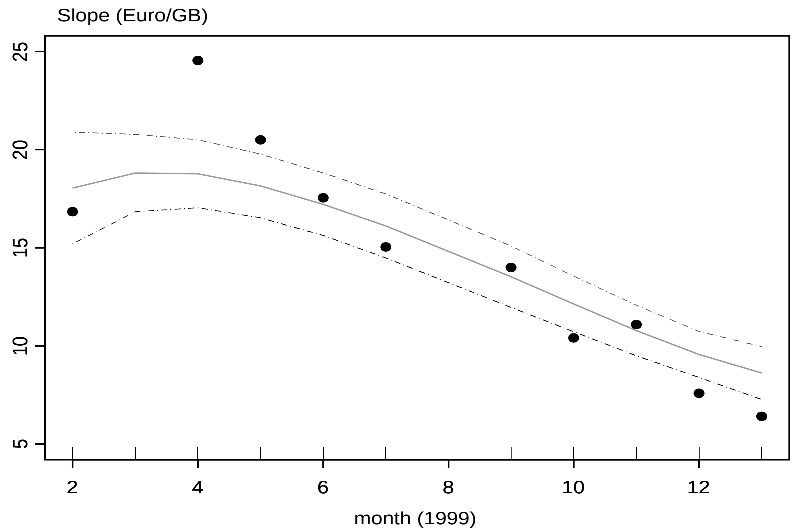
- Effective regressors for VCM: $U = \mathrm{diag}\{x\}B$
- Simple: only rows of B change
- Estimation reduces to (generalized linear) smoothing with U
- Normal (minimize):

$$\begin{aligned} Q &= |y - U\alpha|^2 + \lambda |D\alpha|^2 \\ \hat{\alpha} &= (U'U + \lambda D'D)^{-1} U'y \end{aligned}$$

- Poisson/Binomial (maximize):

$$\begin{aligned} l^* &= l(\alpha; U, y) - \frac{1}{2}\lambda |D_d \alpha|^2 \\ \hat{\alpha}_{t+1} &= (U'\hat{W}_t U + \lambda D'D)^{-1} U'\hat{W}_t \hat{z}_t \end{aligned}$$

Smooth slopes on month index



Cubic P-splines, nseg=10, pord=3, opt $\lambda = 1000$

P-spline VCMs

- Can imagine more VCM terms in (now linear) η
- Plays into hands of P-GAM structure:
  the B_j replaced with U_j in penalized likelihood
- Mixing and matching U's and B's
- Useful for months with missing data
- Useful for month with one observation
- Extrapolation to future months possible

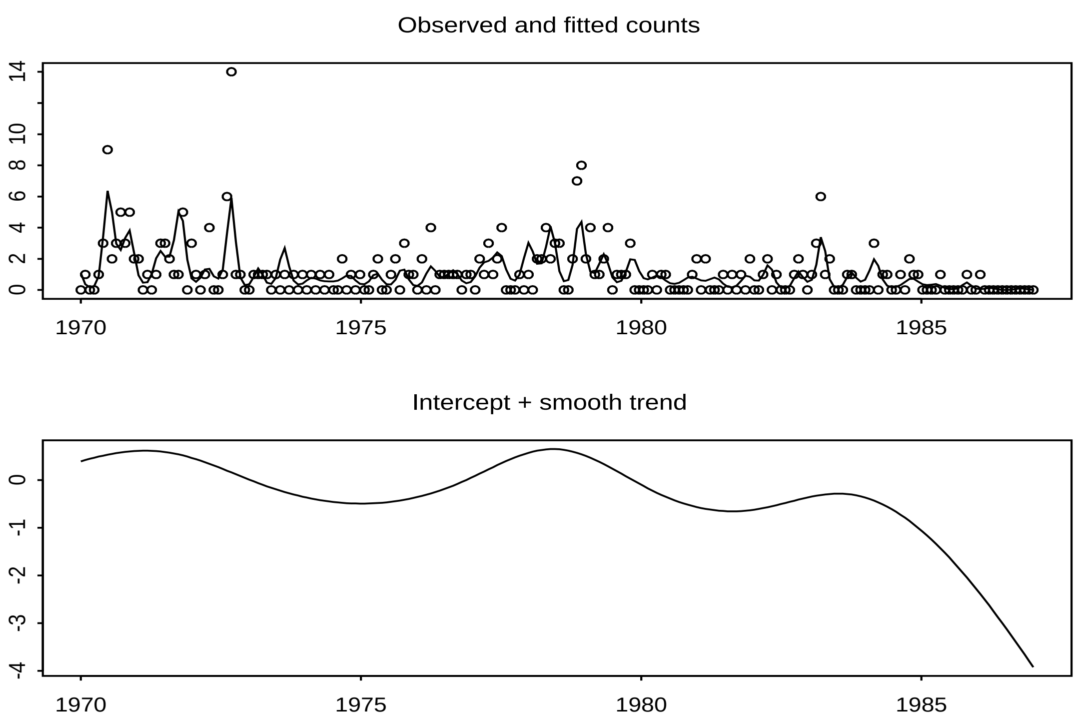
Non-normal example: U.S. polio counts

- Monthly counts: 1970 – 1987 ($m = 216$)
- Assume Poisson response: $g(\mu) = \log(\mu) = \eta$
- Consider *varying* (semi) annual harmonics

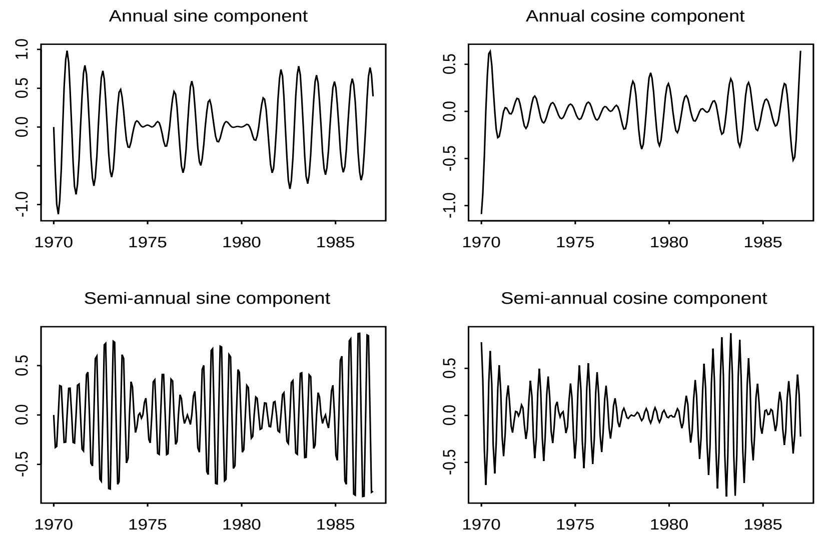
$$\log(\mu_i) = \underbrace{f_0(i)}_{B\alpha_0} + \sum_{k=1}^{2} \left(\sin(k\omega i) \underbrace{f_{1k}}_{B\alpha_1} + \cos(k\omega i) \underbrace{f_{2k}}_{B\alpha_2} \right)$$

- $\omega = 2\pi/12, \quad i$ is month index
- $\eta = [B_0|U_1|U_2] \cdot [\alpha_0'|\alpha_1'|\alpha_2']'$

Polio varying coefficients (nseg=10, pord=2)



Polio sine and cosine components



Software

www.stat.lsu.edu/bmarx

S-PLUS: function that works with existing `gam()`

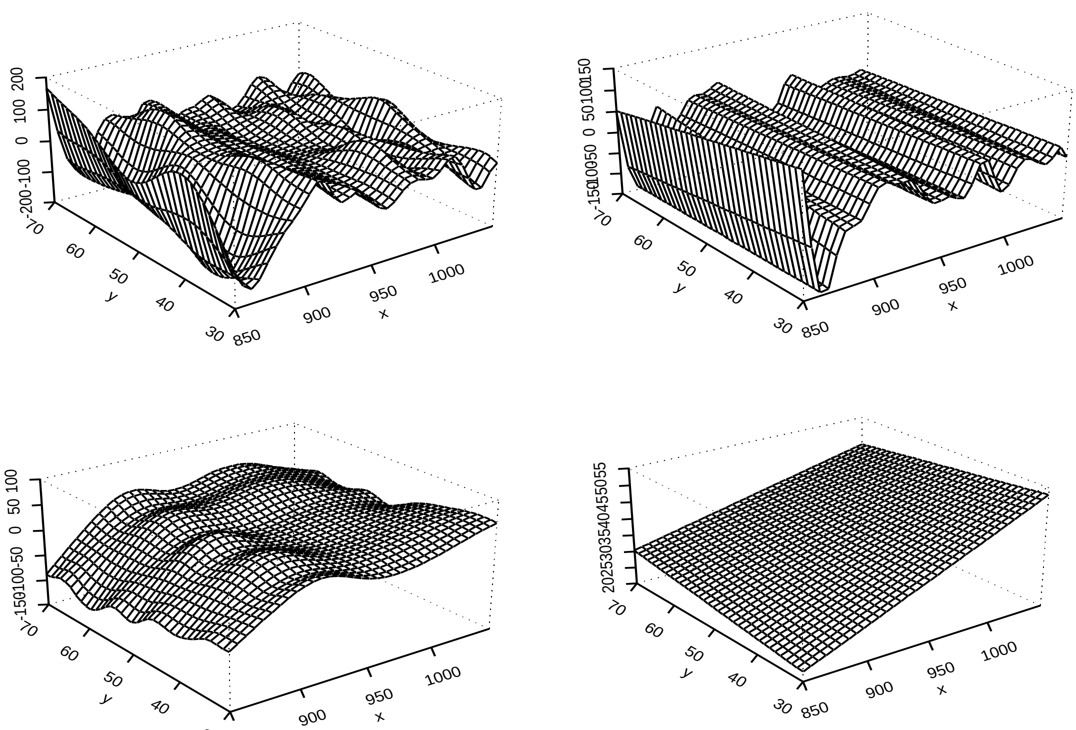
```
p.model1 <- gam( Kyphosis ~ Number +

glass(Age, ps.intervals=10,
           degree=3, order=3, lambda=1) +

glass(Start, ps.intervals=15, degree=2,
             order=1, lambda=.01, varying.index=Age),

family=binomial(link=logit),
data=kyphosis, na.action=na.omit  )
```

- Several defaults, list output, `plot.glass()`
- MATLAB code available for several common models

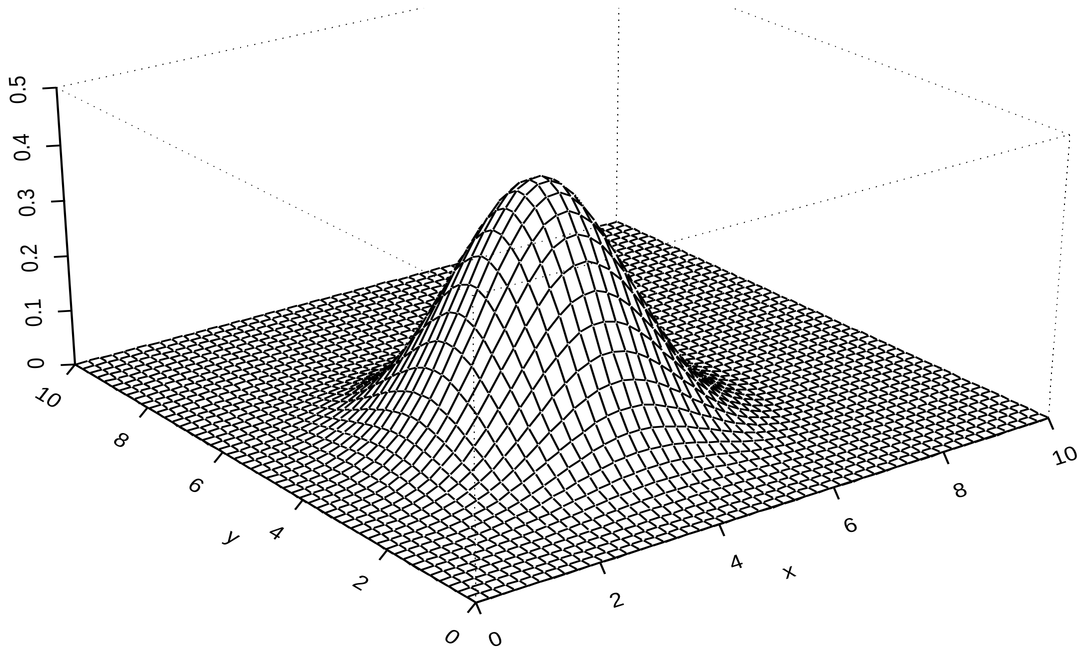
Two-dimensional smoothing with P-splines

- Two steps towards “smooth” surface
- First
  - Use tensor product B-splines: $T_{jk}(x, y) = B_j(x)\breve{B}_k(y)$
  - Equally spaced knots on 2D grid
  - Matrix of coefficients $A = [\alpha_{jk}]$
  - Purposely overfit: make “too wiggly”
- Second
  - Regularize: ensure further smoothness
  - Add penalties on tensor product coefficients
  - Difference penalties on rows/columns of A

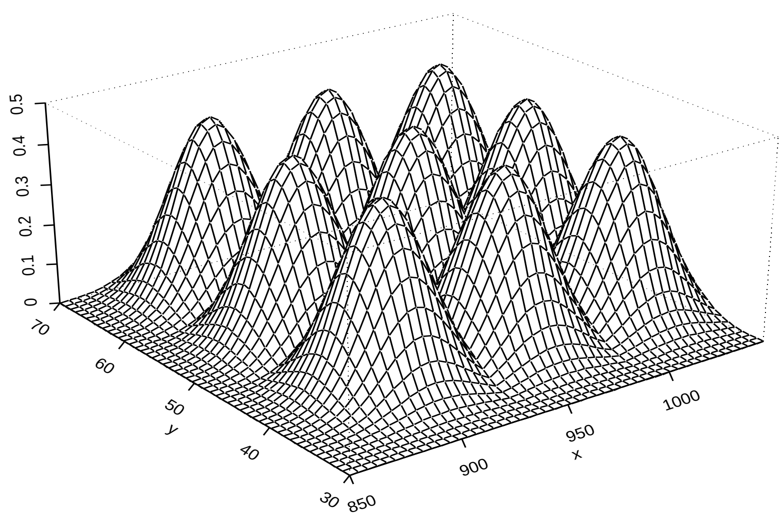
Examples of tensor products surfaces



Surface building block



Egg carton: portion of tensor product basis $(n \times \breve{n})$



Unknown coefficient matrix: $A_{n\times\breve{n}=[\alpha_{jk}]}$

Implementation

- Model contains matrix of coefficients A
- Transform to vector: $\beta = \text{vec}(A)$
- Kronecker product of bases

$$T = B_1 \otimes B_2$$

- T is of dimension $m \times (n\breve{n})$

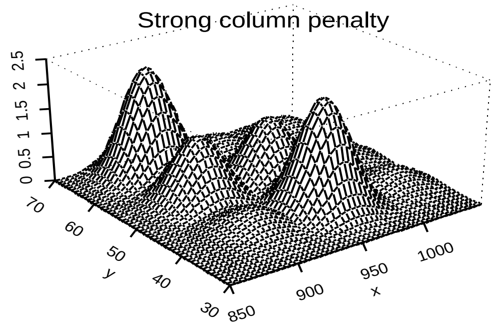
Two-dimensional penalized estimation

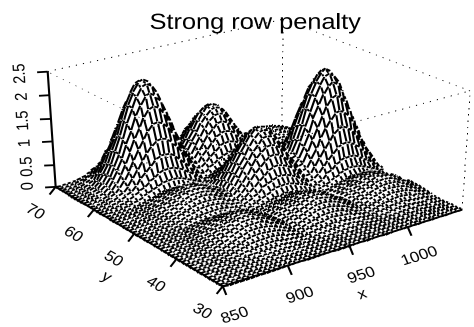
- Objective function

$$\begin{aligned} Q_P &= \text{RSS} + \text{Row Penalty} + \text{Column Penalty} \\ &= \text{RSS} + \lambda_1 \sum_{j=1}^{n} A_{j\bullet} D_d' D_d A_{j\bullet}' + \lambda_2 \sum_{k=1}^{\breve{n}} A_{\bullet k}' D_{\breve{d}}' D_{\breve{d}} A_{\bullet k} \\ &= |z - T\beta|^2 + \lambda_1 |P_1 \beta|^2 + \lambda_2 |P_2 \beta|^2. \end{aligned}$$

- Penalize rows of A with D_d
- Penalize columns of A with $D_{\breve{d}}$
- Number of equations is $n\breve{n}$

Visualization of strong penalty





Details of row and column penalties

- Must also carefully arrange ("stack") penalties
- Block diagonal to break (e.g. row to row) linkages:
  - $P_1 = D \otimes I_{\breve{n}}$
  - $P_2 = I_n \otimes D$

  For example:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \otimes D = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ & & & & & & & \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

Tensor product coefficient estimation

- Results in explicit solution:

$$\hat{\beta} = (T'T + \lambda_1 P_1' P_1 + \lambda_2 P_2' P_2)^{-1} T' z$$

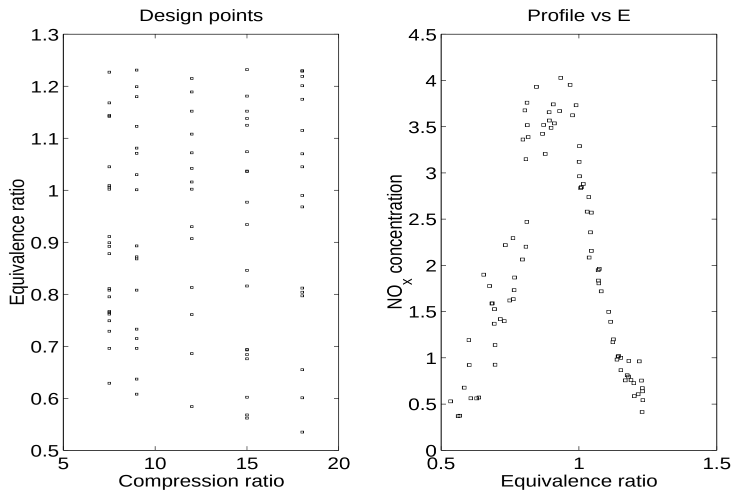
- Surface fitted values expressed in vector: $\hat{z} = T\hat{\beta}$
- In practice usually use higher order penalty

The recipe

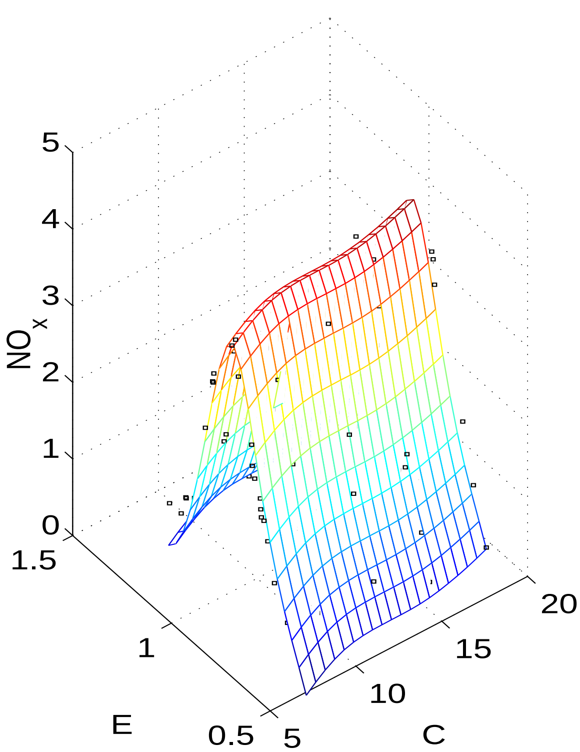
- Use a generous number of equally-spaced tensor product
- For computational efficiency, try to keep $n\breve{n} < 1000$
- Use penalty order $d = 2$ or 3
- Measure performance with CV or (A)IC
- Vary $(\lambda_1,\ \lambda_2)$ on a logarithmic grid search
- Find minimum of performance criterion
- Report $\hat{\beta}$, the P-spline coefficients:
  - a compact form of the estimated coefficient surface

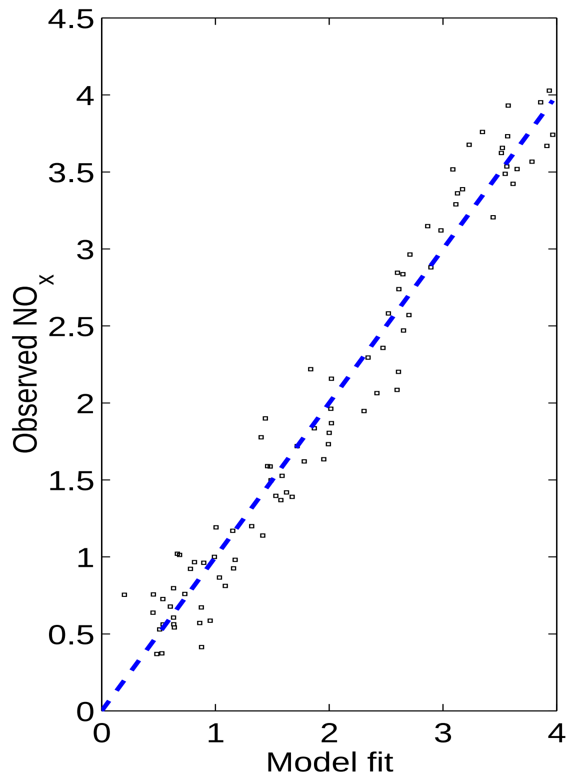
The ethanol data

- Nitrogen oxides in motor exhaust: NO_x (z)
- Compression ratio, C (x), equivalence ratio, E (y)

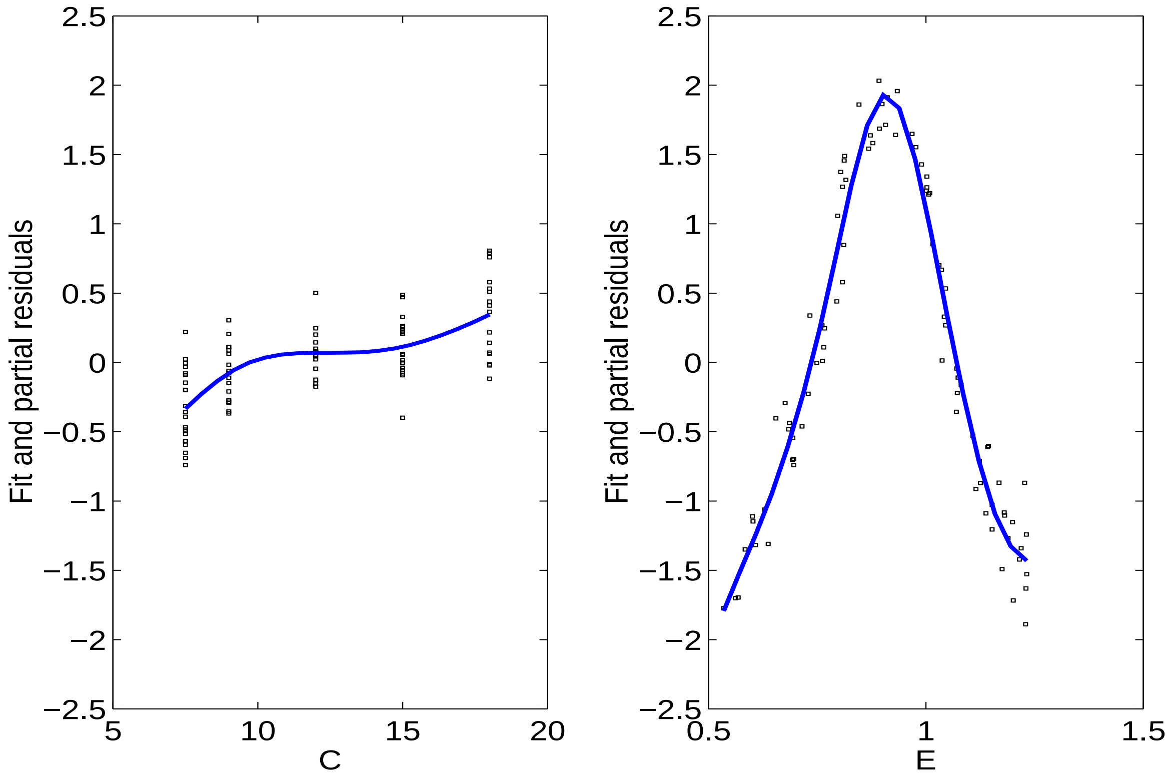


PGAM fit for ethanol data





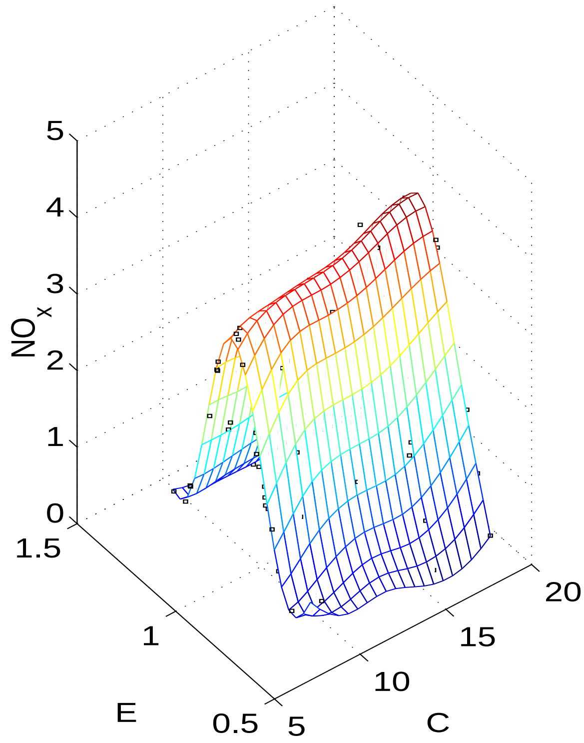
PGAM components for ethanol data

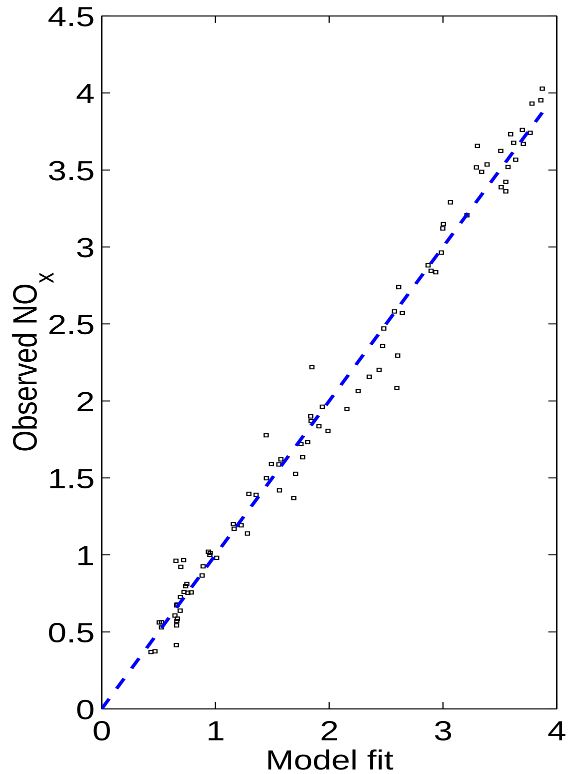


2D smoothing of ethanol data

- Tensor products of cubic B-splines
- Dimension: 64 (8 by 8)
- Fit computed on 400 points
- Time: 0.5 sec (MATLAB, Pentium 150 MHz)
- Residuals (SD) reduced to 60%, compared to GAM

Tensor P-spline fit to ethanol data

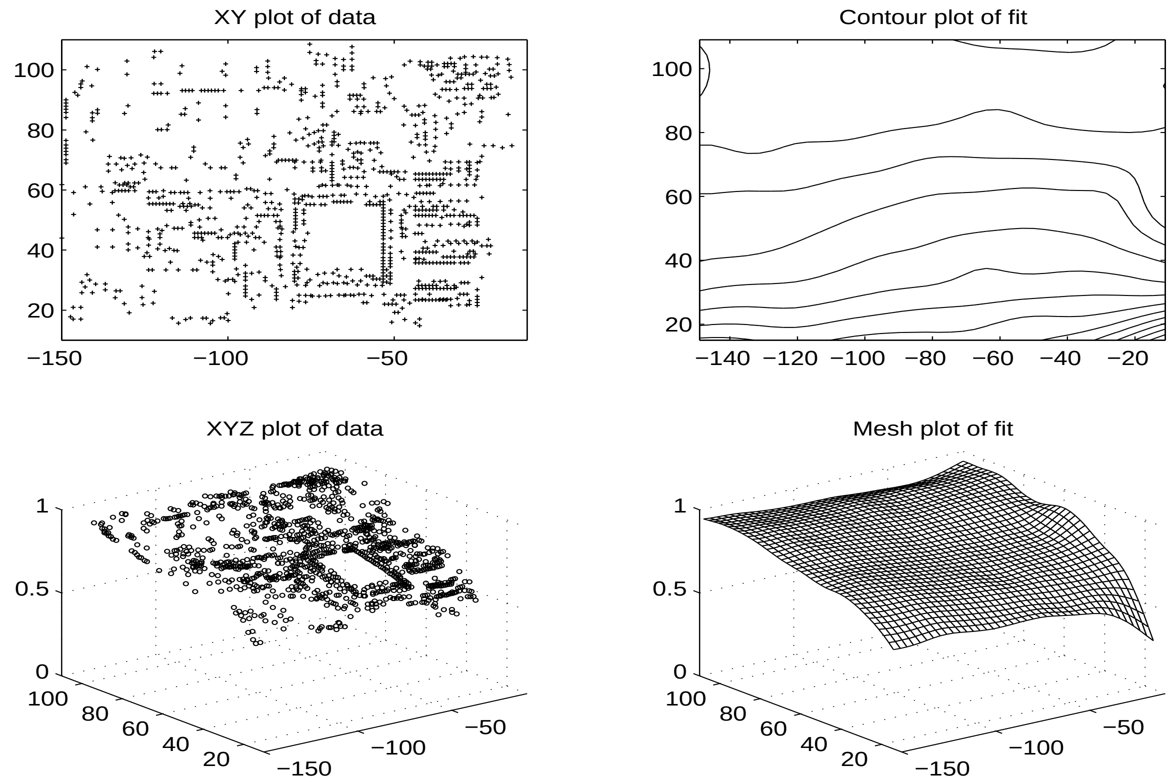




Another 2D application

- Printed circuit board
- Clamping causes warping (approx. 0.5 mm)
- Laser inspection of deformation
- Input: 1127 observations
- Cubic P-spline tensor products: 13 by 13
- Interpolation at 1600 points
- Time: 1.7 sec (MATLAB 6, Pentium III 1 GHz)

Printed circuit board data



Higher dimensions

- Triple (or higher) tensor products possible
- Difference penalty for each dimension
- Many equations: n^3 (n^4)
- Reduce number of B-splines
- Data generally sparse in more dimensions
- Much detail dubious
- (Cajo ter Braak did a project in 3D)

Wrap-up

- Extensions to generalized additive modelling
- Varying coefficient models
- 2-D P-spline smoothing
- *Next*: P-splines in perspective
- Details on P-spline connections with mixed models
- Bayesian connections

Session 6

P-splines in Perspective

Session 6

P-splines in Perspective

The Craft of Smoothing 6

What did you learn?

- Regression on local basis functions (B-splines)
- Penalty to (further) tune smoothness
- Standard errors, limits, interpolation, extrapolation
- Generalized linear smoothing
- Applications: scatterplots, (time) series, densities
- Optimal smoothing: cross-validation, AIC
- All you need for one-dimensional problems
- And a good dose of multi-dimensional smoothing

What else?

- Other interpretations of the penalty
- A Bayesian interpretation
- Mixed model interpretations
- Computation for alternative interpretations
- The effect of autocorrelation
- Comparison to other smoothers

Alternative interpretations of penalties

- Consider penalized least squares : minimize

$$Q = |y - B\alpha|^2 + \lambda|D\alpha|^2$$

- The penalty is rather useful
- But it seems to come out of the blue
- Can we connect it to established models?
- Yes: Bayes, mixed models

Introducing variances

- Rewrite the penalized least squares goal:

$$Q = \frac{|y - B\alpha|^2}{\sigma^2} + \frac{|D\alpha|^2}{\tau^2}$$

- Variance σ^2 of noise e in $y = B\alpha + e$
- Variance τ^2 of $D\alpha$
- First term: log of density of y, conditional on α
- Second term: log of (prior) density of $D\alpha$

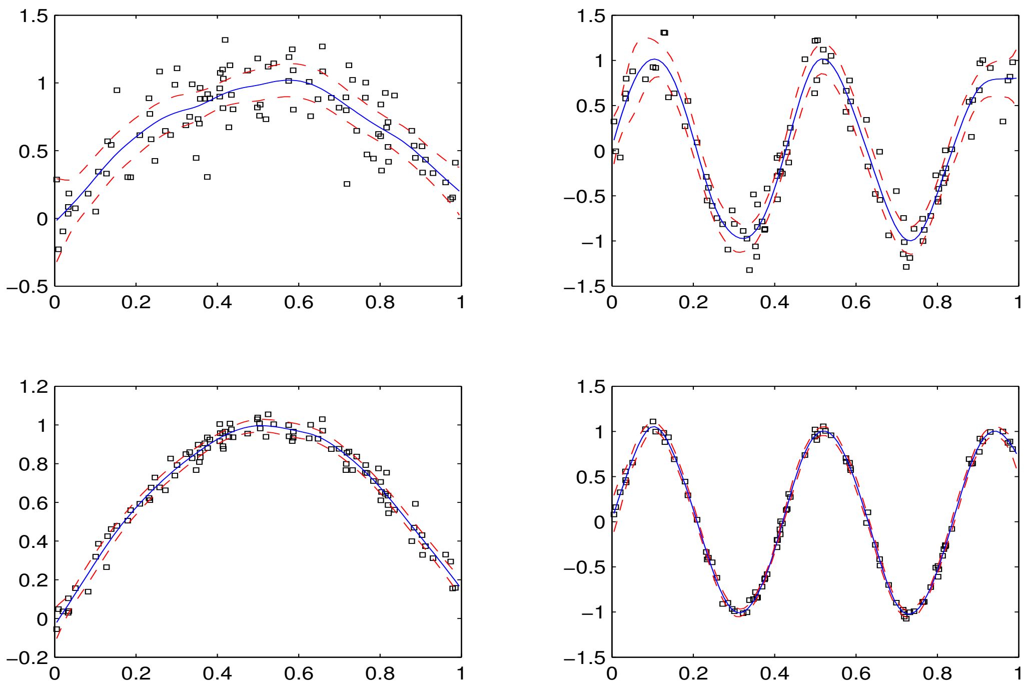
Bayesian simulation

- We look for posterior distributions of α, σ^2, τ^2
- Gibbs sampling
- “Draw” α conditional on σ^2 and τ^2
- “Draw” σ^2 and τ^2, conditional on α
- These are relatively simple subproblems
- Repeat many times, summarize results

Sketch of Bayesian P-splines MCMC steps

```
% Preparations
BB = B' * B;
By = B' * y;

% Update alpha
C = chol(BB / sig2 +  P / tau2);
a = C \ (C' \ (By) /sig2);  % solve system
a = C' \ randn(n, 1) + a;   % Gibbs with right covariance

% Update sigma^2, the observation variance
d1 = y' * y - 2 * a' * By + a' * BB * a;
sig2  = d1 / chi2(1);

% Update tau^2, the mixing variance
e =  D * a;
d2 =  e' * e;
tau2  = d2 / chi2(1);
```


Example of Bayesian P-splines



Pros and cons of Bayesian P-splines

- You fit P-spline thousand of times: much work
- All uncertainties quantified
- This not the case with optimizing AIC, CV
- Theory applies to non-normal smoothing too
- But simulations (of α) are much harder
- Metropolis-Hastings: problems with acceptance rates
- More on this: Lang *et al.*: papers, program BayesX

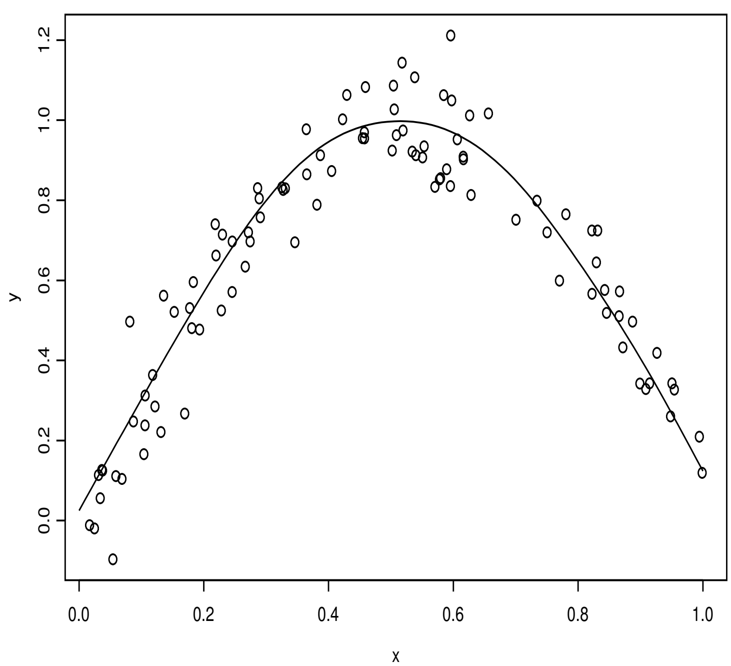
Mixed model

- See penalty as log of "mixing" distribution of $D\alpha$
- This is *not* natural, but very practical
- Mixed model software is good at estimating variance
- $D\alpha$ has degenerate distribution, rewrite the model
- Introduce "fixed" part X and "random" part Z
- $y = B\alpha = X\beta + Za$, with $Z = BD'(DD')^{-1}$
- And X containing powers of x up to $d - 1$
- Now a well behaved: independent components

Mixed model for P-splines in S-PLUS

```
# Based on work by Matt Wand

# Compute fixed (X) and mixed (Z) basis
B = bbase(x, 0, 1, 10, 3)
n = dim(B)[2]
d = 2;
D = diff(diag(n), differences = d)
Q = solve(D %*% t(D), D);
X = outer(x, 0:(d - 1), '^');
Z = B %*% t(Q)

# Fit mixed model
lmf = lme(y ~ X - 1, random = pdIdent(~ Z - 1))
beta.fix <- lmf$coef$fixed
beta.mix <- unlist(lmf$coef$random)
```


Example of P-spline fit with mixed model



EM-type algorithm for P-spline mixed model

- Deviance

$$-2l = m \log \sigma + n \log \tau + |y - B\alpha|^2/\sigma^2 + |D\alpha|^2/\tau^2$$

- ML solution ($\lambda = \sigma^2/\tau^2$)

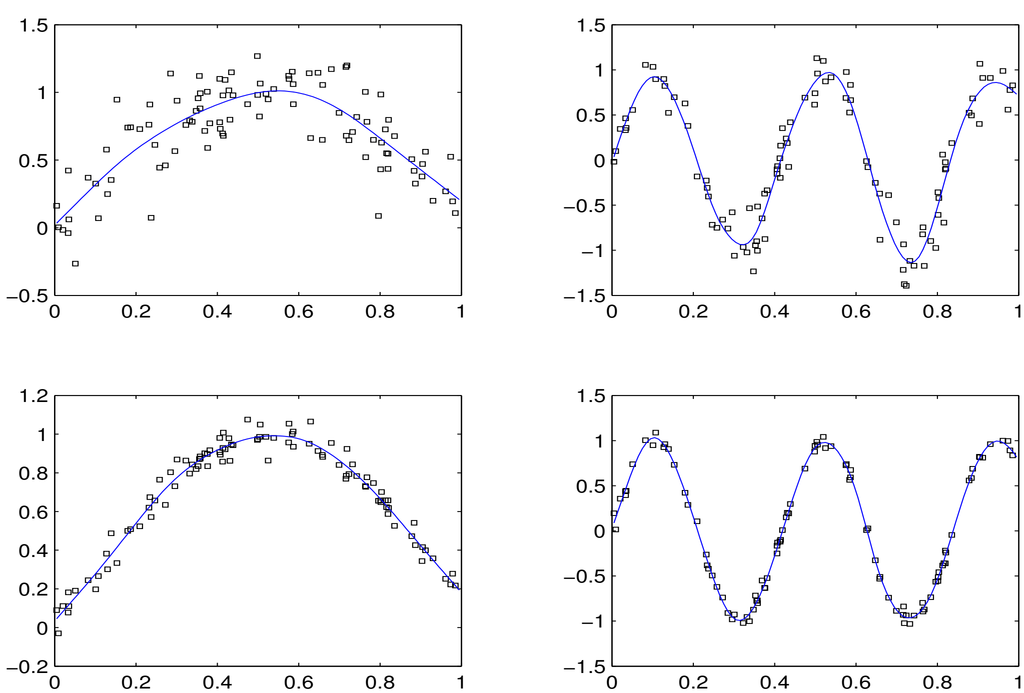
$$(B'B + \lambda D'D)\hat{\alpha} = B'y$$

- One can prove (ED is effective dimension):

$$E(|y - B\hat{\alpha}|^2) = (m - \mathrm{ED})\sigma^2; \quad E(|D\hat{\alpha}|^2) = \mathrm{ED}\tau^2$$

- Use these to estimate $\hat{\sigma}^2$ and $\hat{\tau}^2$ from fit
- Refit with $\lambda = \sigma^2/\tau^2$, repeat

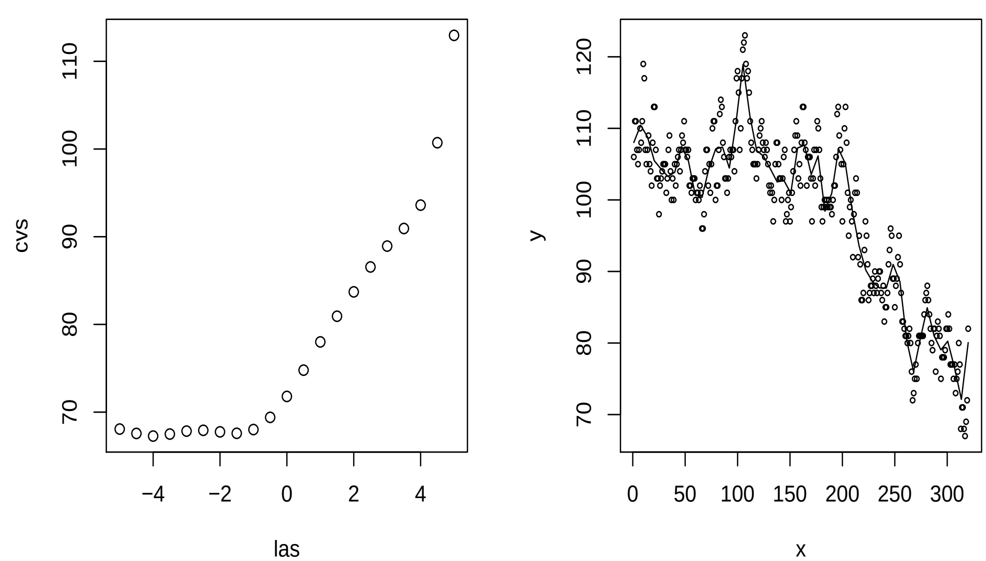
Example of P-spline fit with EM



Cross-validation and autocorrelation

- Cross-validation finds optimal prediction
- For each (left-out) data point
- Using the rest
- CV exploits colored noise
- To improve prediction
- Result a rather wiggly "trend"
- Bayesian and mixed models also assume uncorrelated errors!
- Here we show effect on cross-validation

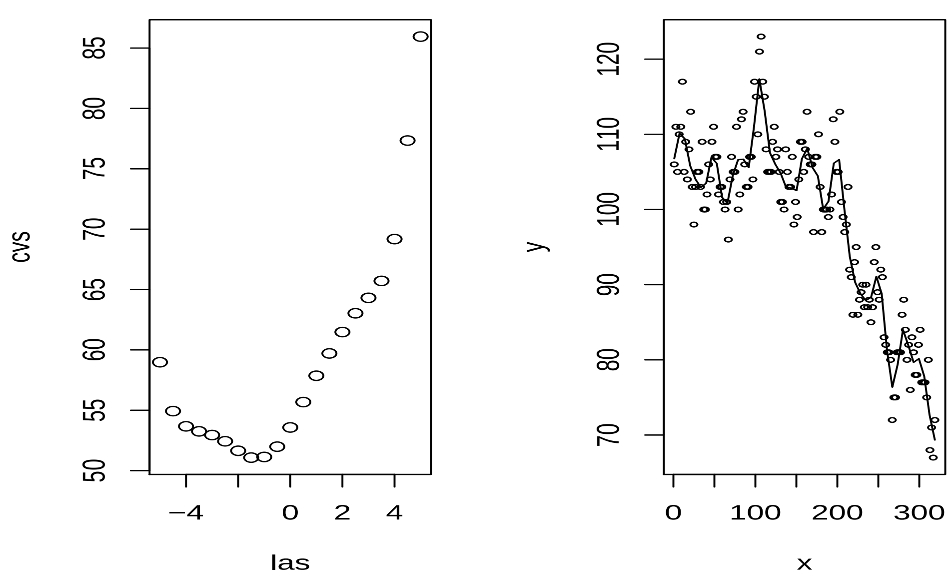
Wood surface: correlated noise



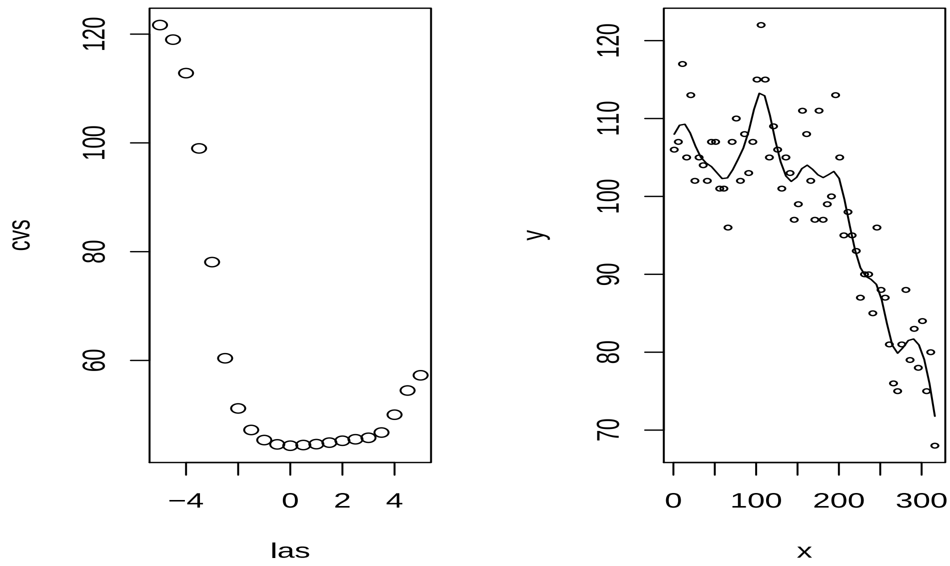
Breaking the correlation

- Use only every second (fifth, tenth, k-th) observation
- Weaker serial correlation over longer distance
- Use $k\lambda$ for smoothing all data
- To restore balance of residuals and penalty
- Refinement: combine k blocks of every k-th obs.

Every second observation



Every fifth observation

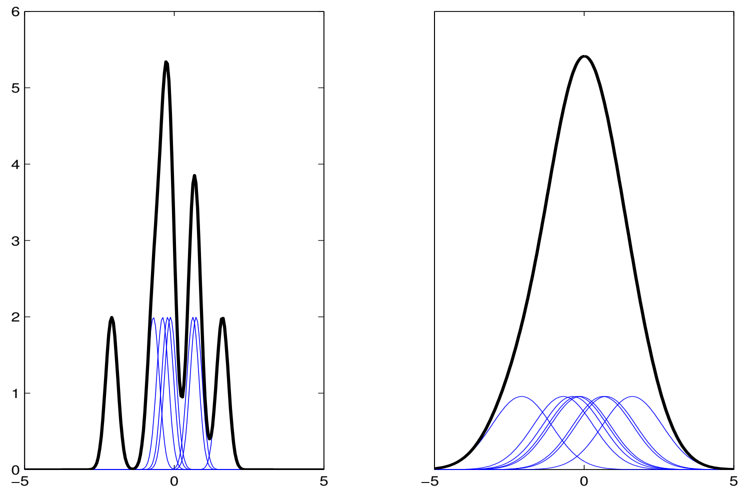


Other smoothers

- Kernels, weighted average
- Local likelihood regression
- Smoothing splines
- Adaptive-knots B-splines
- Wavelets
- “Consumer score card” in handout

Kernel density estimation

- Sum of (gaussian) kernels, centered at data x
- Kernel width determines amount of smoothing



Kernel-weighted average

- Data x and y.
- Weighted average:

$$\hat{f}(u) = \sum_i w(u - x_i) y_i$$

- Weights: Gaussian or similar shape
- Kernel width determines amount of smoothing

Problems of kernel smoothers

- Heavy computation (sum over all data)
- Expensive cross-validation, no effective dimension
- Boundary problems (when domain of x restricted)
- Variance increased (density estimation)
- No compact result
- No building block for GAM
- Useless for penalized signal regression

Local likelihood regression

- Compute weighted regression of y on x
- Fit linear or quadratic curve on interval
- Use kernel weights
- Keep curve fit in middle of interval
- Shift and repeat
- Fits in GLM framework for non-normal data
- Classic application: Savitzky-Golay smoother (1964)

Problems of local likelihood

- Heavy computation
- Expensive cross-validation
- No effective dimension
- No compact result
- No building block for GAM
- Useless for penalized signal regression

Smoothing splines

- Assume continuous function $f(x)$
- Minimize
$$\sum_i [y_i - f(x_i)]^2 + \lambda \int [f''(x)]^2 dx$$
- Continuous roughness penalty
- Result: piecewise cubic f
- Jumps in f''' at data points
- Large system of equations but banded

Problems of smoothing splines

- Fast computation only with specialized software
- Fast cross-validation needs very special software
- Effective dimension needs very special software
- No compact result
- Inefficient building block for GAM

Adaptive-knots B-splines

- P-splines use equally spaced knots
- Penalty tunes smoothness
- Alternative 1: only change number of B-splines
- Discrete control not enough
- Problems with gaps (along x) in data
- Alternative 2: find optimal non-uniform knot spacing
- Complex non-linear problem

Wavelets

- Basis functions of different widths in hierarchy
- Increasing detail: basis size doubles at each deeper level
- One basis function at first level
- Two at second level, four at third level, ...
- All have same shape, but width halved at each level
- Rich variety of basis functions

Problems of wavelets

- Only for equally spaced x
- Power of 2 for number of observations
- No missing data allowed
- No GLM, effective dimension
- Cross-validation misty
- Not very compact result

The “other” P-splines

- We use B-splines and difference penalties
- Ruppert, Wand and Carroll do it differently
- Truncated power functions as “random” part
- A polynomial “fixed” part
- Knots on quantiles of x (not equally spaced)
- Mixed model approach (equivalent to ridge penalty)
- See “Splines, Knot and Penalties” on the CD for critique

The End

P-splines are a good for you

Happy Smoothing!

Further reading

Currie, I., and Durbàn, M. (2003). A note on P-spline additive models with correlated errors. *Computational Statistics* **18**, 251–262.

Currie, I., Durbàn, M., and Eilers, P.H.C. (2003a). Using P-splines to extrapolate two-dimensional Poisson data. In: Proceedings of the 18th International Workshop on Statistical Modelling. Leuven, Belgium. Eds. G. Verbeke, G. Molenberghs, A. Aerts, and S. Fieuws, 97-102.

Currie, I., Durbàn, M., and Eilers, P.H.C. (2003b). Smoothing and forecasting mortality rates. Submitted to *Statistical Modelling*.

Daniel, C. and Wood, F.S. (1980). *Fitting Equations to Data*. Wiley, New York.

de Boor, C. (2001). *A Practical Guide to Splines*. Revised edition. Applied Mathematical Sciences **27**. Springer-Verlag, New York.

Dierckx, P. (1993). *Curve and Surface Fitting with Splines*. Clarendon Press, Oxford.

Dobson, A.J. (2002). *An Introduction to Generalized Linear Models, 2nd Edition*. Chapman and Hall, London.

Eilers, P.H.C (1999). Discussion of VERBYLA *et al.* (1999).

Eilers, P.H.C. and Marx, B.D. (1992). Generalized linear models with P-splines. In: Proceedings of GLIM 92 and 7th International Workshop on Statistical Modelling, Munich, Germany. Lecture Notes in Statistics, Vol. 78, Advances in GLIM and Statistical Modelling, Eds. L. Fahrmeir, B. Francis, R. Gilchrist, G. Tutz. Springer-Verlag, New York: 72-77.

Eilers, P.H.C. and Marx, B.D. (1996). Flexible Smoothing Using B-Splines and Penalized Likelihood (with Comments and Rejoinder). *Statistical Science* **11**(2): 89–121.

Eilers, P.H.C. and Marx, B.D. (2002). Generalized Linear Additive Smooth Structures. *Journal of Computational and Graphical Statistics* **11**(4): 758-783.

Eilers, P.H.C. and Marx, B.D. (2003). Multivariate calibration with temperature interaction using two-dimensional penalized signal regression. *Chemometrics and Intelligent Laboratory Systems*, **66**, 159-174.

Eilers, P.H.C. and Marx, B.D. (2004). Splines, knots, and penalties. Submitted to *Journal of the American Statistical Association*.

Härdle, W. (1990). *Applied Nonparametric Regression*. University Press, Cambridge.

Hastie, T. and Mallows, C. (1993). A Discussion of "A Statistical View of Some Chemometrics Regression Tools" by I.E. Frank and J.H. Friedman. *Technometrics*, **35**: 140–143.

Hastie, T. and Tibshirani, R. (1990). *Generalized Additive Models*. Chapman and Hall, London.

Lang, S. and Brezger, A. (2004) Bayesian P-Splines. *Journal of Computational and Graphical Statistics*, **13**: 183–212.

Marx, B.D. and Eilers, P.H.C. (1998). Direct Generalized Additive Modeling with Penalized Likelihood. *Computational Statistics and Data Analysis* **28**: 193-209.

Marx, B.D. and Eilers, P.H.C. (1999). Generalized Linear Regression on Sampled Signals and Curves: A P-Spline Approach. *Technometrics* **41**: 1-13.

Marx, B.D. and P.H.C. Eilers (2002) . Multivariate calibration stability: a comparison of methods. *Journal of Chemometrics* **16**, 1-12.

Marx, B.D. and P.H.C. Eilers (2004) . Multidimensional signal regression. Submitted to *Technometrics*.

McCullagh, P. and Nelder, J.A. (1989). *Generalized Linear Models* (2nd ed.), Chapman and Hall, London.

Myers, R.H. (1990). *Classical and Modern Regression with Applications, 2nd ed.,* Duxbury Press, Boston.

Nelder, J.A. and Wedderburn, R.W.M. (1972). Generalized Linear Models. *Journal of the Royal Statistical Society* A **135**: 370-384.

O'Sullivan, F. (1986). A Statistical Perspective on Ill-Posed Inverse Problems (with Discussion). *Statistical Science.* **1**, 505–527.

Reinsch, C. (1967). Smoothing by Spline Functions. *Numerische Mathematik* **10**: 177-183.

Ruppert, D. (2002). Selecting the number of knots for penalized splines. *Journal of Computational and Graphical Statistics* **11**(4), 735-757.

Ruppert, D. and Carroll, R.J. (2000). Spatially-Adaptive Penalties for Spline Fitting, *Australian and New Zealand Journal of Statistics* **42**, 205–223.

Ruppert, D., Wand, M.P., and Carroll, R.J. (2003) . *Semiparametric Regression*. Cambridge University Press, New York.

Verbyla, A.P., Cullis, B.R. and Kenward, M.G. (1999). The analysis of designed experiments and longitudinal data by using smoothing splines. *Applied Statistics* **48**, 269–300.

Wand, M. (2000) A comparison of regression spline smoothing procedures. *Computational Statistics* , 443–462.

Whittaker, E.T. (1923) On a new method of graduation. Proc. Edinburgh Math. Soc. **41**, 63-75.

Wood, S.N. (2000) Modelling and smoothing parameter estimation with multiple quadratic penalties. *Journal of the Royal Statistical Society B* **62**, 413–428.

Yee, T. and Wild, C.J. (1996). Vector Generalized Additive Models. *Journal of the Royal Statistical Society* B **58**, 481-493.

Consumer score card for smoothers

This score card is reproduced from our paper in Statistical Science (1996).

Aspect	*KS*	*KSB*	*LR*	*LRB*	*SS*	*SSB*	*RSF*	*RSA*	*PS*
Speed of fitting	—	+	—	+	—	+	+	+	+
Speed of optimization	—	+	—	+	—	+	—	—	+
Boundary effects	—	—	+	+	+	+	+	+	+
Sparse designs	—	—	—	—	+	+	—	+	+
Semi parametric models	—	—	—	—	+	—	+	+	+
Non-normal data	+	+	+	+	+	+	+	+	+
Easy implementation	+	—	+	—	+	—	+	—	+
Parametric limit	—	—	+	+	+	+	+	+	+
Specialized limits	—	—	—	—	+	+	—	—	+
Variance inflation	—	—	+	+	+	+	+	+	+
Adaptive flexibility possible	+	+	+	+	+	+	—	+	+
Adaptive flexibility available	—	—	—	—	—	—	—	+	—
Compact result	—	—	—	—	—	—	+	+	+
Conservation of moments	—	—	+	+	+	+	+	+	+
Easy standard errors	—	—	+	+	—	+	+	+	+

Consumer test of smoothing methods. The abbreviations stand for

KS kernel smoother

KSB kernel smoother with binning

LR local regression

LRB local regression with binning

SS smoothing splines

SSB smoothing splines with band solver

RSF regression splines with fixed knots

RSA regression splines with adaptive knots

PS P-splines

The row "Adaptive flexibility available" means that a software implementation is readily available.