Paul Martins

Beyond Linearity

Notes from ISLR book

Contents

1. Basis function

Using a family of fixed and known functions or transformations $b_1, \dots, b_K$ applied to the predictors X, we fit the model

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \cdots + \beta_K b_K(x_i) + \varepsilon_i \tag{1}$$

This corresponds to polynomial regressions if $b_j(x_i) = x_i^j$ or piecewise constant (step functions) when $b_j(x_i) = I(c_j \leq x_j < c_{j+1})$.

2. Splines

2.1 Regression splines

A regression spline is a combination of the two models cited above: we fit a polynomial for different ranges of x_i. K knots are used to define the ranges of x_i. To make sure a degree d spline is smooth, the first $d-1$ derivatives of the piecewise polynomials need to be continuous at the knots.

Cubic spline Taking $d = 3$, the cubic splines example has 8 degrees of freedom (2×4 β) and 3 constraints (continuity of the 0, 1 and 2 derivatives) which gives a final 5 degrees of freedom. A cubic spline will always have $4+K$ degrees of freedom.

In the basis function framework, we have

$$y_i = \beta_0 + \underbrace{\sum_{d=1}^{3} \beta_d b_d(x_i)}_{\text{polynomial}} + \underbrace{\sum_{k=1}^{K} \beta_{k+3} b_k(x_i)}_{\text{truncated power basis}} \tag{2}$$

$$b_d = x^d, \quad b_k = \begin{cases} (x-\xi_k)^3, & \text{if } x \geq \xi_k \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where ξ_k are the knots. The b_k functions ensure the continuity of the first 2 derivatives. Once again we observe that we only need $K+4$ parameters, hence the number of dof.

The splines can have high variance at the outer range. A natural spline imposes additional boundary constraints (linear at the boundary). Knots are often distributed uniformly and their number (or dof number) is determined with cross-validated RSS.

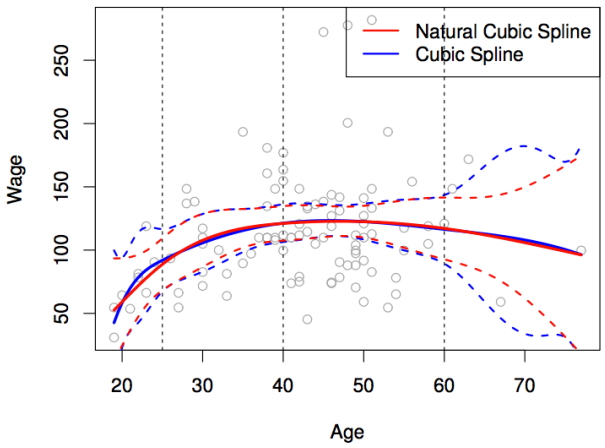

Figure 1. Cubic spline with 3 knots

2.2 Smoothing splines

A smoothing spline g has a knot at every x_i and minimises

$$\sum_{i=1}^{n} \underbrace{(y_i - g(x_i))^2}_{\text{loss}} + \lambda \underbrace{\int g''(t)^2 dt}_{\text{penalty}} \tag{4}$$

The loss part encourages to fit the data well and the penalty penalizes the variability[1]. λ is a parameter that control the smoothness of the spline and the effective degrees of freedom df_λ In other words, λ controls the bias-variance trade-off, or the level of shrinkage.

λ	0	$\rightarrow$	∞
Fit	overfitting	good	underfitting
df_λ	n	$\searrow$	2

3. Local regression

Local regression is a memory-based method as $\hat{f}$ needs all the data for each estimation at x_0. It consists of a multiple weighted linear regression.

1. Define the span $s = k/n$ using k training points x_i which are the closest to x_0
2. Assign a weight $K_{i0} = K(x_i, x_0)$ to each point in the neighborhood so that the furthest away from x_0 gets the smallest weight.
3. Fit a weighted least squares regression of the y_i on the x_i by finding $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize

$$\sum_{i=1}^{n} K_{i0} \cdot (y_i - \beta_0 - \beta_1 x_i)^2 \tag{5}$$

[1] the first derivative represent the local slope, the second represents the roughness/wiggly trend

4 The fitted value at x_0 is given by $\hat{f}(x_0) = \hat{\beta}_0 + \hat{\beta}_1 x_0$

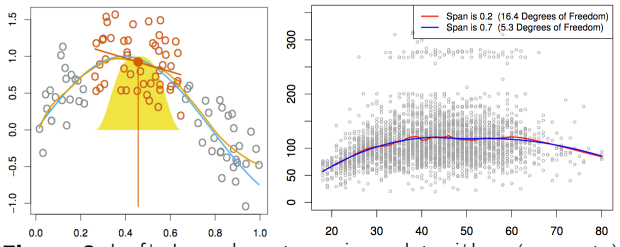

Figure 2. Left: Local regression algorithm (orange) vs true (blue). Right: Local regression with different span values s

4. Generalised additive models (GAM)

GAM offers a framework to extend non-linearity to several predictors

$$y_i = \beta_0 + \sum_{j=1}^{n} f_j(x_{ij}) + \varepsilon_i \tag{6}$$

It adds the contribution of each f_j function calculated on each predictor. Additivity means we can still examine the effects of each predictor on the response independently of other predictors. This can also be used for classifications (ie replacing y_i with $log(p/(1-p))$).