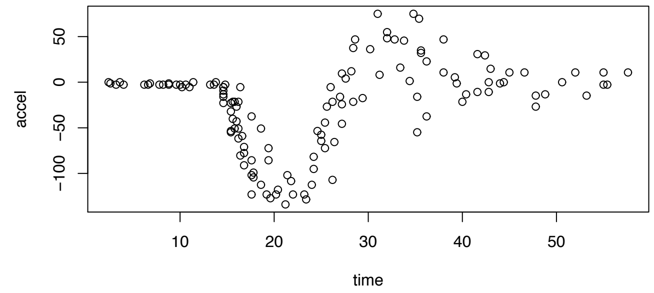
Basis Penalty Smoothers

Simon Wood
Mathematical Sciences, University of Bath, U.K.

Estimating functions

- Here are some ancient data...



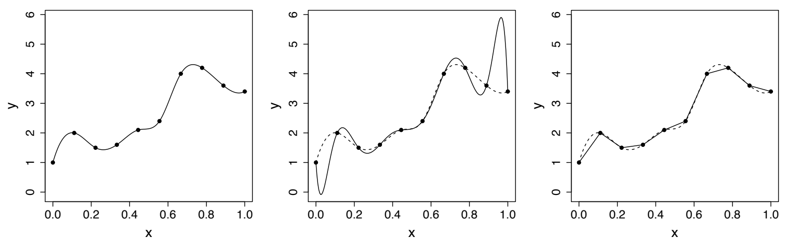
- If f is 'a smooth function', a suitable model might be

$$\texttt{accel}_i = f(\texttt{time}_i) + \epsilon_i.$$

- How to represent f? What function space should we search?
- A space that is good for approximating known functions would be a sensible starting point.

A space for f

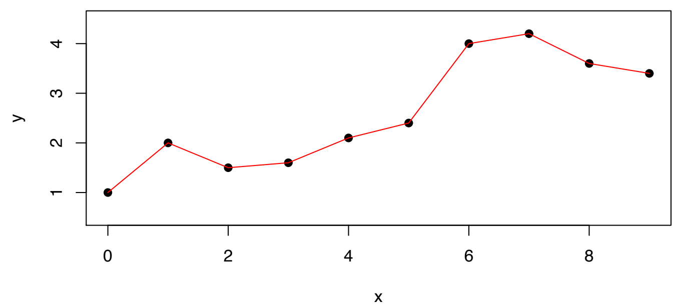
- Taylor's theorem might suggest using the space of polynomials, but look at the middle panel's attempt to approximate the function on the left with a polynomial.



- Trying to pass through the black dots and maintain continuity of all derivatives requires wild oscillation.
- Reducing the continuity requirements gives the better behaved piecewise linear interpolant on the right.

A simple basis for f

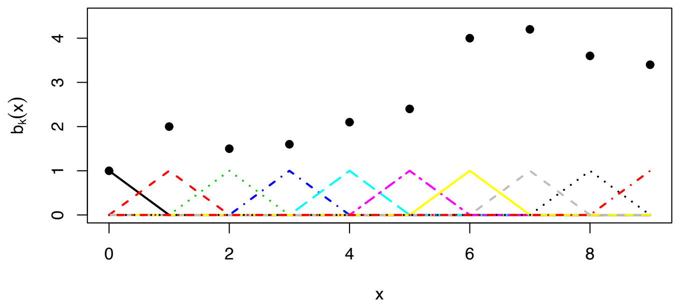
- So, for now, let's represent f as a piecewise linear function, with derivative discontinuities at x_k^*.



- ...this can be written $f(x) = \sum_k \beta_k b_k(x)$, where the b_k are *tent functions*: there is one per $\bullet$. The coefficients β_k give $f(x_k^*)$ directly.

The tent basis

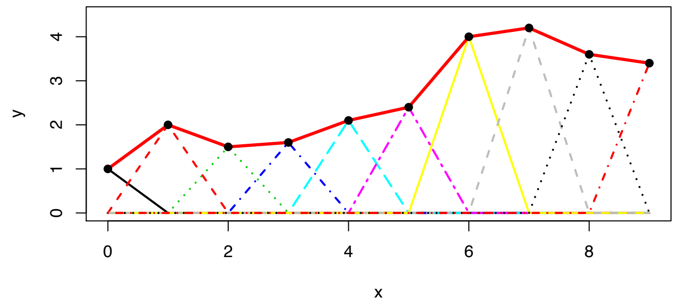
- The k^{th} tent function is 1 at x_k^* and descends linearly to zero at $x_{k\pm}^*$. Elsewhere it is zero.
- The full set look like this...



- Under this definition of $b_k(x)$, we would interpolate x_k^*, y_k^* data by just setting $\beta_k = y_k^*$.

How the tent basis works

- So the function is represented by multiplying each tent function by its coefficient, β_k, and summing the results...



- Given the basis functions and coefficients, we can *predict* the value of f anywhere in the range of the x^* values.

Prediction matrix

- f is defined by the x_k^* values defining the tent basis, and coefficients β_k.
- Now suppose that we want to evaluate the interpolant at a series of values x_i.
- If $\mathbf{f} = [f(x\ ), f(x\ ), \ldots]^T$, then

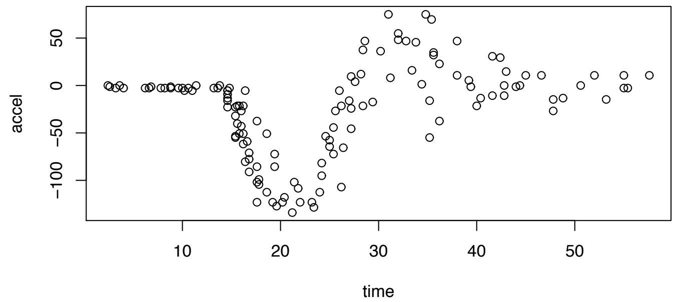
$$\mathbf{f} = \mathbf{X}\beta$$

where the *prediction matrix* is given by

$$\mathbf{X} \quad \begin{bmatrix} b\ \ x & b\ \ x & b\ \ x & . \\ b\ \ x & b\ \ x & . & . \\ . & . & . & . \\ . & . & . & . \end{bmatrix}$$

Regression with a basis

- Returning to these data...



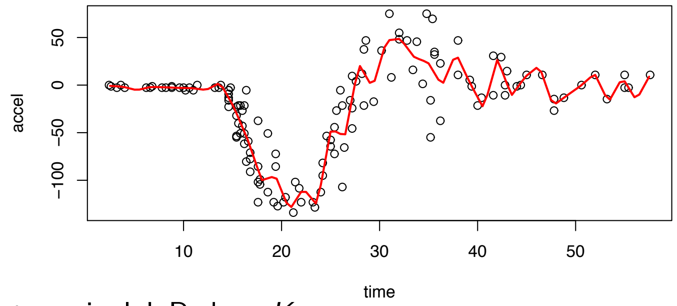
- We can define a tent basis by choosing some t_k^* values spread evenly through the range of observed times.
- Then the model, $a_i = f(t_i) + \epsilon_i$ becomes

$$\mathbf{a} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

...a straightforward linear model.

Estimation in R

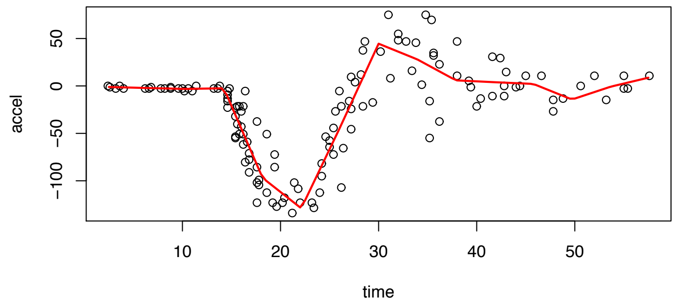
- A few lines of R code are enough to produce **X**. Then `lm` can be used to fit the model.
- Here is the result using K=40 evenly spaced t_k^* (knots).



- Far too wiggly! Reduce K

Reducing K

- After some experimentation, $K = 15$ seems reasonable. . .



- . . . but K selection is a bit fiddly and ad hoc.
  1. Models with different K are not nested, so we can't use hypothesis testing.
  2. We have little choice but to fit with every possible K value if AIC is to be used.
  3. Very difficult to generalize this model selection approach to models with more than one function.

Smoothing

- Using the basis for *regression* was ok, but there are some problems choosing K and deciding where to put the *knots*, x_k^*.
- To overcome these consider using the basis for *smoothing*.
  1. Make K 'large enough' that bias is negligible.
  2. Use even x_k^* spacing.
  3. To avoid overfit, penalize the wiggliness of f using, e.g.

$$\mathcal{P}(f) = \sum_{k}^{K-} (\beta_{k-} \;\; - 2\beta_k + \beta_k \;\;)^2$$

Evaluating the penalty

- To get the penalty in convenient form, note that

$$\begin{bmatrix} \beta \; - \; \beta \;\; \beta \\ \beta \; - \; \beta \;\; \beta \\ . \\ . \end{bmatrix} \begin{bmatrix} - & & , & . & . \\ & - & & .. & \\ . \; . & . & . & . & . \\ . \; . & . & . & . & . \end{bmatrix} \beta \quad \mathbf{D}\beta$$

by definition of **D**

- Hence

$$\mathcal{P}(f) = \beta \;\; \mathbf{D} \;\; \mathbf{D}\beta = \beta \;\; \mathbf{S}\beta$$

by definition of **S**.

Penalized fitting

- Now the penalized least squares estimates are

$$\hat{\beta} = \arg\min_{\beta} \sum_i \{a_i - f(t_i)\} \;\; + \lambda \mathcal{P}(f)$$

  smoothing parameter λ controls the fit-wiggliness tradeoff.
- For computational purposes this is re-written

$$\hat{\beta} = \arg\min_{\beta} \|\mathbf{a} - \mathbf{X}\beta\| \;\; + \lambda\beta \;\; \mathbf{S}\beta.$$

- Formally,

$$\hat{\beta} = (\mathbf{X} \;\; \mathbf{X} + \lambda\mathbf{S})^{-} \;\; \mathbf{X} \;\; \mathbf{a}$$

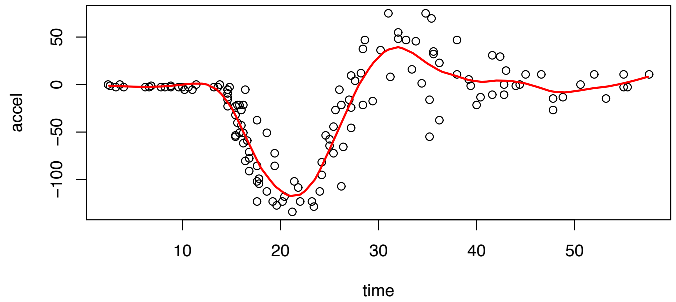
  but direct use of this expression has sub-optimal computational stability.

Computing the smooth fit

- In fact

$$\|\mathbf{a} - \mathbf{X}\beta\| \;\; + \lambda\beta \;\; \mathbf{S}\beta = \left\| \begin{bmatrix} \mathbf{a} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{X} \\ \sqrt{\lambda}\mathbf{D} \end{bmatrix} \beta \right\|$$

- The rhs is the RSS for an augmented linear model, which can be stably fit using `lm`. Here's an example using $K = 40$, but now penalizing...



Issues raised by smoothing

- Notice the dominant role of the penalty in the smoothed f — the discontinuity of the basis is barely visible, the penalty has so smoothed the results.
- But the dramatic effect of penalization raises questions
  1. How do we measure complexity of the model now that penalization has clearly yielded a result much smoother than K=40 would suggest?
  2. What distributional properties will $\hat{f}$ have under penalized estimation?
  3. How do we go about choosing/estimating the degree of penalization (λ)?

The natural basis

- To get started on these questions note that any basis-penalty smoother can be reparameterized so that its basis matrix is orthognal and its penalty is diagonal.
- Let a smoother have model matrix $\mathbf{X}$ and penalty matrix $\mathbf{S}$.
- Form QR decomposition $\mathbf{X} = \mathbf{QR}$, followed by symmetric eigen-decompostion

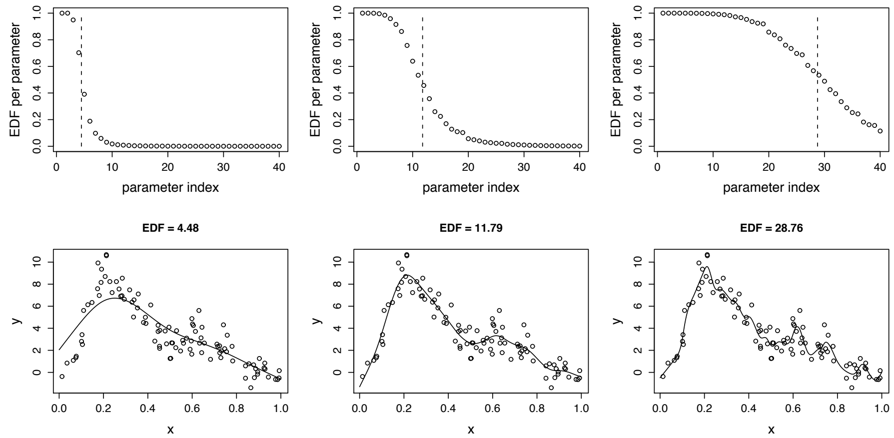
$$\mathbf{R}^{-} \;\; \mathbf{S}\mathbf{R}^{-} \;\; = \mathbf{U}\mathbf{\Lambda}\mathbf{U}$$

- Define $\mathbf{P} = \mathbf{U} \;\; \mathbf{R}$. And reparameterize $\beta' = \mathbf{P}\beta$.
- In the new parameterization the model matrix is $\mathbf{X}' = \mathbf{QU}$, which has orthogonal columns. ($\mathbf{X} = \mathbf{X}'\mathbf{P}$.)
- The penalty matrix is now the diagonal matrix $\mathbf{\Lambda}$ (eigenvalues in decreasing order down leading diagonal).

Effective Degrees of Freedom

- Penalization restricts the freedom of the coefficients to vary. So with 40 coefficients we have < 40 *effective degrees of freedom* (EDF).
- How the penalty restricts the coefficients is best seen in the natural parameterization. (Let $\mathbf{y}$ be the response.)
- Without penalization the coefficients would be $\tilde{\beta}' = \mathbf{X}'^T\mathbf{y}$.
- With penalization the coefficients are $\hat{\beta}' = (\mathbf{I} + \lambda\mathbf{\Lambda})^{-}\ \mathbf{X}'^T\mathbf{y}$.
- i.e. $\hat{\beta}_j = \tilde{\beta}_j(1 + \lambda\Lambda_{jj})^{-}$.
- So $(1 + \lambda\Lambda_{jj})^{-}$ is the *shrinkage factor* for the i^{th} coefficient, and is bounded between 0 and 1. It gives the EDF for $\hat{\beta}_j$.
- So total EDF is $\text{tr}\{(1 + \lambda\Lambda_{jj})^{-}\ \} = \text{tr}(\mathbf{F})$, where $\mathbf{F} = (\mathbf{X}\ \ \mathbf{X} + \lambda\mathbf{S})^{-}\ \mathbf{X}\ \ \mathbf{X}\}$, the 'EDF matrix'.

EDF Illustrated



Smoothing bias

- The formal expression for the penalized least squares estimates is $\hat{\beta} = (\mathbf{X}\ \ \mathbf{X} + \lambda\mathbf{S})^{-}\ \mathbf{X}\ \ \mathbf{y}$
- Hence

$$
\begin{aligned}
E(\hat{\beta}) &= (\mathbf{X}\ \ \mathbf{X} + \lambda\mathbf{S})^{-}\ \mathbf{X}\ \ E(\mathbf{y}) \\
&= (\mathbf{X}\ \ \mathbf{X} + \lambda\mathbf{S})^{-}\ \mathbf{X}\ \ \mathbf{X}\beta \\
&= \mathbf{F}\beta \neq \beta
\end{aligned}
$$

- Smooths are baised!
- i.e. we control model mis-specification bias by using a large K ...but to control the resulting variance we have to penalize ...which leads to smoothing bias.
- The bias makes frequentist inference difficult (including bootstrapping!).

A Bayesian smoothing model

- We penalize because we think that the truth is more likely to be smooth than wiggly.
- Things can be formalized by putting a prior on wiggliness

$$\text{wiggliness prior} \ \propto \exp(-\lambda\beta\ \ \mathbf{S}\beta/(2\sigma\ \ ))$$

- ...equivalent to a prior $\beta \sim N(\mathbf{0}, \mathbf{S}^{-}\sigma\ \ /\lambda)$ where $\mathbf{S}^{-}$ is a generalized inverse of $\mathbf{S}$.
- From the model $\mathbf{y}|\beta \sim N(\mathbf{X}\beta, \mathbf{I}\sigma\ \ )$, so from Bayes' Rule

$$\beta|\mathbf{y} \sim N(\hat{\beta}, (\mathbf{X}\ \ \mathbf{X} + \lambda\mathbf{S})^{-}\ \sigma\ \ )$$

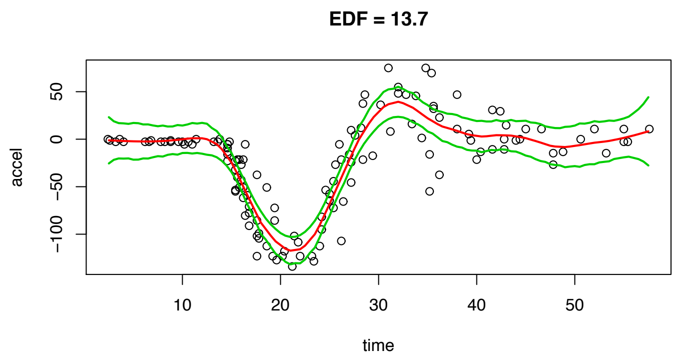
- Finally $\hat{\sigma}\ \ = \|\mathbf{y} - \mathbf{X}\hat{\beta}\|\ \ /\{n - \text{tr}(\mathbf{F})\}$ is useful.

Consequences of the Bayesian model

- The Bayesian model has the same structure as a linear mixed model, and can be computed as such.
- $\beta \sim N(\mathbf{0}, \mathbf{S}^{-}\sigma\ /\lambda) \Rightarrow \mathbf{f} \sim N(\mathbf{0}, (\mathbf{XSX}\ )^{-}\sigma\ /\lambda)$, i.e. f is equivalent to a Gaussian random field with covariance matrix $(\mathbf{XSX}\ )^{-}\sigma\ /\lambda$.
- But even if we compute f using mixed model technology, we are really being Bayesian in most cases...
- ...usually we do not expect f to be re-drawn from the prior on each replication of the response data, as a true random effect would be.

The Bayesian model in action

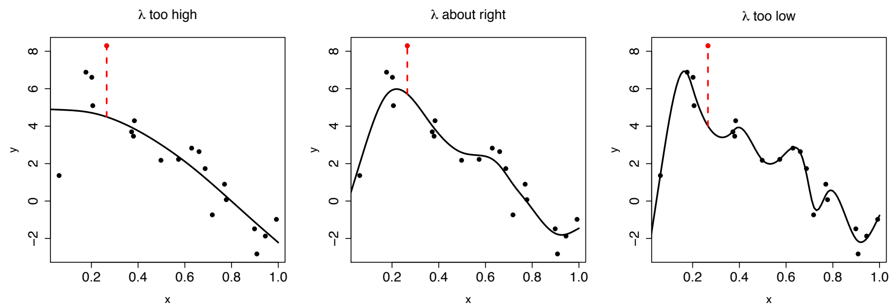
- An argument due to Nychka (1988) shows that the intervals for f based on the Bayesian posterior have good across the function frequentist coverage, because the Bayesian covariance matrix can be viewed as including a squared bias component.
- Here is an example of such an interval



Smoothness selection approaches

- The smoothing model $y_i = f(x_i) + \epsilon_i$, $\epsilon_i \sim N(0, \sigma\ )$, is represented via a basis expansion of f, with coefficients β.
- The β estimates are $\hat{\beta} = \arg\min_\beta \|\mathbf{y} - \mathbf{X}\beta\|\ \ + \lambda\beta\ \ \mathbf{S}\beta$ where $\mathbf{X}$ is the model matrix derived from the basis, and $\mathbf{S}$ is the wiggliness penalty matrix.
- λ controls smoothness — how should it be chosen?
- There are 3 main statistical approaches
  1. Choose λ to minimize error in predicting new data.
  2. Treat smooths as random effects, following the Bayesian smoothing model, and estimate λ as a variance parameter using a marginal likelihood approach.
  3. Go fully Bayesian by completing the Bayesian model with a prior on λ (requires simulation and not pursued here).

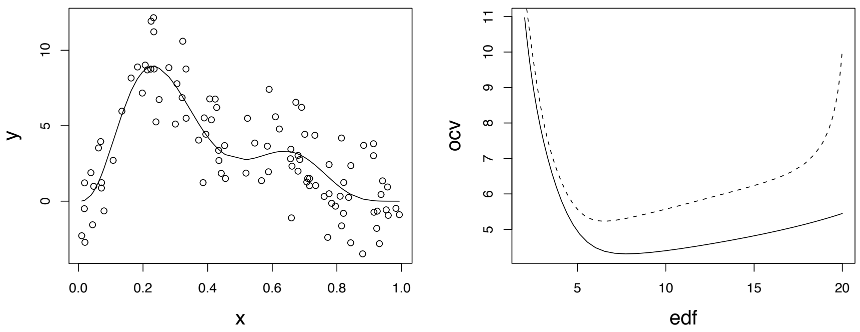
Prediction error: cross validation



1. Choose λ to try to minimize the error predicting new data.
2. Minimize the average error in predicting single datapoints *omitted* from the fit. Each datum left out once in average.
3. If $\mathbf{A} = \mathbf{X}(\mathbf{X}\ \ \mathbf{X} + \lambda\mathbf{S})^{-}\ \mathbf{X}\ \ $, it turns out that

$$\mathcal{V}_o(\lambda) = \frac{1}{n}\sum_i (y_i - \hat{\mu}_i^{-i}\ )^2 = \frac{1}{n}\sum_i \frac{(y_i - \hat{\mu}_i)^2}{(1 - A_{ii})^2}$$

OCV not invariant



- OCV is not invariant in an odd way. If $\mathbf{Q}$ is orthogonal then fitting objective

$$\|\mathbf{Qy} - \mathbf{QX}\boldsymbol{\beta}\| \ + \lambda\boldsymbol{\beta} \ \mathbf{S}\boldsymbol{\beta}$$

yields identical inferences about $\boldsymbol{\beta}$ as the original objective, but it gives a different $\mathcal{V}_o$.

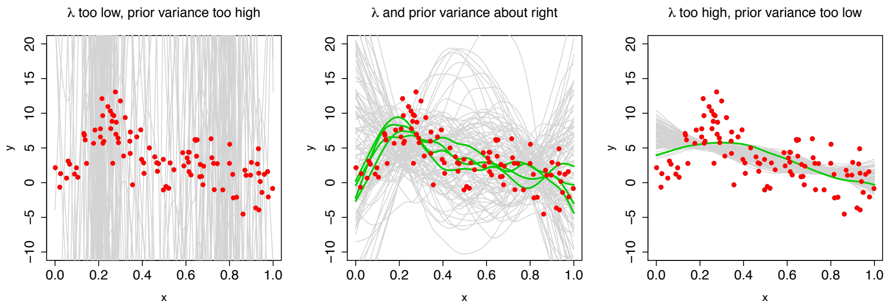
GCV: generalized cross validation

- If we find the $\mathbf{Q}$ that causes the leading diagonal elements of $\mathbf{A}$ to be constant, and then perform OCV, the result is the invariant alternative GCV:

$$\mathcal{V}_g = \frac{n\|\mathbf{y} - \hat{\boldsymbol{\mu}}\|}{\{n - \text{tr}(\mathbf{A})\}}$$

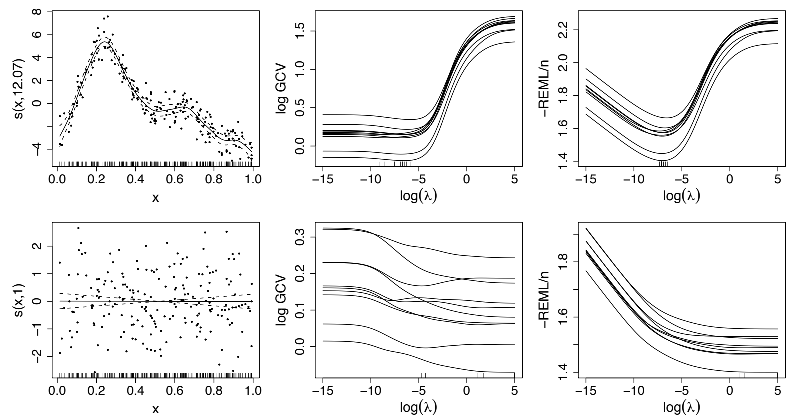
- It is easy to show that $\text{tr}(\mathbf{A}) = \text{tr}(\mathbf{F})$, where $\mathbf{F}$ is the degrees of freedom matrix.
- In addition to invariance, GCV is much easier to optimize efficiently in the multiple smoothing parameter case.

Marginal Likelihood smoothness selection



1. Choose λ to maximize the average likelihood of random draws from the prior implied by λ.
2. If λ too low, then almost all draws are too variable to have high likelihood. If λ too high, then draws all underfit and have low likelihood. The right λ maximizes the proportion of draws close enough to data to give high likelihood.
3. Formally, maximize e.g. $\mathcal{V}_r(\lambda) = \log \int f(\mathbf{y}|\boldsymbol{\beta}) f_\lambda(\boldsymbol{\beta}) d\boldsymbol{\beta}$. - Marginal Likelihood.

Prediction error vs. likelihood $\boldsymbol{\lambda}$ estimation



1. Pictures show GCV and REML scores for different replicates from same truth.
2. Compared to REML, GCV penalizes overfit only weakly, and so is more likely to occasionally undersmooth.

Summary

- We can construct smoothers from sets of basis functions, with associated quadratic penalties.
- Estimation is then by quadratically penalized least squares.
- Penalization reduces freedom to vary: we need a notion of effective degrees of freedom.
- A Bayesian view of smoothing is useful for further inference.
- The appropriate amount of penalization can be estimated by marginal likelihood or prediction error methods.