Lecture 16: High-dimensional regression, non-linear regression

Reading: Sections 6.4, 7.1

STATS 202: Data mining and analysis

October 30, 2019

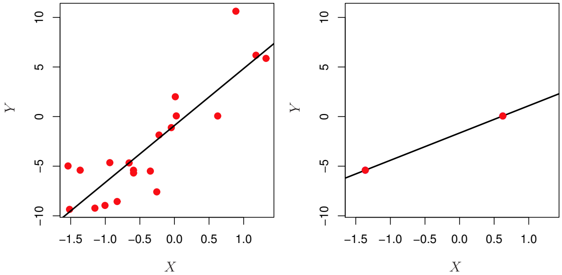
High-dimensional regression

- Most of the methods we've discussed work best when n is much larger than p.
- However, the case $p \gg n$ is now common, due to experimental advances and cheaper computers:

  1. **Medicine:** Instead of regressing heart disease onto just a few clinical observations (blood pressure, salt consumption, age), we use in addition 500,000 single nucleotide polymorphisms.
  2. **Marketing:** Using search terms to understand online shopping patterns. A *bag of words* model defines one feature for every possible search term, which counts the number of times the term appears in a person's search. There can be as many features as words in the dictionary.

Some problems we have talked about

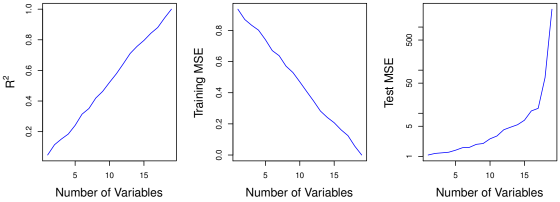
- When $n = p$, we can find a fit that goes through every point.



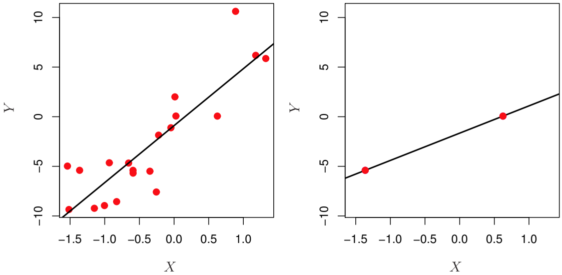
- Least-squares regression doesn't have a unique solution when $p > n$.
- Fortunately, we can use regularization methods, such as variable selection, ridge regression and the lasso.

Some problems we have talked about

- We know that least-squares regression doesn't work when $p > n$.
- We can use regularization methods, such as variable selection, ridge regression and the lasso.
- When $n = p$, we can find a fit that goes through every point.
- Measures of training error are a bad proxy for the test error.

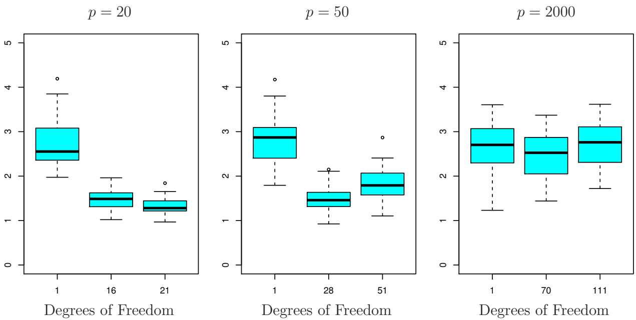


Some new problems



- Furthermore, it becomes hard to estimate the noise σ^2. (Typical unbiased estimate is $\hat{\sigma}^2 = \frac{1}{n-p-1} RSS$ for regression with intercept and p additional predictors.)
- Measures of model fit C_p, AIC, and BIC fail without reasonable noise estimate.

Some new problems



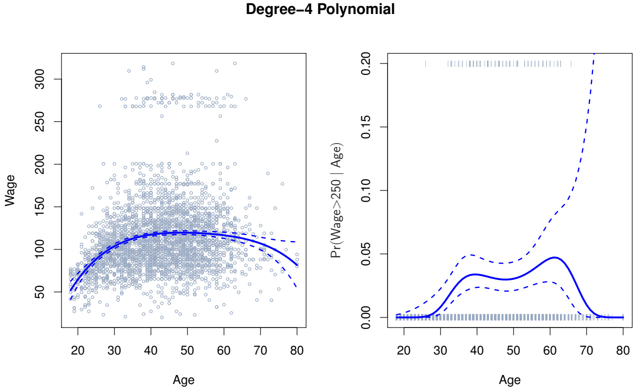
- Simulation with $n = 100$ observations.
- In each case, only 20 predictors are associated to the response.
- Plots show the test error of the Lasso.
- **Message:** Adding irrelevant predictors hurts the performance of the regression!

Interpreting coefficients when $p > n$

- When $p > n$, we always have multicollinearity: some predictors are a linear combination of other predictors.
- Typically, there isn't a single best set of variables for prediction: many sets are equally good.
- The Lasso and Ridge regression will choose one set of coefficients.
- **Take-away:** Don't overstate the importance of the predictors selected in high dimensions.

Beyond linearity

Problem: How do we model a non-linear relationship between inputs and output?



Left: Regression of `wage` onto `age`.
Right: Logistic regression for classes `wage`> 250 and `wage`≤ 250
(Dotted lines are ± 2 standard errors)

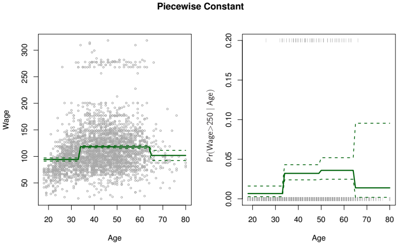
Basis functions

Strategy: Leverage linear regression technology

- Define a model with derived features $f_j(X)$:

$$Y = \beta_0 + \beta_1 f_1(X) + \beta_2 f_2(X) + \cdots + \beta_d f_d(X).$$

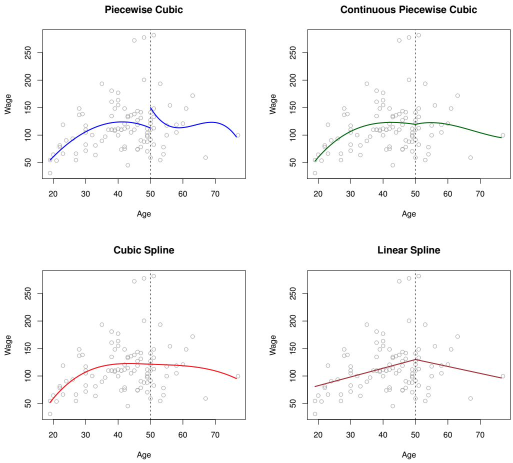
- Fit this model through least-squares regression (possibly using regularization).
- Options for features / basis functions $f_1, \ldots, f_d$:
  1. Polynomials, $f_i(x) = x^i$.
  2. Indicator functions, $f_i(x) = \mathbf{1}(c_i \leq x < c_{i+1})$.



Basis functions

- Options for features / basis functions $f_1, \ldots, f_d$:

  3. Piecewise polynomials:



Cubic splines

- Define a set of **knots** (a.k.a. **break points**) $\xi_1 < \xi_2 < \cdots < \xi_K$.
- We want the function $Y = f(X)$ to:
  1. Be a cubic polynomial between every pair of knots ξ_i, ξ_{i+1}.
  2. Be continuous at each knot.
  3. Have continuous first and second derivatives at each knot.
- It turns out, we can write f in terms of $K + 3$ basis functions:

  $$f(X) = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \beta_4 h(X, \xi_1) + \cdots + \beta_{K+3} h(X, \xi_K)$$

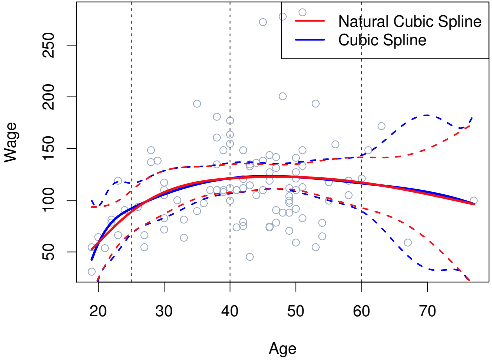
  where,

  $$h(x, \xi) = \begin{cases} (x - \xi)^3 & \text{if } x > \xi \\ 0 & \text{otherwise} \end{cases}$$

- Popular because (allegedly) most human eyes cannot detect discontinuities of third derivate at knots.

Natural cubic splines

Spline which is linear instead of cubic for $X < \xi_1$, $X > \xi_K$.

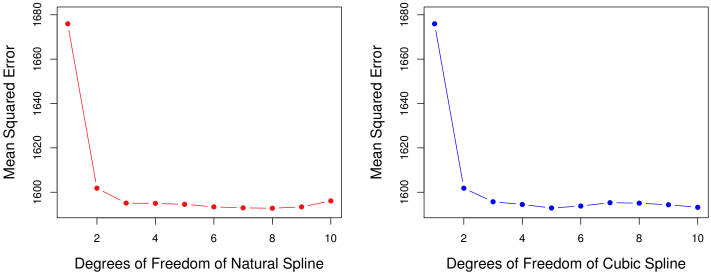


The predictions are more stable for extreme values of X.

Choosing the number and locations of knots

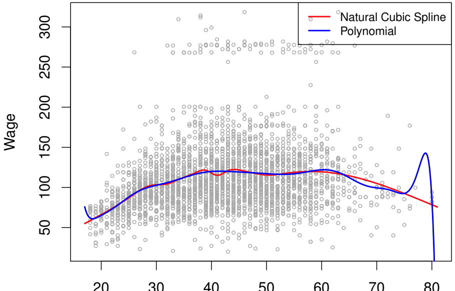
The locations of the knots are typically quantiles of X.

The number of knots, K, is chosen by cross validation:



Natural cubic splines vs. polynomial regression

- Splines can fit complex functions with few parameters: low degree polynomial, but can fit complex functions by adding more knots.
- Polynomials require high degree terms to be flexible.
- High-degree polynomials can be unstable at the edges of the dataset.
- Degree 15 polynomial vs. spline with 15 degrees of freedom:



Smoothing splines

Find the function f which minimizes

$$\sum_{i=1}^{n}(y_i - f(x_i))^2 + \lambda \int f''(x)^2 dx$$

- The RSS of the model.
- A penalty for the roughness of the function.

Facts:

- The minimizer $\hat{f}$ is a natural cubic spline, with knots at each sample point $x_1, \ldots, x_n$.
- Obtaining $\hat{f}$ is similar to a Ridge regression.

Deriving a smoothing spline

1. Show that if you fix the values $f(x_1), \ldots, f(x_2)$, the roughness

$$\int f''(x)^2 dx$$

   is minimized by a natural cubic spline.

2. Deduce that the solution to the smoothing spline problem is a natural cubic spline, which can be written in terms of its basis functions.

$$f(x) = \beta_0 + \beta_1 f_1(x) + \ldots \beta_{n+3} f_{n+3}(x)$$

3. Letting $\mathbf{N}$ be a matrix with $\mathbf{N}(i,j) = f_j(x_i)$, we can write the objective function:

$$(y - \mathbf{N}\beta)^T (y - \mathbf{N}\beta) + \lambda \beta^T \Omega_{\mathbf{N}} \beta,$$

   where $\Omega_{\mathbf{N}}(i,j) = \int f_i''(t) f_j''(t) dt$.

Deriving a smoothing spline

4. Similar to the ridge regression setting, the coefficients $\hat{\beta}$ which minimize
$$(y - \mathbf{N}\beta)^T(y - \mathbf{N}\beta) + \lambda\beta^T\Omega_{\mathbf{N}}\beta,$$
are $\hat{\beta} = (\mathbf{N}^T\mathbf{N} + \lambda\Omega_{\mathbf{N}})^{-1}\mathbf{N}^T y$.

5. Note that the predicted values are a linear function of the observed values:
$$\hat{y} = \underbrace{\mathbf{N}(\mathbf{N}^T\mathbf{N} + \lambda\Omega_{\mathbf{N}})^{-1}\mathbf{N}^T}_{\mathbf{S}_\lambda} y$$