STAT 5630, Fall 2019

Splines

Xiwei Tang, Ph.D. <xt4yj@virginia.edu>

University of Virginia
November 12, 2019

Outline

- From Linear to Local Methods
- Piecewise Polynomials and Splines
- B-Splines

Linear vs. Nonlinear models

- For most of our lectures up to now, we focused on linear models.
    - Convenient and easy to fit
    - Easy to interpret
    - An approximation to the true underlying function $f(x)$
    - When n is small and/or p is large, linear models tend not to overfit
- Nonlinear models are more flexible and may lead to better fitting
- Our first encounter with nonlinear functions is the SVM with kernel trick, which is equivalent to (some) basis expansions.
- The concept in this lecture is mainly about nonlinear functions of a univariate variable.

Linear vs. Nonlinear models

- Additive model: stepping outside the linear model, lets assume that our model has the form

$$f(x) = \sum_{j=1}^{p} f_j(x_j)$$

- This allows some flexibility since f_j does not need to be $\beta_j x_j$.
- For most part of today's lecture, we focus on how to estimate the functions f_j's, which is are univariate functions of x_j's.
- In particular, we consider a linear basis expansion of each f_j, i.e.,

$$f_j(x) = \sum_{m=1}^{M_j} \beta_{jm} h_{mj}(x_j)$$

- h_{mj} are the basis functions, maybe different for each covariate j (we could also use the notation $\phi_m(x_j)$).

Linear vs. Nonlinear models

- Once we have determined the basis functions h_m, the model is again linear (just not in the original covariates)
- Some typical choices of h
    - $h_m(x) = x$: the original linear model
    - $h_m(x) = x^2$, $x^3, \ldots$: polynomials
    - $h_m(x) = \log(x)$, $\sqrt{x}, \ldots$: other nonlinear transformations
    - $h_m(x) = \mathbf{1}\{L_m < x < U_m\}$: indicator for a region of X

Piecewise Polynomials and Splines

Piecewise Polynomials

- The approach is straight forward: we find a collection of basis functions, and calculate the $h_m(x_i)$ values of each subject on these basis, and treat them as values of news predictors. A linear function can be then used to fit the model.
- For example, consider the piecewise constant:

$$h_1(x) = \mathbf{1}\{x < \xi_1\}, \quad h_2(x) = \mathbf{1}\{\xi_1 \leq x < \xi_2\}, \quad h_3(x) = \mathbf{1}\{\xi_2 \leq x\}$$

- ξ_1 and ξ_2 are called knots
- Hence the model becomes

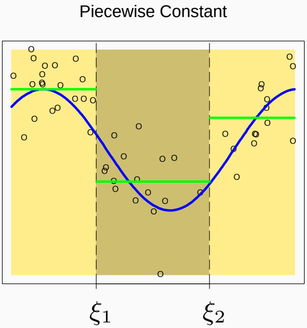
$$f(X) = \sum_{i=1}^{3} \beta_m h_m(X)$$

- This is essentially fitting a constant function at each region, so $\beta_m = \overline{Y}_m$. This is similar to a regression tree model.

Piecewise Polynomials

- We can also fit a linear function at each region by considering three additional basis functions:

$$h_4(x) = x\mathbf{1}\{x < \xi_1\},\ h_5(x) = x\mathbf{1}\{\xi_1 \le x < \xi_2\},\ h_6(x) = x\mathbf{1}\{\xi_2 \le x\}$$



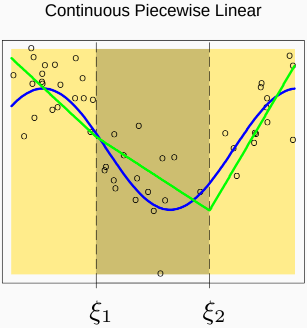
Piecewise Polynomials

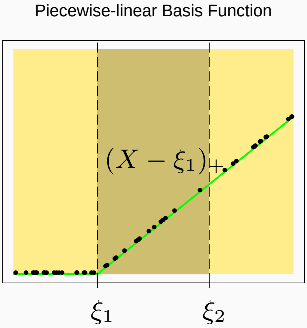
- However, the prediction functions are not continuous. Hence we might want some restrictions on the parameter estimates to force it. For example

$$f(\xi_1^-) = f(\xi_1^+)$$

implies $\beta_1 + \xi_1\beta_4 = \beta_2 + \xi_1\beta_5$

- This leads to a continuous fitting:





Piecewise Polynomials

- Because of the two constrains, there are only 4 parameters instead of 6
- The trick to this model fitting is to incorporate the constrains into the basis functions (or an equivalent set of basis):

$$h_1(x) = 1,\ h_2(x) = x,\ h_3(x) = (x - \xi_1)_+,\ h_4(x) = (x - \xi_2)_+,$$

  where $(\cdot)^+$ denotes the positive part.
- We can then check that any linear combination of these four functions lead to
  - Continuous everywhere
  - Linear everywhere except the knots
  - Has a different slope for each region
- This can be easily done using R function bs in the package splines.

Cubic Splines

- Another common choice is cubic splines, which uses cubic functions within each region. However, continuity of the first and second order at the knots is forced.
- For each knot ξ, we need the following 4 basis functions:

$$h_1(x) = 1,\ h_2(x) = x,\ h_3(x) = x^2,\ h_4(x) = (x - \xi)^3.$$

- Cubic spline function with K knots:

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \sum_{k=1}^{K} b_k (x - \xi_k)_+^3$$

- The degrees of freedom for a cubic spline:

$(\#\,\text{regions}) \times (4\,\text{per region}) - (\#\,\text{knots}) \times (3\,\text{constraints per knot})$

- The (third order) knot discontinuity is not really visible

B-Splines

B-spline basis

- The previous definition of the splines are known as regression splines
- An alternative (computationally more efficient) way of defining the spline basis is proposed by de Boor (1978)
- Each basis function is nonzero over at most (degree + 1) consecutive intervals
- The order of a spline is $M = \text{degree} + 1$
- The resulting design matrix is banded

B-spline basis

- Create augmented knot sequence τ:

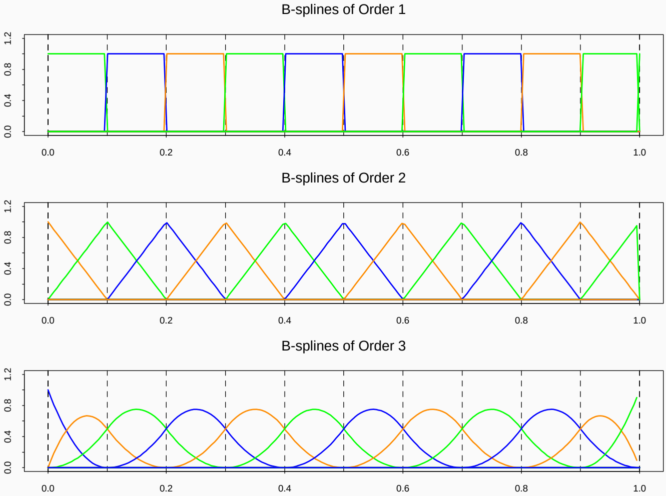
$$\tau_1 = \cdots = \tau_M = \xi_0$$
$$\tau_{M+j} = \xi_j, \quad j = 1, \ldots K$$
$$\tau_{M+K+1} = \cdots = \tau_{2M+K+1} = \xi_{K+1}$$

  where ξ_0 and ξ_{K+1} are the left and right boundary points.
- Denote $B_{i,m}(x)$ the ith B-spline basis function of order m for the knot sequence τ, $m \leq M$. We recursively calculate them as follows:

$$B_{i,1}(x) = \begin{cases} 1 & \text{if} \quad \tau_i \leq x < \tau_{i+1} \\ 0 & \text{o.w.} \end{cases}$$

$$B_{i,m}(x) = \frac{x - \tau_i}{\tau_{i+m-1} - \tau_i} B_{i,m-1}(x) + \frac{\tau_{i+m} - x}{\tau_{i+m} - \tau_{i+1}} B_{i+1,m-1}(x)$$

B-spline basis



Generating B-spline basis in R

```
> library(splines)
> bs(x, df = NULL, knots = NULL, degree = 3, intercept = FALSE)
```

- df : degrees of freedom (the total number of basis)
- knots : specify knots. By default, these will be the quantiles of x
- degree : degree of piecewise polynomial, default 3 (cubic splines)
- intercept : if TRUE , an intercept is included, default FALSE
- Return a matrix of dimension $n\times$ df

Natural Cubic Splines

- polynomials fit to data tends to be erratic near the boundaries, and extrapolation can be dangerous
- Natural cubic splines (NCS) forces the second and third derivatives to be zero at the boundaries, i.e., $\min(x)$ and $\max(x)$
- Hence, the fitted model is linear beyond the two extreme knots $(-\infty, \xi_1]$ and $[\xi_K, \infty)$
- Assuming linearity near the boundary is reasonable since there is less information available
- The constraints frees up 4 degrees of freedom. The degrees of freedom of NCS is just the number of knots K.

Extrapolating beyond the boundaries

United States birth rate data