Stat 230, Spring 2013
Homework 11: Model Selection, Principal Components

Due Wednesday 4/24/13 at 11:55pm on bspace.
Note: depending on how efficient your code is the simulations may take a little while. You can do three things (that I can think of) to deal with this problem:

1. Start early so you have time if it takes you a long time.
2. Rewrite your code so it works more efficiently.
3. Do 1000 replications instead of 10,000. I'd much rather have you get some results for each part rather than precise results for only part of the assignment.

PART 1. Model Selection
The file HW11.rda has the code I used to generate the data for HW 10. Through simulation we will get an idea of how well each of the penalty methods discussed in lecture (Mallow's Cp, AIC, and BIC) performs relative to cross validation. For HW10, I used this function to generate the data with the argument `wh=20` so that the full model was actually all 20 variables, but some were very weakly correlated with the response.

You will use the arguments as given, so you have some chance of fitting noise and you may or may not end up including some of the variables with very weak relationships with the response.

Perform 10000 replications using each of the following values of n: 100, 1000, and 10000. For each n, find the "best" model using each criteria for model selection (cross validation using RSS, Mallow's Cp, AIC, and BIC). For each replication you'll want to store the value of the criterion used for each of the 20 different model sizes, so your end results should be saved in, for example, a 10,000 by 20 matrix or data frame (for each n).

For each value of n, you will make 2 plots.

1. Make a plot that has four lines with points (use `type="b"`), one for each method. The horizontal axis will be for p. The vertical axis will be for the criteria used for each method. Take the mean of the value across all replications, take the log for the CV RSS and Mallow's Cp, then divide all 4 by the value at $p = 16$ (intercept plus 15 of the 20 variables). At least they can be reasonably shown on the same plot then, and you can see how shallow/sharp the different lines are.
2. Make a plot that shows the empirical distribution of model sizes for the 10,000 replications, for each method. You can do this in a few different ways: a mosaic plot, side-by-side box plots, histograms stacked vertically, or any other visualization that you think shows any differences in the distributions as well as possble.

Each replication results in a "best" model. For each replication return the vector of coefficients, with a 0 in the correct place for each vector left out of the model. For the

vector of coefficients, calculate `mean`$(\hat{\beta}-\beta)$ to get an estimate of the bias in the coefficients. Also get `sd`(β) and a count of the number of times the coefficient was left out (for the first 15) and put in (for the last 5). Each of those is a vector of length 20 (or 21 if you include the intercept, which you don't need to). Comment briefly on differences between the four methods. You might want to summarize those vectors to make the comparison more clear, but keep in mind that most of the interesting differences are likely to show up in the last 10 vectors.

PART 2. Correlated Explanatory Variables

Rewrite the `makedata()` function so that the first 15 vectors of X have the following correlation structure. Group the 15 vectors into 5 groups of 3 (at random). Within each group of 3, each vector will have the same correlation with the other 2 vectors. For the 5 groups, those correlations will be .1, .3, .5, .7, and .9. They should still be standard normal. They can be approximate (ie $X[,2] = 0.1 * X[,1] + \text{sqrt}(1-.1^2) * X[,2]$ would work for the second vector). Recall that we did this for generating correlated errors in HW 4. Randomize the order of the 20 vectors before assigning the coefficients. Other than this change in X, the rest of the `makedata()` function can be left alone.

Repeat PART 1 using this new `makedata()` function.

PART 3. Principal Components

Using the same `makedata()` function as in PART 2, construct 20 principal components based on X. Plot the eigenvalues and observe that they get smaller. Based on visual inspection, how many would you include? (No right answer here.)

Repeat PART 1 using these principal components. You don't need to make the table for the different statistics. How does using principal components affect model size?