6.2 Fitting Generalised Additive Models

Fitting Generalised Additive Models

For simplicity let us assume a model with two explanatory variables, x and z and the response variable y with a simple additive structure

$$y_i = f_1(x_i) + f_2(z_i) + \varepsilon_i.$$

The f_j are smooth functions, and the ε_i are i.i.d. $N(0, \sigma^2)$ random variables.

There are two main characteristics of this model:

1. The assumption of additive effects is a fairly strong one: $f_1(x) + f_2(z)$ is a quite restrictive special case of a function of two variables $f(x, z)$.
2. The fact that the model contains more than one function introduces an **identifiability** problem: f_1 and f_2 are each only estimable to within an additive constant. Any constant could be simultaneously added to f_1 and subtracted from f_2 without changing the model predictions.

Provided the identifiability issue is addressed, the additive model can be represented using penalised regression splines, estimated by penalised least squares and the degree of smoothing estimated by cross-validation, in the same way as the simple univariate model.

Generalized smoothing splines

Before we continue with the description of fitting additive model we briefly introduce **generalised smoothing splines**.

The generalised smoothing spline approach is built around the theory of **reproducing kernel Hilbert spaces**.

We consider a space of functions $\mathcal{F}$, where an inner product of two functions is

$$\langle g, f \rangle = \int g^{''}(x) f^{''}(x) dx,$$

and consequently a norm on the space is the cubic spline penalty can be written as

$$\langle f, f \rangle = \int f^{''}(x)^2 dx.$$

Note that functions for which the norm is zero are excluded from $\mathcal{F}$.

The **Reisz representation theorem** says that there exists a function $R_z \in \mathcal{F}$ such that $f(z) = \langle R_z, f \rangle$, for any $f \in \mathcal{F}$. That is, if we want to evaluate f at some particular value z, then one way of doing it is to take the inner product of the function f, with **representor of evaluation** function R_z.

Suppose further that we construct a function of two variables $R(z, x)$, such that for any z, $R(z, x) = R_z(x)$. This function is known as the **reproducing kernel** of the space, since $\langle R(z, \cdot), R(\cdot, t) \rangle = R(z, t)$, that is

$$\int \frac{\partial^2 R(z, x)}{\partial x^2} \frac{\partial^2 R(x, t)}{\partial x^2} dx = R(z, t),$$

by the Reisz representation theorem.

From this general approach to splines the following **cubic spline basis** can be derived (see Gu (2002)):

$$b_1(x) = 1, b_2(x) = x \text{ and } b_{i+2} = R(x, x_i^*)$$

for $i = 1, \ldots, q-2$, where x_i^* are the knot locations and

$$R(x, z) = \frac{1}{4}\left[\left(z - \frac{1}{2}\right)^2 - \frac{1}{12}\right]\left[\left(x - \frac{1}{2}\right)^2 - \frac{1}{12}\right] - \frac{1}{24}\left[\left(|x - z| - \frac{1}{2}\right)^4 - \frac{1}{2}\left(|x - z| - \frac{1}{2}\right.\right.$$

Penalized regression spline representation of an additive model

Each smooth function f_1 and f_2 in the above additive model can be represented using a penalized regression spline basis resulting from the general approach to splines as outlined above.

$$f_1(x) = \delta_1 + \delta_2 x + \sum_{j=1}^{q_1 - 2} \delta_{j+2} R(x, x_j^*)$$

and

$$f_2(z) = \gamma_1 + \gamma_2 z + \sum_{j=1}^{q_2 - 2} \gamma_{j+2} R(z, z_j^*)$$

where δ_j and γ_j are the unknown parameters for f_1 and f_2 respectively. q_1 and q_2 are the number of unknown parameters for f_1 and f_2 respectively, while x_j^* and z_j^* are the knot locations for the two functions.

To deal with the previously mentioned identifiability problem of the additive model we set $\gamma_1 = 0$. Then the additive model can be written in the linear model form

$$y = \boldsymbol{X}\beta + \varepsilon,$$

where the ith row of the design matrix is now

$$X_i = [1, x_i, R(x_i, x_1^*), R(x_i, x_2^*), \ldots, R(x_i, x_{q_1-2}^*), z_i, R(z_i, z_1^*), \ldots, R(z_i, z_{q_2-2}^*)]$$

and the parameter vector is

$$\beta = [\delta_1, \delta_2, \ldots, \delta q_1, \gamma_2, \gamma_3, \ldots, \gamma_{q_2}]^\top.$$

The wiggliness of the functions can also be measured by

$$\int f_1^{''}(x)^2 dx = \beta^\top \boldsymbol{S}_1 \beta$$

and

$$\int f_2^{''}(x)^2 dx = \beta^\top \boldsymbol{S}_2 \beta$$

where $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are zero everywhere except for $(\boldsymbol{S}_1)_{i+2,j+2} = R(x_i^*, x_j^*)$ for $i, j = 1, \ldots, q_1 - 2$ and $(\boldsymbol{S}_2)_{i+q_1+1,j+q_1+1} = R(z_i^*, z_j^*)$ for $i, j = 1, \ldots, q_2 - 2$.

Fitting additive model by penalized least squares

The parameter β of our additive model is obtained by minimization of the penealized least squares criterion

$$\|y - \boldsymbol{X}\beta\|^2 + \lambda_1 \beta^\top \boldsymbol{S}_1 \beta + \lambda_2 \beta^\top \boldsymbol{S}_2 \beta,$$

where λ_1 and λ_2 control the weight to be given to the objective of making f_1 and f_2 smooth, relative to the objective of closely fitting the response data.

Defining $\boldsymbol{S} = \lambda_1 \boldsymbol{S}_1 + \lambda_2 \boldsymbol{S}_2$ the parameter estimates are calculated as

$$\hat{\beta} = (\boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{S})^{-1} \boldsymbol{X}^\top y.$$

Additionally, the hat matrix is here

$$\boldsymbol{H}_\lambda = \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{S})^{-1} \boldsymbol{X}^\top,$$

where λ subscript is here a vector $\lambda = (\lambda_1, \lambda_2)^\top$.

Residual variance

In the identity link, normal errors case, by analogy with linear regression, σ^2 could be estimated by the residual sum of squares divided by the residual degrees of freedom:

The variance σ^2 of the residual error is typically estimated by

$$\hat{\sigma}^2 = \frac{\|\boldsymbol{y} - \boldsymbol{H}_\lambda \boldsymbol{y}\|^2}{n - \mathrm{tr}(\boldsymbol{H}_\lambda)}$$

where $\boldsymbol{H}_\lambda$ is the hat matrix defined by $\boldsymbol{H}_\lambda \boldsymbol{y} = \hat{\boldsymbol{y}}$.

In fact this estimator of σ^2 is not unbiased since it can be shown

$$\mathrm{E}\left(\|\boldsymbol{y} - \boldsymbol{H}_\lambda \boldsymbol{y}\|^2\right) = \sigma^2 [n - 2\mathrm{tr}(\boldsymbol{H}_\lambda) + \mathrm{tr}(\boldsymbol{H}_\lambda^\top \boldsymbol{H}_\lambda)] \boldsymbol{b}^\top \boldsymbol{b},$$

where $\boldsymbol{b} = \boldsymbol{\mu} - \boldsymbol{H}_\lambda \boldsymbol{\mu}$ represents the smoothing bias. Here $\boldsymbol{\mu} = \boldsymbol{X}\boldsymbol{\beta}$.

Backfitting

Backfitting is a beautifully simple way of fitting additive models (AM), which allows an enormous range of possibilities for representing the component functions of an AM, including, e.g. loess smooths and regression trees.

The basic idea behind backfitting is to estimate each smooth component of an additive model by iteratively smoothing *partial residuals* from the AM. The partial residuals relating to the j-th smooth term are the residuals resulting from subtracting all the current model term estimates from the response variable, except for the estimate of the j-th smooth.

Here is a more formal description of the algorithm. Suppose that the objective is to estimate the AM

$$y_i = \alpha + \sum_{j=1}^{m} f_j(x_{ji}) + \epsilon_i$$

where the f_j are some functions and the predictors x_j may sometimes be vector predictors (covariates). Let $\hat{\boldsymbol{f}}_j$ denote the vector whose j-th element is the estimate of $f_j(x_{ji})$.

The basic backfitting algorithm is as follows:

1. Set $\hat{\alpha} = \bar{y}$, and it never changes. Set $\hat{\boldsymbol{f}}_j = \mathbf{0}$ for $j = 1, \ldots, m$.
2. Cycle through $j = 1, \ldots, m$ and

   (a) Calculate partial residuals: $\boldsymbol{e}_p^j = \boldsymbol{y} - \hat{\alpha} - \sum_{k \neq j} \hat{\boldsymbol{f}}_k$

   (b) Set $\hat{\boldsymbol{f}}_j$ equal to the result of smoothing $\boldsymbol{e}_p^j$ with respect to x_j until the $\hat{\boldsymbol{f}}_j$ stop changing noticeably.

Backfitting runs very fast and works with virtually every combination of smoothers.

Example: Backfitting (Wood (2006))

This example implements the basic backfitting algorithm using the R function **`smooth.spline`** as the smoothing method. Here, **`edf`**$[\mathtt{j}]$ contains the required degrees of freedom for the jth smooth, **`x`** is an m column array, with jth column containing the (single) covariate for the jth smooth, and the response is **`y`**.

```
## This code requires inputs x,y,m,edf to work
f<-x*0;
alpha<-mean(y);
ok <- TRUE
while (ok) { # backfitting loop
for (i in 1:m) { # loop through the smooth terms
ep <- y - rowSums(f[,-i]) - alpha
b <- smooth.spline(x[,i],ep,df=edf[i])
f[,i] <- predict(b,x[,i])$y
}
rss <- sum((y-rowSums(f))^2)
if (abs(rss-rss0)<1e-6*rss)
  ok <- FALSE
rss0 <- rss
}
```

Fitting generalized additive models

The GAM is fitted by penalised likelihood maximisation. In principle, this is achieved by **penalised iteratively re-weighted least squares** (P-IRLS) via the following procedure.

1. Given the current parameter estimate $\boldsymbol{\beta}^{(k)}$ and the corresponding estimated mean response vector $\boldsymbol{\mu}^{(k)}$, calculate weights w_{ii} and response $\boldsymbol{z}$ as in the case of IRLS in the generalized linear model framework.

2. Minimize

$$\|\sqrt{\boldsymbol{W}}(\boldsymbol{z}-\boldsymbol{X}\boldsymbol{\beta})\|^2+\lambda_1\boldsymbol{\beta}^\top\boldsymbol{S}_1\boldsymbol{\beta}+\lambda_2\boldsymbol{\beta}^\top\boldsymbol{S}_2\boldsymbol{\beta} \tag{6.2.1}$$

with respect to $\boldsymbol{\beta}$ to obtain $\boldsymbol{\beta}^{(k+1)}$. $\boldsymbol{W}$ is a diagonal matrix such that $W_{ii}=w_{ii}$.

3. Update $\boldsymbol{\beta}$ and thus $\boldsymbol{W}$ and $\boldsymbol{z}$. If the difference between old and new is small, terminate, otherwise start over.

At convergence of the above algorithm, the expression

$$\|\sqrt{\boldsymbol{W}}(\boldsymbol{z}-\boldsymbol{X}\boldsymbol{\beta})\|^2+\sum_{k=1}^{K}\lambda_k\boldsymbol{\beta}^\top\boldsymbol{S}k\boldsymbol{\beta},$$

when viewed as a function of $\boldsymbol{\beta}$, is (up to a constant) a quadratic approximation of the penalized deviance

$$D(\beta)+\sum_{k=1}^{K}\lambda_k\boldsymbol{\beta}^\top\boldsymbol{S}_k\boldsymbol{\beta}.$$

Smoothing parameter estimation criteria

A GCV score for smoothing parameter selection is hence

$$\mathcal{V}_g^w=\frac{n\|\sqrt{\boldsymbol{W}}(\boldsymbol{z}-\boldsymbol{X}\boldsymbol{\beta})\|^2}{[n-\operatorname{tr}(\boldsymbol{H}_\lambda)]^2}.$$

This approximation is only valid locally to the λ used to find $\boldsymbol{z}$ and $\boldsymbol{W}$.

A globally applicable GCV score can be obtained

$$\mathcal{V}_g=\frac{nD(\hat{\beta})}{(n-\operatorname{tr}(\boldsymbol{H}_\lambda))^2}.$$

In case the scaling parameter of the GLM is known (or if there is none), the **UBRE (unbiased risk estimator)**

$$\mathcal{V}_u^w=\frac{1}{n}\|\sqrt{\boldsymbol{W}}(\boldsymbol{z}-\boldsymbol{X}\boldsymbol{\beta})\|^2\sigma^2+\frac{2}{n}\operatorname{tr}(\boldsymbol{H}_\lambda)\sigma^2$$

may be used as a criterion for the selection of the λ_k. Comparison with AIC and setting $d=\operatorname{tr}(\boldsymbol{H}_\lambda)$ shows that this is effectively a linear transformation of the AIC (or Mallow's C_p).

More generally we, therefore, have the UBRE equal to

$$\mathcal{V}_u = \frac{1}{n} D(\hat{\beta}) - \sigma^2 + \frac{2}{n} \mathrm{tr}(\boldsymbol{H}_\lambda)\sigma^2.$$

Activity: Fitting AMs by penalized least squares

Question *Submitted Mar 17th 2023 at 1:06:24 am*

a) Show that the additive model penalized least squares criterion

$$\|y - \boldsymbol{X}\beta\|^2 + \sum_{k=1}^{K} \lambda_k \beta^\top \boldsymbol{S}_k \beta$$

can be replaced with an ordinary regression criterion on an augmented data set.

b) For the augmented design matrix $\tilde{\boldsymbol{X}}$ from a) show that the sum of the first n elements on the leading diagonal of

$$\tilde{\boldsymbol{X}}(\tilde{\boldsymbol{X}}^\top \tilde{\boldsymbol{X}})^{-1} \tilde{\boldsymbol{X}}^\top$$

is

$$\mathrm{tr}(\boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{S})^{-1} \boldsymbol{X}^\top).$$

Here n is the number of rows in $\boldsymbol{X}$.

asdf

Activity in R: Backfitting

You will now explore backfitting in the context of multiple linear regression.

Suppose that we would like to perform multiple linear regression, but we do not have software to do so. Instead, we only have software to perform simple linear regression. Therefore, we take the following iterative approach: we repeatedly hold all but one coefficient estimate fixed at its current value, and update only that coefficient estimate using simple linear regression. The process is continued until convergence. We now try this out on a toy example.

(a) Generate a response Y and two predictors X_1 and X_2 , with $n = 100$.

(b) Initialize $\hat{\beta}_1$ to take on a value of your choice.

(c) Keeping $\hat{\beta}_1$ fixed, fit the model

$$Y - \hat{\beta}_1 X_1 = \beta_0 + \beta_2 X_2 + \varepsilon.$$

You can do this as follows:

```
a =y - beta1 * x1
beta2 = lm( a~x2 )$coef[2]
```

(d) Keeping $\hat{\beta}_2$ fixed, fit the model

$$Y - \hat{\beta}_2 X_2 = \beta_0 + \beta_1 X_1 + \varepsilon.$$

You can do this as follows:

```
a =y - beta2 * x2
beta1 = lm( a~x1 )$coef [2]
```

(e) Write a for loop to repeat (c) and (d) 1,000 times. Report the estimates of $\hat{\beta}_0$, $\hat{\beta}_1$, and $\hat{\beta}_2$ at each iteration of the for loop. Create a plot in which each of these values is displayed, with $\hat{\beta}_0$, $\hat{\beta}_1$, and $\hat{\beta}_2$ each shown in a different color.

(f) Compare your answer in (e) to the results of simply performing multiple linear regression to predict Y using X_1 and X_2. Use the $\texttt{abline}()$ function to overlay those multiple linear regression coefficient estimates on the plot obtained in (e).

Additional Activity

Question 1 *Submitted Mar 17th 2023 at 1:08:08 am*

Is the following statement correct: *The additive model can be written in the linear model form*

$$y = \boldsymbol{X}\beta + \varepsilon,$$

where the ith row of the design matrix is now

$$X_i = [1, x_i, R(x_i, x_1^*), R(x_i, x_2^*), \ldots, R(x_i, x_{q_1-2}^*), z_i, R(z_i, z_1^*), \ldots, R(z_i, z_{q_2-2}^*)]$$

for an appropriately chosen funtion R and the parameter vector is

$$\beta = [\delta_1, \delta_2, \ldots, \delta q_1, \gamma_2, \gamma_3, \ldots, \gamma_{q_2}]^\top.$$

● Yes

○ No

Question 2 *Submitted Mar 17th 2023 at 1:08:17 am*

Which of the following steps of the penalised iteratively re-weighted least squares (P-IRLS) algorithm for fitting generalised additive models are NOT correct:

☑ Given the current parameter estimate $\boldsymbol{\lambda}^{(k)}$ and the corresponding estimated mean response vector $\boldsymbol{\mu}^{(k)}$, calculate weights w_{ii} and response $\boldsymbol{z}$ as in the case of IRLS in the generalized linear model framework.

☑ Minimize

$$\|\sqrt{\boldsymbol{W}}(\boldsymbol{z} - \boldsymbol{X}\boldsymbol{\beta})\|^2 + \lambda_1 \boldsymbol{\beta}^\top \boldsymbol{S}_1 \boldsymbol{\beta} + \lambda_2 \boldsymbol{\beta}^\top \boldsymbol{S}_2 \boldsymbol{\beta}$$

with respect to $\boldsymbol{\lambda}$ to obtain $\boldsymbol{\lambda}^{(k+1)}$. $\boldsymbol{W}$ is a diagonal matrix such that $W_{ii} = w_{ii}$.

☑ Update $\boldsymbol{\lambda}$ and thus $\boldsymbol{W}$ and $\boldsymbol{z}$. If difference between old and new is small, terminate, otherwise start over.