An Introduction to GAMs based on penalized regression splines

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

Generalized Additive Models (GAM)

- A GAM has a form something like:

$$g\{\mathbb{E}(y_i)\} = \eta_i = \mathbf{X}_i^* \boldsymbol{\beta}^* + f_1(x_{1i}) + f_2(x_{2i}, x_{3i}) + f_3(x_{4i}) + \cdots$$

  - g is a known *link function*.
  - y_i independent with some exponential family distribution. Crucially this $\Rightarrow \text{var}(y_i) = V(\mathbb{E}(y_i))\phi$, where V is a distribution dependent known function.
  - f_j are smooth unknown functions (subject to centering conditions).
  - $\mathbf{X}^* \boldsymbol{\beta}^*$ is parametric bit.
- i.e. a GAM is a GLM where the linear predictor depends on smooth functions of covariates.

GAM Representation and estimation

- Originally GAMs were estimated by backfitting, with any scatterplot smoother used to estimate the f_j.
- ...but it was difficult to estimate the degree of smoothness.
- Now the tendency is to represent the f_j using basis expansions of *moderate size*, and to apply tuneable quadratic penalties to the model likelihood, to avoid overfit.
- ...this makes it easier to estimate degree of smoothness, by estimating the tuning parameters/ smoothing parameters.

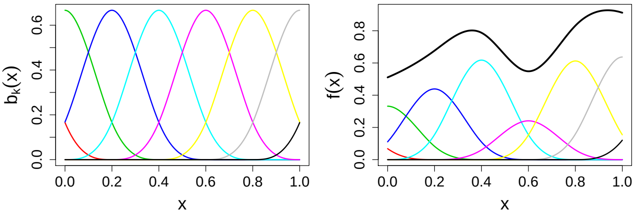
Example basis-penalty: P-splines

- Eilers and Marx have popularized the use of B-spline bases with discrete penalties.
  - If $b_k(x)$ is a B-spline and β_k an unknown coefficient, then

$$f(x) = \sum_{k}^{K} \beta_k b_k(x).$$

  - Wiggliness can be penalized by e.g.

$$\mathcal{P} = \sum_{k=2}^{K-1} (\beta_{j-1} - 2\beta_j + \beta_{j+1})^2 = \boldsymbol{\beta}^{\mathrm{T}} \mathbf{S} \boldsymbol{\beta}.$$



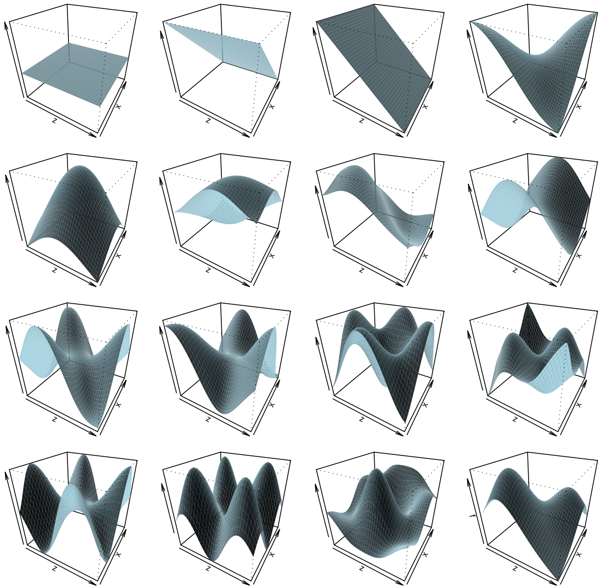
Other reduced rank splines

- Reduced rank versions of splines with derivative based penalties often have slightly better MSE performance.
- e.g. Choose a set of *knots*, x_k^* spread nicely in the range of the covariate, x, and obtain the cubic spline basis based on the x_k^*. i.e. the basis that arises by minimizing, e.g.

$$\sum_k \{y_k^* - f(x_k^*)\}^2 + \lambda \int f''(x)^2 dx \quad \text{w.r.t. } f.$$

- Choosing the knot locations for any penalized spline type smoother is rather arbitrary. It can be avoided by taking a reduced rank eigen approximation to a full spline, (actually get an optimal low rank basis this way).

Rank 15 Eigen Approx to 2D TPS



Other basis penalty smoothers

- Many other basis-penalty smoothers are possible.
- Tensor product smooths are basis-penalty smooths of several covariates, constructed (automatically) from smooths of single covariates.
- Tensor product smooths are immune to covariate rescaling, provided that they are multiply penalized.
- Finite area smoothing is also possible (look out for *Soap film smoothing*).

Estimation

- Whatever the basis, the GAM becomes $g\{\mathbb{E}(y_i)\} = \mathbf{X}_i\boldsymbol{\beta}$, a richly parameterized GLM.
- To avoid overfit, estimate $\boldsymbol{\beta}$ to minimize

$$D(\boldsymbol{\beta}) + \sum_j \lambda_j \boldsymbol{\beta}^{\mathrm{T}}\mathbf{S}_j\boldsymbol{\beta}$$

  — the penalized deviance. λ_j control fit-smoothness (variance-bias) tradeoff.
- Can get this objective 'directly' or by putting a prior on function wiggliness $\propto \exp(-\sum \lambda_j \boldsymbol{\beta}^{\mathrm{T}}\mathbf{S}_j\boldsymbol{\beta}/2)$.
- So GAM is also a GLMM and λ_j are variance parameters.
- Given λ_j actual $\boldsymbol{\beta}$ fitting is by a Penalized version of IRLS (Fisher scoring or full Newton), or by MCMC.

Smoothness selection

- Various criteria can be minimized for λ selection/estimation
- Cross validation leads to a GCV criterion

$$\mathcal{V}_g = D(\hat{\beta}) / \left\{n - \text{tr}(\mathbf{A})\right\}^2$$

- AIC or Mallows' *Cp* leads to $\mathcal{V}_a = D(\hat{\beta}) + 2\text{tr}(\mathbf{A})\phi$.
- Taking the Bayesian/mixed model approach seriously, a REML based criteria is

$$\mathcal{V}_r = D(\hat{\beta})/\phi + \hat{\beta}^{\mathrm{T}}\mathbf{S}\beta/\phi + \log|\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X} + \mathbf{S}| - \log|\mathbf{S}|_+ - 2l_s$$

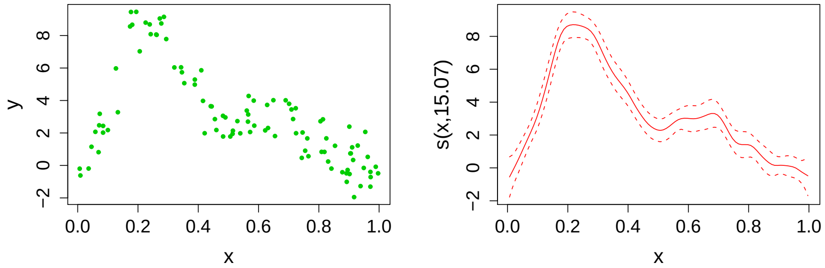
- ... $\mathbf{W}$ is the diagonal matrix of IRLS weights, $\mathbf{S} = \sum_j \lambda_j \mathbf{S}_j$, $\mathbf{A} = \mathbf{X}(\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X} + \mathbf{S})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{W}$, the trace of which is the model EDF and l_s is the saturated log likelihood.

Numerical Methods for optimizing $\boldsymbol{\lambda}$

- All criteria can reliably be optimized by Newton's method (outer to PIRLS $\boldsymbol{\beta}$ estimation).
- Need derivatives of $\mathcal{V}$ wrt $\log(\lambda_j)$for this...
  1. Get derivatives of $\hat{\boldsymbol{\beta}}$ w.r.t. $\log(\lambda_j)$ by differentiating PIRLS or by Implicit Function Theorem approach.
  2. Given these we can get the derivatives of **W** and hence tr(**A**) w.r.t. the $\log(\lambda_j)$ as well as the derivatives of D.
- Derivatives of GCV, AIC and REML have *very* similar ingredients.
- Some care is needed to ensure maximum efficiency and *stability*.
- MCMC and boosting offer alternatives for 'estimating' $\boldsymbol{\lambda}, \boldsymbol{\beta}$.

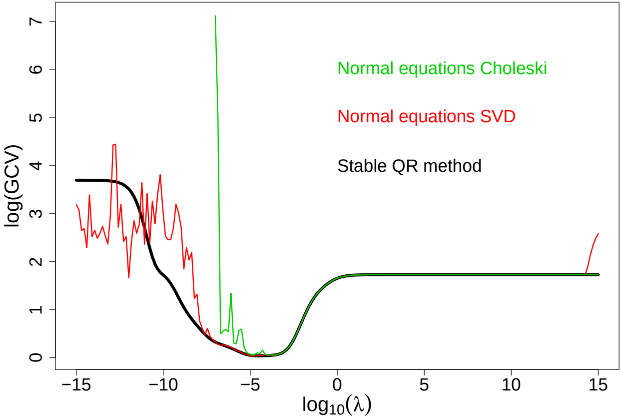
Why worry about stability? A simple example

The x,y, data on the left were modelled using the cubic spline on the right (full TPS basis, λ chosen by GCV).



The next slide compares GCV calculation based on the naïve 'normal equations' calculation $\hat{\beta} = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{S})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}$ with a stable QR based alternative…

Stability matters for λ selection!



…automatic minimization of the red or green versions of GCV is not a good idea.

GAM inference

- The best calibrated inference, in a frequentist sense, seems to arise by taking a Bayesian approach.
- Recall the prior on function wiggliness

$$\propto \exp\left(-\frac{1}{2}\sum \lambda_j \boldsymbol{\beta}^{\mathrm{T}}\mathbf{S}_j\boldsymbol{\beta}\right)$$

  — an improper Gaussian on $\boldsymbol{\beta}$.
- Bayes' rule and some asymptotics then $\Rightarrow$

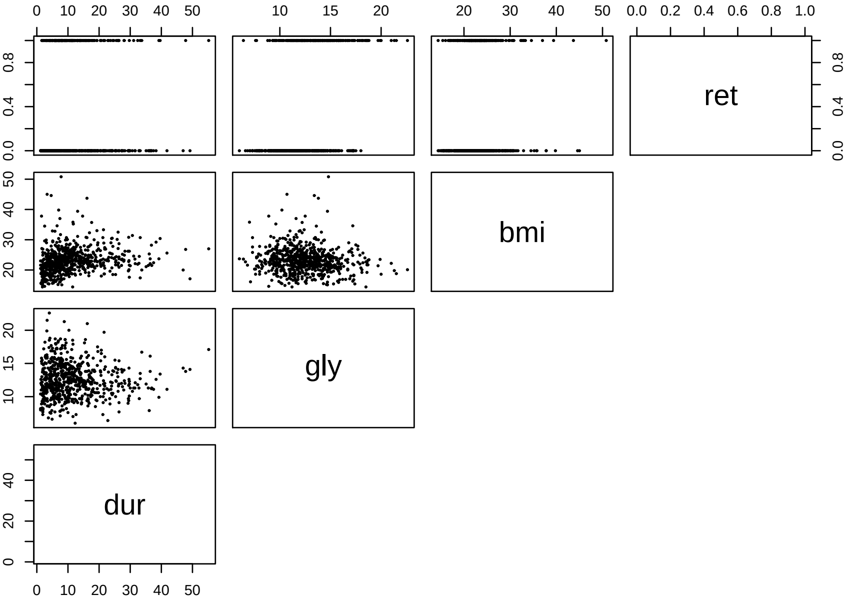
$$\boldsymbol{\beta}|\mathbf{y} \sim N(\hat{\boldsymbol{\beta}}, (\mathbf{X}^{\mathrm{T}}\mathbf{W}\mathbf{X} + \sum \lambda_j \mathbf{S}_j)^{-1}\phi)$$

- Posterior $\Rightarrow$ e.g. CIs for f_j, but can also simulate from posterior very cheaply, to make inferences about anything the GAM predicts.

GAM inference II

- The Bayesian CIs have good across the function frequentist coverage probabilities, provided the smoothing bias is somewhat less than the variance.
- Neglect of smoothing parameter uncertainty is not very important here.
- An extension of Nychka (1988; JASA) is the key to understanding these results.
- P-values for testing model components for equality to zero are also possible, by 'inverting' Bayesian CI for the component. P-value properties are less good than CIs.

Example: retinopathy data



Retinopathy models?

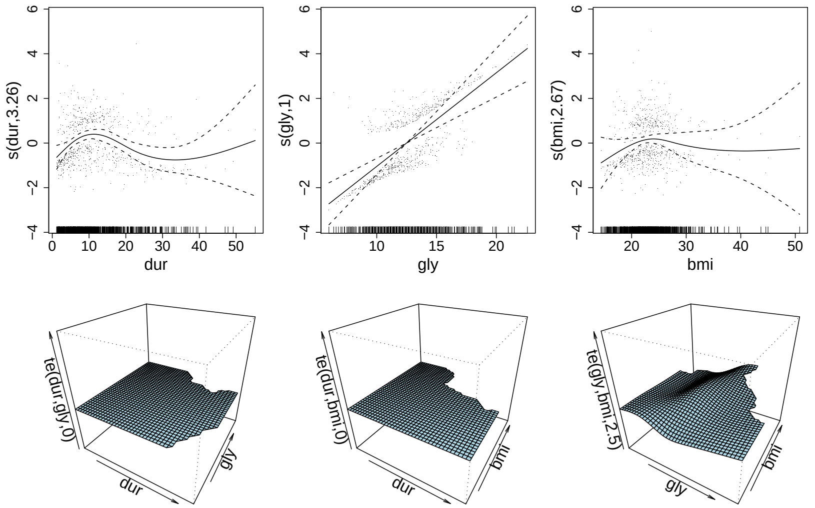
- Question: How is development of `ret`inopathy in diabetics related to `dur`ation of disease at baseline, body mass index (`bmi`) and percentage `gly`cosylated haemoglobin?
- A possible model is

$$\begin{aligned}\text{logit}\{\mathbb{E}(\texttt{ret})\} = & f_1(\texttt{dur}) + f_2(\texttt{bmi}) + f_3(\texttt{gly}) \\ & + f_1(\texttt{dur}, \texttt{bmi}) + f_2(\texttt{dur}, \texttt{gly}) + f_3(\texttt{gly}, \texttt{bmi})\end{aligned}$$

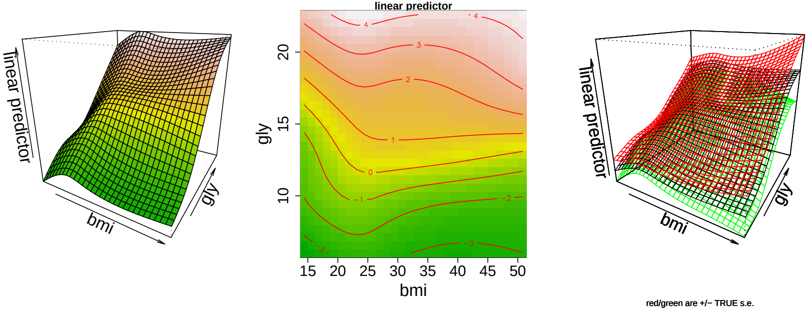
  where `ret` $\sim$ Bernoulli.
- In R, model is fit with something like

```
gam(ret ~ te(dur)+te(gly)+te(bmi)+
te(dur,gly)+te(dur,bmi)+te(gly,bmi),
family=binomial)
```

Retinopathy Estimated effects



Retinopathy GLY-BMI interaction

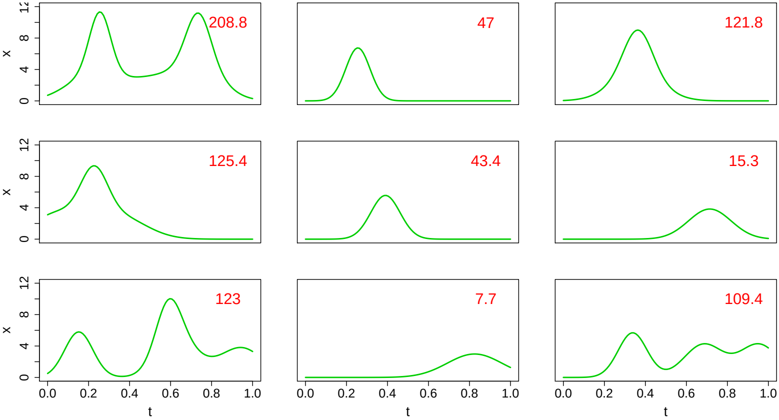


GAM 'extensions'

- To obtain a satisfactory framework for generalized additive modelling has required solving a rather more general estimation problem ...
- GAM framework can cope with *any* quadratically penalized GLM where smoothing parameters enter the objective linearly. Consequently the following examples extensions can all be used without new theory...
  - Varying coefficient models, where a coefficient in a GLM is allowed to vary smoothly with another covariate.
  - Model terms involving any linear functional of a smooth function, for example functional GLMs.
  - Simple random effects, since a random effect can be treated as a smooth.
  - Adaptive smooths can be constructed by using multiple penalties for a smooth.

Example: functional covariates

- Consider data on 150 functions, $x_i(t)$, (each observed at $\mathbf{t}^{\mathrm{T}} = (t_1, \ldots, t_{200})$), with corresponding noisy univariate response, y_i.
- First 9 $(x_i(t), y_i)$ pairs are ...



F-GLM

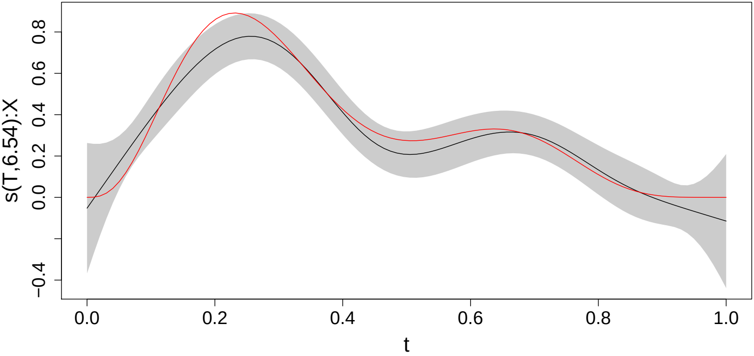
- An appropriate model might be the *functional GLM*

$$g\{\mathbb{E}(y_i)\} = \int f(t)x_i(t)dt$$

  where predictor x_i is a *known function* and $f(t)$ is an unknown smooth *regression coefficient function*.
- Typically f and x_i are discretized so that $g\{\mathbb{E}(y_i)\} = \mathbf{f}^{\mathrm{T}}\mathbf{x}_i$ where $\mathbf{f}^{\mathrm{T}} = [f(t_1), f(t_2)\ldots]$ and $\mathbf{x}_i^{\mathrm{T}} = [x_i(t_1), x_i(t_2)\ldots]$.
- Generically this is an example of dependence on a linear functional of a smooth.
- R package `mgcv` has a simple 'summation convention' mechanism to handle such terms…

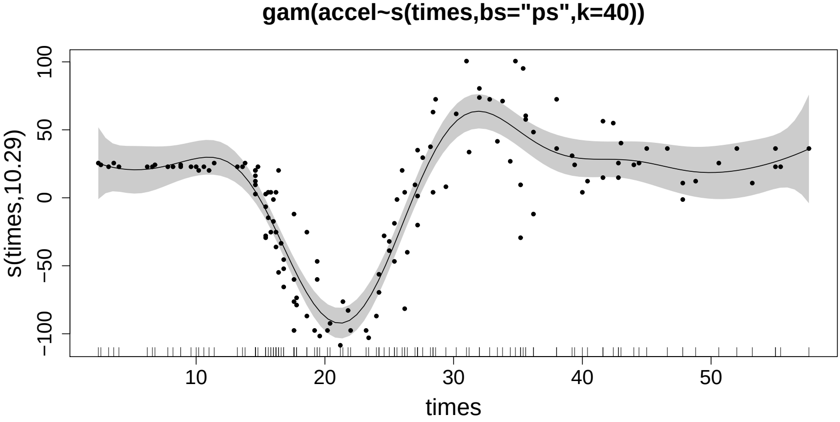
FGLM fitting

- Want to estimate smooth, f, in model $y_i = \int f(t)x_i(t)dt + \epsilon_i$.
- `gam(y~s(T,by=X))` will do this, if `T` and `X` are matrices.
- i^{th} row of `X` is the observed (discretized) function $x_i(t)$. Each row of `T` is a replicate of the observation time vector **t**.



Adaptive smoothing

- Perhaps I don't like this P-spline smooth of 'the' motorcycle crash data...



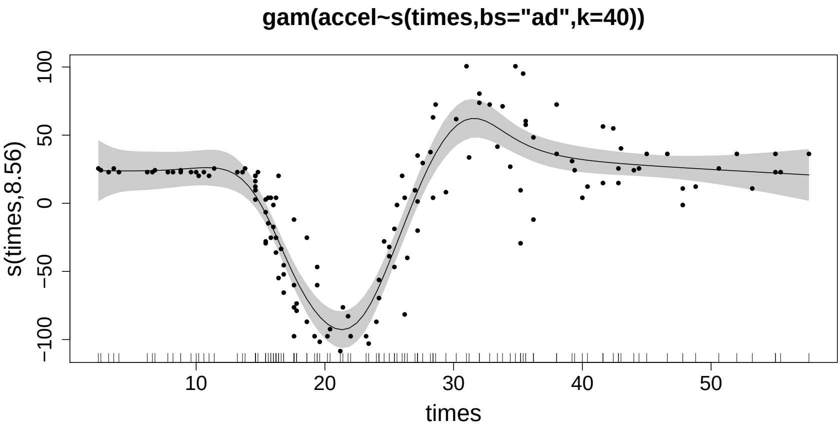
- Should I really use adaptive smoothing?

Adaptive smoothing 2

- P-splines and the preceding GAM framework make it *very* easy to do adaptive smoothing.
- Use a B-spline basis $f(x) = \sum \beta_j b_j(x)$, with an adaptive penalty $\mathcal{P} = \sum_{k=2}^{K-1} c_k(\beta_{k-1} - 2\beta_k + \beta_{k+1})^2$, where c_k varies smoothly with k and hence x.
- Defining $d_k = \beta_{k-1} - 2\beta_k + \beta_{k+1}$, and $\mathbf{D}$ to be the matrix such that $\mathbf{d} = \mathbf{D}\boldsymbol{\beta}$, we have $\mathcal{P} = \boldsymbol{\beta}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathrm{diag}(\mathbf{c})\mathbf{D}\boldsymbol{\beta}$.
- Now use a (B-spline) basis expansion for $\mathbf{c}$ so that $\mathbf{c} = \mathbf{C}\boldsymbol{\lambda}$.
- Then $\mathcal{P} = \sum_j \lambda_j \boldsymbol{\beta}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathrm{diag}(\mathbf{C}_{\cdot j})\mathbf{D}\boldsymbol{\beta}$.
- i.e. the adaptive P-spline is just a P-spline with multiple penalties.

Adaptive smoothing 3

- R package `mgcv` has an adaptive P-spline class. Using it does give some improvement ...



Conclusions

- Penalized regression splines are the starting point for a fairly complete framework for Generalized Additive Modelling.
- The numerical methods and theory developed for this framework are applicable to any quadratically penalized GLM, so many extensions of 'standard' GAMs are possible.
- The R package `mgcv` tries to exploit the generality of the framework, so that almost any quadratically penalized GLM can readily be used.

References

- Hastie and Tibshirani (1986) invented GAMs. The work of Wahba (e.g. 1990) and Gu (e.g. 2002) heavily influenced the work presented here. Duchon (1977) invented thin plate splines. The Retinopathy data are from Gu.
- Penalized regression splines go back to Wahba (1980), but were given real impetus by Eilers and Marx (1996) and in a GAM context by Marx and Eilers (1998).
- See Wood (2006) *Generalized Additive Models: An Introduction with R*, CRC for more information. Wood (2008; JRSSB) is more up to date on numerical methods.
- The `mgcv` package in R implements everything covered here.