A Crash Course on P-splines

Paul Eilers & Brian Marx

Channel Network Conference Nijmegen
April 2015

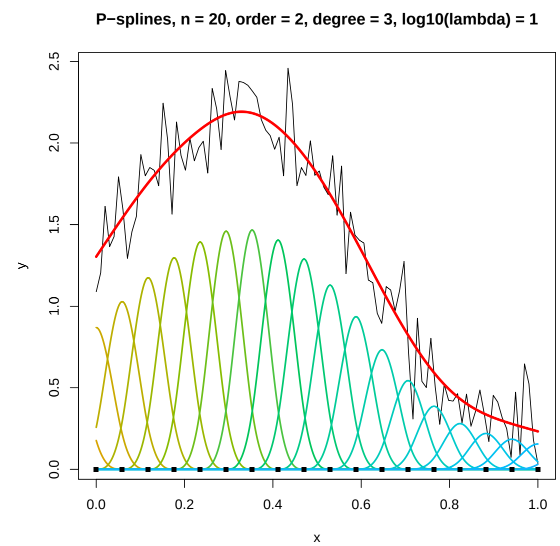
A Crash Course on P-splines

Introduction

What are P-splines?

- A flexible tool for smoothing
- Based on regression
- Local basis functions: B-splines
- No efforts to optimize the basis
- Just a large bunch of B-splines
- And a penalty to tune smoothness
- (Software demo: `PSPlay_psplines`)

Plot from `PSPlay_psplines` program



The roots of P-splines

- Eilers and Marx: Statistical Science, 1996
- In fact not a very revolutionary proposal
- A simplification of O'Sullivan's ideas
- But the time seemed right
- Now almost 1000 citations
- Many from applied areas (what really counts for us)
- E&M evangelized heavily
- A variety of applications, many in this course

The plan of the course

- We start with penalties
- They are the core ingredient
- Splines come later
- They just "add the flesh to the skeleton"
- Basic (generalized) linear smoothing
- Extensions: generalized additive models, 2-D smoothing
- Bayesian and mixed model interpretations
- Specialized penalties

Part 1

The power of penalties

Discrete smoothing

- Given: data series y_i, $i = 1, \ldots, m$
- Wanted: a smooth series z
- Two (conflicting) goals: fidelity to y *and* smoothness
- Fidelity, sum of squares: $S = \sum_i (y_i - z_i)^2$
- How to quantify smoothness?
- Use roughness instead: $R = \sum_i (z_i - z_{i-1})^2$
- Simplification of Whittaker's (1923) "graduation"

Penalized least squares

- Combine fidelity and roughness

$$Q = S + \lambda R = \sum_i (y_i - z_i)^2 + \lambda \sum_i (z_i - z_{i-1})^2$$

- Parameter λ sets the balance
- Operator notation: $\Delta z_i = z_i - z_{i-1}$

$$Q = \sum_i (y_i - z_i)^2 + \lambda \sum_i (\Delta z_i)^2$$

Matrix-vector notation

- Penalized least squares objective function

$$Q = ||y - z||^2 + \lambda ||Dz||^2$$

- Differencing matrix D, such that $Dz = \Delta z$

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

- Explicit solution: $\hat{z} = (I + \lambda D'D)^{-1} y$

Implementation in R

```
m <- length(y)
E <- diag(m)      # Identity matrix
D <- diff(E)      # Difference operator
G <- E + lambda * t(D) %*%  D
z <- solve(G, y)  # Solve the equations
```

Notes on computation

- Linear system of equations
- m equations in m unknowns
- Practical limit with standard algorithm: $m \approx 4000$
- System is extremely sparse (bandwidth = 3)
- Specialized algorithms easily handle $m > 10^6$
- Computation time then linear in m

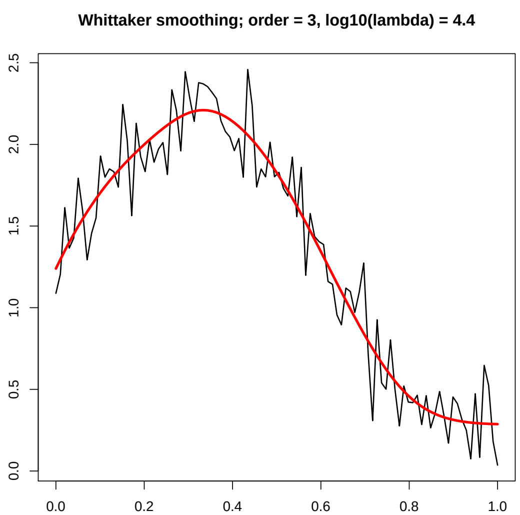
Higher order penalties

- Higher order differences are easily defined
- Second order: $\Delta^2 z_i = \Delta(\Delta z_i) = (z_i - z_{i-1}) - (z_{i-1} - z_{i-2})$
- Second order differencing matrix

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

- Higher orders are straightforward
- In R: `D = diff(diag(m), diff = d)`

Plot from `PSPlay_discrete` program



The effects of higher orders

- Smoother curves
- Polynomial limits for large λ
- Degree of interpolation
- Degree of extrapolation
- Conservation of moments (will be explained later)
- (Software demo: `PSPlay_discrete`)

Limits

- Consider large λ in $Q = ||y - z||^2 + \lambda ||Dz||^2$
- Penalty is overwhelming, hence essentially $Dz = \Delta z = 0$
- This is the case if $z_i - z_{i-1} = 0$, hence $z_i = c$, a constant
- Generally: $\Delta^d z = 0$ if z is order $d - 1$ polynomial in i
- Linear limit when $d = 2$, quadratic when $d = 3$, ...
- It is also the least squares polynomial

Interpolation and extrapolation

- Let y_i be missing for some i
- Use weights w_i (0 if missing, 1 if not)
- Fill in arbitrary values (say 0) for missing y
- Minimize, with $W = \text{diag}(w)$

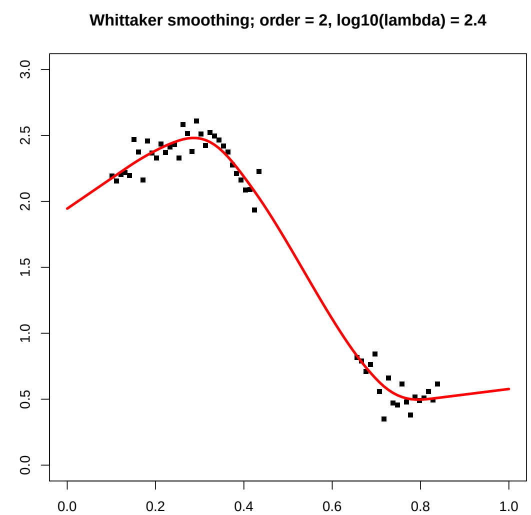
$$Q = (y - z)'W(y - z) + \lambda ||Dz||^2$$

- Trivial changes: $\hat{z} = (W + \lambda D'D)^{-1}Wy$

Interpolation and extrapolation, continued

- Interpolation is by polynomial in i
- Order $2d - 1$
- Extrapolation: introduce "missing" data at the end(s)
- Extrapolation is by polynomial in i
- Order $d - 1$
- (Software demo: `PSPlay_interpolation`)

Plot from `PSPlay_interpolate` program



Non-normal data

- We measured fidelity by the sum of squares
- This is reasonable for (approximately) normal data
- Which means: trend plus normal disturbances
- How will we handle counts?
- Or binomial data?
- Use penalized (log-)likelihood
- Along the lines of the generalized linear model (GLM)

Smoothing of counts

- Given: a series y of counts
- We model a smooth linear predictor η
- Assumption: $y_i \sim \text{Pois}(\mu_i)$, with $\eta_i = \log \mu_i$
- The roughness penalty is the same
- But fidelity measured by deviance (-2 LL):

$$Q = 2\sum_i (\mu_i - y_i \eta_i) + \lambda \sum_i (\Delta^d \eta_i)^2$$

Linearization and weighted least squares

- Derivatives of Q give penalized likelihood equations

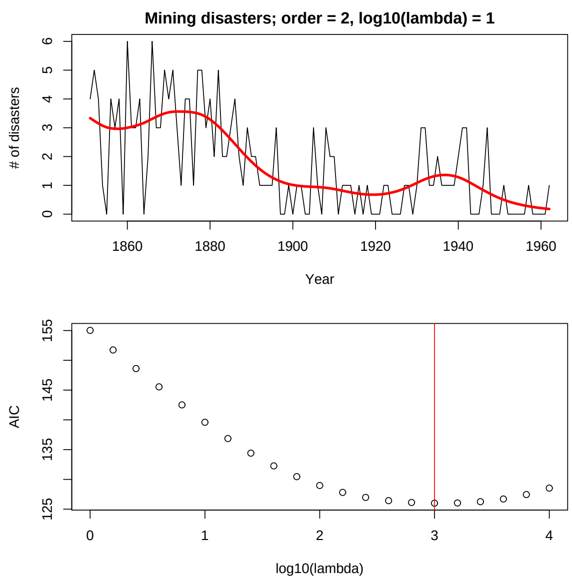
$$\lambda D'Dz = y - e^{\eta} = y - \mu$$

- Non-linear, but the Taylor approximation gives

$$(\tilde{M} + \lambda D'D)\eta = y - \tilde{\mu} + \tilde{M}\tilde{\eta}$$

- Current approximation $\tilde{\eta}$, and $\tilde{M} = \text{diag}(\tilde{\mu})$
- Repeat until (quick) convergence
- Start from $\tilde{\eta} = \log(y + 1)$

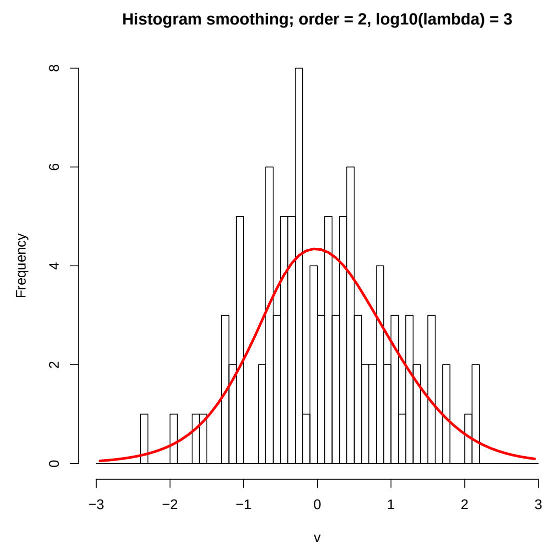
Example: coal mining accidents



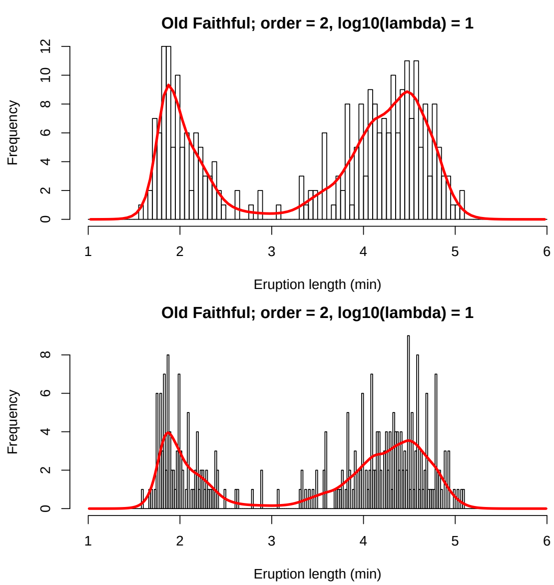
A useful application: histogram smoothing

- The "Poisson smoother" is ideal for histograms
- Bins can be very narrow
- Still a smooth realistic (discretized) density estimate
- Conservation of moments
- $\sum_i y_i x_i^k = \sum_i \hat{\mu}_i x_i^k$ for integer $k < d$ (bin midpoints in x)
- With $d = 3$, mean and variance don't change
- Whatever the amount of smoothing
- (Software demo: `PSPlay_histogram`)

Plot from `PSPlay_histogram` program



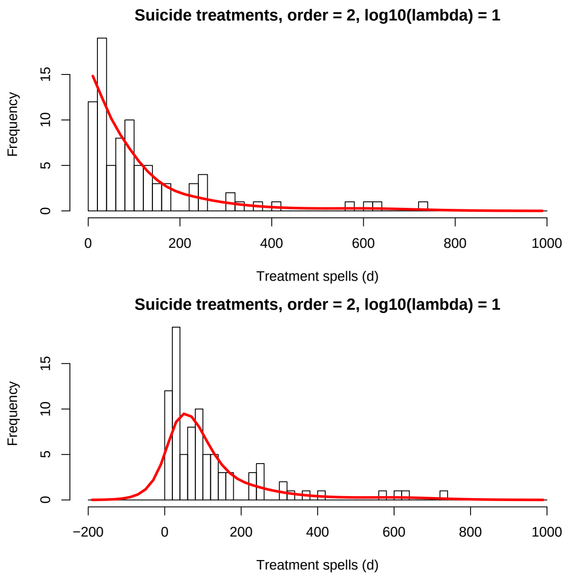
Smoothing old Faithful



Respect the boundaries

- Extend the histogram with enough zero counts
- But some data are inherently bounded
- Non-zero, or between 0 and 1
- Then you should limit the domain accordingly
- Otherwise you will smooth in the "no go" area
- Example: suicide treatment data
- Inherently non-negative durations

Smoothing the suicide treatment data



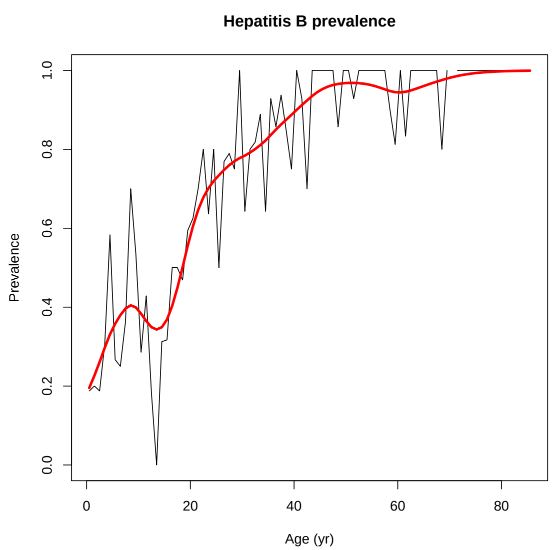
Binomial data

- Given: sample sizes s, “successes” y
- Smooth curve wanted for p, probability of succes
- We model the logit:

$$z = \log \frac{p}{1-p}; \quad p = \frac{e^z}{1+e^z} = \frac{1}{1+e^{-z}}$$

- Linearization as for counts
- Start from logit of $(y+1)/(s+2)$
- No surprises, details skipped

Example: hepatitis B prevalence (Keiding)



Optimal smoothing

- We can smooth almost anything (in GLM sense)
- How much should we smooth?
- Let the data decide
- Cross-validation, AIC (BIC)
- Essentially we measure prediction performance
- On new or left-out data

Leave-one-out cross-validation

- Leave out y_i (make w_i zero)
- Interpolate a value for it: $\hat{y}_{-i}$
- Do this for all observations in turn
- You get a series of "predictions"
- How good are they?
- Use $CV = \sum(y_i - \hat{y}_{-i})^2$, or $RMSCV = \sqrt{CV/m}$
- Search for λ that minimizes CV

Speeding up the computations

- LOO CV looks expensive (repeat smoothing m times)
- It is, if done without care
- But there is a better way
- We have $\hat{y} = (W + \lambda D'D)^{-1}Wy = Hy$
- We call H the hat matrix; property: $h_{ij} = \partial\hat{y}_i/\partial y_j$
- One can prove: $y_i - \hat{y}_{-i} = (y_i - \hat{y}_i)/(1 - h_{ii})$
- Smooth once (for each λ), compute all $\hat{y}_{-i}$ at the same time

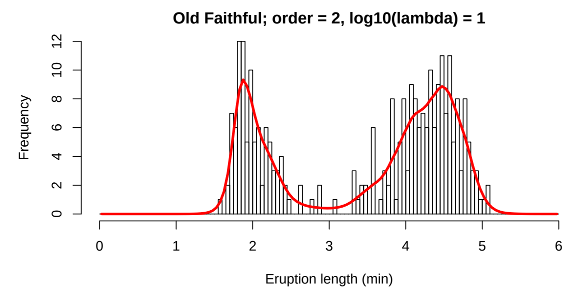
Akaike's information criterion

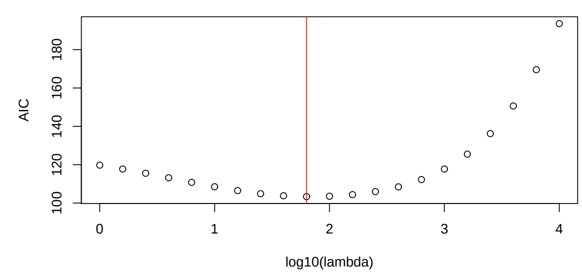
- Definition: $AIC = \text{Deviance} + 2ED = -2LL + 2ED$
- Here ED is the effective model dimension
- Useful definition:

$$ED = \sum_i \partial\hat{\mu}_i/\partial y_i = \sum_i h_{ii} = \text{tr}(H)$$

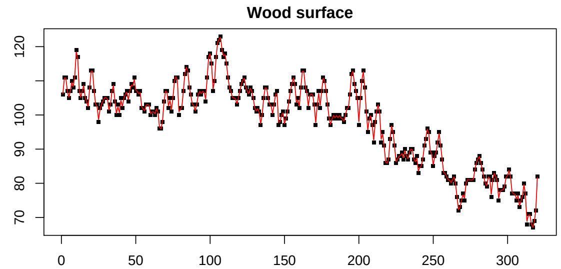
- This defines a hat matrix for generalized linear smoothing
- Vary λ on a grid to find minimum of AIC
- Minimization routine can be used too
- But it is useful to see the curve of AIC vs. $\log\lambda$

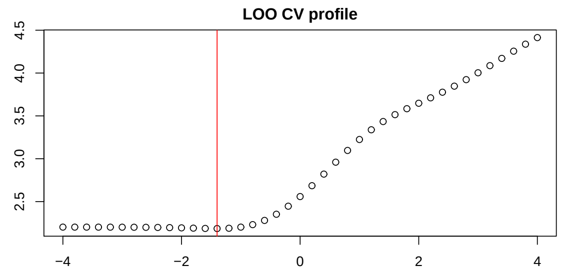
A convincing example: Old Faithful





A worrying example: a wood surface





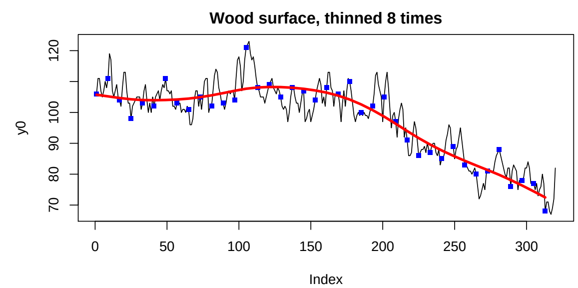
What went wrong?

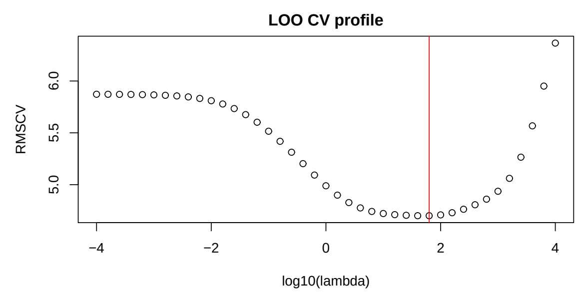
- The (silent) assumption: trend plus independent noise
- Here the noise is correlated
- LOO CV means: best prediction of left-out data
- Light smoothing gives better predictions
- That is not what we had in mind
- The smooth trend is not automatically detected

Two solutions

- The elegant solution: model correlated noise
- This has been done (Currie and Durban)
- A lot of extra work
- Simple alternative: take every fifth (tenth) observation
- Thinning observations breaks correlation
- Scale final λ by f^{2d}
- If f is the thinning factor

Thinning to break correlation





Similar problems with histograms

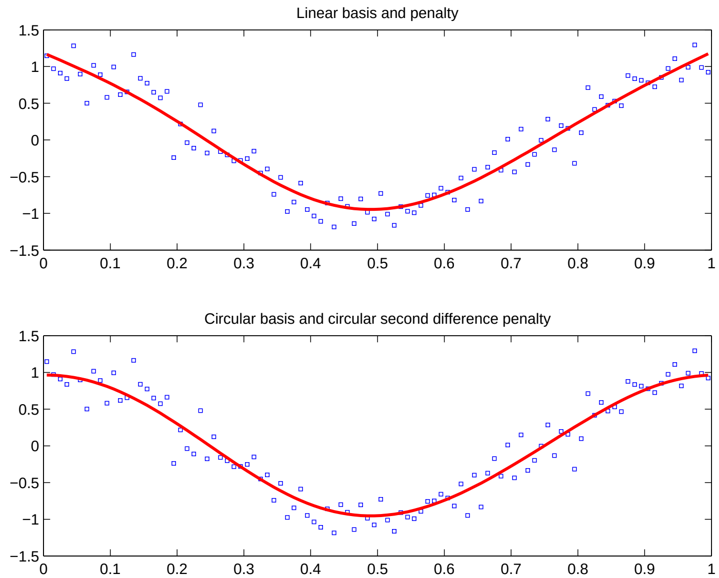
- If counts are a time series, AIC can fail
- Again serial correlation is the cause
- Other histograms show digit preference
- People read an analog scale or estimate a number
- Examples: blood pressure in mm (mostly even numbers)
- Age, or birth date: rounding to multiples of five.
- Solution: model digit preference (non-trivial)
- Or use your carpenter's eye

Circular smoothing

- Sometimes the data are circular
- Because we look at one period (or more)
- Then we wish that both ends connect smoothly
- Modify difference matrix with extra row(s), like

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

Illustration of circular smoothing



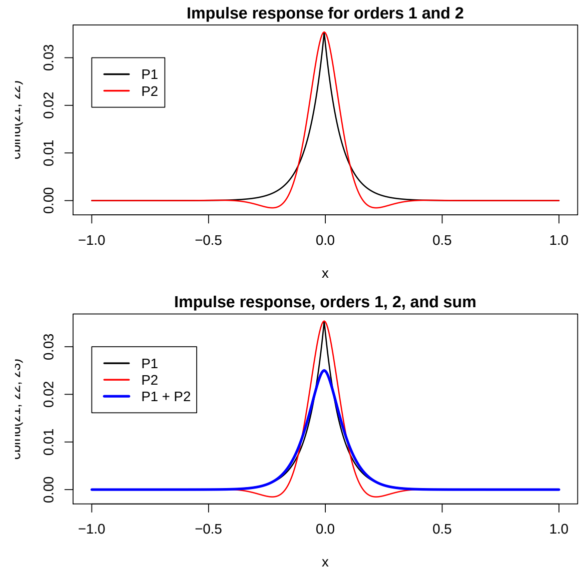
Designer penalties

- By now you should have got the message
- The penalty pushes the result in the desired direction
- For special cases special penalties may be needed
- Example 1: a non-negative impulse response
- Example 2: harmonic smoothing
- Example 3: monotone smoothing

Impulse response

- Consider special “data”
- All zeros, but one 1 (an impulse)
- The result of smoothing we call the impulse response
- It shows how data get “smeared our”
- For $d = 2$, it has negative side lobes
- This might not be desirable
- Solution: use penalty $\lambda^2 ||D_2 z||^2 + 2\lambda ||D_1 z||^2$
- Here D_1 (D_2) forms first (second) differences

Illustration of positive impulse response



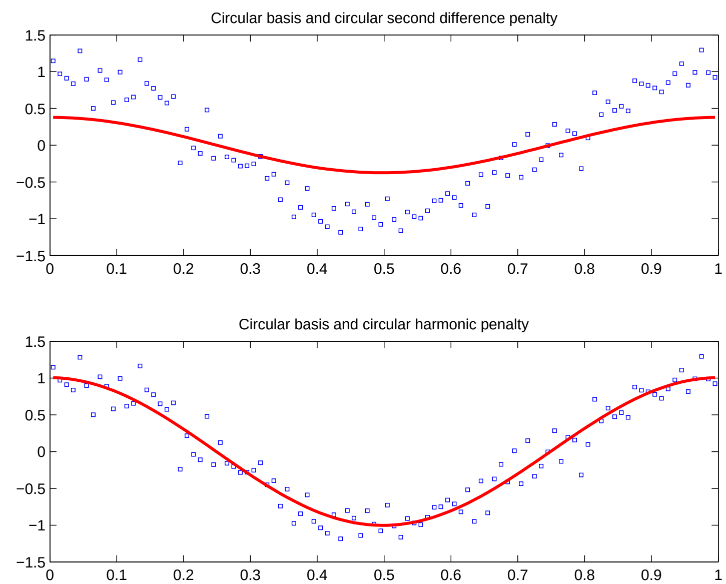
Harmonic smoothing

- Assume periodic data, period p
- Wanted: smooth limit that approaches (co)sine
- Solution: a specialized penalty

$$R = \sum_i (z_i - 2\phi z_{i-1} + z_{i-2})^2$$

- Where $\phi = \cos(2\pi/p)$

Illustration of harmonic smoothing



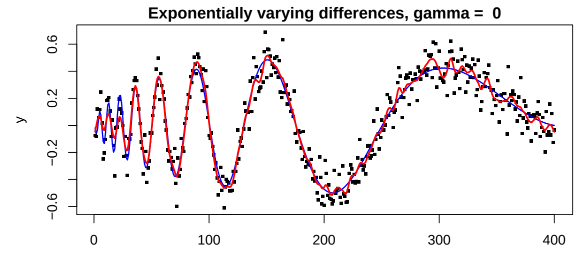
Varying penalties

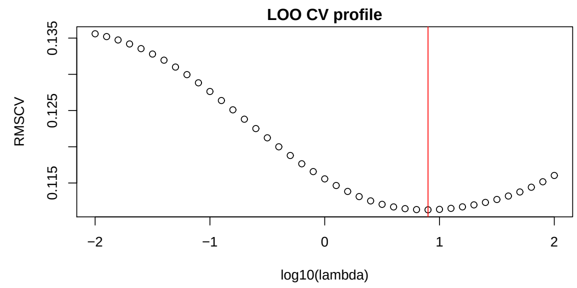
- Our penalties had the same weight everywhere
- But we can change that:

$$R = \lambda \sum_i v_i (\Delta^d z_i)^2$$

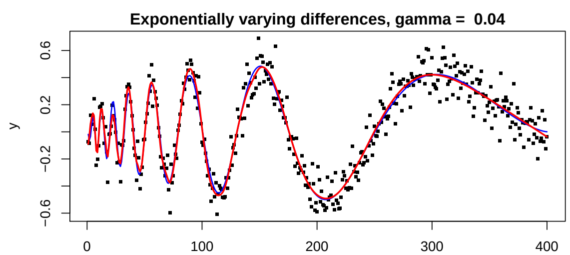
- Or, with $V = \mathrm{diag}(v)$, $R = z' D' V D z$
- New problem: how to choose v?
- Simple choice: $v_i = \exp(\gamma i)$
- Optimize λ and γ

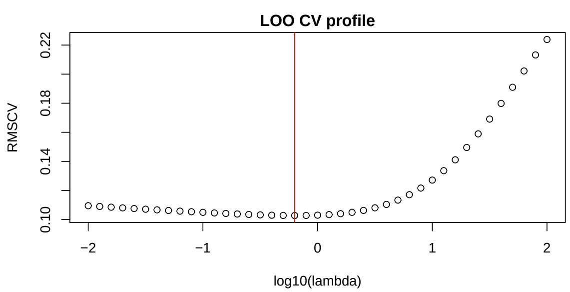
Swept sine, constant penalty





Swept sine, exponentially varying penalty

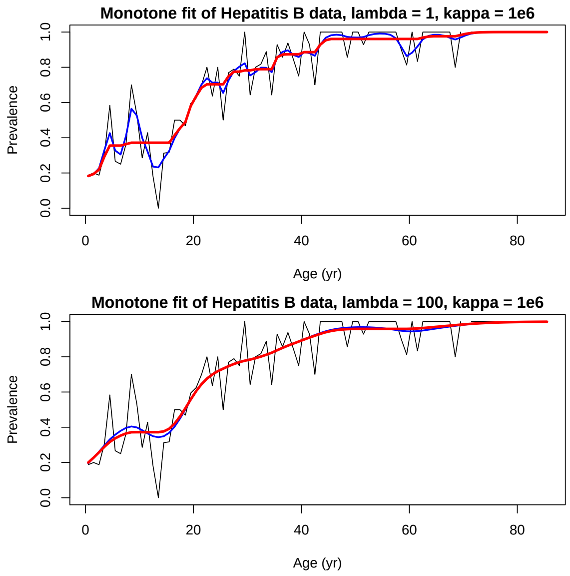




Asymmetric penalties and monotone smoothing

- Sometimes we want a smooth increasing result
- Smoothing alone does not guarantee a monotone shape
- We need a little help
- Additional asymmetric penalty $P = \kappa \sum_i v_i (z_i - z_{i-1})^2$
- With $v_i = 1$ if $z_i < z_{i-1}$ and $v_i = 0$ otherwise
- The penalty only works where monotonicity is violated
- With large κ we get the desired result
- This idea also works for convex smoothing

Example of monotone smoothing



Wrap-up

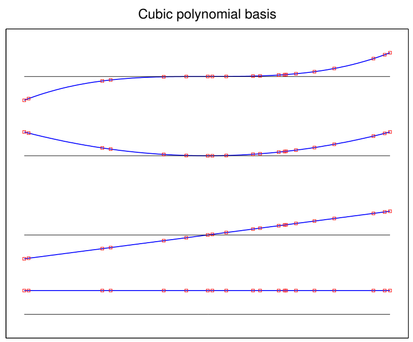
- The discrete smoother is simple and powerful
- It can be used for normal and non-normal data
- Penalty pushes solution in desired direction
- Penalty fills gaps in the data
- Desirable limits: polynomial or (co)sine
- “Designer penalties” open up new terrain
- Data have to be equally spaced (but gaps are allowed)
- Next session: the real thing, combination with B-splines

Part 2

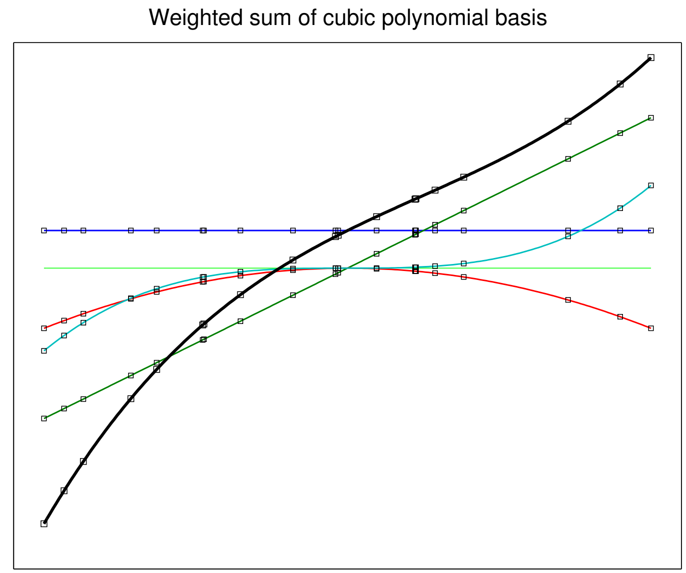
The splendor of splines

Basis functions for polynomial curve fit

- Regression model $\mu = X\alpha$
- Columns of matrix X: basis functions. Polynomial basis

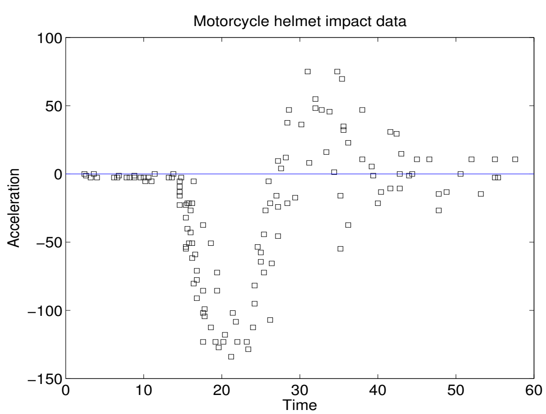


Basis functions scaled and added



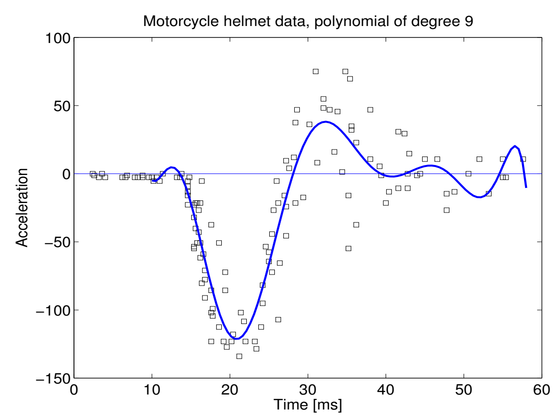
The motorcycle data

- Simulated crash experiment, a clasic in smoothing
- Acceleration of motorcycle helmets measured



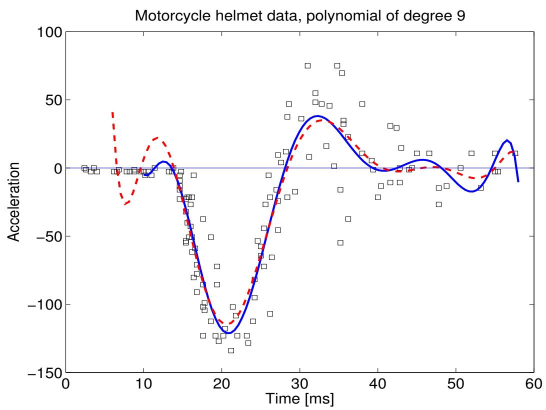
Polynomial fit to motorcycle data

- High degree (here 9) needed for decent curve fit
- Bad numerical condition (use orthogonal polynomials)



Sensitivity to data changes

- Longer left part (near zero)
- Notice the wiggles, also at the right

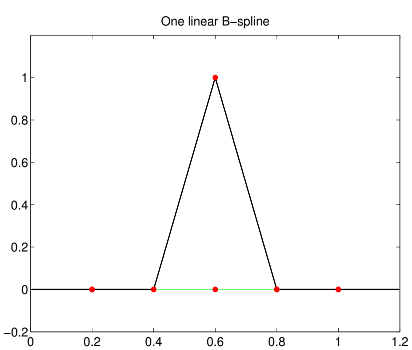


The trouble with polynomials

- High degree (10 or more) may be needed
- Basis functions (powers of x) are global
- Moving one end (vertically) moves the other end too
- Good fit at one end spoils it at the other end
- Unexpected, but unavoidable, wiggles
- The higher the degree the more sensitive
- Polynomials are not a great choice
- We switch to B-splines

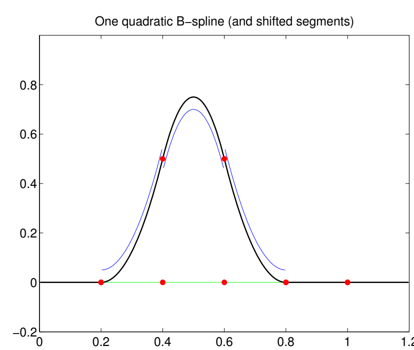
One linear B-spline

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



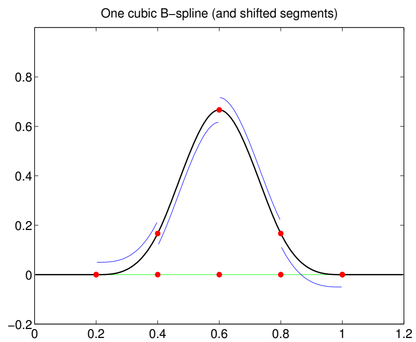
One quadratic B-spline

- 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

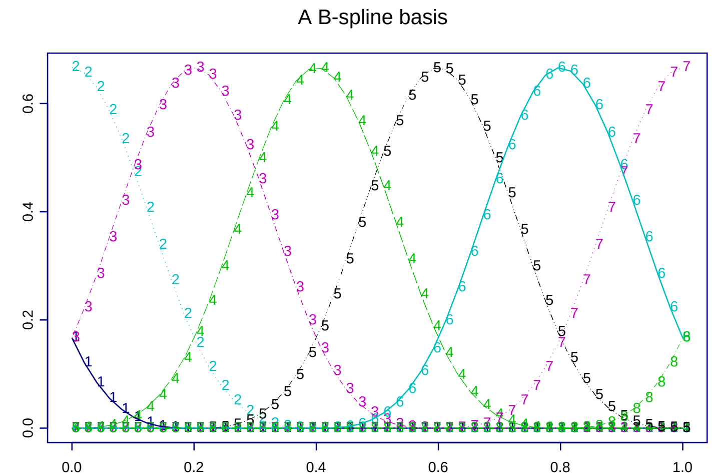


One cubic B-spline

- 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



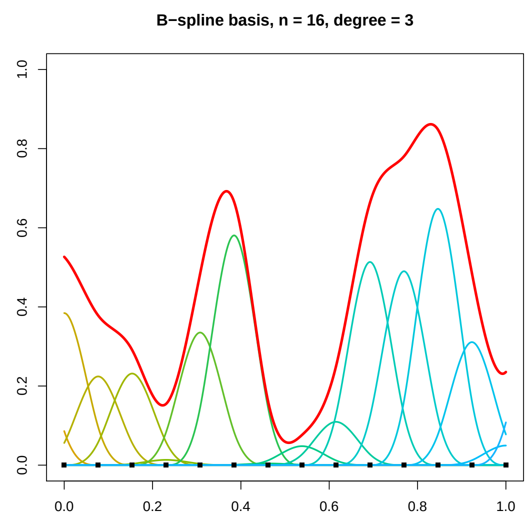
B-spline basis

- Basis matrix B
- Columns are B-splines

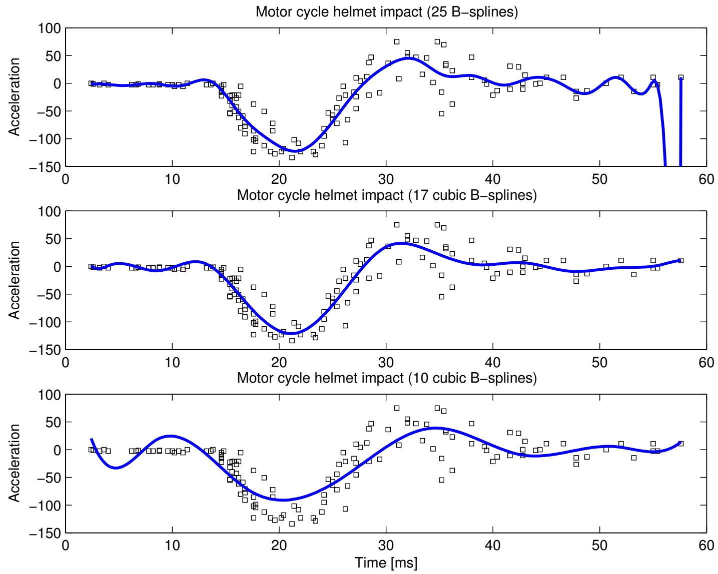
$$\begin{bmatrix} B_1(x_1) & B_2(x_1) & B_3(x_1) & \dots & B_n(x_1) \\ B_1(x_2) & B_2(x_2) & B_3(x_2) & \dots & B_n(x_2) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ B_1(x_m) & B_2(x_m) & B_3(x_m) & \dots & B_n(x_m) \end{bmatrix}$$

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

Plot from `PSPlay_bsplines` program



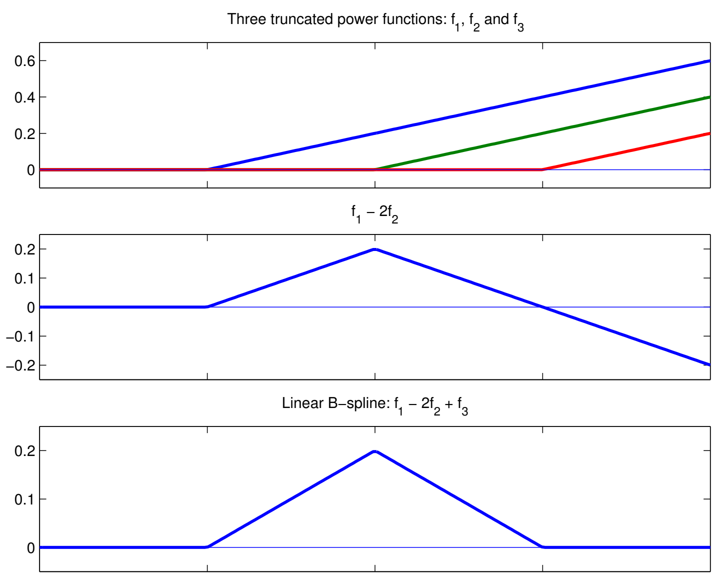
B-splines fit to motorcycle data



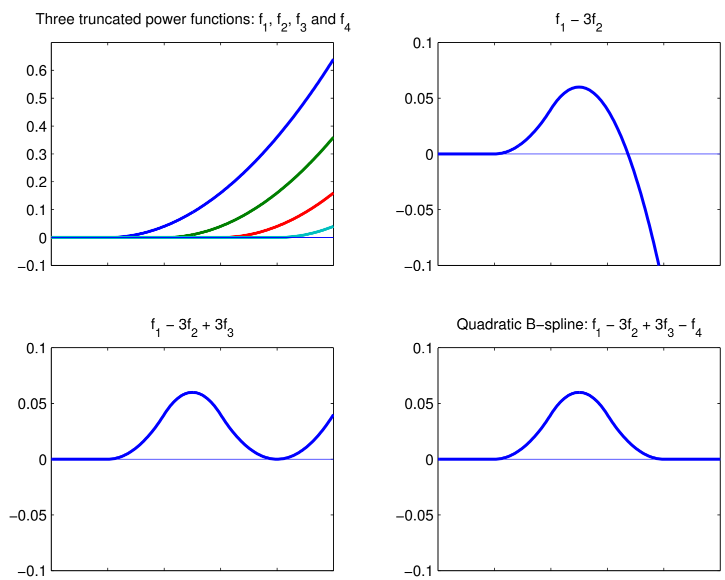
How to compute B-splines

- You can work from first principles
- Compute parameters of the polynomial segments
- Nine (3 times 3) coefficients, 8 constraints, height arbitrary
- Easier: recursive formula De Boor
- Even more easy: 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
- But we only consider equal spacing
- Number of knots determines width and number of B-splines

P-splines on one slide

- Do regression on (cubic) B-splines
- Use equally spaced knots
- Take a large number of them (10, 20, 50)
- Put a difference penalty (order 2 or 3) on the coefficients
- Tune smoothness with λ (penalty weight)
- Don't try to optimize the number of B-splines
- Relatively small system of equations (10, 20, 50)
- Arbitrary distribution of x allowed

Technical details of P-splines

- Minimize (with basis B)

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

- Explicit solution:

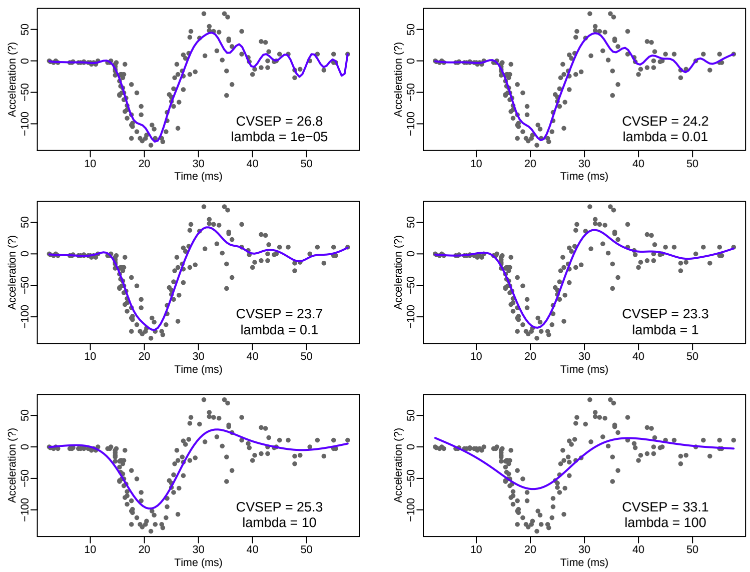
$$\hat{\alpha} = (B'B + \lambda D'D)^{-1} B'y$$

- Hat matrix $H = (B'B + \lambda D'D)^{-1} B'$
- For a nice curve, compute B^* on nice grid x^*
- Plot $B^*\hat{\alpha}$ vs x^*

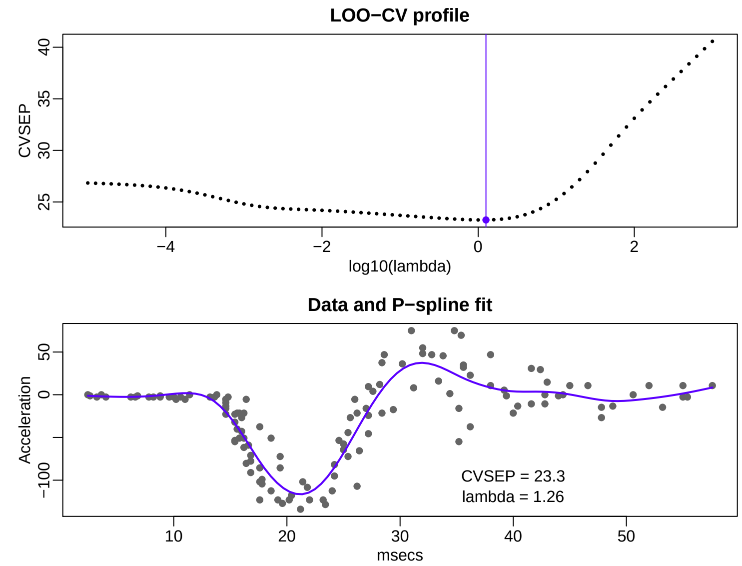
Properties of P-splines

- Penalty $\sum_j (\Delta^d \alpha_j)^2$
- Limit for strong smoothing is a polynomial of degree $d - 1$
- Interpolation: polynomial of degree $2d - 1$
- Extrapolation: polynomial of degree $d - 1$
- Conservation of moments of degree up to $d - 1$
- Many more B-splines then observations allowed
- The penalty does the work!
- (Software demo: `PSPlay_psplines`)

Motorcycle helmet data



Optimal P-spline fit based on CVSEP



Standard errors

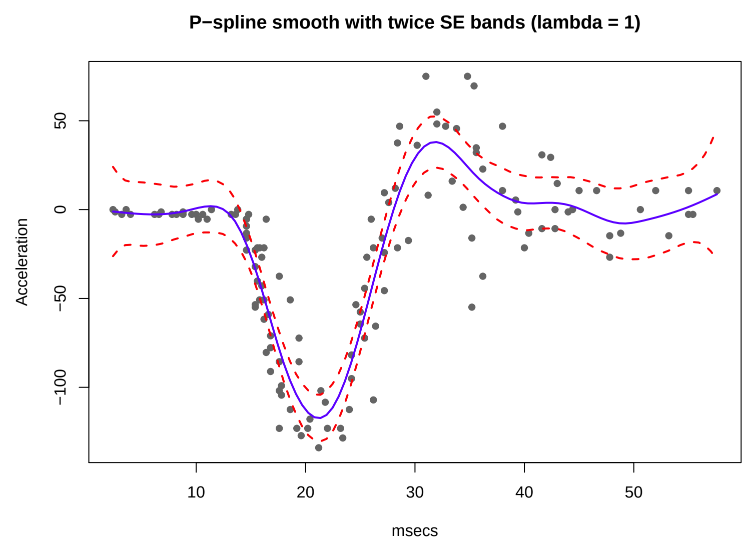
- Sandwich estimator

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

- 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



Generalized linear smoothing

- It is just like a GLM (generalized linear model)
- With the penalty sneaked in
- Poisson example for counts y
- Linear predictor $\eta = B\alpha$, expectations $\mu = e^{\eta}$
- Assumption $y_i \sim \text{Pois}(\mu_i)$ (independent)
- From penalized Poisson log-likelihood follows iteration with

$$(B'\tilde{M}B + \lambda D'D)\alpha = B'(y - \tilde{\mu} + \tilde{M}B\tilde{\alpha})$$

- Here $M = \text{diag}(\mu)$

Alternative interpretations of penalties

- Consider penalized least squares : minimize

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

- That penalty is rather useful
- But it seems to come out of the blue
- Can we connect it to established models?
- Yes: Bayes, or 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 contrast $D\alpha$
- First term: log of density of y, conditional on α
- Second term: log of (prior) density of $D\alpha$
- So λ is a ratio of variances: $\lambda = \sigma^2/\tau^2$

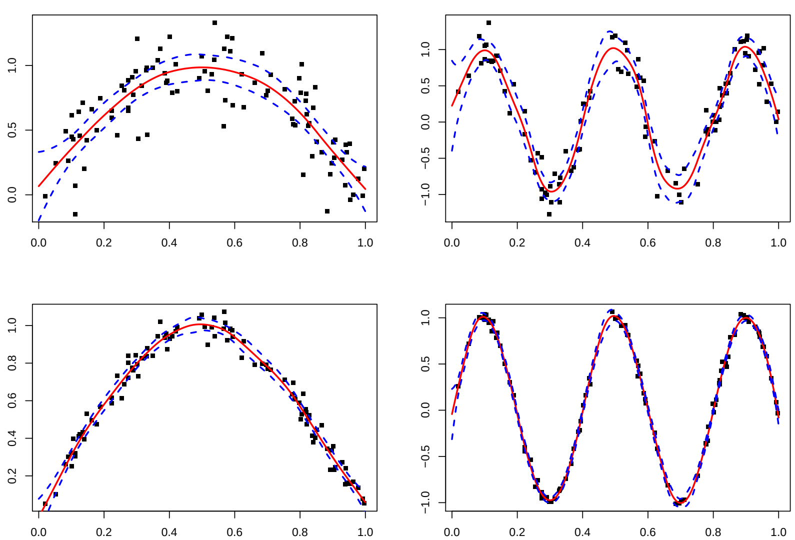
Bayesian simulation

- We look for posterior distributions of α, σ^2, τ^2
- Use 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

```
# Prepare some useful summaries
BB = t(B) %*% B; By = t(B) %*% y; yy = t(y) %*% y; P = t(D) %*% D

# Run a Markov chain (loop not shown):

# Update coefficients
U = BB / sig2 + P / tau2
Ch = chol(U)
a0 = solve(Ch, solve(t(Ch),  By)) / sig2;
a = solve(Ch, rnorm(length(a0))) + a0;

# Update error variance
r2 = yy - 2 * t(a) %*% By + t(a) %*% BB %*% a;
sig2 = as.single(r2 / rchisq(1, m));

# Update roughness variance
r = D %*% a;
tau2 = as.single(t(r) %*% r / rchisq(1, nb - 2));
```


Example of Bayesian P-splines



Pros and cons of Bayesian P-splines

- You fit P-splines thousand of times: much work
- But all uncertainties are quantified
- This not the case when optimizing AIC, CV
- Theory applies to non-normal smoothing too
- But simulations (of α) are much harder
- Metropolis-Hastings: acceptance rates need tuning
- More on this: Lang *et al.*: papers, program BayesX
- More modern tools: Langevin sampler, INLA

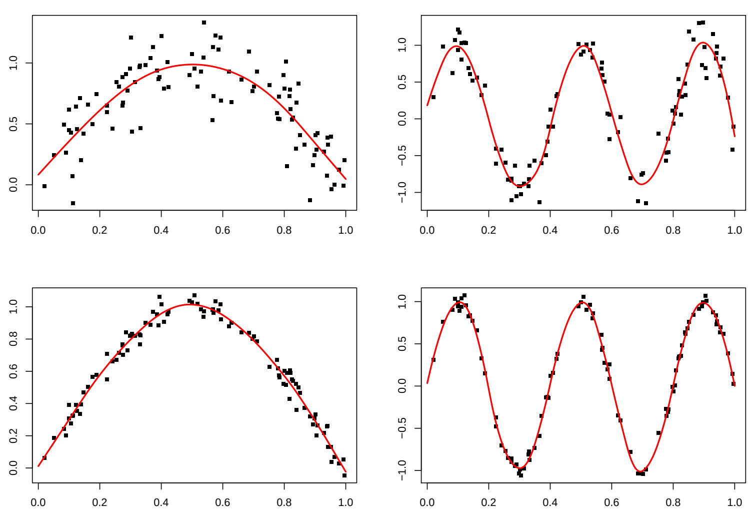
Mixed model

- See penalty as log of "mixing" distribution of $D\alpha$
- Mixed model software is good at estimating variances
- $D\alpha$ has singular distribution, rewrite the model
- Introduce "fixed" part X and "random" part Z
- $y = B\alpha = X\alpha + Za$, 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 R

```
# 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 ($\lambda = \sigma^2/\tau^2$)

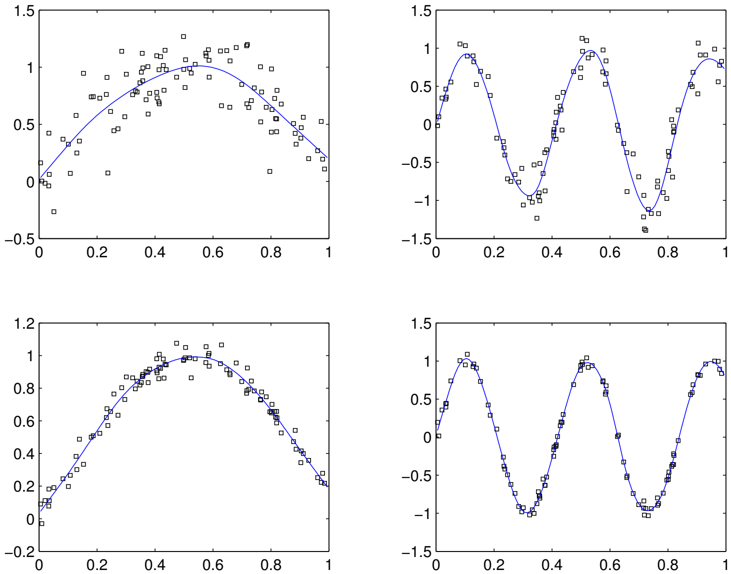
$$(B'B + \lambda D'D)\hat{\alpha} = B'y$$

- 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



Handling a penalty by data augmentation

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

- Solve linear system

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

- Equivalent: regression with augmented data:

$$B_+ = \begin{bmatrix} B \\ \sqrt{\lambda}D \end{bmatrix}; \quad y_+ = \begin{bmatrix} y \\ 0 \end{bmatrix};$$

P-splines with L_1 (P1-splines)

- L_1 norm: sum of absolute values
- L_1 regression on B-spline basis $B(x)$, with L_1 difference penalty

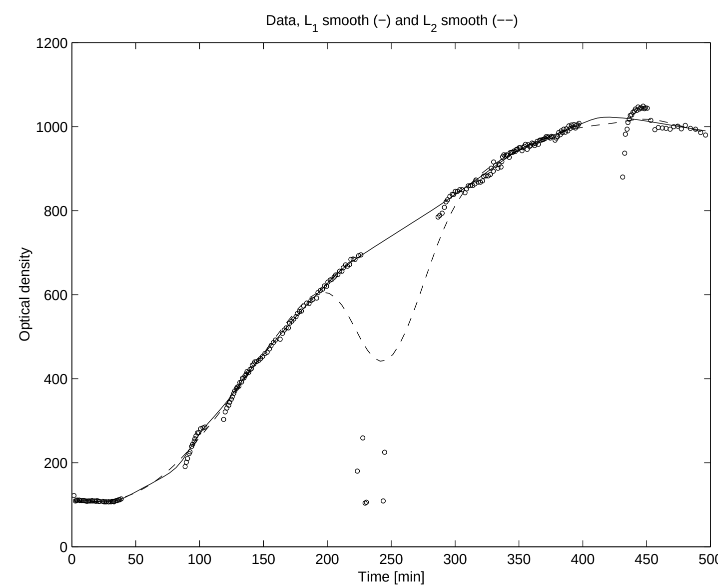
$$Q = |y - B\alpha| + \lambda|D\alpha|$$

- Equivalent data augmentation:

$$B_+ = \begin{bmatrix} B \\ \lambda D \end{bmatrix}; \quad y_+ = \begin{bmatrix} y \\ 0 \end{bmatrix};$$

- Solve with linear programming
- Use `l1fit()` or `rq()` (package `quantreg`) in R

P1-splines are robust



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
- We want to avoid two-dimensional smoothing
- 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

The old way: backfitting for GAM

- 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

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, direct solution
- Fast computation
- Equations of moderate size, compact result (α^*)
- Explicit computation of hat matrix:
- Easy to compute CV, ED, AIC
- Easy standard errors
- No iterations, no convergence criteria to set
- Implemented in Simon Wood's `mgcv` package

Features of P-spline GAMs

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

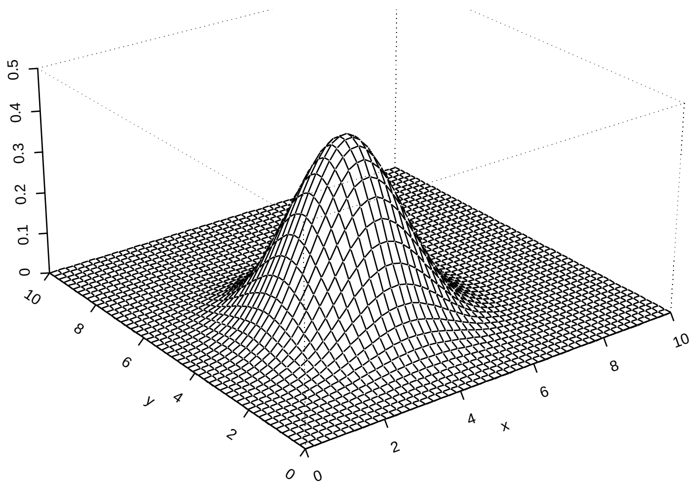
$$B_j(B'\hat{W}B + P)^{-1}B'\hat{W}B(B'\hat{W}B + P)^{-1}B'_j$$

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

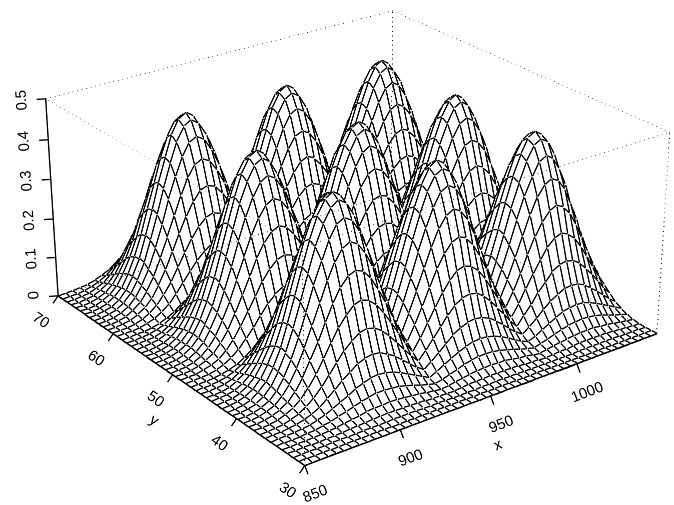
Two-dimensional smoothing with P-splines

- 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 $A = [\alpha_{jk}]$
- Difference penalties on coefficients
- Penalties on rows/columns of A

Surface building block



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



Implementation of the basis

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

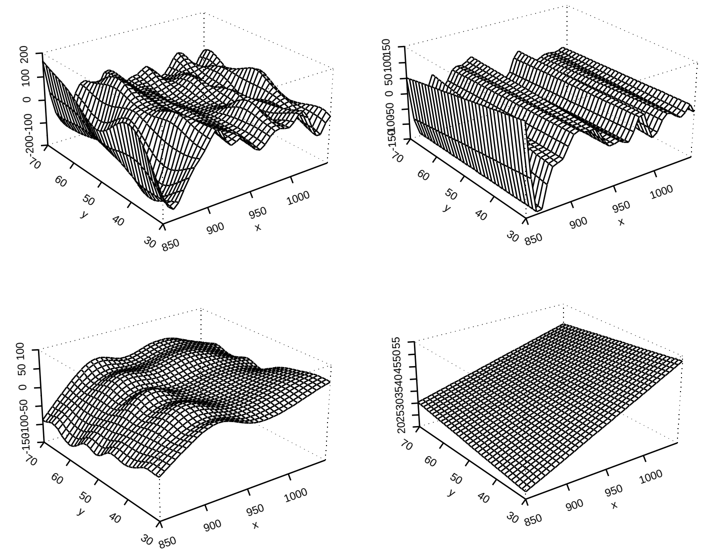
$$\begin{aligned} Q_P &= \text{RSS} + \text{Row Penalty} + \text{Column Penalty} \\ &= \text{RSS} + \lambda_1 \sum_{j=1}^{n} A_{j\bullet} D_d' D_d A_{j\bullet}' + \lambda_2 \sum_{k=1}^{\breve{n}} A_{\bullet k}' D_{\breve{d}}' D_{\breve{d}} A_{\bullet k} \\ &= |z - T\alpha|^2 + \lambda_1 |P_1 \alpha|^2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

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

Details of row and column penalties

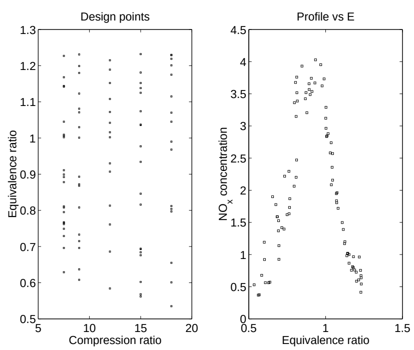
- Must also carefully arrange ("stack") penalties
- Block diagonal to break (e.g. row to row) linkages:
- $P_1 = D \otimes I_{\breve{n}}$
- $P_2 = I_n \otimes D$

Examples of tensor products surfaces

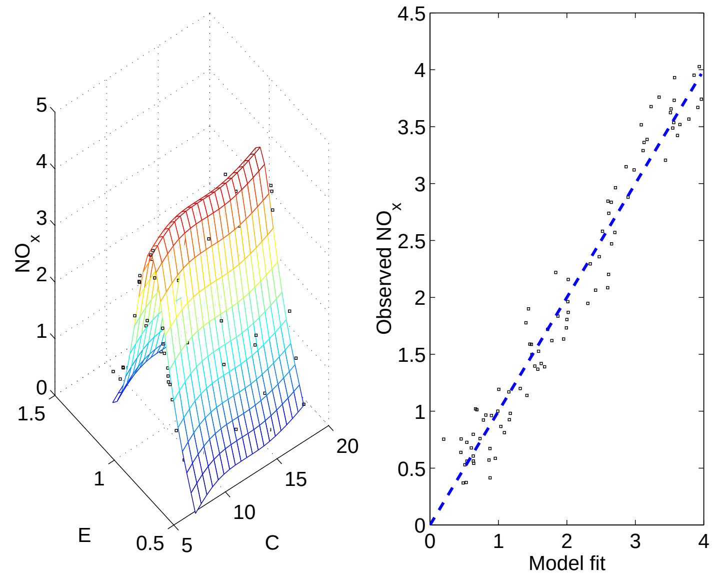


The ethanol data

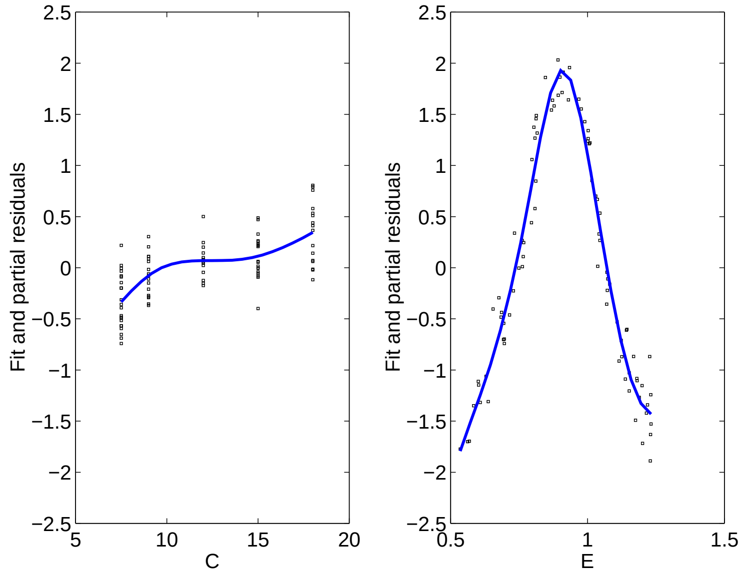
- Nitrogen oxides in motor exhaust: NO_x (z)
- Compression ratio, C (x), equivalence ratio, E (y)



PGAM fit for ethanol data



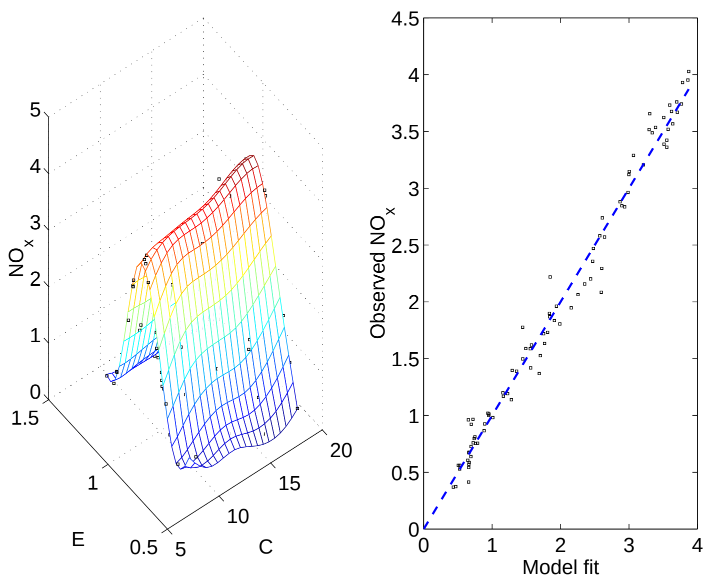
PGAM components for ethanol data



2D smoothing of ethanol data

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

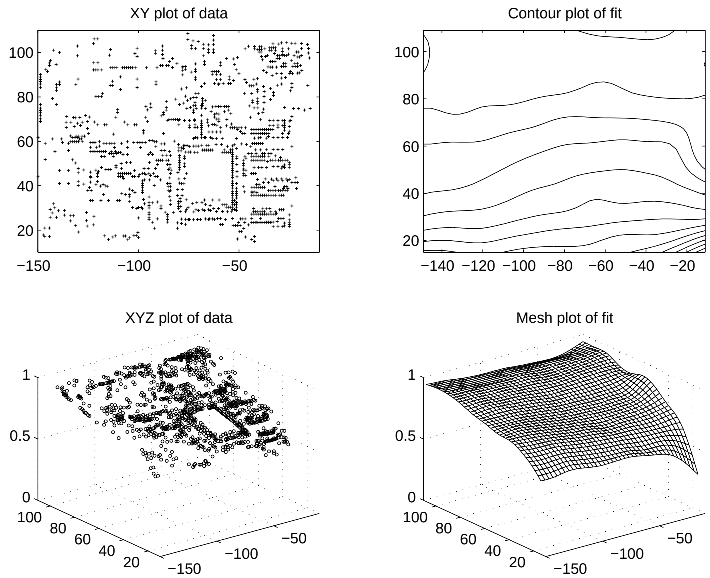
Tensor P-spline fit to ethanol data



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

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
- Special algorithm for (possibly incomplete) data on grids
- Speed-up 10 to 1000 times

Wrap-up

- P-splines are useful
- They are beautiful too
- People like them: many citations
- The penalties form the skeleton
- The B-splines put the flesh on it
- See back of handout for further reading

About software

- We (PE and BD) did not write a package
- Too busy exploring new applications ;-)
- Some scripts on `stat.lsu.edu/bmarx`
- Simon Wood's `mgcv` package offers a lot
- I'm always willing to help
- And to share my software
- `p.eilers@erasmusmc.nl`

Further reading

Bollaerts, K., Eilers, P.H.C. and van Mechelen, I. (2006) Simple and Multiple P-splines regression with Shape Constraints. *British Journal of Mathematical and Statistical Psychology 59*, 451–469.

Camarda, C.G. (2012) MortalitySmooth: An R Package for Smoothing Poisson Counts with P-Splines. *The Journal of Statistical Software 50*.

Currie, I.D. (2013) Smoothing constrained generalized linear models with an application to the Lee-Carter model *Statistical Modelling 13*, 69–93.

Currie, I., Durbán, M., and Eilers, P.H.C. (2003) 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. (2004) Smoothing and forecasting mortality rates. *Statistical Modelling 4*, 279–298.

Currie, I.D.; Durban, M. and Eilers, P.H.C. (2006) Generalized linear array models with applications to multidimensional smoothing. *J. R. Statist. Soc. B 68* 259–280.

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.

Delwarde, A.; Denuit, M. and Eilers, P. (2007) Smoothing the Lee-Carter and Poisson log-bilinear models for mortality forecasting: a penalized log-likelihood approach. *Statistical Modelling 7*, 29–48.

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

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

Eilers, P.H.C (2003) A perfect smoother. *Analytical chemistry 75*, 3631–3636.

Ill-posed problems with counts, the composite link model and penalized likelihood. *Statistical Modelling 7*, 239–254.

Eilers, P.H.C., Currie, I.D. and Durban, M. (2006) Fast and Compact Smoothing on Multi-dimensional Grids. *Computational Statistics and Data Analysis 50* 61–76.

Eilers, P.H.C.; Gampe, J.; Marx, B.D. and Rau, R. (2008) Modulation models for seasonal time series and incidence tables. *Statistics in Medicine 27*, 3430–3441.

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*, 89–121.

Eilers, P.H.C. and Marx, B.D. (2002) Generalized Linear Additive Smooth Structures. *Journal of Computational and Graphical Statistics 11*, 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. 2010 Splines, knots, and penalties, *Wiley Interdisciplinary Reviews: Computational Statistics 2*, 637–653.

Frasso, G. and Eilers, P.H.C. (2015) L- and V-curves for optimal smoothing. *Statistical Modelling 15*, 91–111.

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.

Kauermann, G.; Krivobokova, T. and Semmler W. (2011) Filtering Time Series with Penalized Splines. *Studies in Nonlinear Dynamics & Econometrics 15(2)*, Article 2.

Krivobokova, T. and Kauermann, G. (2007) A Note on Penalized Spline Smoothing With Correlated Errors. *Journal of the American Statistical Association 102*, 1328–1337.

Lambert, P. and Eilers, P.H.C. (2009) Bayesian density estimation from grouped continuous data. *Computational Statistics & Data Analysis 53*(4), 1388–1399.

Lang S. and Bresger D. (2004) Bayesian p-splines. *Journal of Computational and Graphical Statistics 13*, 183–212.

Lee, D.-J. and Durbán, M. (2011) P-spline ANOVA-type interaction models for spatio-temporal smoothing. *Statistical Modelling 11*, 49–69.

Lee, D.-J.; Durban M. and Eilers, P. (2013) Efficient two-dimensional smoothing with P-spline ANOVA mixed models and nested bases, *Computational Statistics & Data Analysis 61*, 22–37.

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 Eilers, P.H.C. (2002) Multivariate calibration stability: a comparison of methods. *Journal of Chemometrics 16*, 1–12.

Marx, B.D. and Eilers, P.H.C. (2004) Multidimensional signal regression. *Technometrics 47*, 13–22.

Eilers, P.H.C.; Li, B. and Marx, B.D. (2009) Multivariate calibration with single-index signal regression. *Chemometrics and Intelligent Laboratory Systems 96*, 196–202.

Marx, B. D.; Eilers, P.H.C. and Li, B. (2011) Multidimensional single-index signal regression. *Chemometrics and Intelligent Laboratory Systems 109*, 120–130.

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.

Perperoglou, A. and Eilers, P.H.C. (2010) Penalized regression with individual deviance effects. *Computational Statistics 25*, 341–361.

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

Rodrguez-Álvarez, M.X; Lee, D.-J.; Kneib, T.; Durbán, M and Eilers, P. (2014) Fast smoothing parameter separation in multidimensional generalized P-splines: the SAP algorithm. To appear in *Statistics and Computing*.

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

Schall, R. (1991) Estimation in generalized linear models with random effects. *Biometrika 78*, 719–727.

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.

Welham S.J., Cullis B.R., Kenward M.G. and Thompson R. (2007) A comparison of mixed model splines for curve fitting. *Australian and New Zealand Journal of Statistics 49*, 1–23.

Welham, S.J. and Thompson, R. (2009) A note on bimodality in the log-likelihood function for penalized spline mixed models. *Computational Statistics and Data Analysis 53*, 920–931.

Whittaker, E.T. (1923) On a new method of graduation. *Proceedings of the Edinburgh Mathematical Society 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.

Wood, S.N. (2006) *Generalized Additive Models. An Introduction with R.* Chapman and Hall.

Xiao, L.: Li, Y. and Ruppert, D. (2013) Fast Bivariate P-splines: the Sandwich Smoother, *JRSS-B 75*, 577–599.

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.