Smooth modelling with regular likelihoods

Simon Wood
School of Mathematics, University of Bristol, U.K.

In collaboration with Natalya Pya and Benjamin Säfken.

Funded by EPSRC
Generalized additive models relate a response, $y$, to predictors, $x_j$ (vector?) with a structure something like…

\[
\eta = f_1(x_1) + f_2(x_2) + \ldots
\]

\[
y_i \sim EF(g^{-1}(\eta_i), \phi)
\]  

(1)

the $f_j$ are unknown smooth functions, and $g$ a known link.

More generally if $A\theta$ is a parametric linear predictor and $L_j$ a linear operator

\[
\eta = A\theta + \sum_j L_j f_j(x_j)
\]

\[
y_i \sim EF(g^{-1}(\eta_i), \phi)
\]  

(2)
Example: Retinopathy data

How is probability of retinopathy related to predictors?
Example: Retinopathy model

A possible model is

$$\text{logit}\{\mathbb{E}(\text{ret})\} = f_1(\text{dur}) + f_2(\text{bmi}) + f_3(\text{gly}) + f_4(\text{dur}, \text{bmi}) + f_5(\text{dur}, \text{gly}) + f_6(\text{gly}, \text{bmi})$$

where $\text{ret} \sim \text{Bernoulli}$.

In R (mgcv), model is fit with something like

```r
gam(ret ~ s(dur) + s(gly) + s(bmi) +
    ti(dur,gly) + ti(dur,bmi) + ti(gly,bmi),
    family=binomial)
```
Retinopathy Estimated effects

---

```
Retinopathy Estimated effects

![Graphs showing estimated effects of duration (dur), glycemia (gly), and BMI (bmi) on retinopathy.]
```

---

**Summary:**

The graphs depict the estimated effects of duration (dur), glycemia (gly), and BMI (bmi) on retinopathy. The plots illustrate the relationship between these variables and the outcome, with curves and scatter plots indicating the distribution and trends. The 3D visualizations further highlight the interaction effects within these variables, providing a comprehensive view of the estimated impacts on retinopathy.
Model computational representation

- Represent each smooth with a linear (spline) basis expansion

\[ f_j(x_j) = \sum_i^{p} \beta_{ij} b_{ij}(x_j), \]

- \( b_{ij}(x) \) are known basis functions, \( \beta_{ij} \) unknown coefficients.
- \( p \ll n \), but more than enough to avoid oversmoothing.
- So GAM becomes an overparameterized GLM.
- To avoid overfit, associate a wiggliness penalty \( \beta^T S_j \beta \) with each \( f_j \). e.g. \( \beta^T S_j \beta = \int f_j''(x)^2 dx \).
Example smoothers

- The basis-penalty approach allows a rich variety of smooth model components. e.g. simple Gaussian random effects, as well as ...
Model estimation

- Empirical Bayes.
  - Likelihood given by exponential family \( y | \beta \sim f(y | \beta) \).
  - Prior given by penalties \( \beta \sim N(0, (\sum_j \lambda_j S_j)^{-\phi}) \).
- Estimates/ posterior modes given smoothing parameters \( \lambda_j \)

\[
\hat{\beta} = \arg \max_{\beta} \log f(y | \beta) + \log f(\beta) = \arg \max_{\beta} I(\beta) - \frac{1}{2\phi} \sum_j \lambda_j \beta^T S_j \beta
\]

- Use Laplace approximate marginal likelihood to estimate \( \lambda \)

\[
\hat{\lambda} \approx \arg \max_{\lambda} \log \int f(y | \beta) f(\beta) d\beta
\]

(or use prediction error criterion, GCV, AIC etc)
How marginal likelihood works

- Draw $\beta$ from prior implied by $\lambda$. Find average value of likelihood for these draws.
- Choose $\lambda$ to maximize this average likelihood.
- Formally, maximize $\int f(y|\beta)f(\beta)d\beta$ w.r.t. $\lambda$. 
Asymptotic considerations

- Is the reduced rank spline approach ok? Consider cubic \textit{regression} spline with $k$ equally spaced ($h$) knots.
- Average variance is $n^{-1} \text{tr}(X(X^TX)^{-1}X^T)\sigma^2 = \sigma^2 k/n$.
- Squared bias is $O(h^8) = O(k^{-8})$, from approximation error of spline (e.g. deBoor, 1978).
- Need squared bias to have same order as variance for neither to dominate as $n \to \infty$.
- i.e. $k \propto n^{1/9}$, so that MSE is $O(n^{-8/9})$.
- Consistency is maintained under penalization with (Laplace Approximate) ML smoothness selection, but need $k = O(n^{1/5})$ (or faster) to penalize in $n \to \infty$ limit. Then MSE seems to be $O(n^{-4/5})$. 
Why the GAM framework is useful

- Flexibility in specification of linear predictor: many types of smooth plus Gaussian random effects can be combined.
- Numerically robust and general methods can for fitting with smoothness estimation are possible (e.g. Wood, 2000, 2011).
- Theoretically well founded (e.g. Kauermann et al. 2009)
- The Bayesian view of smoothing gives well calibrated interval estimates (Nychka, 1988, Marra and Wood, 2012).
- Model selection via extra model selection penalties (e.g. Marra and Wood, 2011), or decent p-value approximations is available (Wood, 2013a,b), but AIC is more problematic (Greven and Kneib, 2010).
Extending the model

- Many extensions have been proposed...
  1. GAM type linear predictor with non-exponential family distribution for independent response variable. Examples: scaled t for heavy tails, beta regression for proportions, ordered categorical models, Tweedie and negative binomial distributions
  2. GAM type linear predictor without a separable log-likelihood. E.g. Cox Proportional hazards and Cox Process.
  3. Models with multiple GAM type linear predictors: E.g. GAMLSS (Rigby & Stasinopoulus, 2005), or multivariate additive models (e.g. Yee and Wild, 1996).

- ...no single framework available: generally can’t do everything you could do with an exponential family GAM.
Aims

- To produce a general framework for the various GAM extensions that is as useful as the GAM framework. i.e.
  1. Allow numerically reliable smoothing parameter estimation for a wide range of smooths, including those with multiple smoothing penalties.
  2. Allow access to the same inferential machinery that is available for GAMs.

- To extend the inferential machinery to include a better performing AIC for smooth models.

- General framework should reduce implementation of new model classes to the implementation of some standard derivatives of the log likelihood.
Consider models, for data $\mathbf{y}$, in which the log likelihood, $l$, depends on smooth functions $f_j$.

Represent the $f_j$ using intermediate rank basis expansions with quadratic penalties, so that coefficient estimates, given smoothing parameters, $\lambda$, are

\[ \hat{\beta} = \arg \max_{\beta} L(\beta) = l(\beta) - \frac{1}{2} \sum_j \lambda_j \beta^T S^j \beta. \]

Writing $S_\lambda = \sum_j \lambda_j S^j$, we can motivate the above by a prior $\beta \sim N(0, S^{-}_\lambda)$.

Can then estimate $\lambda$ by Marginal Likelihood.
Estimation methods: Laplace approximate ML

- We want to estimate log smoothing parameters as

\[ \hat{\rho} \simeq \arg \max_{\rho} \log \int f(y|\beta)f(\beta)d\beta \]

- Can’t generally do the integral.
- Approximate intergrand by exponential of 2nd order Taylor approx. of log intergrand (almost a multivariate normal).
- We get the Laplace Approximate ML

\[ \mathcal{V}(\lambda) = \mathcal{L}(\hat{\beta}) + \frac{1}{2} \log |S^\lambda|_+ - \frac{1}{2} \log |\mathcal{H}| + \frac{M_p}{2} \log(2\pi). \]

where \(-\mathcal{H}\) is Hessian of penalized log likelihood, \(\mathcal{L}\), at \(\hat{\beta}\).
Estimation methods: numerical strategy

- Nested Newton optimization...
- Optimize LAML $\mathcal{V}$ w.r.t. $\mathbf{\rho}$ by Newton’s method.
- Each trial set of $\mathbf{\rho}$ values require corresponding $\hat{\mathbf{\beta}}$, also by Newton’s method. Efficient as starting values excellent after first trial!
- Use implicit differentiation to obtain $d\hat{\mathbf{\beta}}/d\mathbf{\rho}$ etc, and hence derivatives of LAML w.r.t. $\mathbf{\rho}$.
- Problem: naïve computation is unstable. Worst parts are the log determinants, e.g. $\log|\sum_j \lambda_j \mathbf{S}^j|_+$ ...
- Some $\lambda_j \to \infty$ is legitimate, which can lead to taking logs of ‘numerical zeroes’ in the log determinant terms.
Simplified notation

- Some expressions required to actually implement general methods become incomprehensibly complex without a reasonable notation . . .
  1. Let differentiation with respect to (Greek) parameters be denoted $f^j_\beta = \partial f / \partial \beta_j$, $g^{jk}_{\alpha \beta} = \partial^2 g / \partial \alpha_j \partial \beta_k$ etc.
  2. Superscripts without a subscript are labels.
  3. Repeated Roman indices occurring on only one side of an equation should be summed over. e.g. $a_j = b^i_\beta w_{ij} = \sum_i b^i_\beta w_{ij}$

- e.g. $\mathcal{L}^{ij}_{\beta \beta}$ is the Hessian of the penalized log-likelihood, $p_i = \mathcal{L}^{ij}_{\beta \beta} \mathcal{L}_\beta^j$ is the product of the Hessian and the gradient.
General case Newton step computation for $\rho = \log \lambda$

1. Reparameterize \textit{each smooth} so that $\log |S^\lambda|_+$ is stable.
2. Find $\hat{\beta}$ by stabilized Newton method.
3. Drop unidentifiable $\hat{\beta}$ elements at convergence. If necessary repeat Newton optimization to allow other coefficients to adjust. Let $L_{ij}^{\hat{\beta}\hat{\beta}}$ be the inverse Hessian of $L$ at $\hat{\beta}$.
4. Compute $d\hat{\beta}_i/d\rho_k = L_{ij}^{\hat{\beta}\hat{\beta}} \lambda_k S_{ji}^k \hat{\beta}_l$.
5. Given $d\hat{\beta}/d\rho_k$ compute $l_{ijkl}^{\hat{\beta}\hat{\beta}}$.
6. Given $l_{ijkl}^{\hat{\beta}\hat{\beta}}$, compute $d^2\hat{\beta}/d\rho_k d\rho_l$.
7. Compute $L_{kj}^{\hat{\beta}\hat{\beta}} l_{ij}^{kp\rho} (\text{model specific, } O(Mnp^2))$.
8. The derivatives of $\mathcal{V}$ can now be computed.
Implicit differentiation example

- $\mathcal{L}_\beta^i = l^i_\beta - \lambda_k S^k_{ij} \hat{\beta}_j = 0$

- Differentiating w.r.t. $\rho_k = \log \lambda_k$ yields

\[
\mathcal{L}_\beta^i = l^i_\beta \frac{d\hat{\beta}_j}{d\rho_k} - \lambda_k S^k_{ij} \hat{\beta}_j - \lambda_l S^l_{ij} \frac{d\hat{\beta}_j}{d\rho_k} = 0.
\]

- Re-arranging and defining $\mathcal{L}_{\beta j}^{\hat{\beta}}$ as inverse of $\mathcal{L}_{\beta j}^{i \hat{\beta}}$,

\[
\frac{d\hat{\beta}_i}{d\rho_k} = \mathcal{L}_{\beta j}^{\hat{\beta}} \lambda_k S_{ji}^k \hat{\beta}_l,
\]

- Now can compute $l^i_\beta l^j_\beta \rho = \frac{l^i_\beta l^k_\beta}{\lambda_k S_{ji}^k} d\hat{\beta}_k / d\rho_l$ etc.

- In general require log likelihood derivatives to 4th order.
General case examples

- Cox PH, Cox Process and multivariate GAMs...
- Also GAMLSS models (Rigby & Stasinopoulus, 2005)...
  - log likelihood for the \( i^{\text{th}} \) datum is \( l(y_i, \eta_i^1, \eta_i^2, \ldots) \) where the \( \eta^k = X^k \beta^k \) are \( K \) linear predictors.
  - Newton estimation of \( \hat{\beta} \) + implicit differentiation requires
    \[
    \frac{\partial}{\partial \beta^l} = l_{\eta^l} X_{ij}^l, \quad \frac{\partial}{\partial \beta^l \beta^m} = l_{\eta^l \eta^m} X_{ij}^l X_{ik}^m.
    \]
  - First derivatives of \( \mathcal{V} \) then require
    \[
    \frac{\partial^2}{\partial \beta^l \beta^m \rho^p} = \frac{\partial^2}{\partial \rho^q \beta^r} = l_{\hat{\eta}^l \hat{\eta}^m \hat{\eta}^q} X_{ij}^l X_{ik}^m X_{ir}^q \frac{d\hat{\beta}^q}{d\rho^p} = \frac{d\hat{\eta}^q_i}{d\rho^p} X_{ij}^l X_{ik}^m.
    \]
- Second derivatives not much worse. Generically require mixed partials of log density up to 4th order.
Less general case and software

- For a single linear predictor and a log likelihood that is the sum of separate components for each observation, then methods that exploit the regression model structure are better. Details omitted!
- These are the cases where we really have a GAM, except that the response is not exponential family distributed.
- e.g. Beta regression, ordered categorical regression, Tweedie or negative binomial with estimation of all parameters etc.
- Methods implemented in mgcv 1.8-x. e.g.
  \[
  \text{gam}(\text{time} \sim s(x) + s(z), \text{family}=\text{cox.ph}, \text{weights}=.\text{censor})
  \]
  \[
  \text{gam}(\text{list}(y \sim s(x) + s(z), \sim s(v)+s(w)), \text{family}=\text{ziplss})
  \]
- ...follow up as for regular GAM.
Simple GAMLSS example

▸ CI clearly wrong for adaptive smooth on the right

▸ Let variance vary smoothly with time.

\[
gam(\text{list}(\text{accel} \sim \text{s(times,bs="ad")}, \sim \text{s(times,bs="ad")})),
\text{data=mc,family=gaulss(link=c("identity","identity")))}
\]
GAMLSS results

- Top is new method (adaptive). Bottom is gamlss (adaptive not possible).
Distribution of model coefficients

- Given log smoothing parameters, $\rho = \log \lambda$, we have Bayesian large sample result (where $\mathcal{I}$ is information matrix)

\[
\beta | \mathbf{y}, \rho \sim N(\hat{\beta}_\rho, (\mathcal{I} + S_\rho)^{-1}).
\]

- Can also allow for smoothing parameter uncertainty.

- Let $\mathbf{R}^T \mathbf{R} = (\mathcal{I} + S_\rho)^{-1}$ and $J_{ij} = \partial \hat{\beta}_i / \partial \rho_j$, while $V_\rho$ is negative inverse Hessian of ML.

\[
\beta | \mathbf{y} \overset{d}{=} \hat{\beta}_\rho + \mathbf{J}(\rho - \hat{\rho}) + \mathbf{R}^T \mathbf{z} + \sum_k \frac{\partial \mathbf{R}^T \mathbf{z}}{\partial \rho_k} \bigg|_{\hat{\rho}} (\rho_k - \hat{\rho}_k) + r
\]

where $\mathbf{z}$ is i.i.d. $N(0, 1)$ vector and $r$ a lower order remainder.

- So in the large sample limit $\beta | \mathbf{y} \sim N(\hat{\beta}_\rho, V'_\beta)$, where $V'_\beta = (\mathcal{I} + S_\rho)^{-1} + V''$, and $V''$ is a computable correction.
A better AIC

- Greven and Kneib (2010, Biometrika) show that usual generalizations of AIC can perform poorly in penalized regression, especially with random effects. Issue seems to be neglect of smoothing parameter uncertainty.
- Penalization does not alter AIC derivation up to
  \[ \text{AIC} = -2l(\hat{\beta}) + 2\mathbb{E}\left\{ (\hat{\beta} - \beta_d)^T I_d(\hat{\beta} - \beta_d) \right\} \]
  \ldots but it does alter the expectation term (variance ↓ bias ↑).
- Accounting for penalization, including smoothing parameter uncertainty, we end up with
  \[ \text{AIC} = -2l(\hat{\beta}) + 2\text{tr}(I V'_\beta). \]
  and a heuristic upper bound on the penalty term of
  \[ 2\text{tr}(2 I V_\beta - I V_\beta I V_\beta). \]
Better AIC example

- How often AIC selects a term as term 'size' increases...

- **Top** Left, for a random effect, middle the different effective degrees of freedom for the left plot. Right for a smooth. Dotted is the new AIC computation.

- **Bottom** As top left for binary, beta and Cox PH data.
Try proportional hazards with smooth effects for age and number of affected lymph nodes.
Colon cancer model

gam(time~s(age,by=sex)+sex+s(nodes)+perfor+
rx+obstruct+adhere,family=coxph(),data=col,
weights=status)

- Nodes, treatment (rx), age, sex and adhere significant.
- Should look at treatment node interaction.
Prostate diagnosis: ordered categorical signal regression

- Here are some mass spectra for samples from patients with normal, enlarged and cancerous prostate.

- Signal regression model for prediction? Category determined by a logistically distributed latent variable with mean

\[ \mu_i = \alpha + \int f(D)\nu_i(D)dD \]

- Smooth adaptive \( f \) and category cut points to be estimated.
Prostate diagnosis: ordered categorical signal regression

- In R/mgcv call is just
  \[ \text{gam}(hs \sim s(D, by=M, bs="ad", k=100), family=\text{ocat}(R=3)) \]
  
  D and M are matrices. Each row contains Masses and Spectral intensity, for one subject.

- Results are...

  ![Graphs](https://via.placeholder.com/150)

  ...OK, but a very application specific algorithm achieves better classification.
Fuel efficiency of cars
Multivariate additive model

- Correlated bivariate normal response.
- Component means given by smooth additive predictors. Best model very simple (and somewhat unexpected)
Conclusions

- It is possible to produce a quite general simulation free framework that allows computationally reliable inference for a very wide range of GAM extensions.
- The framework has the advantage of allowing access to much of the inferential machinery available for exponential family GAMs.
- A side effect of the approach is that we can fix AIC for this penalized regression context.
- The disadvantage is the implementational one of requiring 3\textsuperscript{rd} or 4\textsuperscript{th} order derivatives of the likelihood, w.r.t. its parameters.
- In simulations (not shown) the approach is faster and somewhat more reliable than available alternatives (but for many models in the class there is nothing else available).
The problem with log determinants

- Consider $5 \times 5$ matrix $\mathbf{C}$ with QR decomposition $\mathbf{C} = \mathbf{QR}$
- $|\mathbf{C}| = |\mathbf{R}| = \prod_i R_{ii}$.
- Suppose that $\mathbf{C} = \mathbf{A} + \mathbf{B}$. $\mathbf{A}$ rank 2, element size $O(a)$, $\mathbf{B}$ rank 3, element size $O(b)$ and $a \gg b$.
- Schematic non-zero structure of $\mathbf{C} = \mathbf{A} + \mathbf{B}$.

$$
\begin{bmatrix}
\bullet & \bullet & \bullet & \bullet & \bullet \\
\bullet & \bullet & \bullet & \bullet & \cdot \\
\bullet & \bullet & \bullet & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\end{bmatrix}
= 
\begin{bmatrix}
\bullet & \bullet & \bullet & \bullet & \bullet \\
\bullet & \bullet & \bullet & \bullet & \cdot \\
\bullet & \bullet & \bullet & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\end{bmatrix}
+ 
\begin{bmatrix}
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot \\
\end{bmatrix}
$$

where $\bullet$ shows the $O(a)$ elements and $\cdot$ those of $O(b)$. 
QR decomposition applies successive householder reflections to $\mathbf{C}$, each zeroing the subdiagonal elements of successive columns of $\mathbf{C}$.

Let $\mathbf{Q}_2^T$ be product of first 2 reflections. Consider the QR decomposition after 2 steps. $\mathbf{Q}_2^T \mathbf{C} = \mathbf{Q}_2^T \mathbf{A} + \mathbf{Q}_2^T \mathbf{B}$ is

$$
\begin{bmatrix}
\bullet & \bullet & \bullet & \cdots \\
\bullet & \bullet & \bullet & \cdots \\
d_1 & \cdots \\
d_2 & \cdots \\
d_3 & \cdots 
\end{bmatrix}
= \begin{bmatrix}
\bullet & \bullet & \bullet \\
\bullet & \bullet & \bullet \\
d_1' & \cdots \\
d_2' & \cdots \\
d_3' & \cdots 
\end{bmatrix}
+ \begin{bmatrix}
\cdots & \cdots & \cdots \\
\cdots & \cdots & \cdots \\
d_1'' & \cdots \\
d_2'' & \cdots \\
d_3'' & \cdots 
\end{bmatrix}
$$

Because $\mathbf{A}$ is rank 2, $d'$ should be 0, and $d$ should be $d''$.

Computationally $d' = O(\epsilon a)$ where $\epsilon$ is the machine precision.

If $b$ approaches $O(\epsilon a)$, we suffer catastrophic loss of precision in $d$, $R_{33}$ and the computed value of $|\mathbf{C}|$. 

Consider the similarity transform $\mathbf{UCU}^T = \mathbf{UAU}^T + \mathbf{UBU}^T$, which produces the following schematic

\[
\begin{bmatrix}
\ddots & \ddots & \cdots & \cdots & \cdots \\
\ddots & \ddots & \cdots & \cdots & \cdots \\
\ddots & \ddots & \cdots & \cdots & \cdots \\
\ddots & \ddots & \cdots & \cdots & \cdots \\
\ddots & \ddots & \cdots & \cdots & \cdots \\
\end{bmatrix} =
\begin{bmatrix}
\mathbf{U}
\end{bmatrix}
+ \begin{bmatrix}
\mathbf{U}
\end{bmatrix}
\]

...the computation of $\mathbf{UCU}^T$ is by adding $\mathbf{UBU}^T$ to $\mathbf{UAU}^T$ with the theoretically zero elements set to exact zeroes.

$|\mathbf{UCU}^T| = |\mathbf{C}|$, but computation no longer suffers from the precision loss problem, no-matter how disparate $a$ and $b$ are.

Can generalize to any number of components (Appendix B, Wood, 2011).