Splines (Semiparametric Regression) Demo

Data Mining

Prof. Dawn Woodard
School of ORIE
Cornell University

Demo on Wage Data

Goal: predict income based on age, gender, educational level, etc.

- Potentially useful for:

The Wage dataset in the ISLR package includes the following for 3000 individuals from the mid-Atlantic states:

- year: year in which wage was recorded
- age: age in years
- sex
- maritl: marital status
- race
- educational level (categorical)
- wage: worker's pretax income in thousands
- etc.

Demo on Wage Data

- This example is from James, Witten, Hastie, Tibshirani (2013) and is used by permission.
- install & load the "ISLR" package
- call "help(Wage)" to get more information about the Wage dataset.
- create a scatterplot of wage vs. age (wage on the y-axis). It is clearly a nonlinear, and possibly a nonmonotonic, relationship.
- Try fitting a degree 4 polynomial regression model for wage as a function of age. Code:

```
Wage$age2 = Wage$age^2
Wage$age3 = Wage$age^3
Wage$age4 = Wage$age^4
wageFit = lm( formula = wage ~ age + age2 + age3 + age4,
data =Wage )
```

- Call "summary" on the resulting lm object to see the coefficient estimates

Demo on Wage Data

Focus on the coefficient estimates and the estimate of σ:

```
Call:
lm(formula = wage ~ age + age2 + age3 + age4, data = Wage)

Residuals:
    Min      1Q  Median      3Q     Max
-98.707 -24.626  -4.993  15.217 203.693

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.842e+02  6.004e+01  -3.067 0.002180 **
age          2.125e+01  5.887e+00   3.609 0.000312 ***
age2        -5.639e-01  2.061e-01  -2.736 0.006261 **
age3         6.811e-03  3.066e-03   2.221 0.026398 *
age4        -3.204e-05  1.641e-05  -1.952 0.051039 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 39.91 on 2995 degrees of freedom
Multiple R-squared:  0.08626,   Adjusted R-squared:  0.08504
F-statistic: 70.69 on 4 and 2995 DF,  p-value: < 2.2e-16
```

Demo on Wage Data

- The estimated standard deviation of wage conditional on age is

  (A) 0-30,000
  (B) 30,001-60,000
  (C) 60,001-90,000
  (D) 90,001-120,000

- Let's create a scatterplot of Wage vs. age, with the predicted curve superimposed. First, evaluate the curve at a grid of age values:

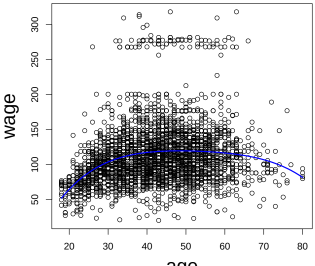
- Then predict the wage at this grid of values, using your model:
  predDat = data.frame(age = age.grid, age2 =age.grid∧2, age3 = age.grid∧3, age4 = age.grid∧4)
  preds = predict(object = wageFit, newdata = predDat)

Demo on Wage Data

Create a scatterplot of wage vs. age:
plot(x = Wage$age, y = Wage$wage, xlab = “age”, ylab = “wage”)

Superimpose the prediction curve:
lines(x = age.grid, y = preds, col = “blue”, lwd = 2)

The “lines” function superimposes a curve (defined by a sequence of x,y pairs) on a plot.
lwd controls:
col controls:



Demo on Wage Data

- We haven't checked the assumptions of the linear regression model here. The assumption that Y is normally distributed conditional on x clearly doesn't hold. We can still use the model for predictions, though.

Demo on Wage Data

- We haven't checked the assumptions of the linear regression model here. The assumption that Y is normally distributed conditional on x clearly doesn't hold. We can still use the model for predictions, though.

Demo on Wage Data

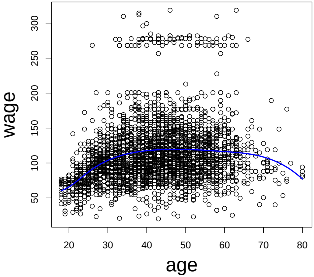
- We'll use the "splines" library; install and load it.
- The code to fit a cubic spline model is simple:
  wageFit = lm(formula = wage ∼ bs(age, knots = c(25, 40, 60)), data = Wage)
- The "bs" function specifies:
- "knots" specifies
- Get predictions and plot the curve using the same code from before:
  preds = predict(object = wageFit, newdata = predDat)
  plot(x = Wage$age, y = Wage$wage, xlab = "age", ylab = "wage")
  lines(x = age.grid, y = preds, col = "blue", lwd = 2)

Demo on Wage Data

```
Call:
lm(formula = wage ~ bs(age, knots = c(25, 40, 60)), data = Wage)

Residuals:
    Min      1Q  Median      3Q     Max 
-98.832 -24.537  -5.049  15.209 203.207 

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)    
(Intercept)                       60.494      9.460   6.394 1.86e-10 ***
bs(age, knots = c(25, 40, 60))1    3.980     12.538   0.317 0.750899    
bs(age, knots = c(25, 40, 60))2   44.631      9.626   4.636 3.70e-06 ***
bs(age, knots = c(25, 40, 60))3   62.839     10.755   5.843 5.69e-09 ***
bs(age, knots = c(25, 40, 60))4   55.991     10.706   5.230 1.81e-07 ***
bs(age, knots = c(25, 40, 60))5   50.688     14.402   3.520 0.000439 ***
bs(age, knots = c(25, 40, 60))6   16.606     19.126   0.868 0.385338    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 39.92 on 2993 degrees of freedom
Multiple R-squared:  0.08642,   Adjusted R-squared:  0.08459 
F-statistic: 47.19 on 6 and 2993 DF,  p-value: < 2.2e-16
```

Demo on Wage Data

The curve looks very similar to that from the degree-4 polynomial model:



However, it's usually not obvious how to choose what degree of polynomial to use, and polynomial fits often don't work as well in other contexts. The cubic spline model is very general-purpose and often works well.

Demo on Wage Data

- Instead of specifying the knots you can let R pick them using the equal-quantiles approach we discussed in class.
- When you call the “bs” function, specify the “df” argument instead of the “knots” argument:
  wageFit = lm(formula = wage$\sim$ bs(age, df = 6), data = Wage)
-
- We can check where R chose to put the knots:
  attr(bs(Wage$age, df = 6), “knots”)

Demo on Wage Data

Recalculate the predictions and plot the predicted curve. How much did the curve change?

(A) Not much

(B) Some

(C) A lot

Part of the reason why cubic splines are desirable is that:

Demo on Wage Data

- It's easy to handle multiple continuous predictors. One way is to allow the expected value of Y to be nonlinear in only one of those predictors. Then the spline terms are used only for that predictor. Example with 2 predictors, nonlinear in x_1, using 2 knots:

- The only other continuous predictor in this dataset is "year". Create a scatterplot of wage vs. year to see whether the relationship looks linear or nonlinear.

Demo on Wage Data

- Fit a model that allowes the expectation of wage to be nonlinear in age but linear in year:

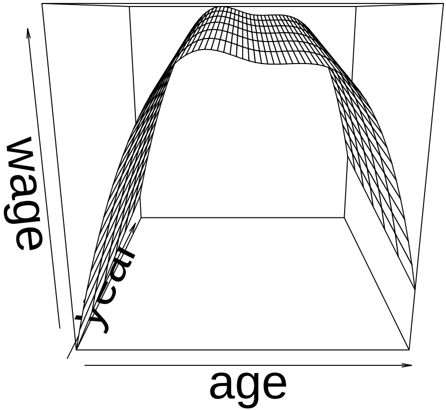
  wageFit = lm(formula = wage∼ bs(age, df = 6) + year, data = Wage)

Demo on Wage Data

```
Call:
lm(formula = wage ~ bs(age, df = 6) + year, data = Wage)

Residuals:
    Min      1Q  Median      3Q     Max 
-98.869 -24.627  -4.986  15.783 200.759 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)      -2611.3999   721.8512  -3.618 0.000302 ***
bs(age, df = 6)1    26.8529    12.4111   2.164 0.030573 *  
bs(age, df = 6)2    53.4152     7.1146   7.508 7.89e-14 ***
bs(age, df = 6)3    65.6612     8.3060   7.905 3.73e-15 ***
bs(age, df = 6)4    54.3638     8.7144   6.238 5.04e-10 ***
bs(age, df = 6)5    71.5056    13.7170   5.213 1.99e-07 ***
bs(age, df = 6)6    14.3598    16.1749   0.888 0.374729    
year                 1.3303     0.3599   3.696 0.000223 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 39.82 on 2992 degrees of freedom
Multiple R-squared:  0.09144,   Adjusted R-squared:  0.08931 
F-statistic: 43.02 on 7 and 2992 DF,  p-value: < 2.2e-16
```

Demo on Wage Data

Create a 3D plot of the predicted wage surface as a function of age and year. The code is a bit tricky, and is up on Blackboard in the “Code” section. Try it!



Demo on Wage Data

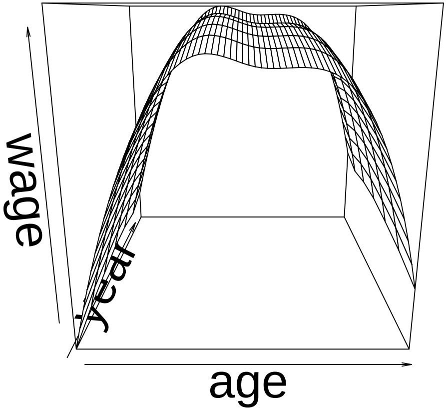
- With two continuous predictors, one can also allow the expected value of Y to be nonlinear in each of the predictors. Then the spline terms are used for both predictors. Example with 2 predictors using 2 knots in each:

- Fit a model that allowes the expectation of wage to be nonlinear in age and nonlinear in year. I use only 2 knots for year here:

  wageFit = lm(formula = wage$\sim$ bs(age, df = 6) + bs(year, df = 5), data = Wage)

Demo on Wage Data

Rerun the code on Blackboard:



Demo on Wage Data

It's easier to see the nonlinearity of the relationship of wage and year by rotating the plot 90 degrees. Add the argument, “theta = 90” to the call to the “persp” function. What do you notice?