Homework Assignment 6

Max H. Farrell – Chicago Booth
BUS41100 Applied Regression Analysis

Due at the beginning of class of week 9

1 Flight Delays

Return to the same data we used for homework 5. See description of the data and the variables there. We are going to use the methods from class to build prediction models for *departure* delays. We will predict both the delay in minutes and the binary

Do the same data preparation you did in homework 5, up to part **(f)**. Create two binary variables, one flagging any departure delay and one flagging delays of at least 20 minutes. We have three outcomes to predict now: two binary and one continuous.

First we need to create the universe of X variables to choose from.

(a) Not all variables can be used in the model building step, for several reasons. Examine the variables `pressure`, `tailnum`, `minute`, and `date` and explain why they should be excluded from model building. Next, look for other variables that should be excluded for similar reasons.

(b) Next, not all variables should enter in the form they are given. If we put `day` into the model as-is, how will it enter? What should you do instead? Look at `plot(day,dep_delay)` to help decide. What is wrong with including `dep_time`? Do a similar exercise for other variables. *Coding tip: for plotting use a small sample to make it go faster, like 10% of the data.*

(c) When building a model to predict a delay of at least twenty minutes, explain why we should not include `dep_delay` in the universe of variables. What else should we exclude for the same reason?

Now the universe of variables is complete, we proceed to model selection. Throughout we will use forward stepwise selection based on BIC. Split the data into training and testing samples (size and sampling scheme is up to you).

(d) For all three outcomes, search over only main effects. What do you find?

(e) *This part is to illustrate that it matters how you do the model selection and you need to understand what the computer is really doing.*

We are going to explore interaction terms next. Remember what is under your control in this process: where to start, what to search over, which direction, and how to measure the steps. There are two ways we can think about doing stepwise selection over interaction terms: (1) start with the empty model and search over all main effects and all two-way interactions, or (2) start with all the main effects and search over all two-way interactions.

For `dep_delay` only, do both of these search procedures and compare your results. Explain what you found particularly in light of what happened in part **(d)**.

Coding tip: if you have N main effects in part **(d)**, *try running the stepwise procedure for N+1 or N+2 steps by adding the argument* `steps=` *to the* `step` *function.*

(f) For all three outcomes, use stepwise selection to figure out which interaction terms to add. *Coding tip: use a small sample, like 10%, of the data to start with, so you can debug the code. Once the code is correct, start it running on the full training data and do something else for a while.*

(g) Compare the variables selected. What do you find is different between the different outcomes and why do you think that is?

(h) Compare all the models you have, for all three outcomes, in terms of their prediction performance. For each outcome, which model is best?

(i) Now, using your findings, advocate for *one* model to be used to make decisions about predicting delays. That is, pick only one way of measuring the delay and one model for that outcome.

2 Community Crime

The file `CommunityCrime.csv` has violent crime rates for 1994 communities across the US and 25 descriptive variables. Our goal is to find a good-performing parsimonious model to predict the log crime rate. The demographic variables include:

- `householdsize:` mean people per household
- `PctUnemployed:` % of people 16 and over unemployed
- `PctFam2Par:` % of families (w/ kids) having two parents
- `PctRecentImmig:` % immigrated within 3 years
- `PctHousOccup:` % of housing occupied
- `RentMedian:` rental housing – median rent
- `PctUsePubTrans:` % of people who use public transit

The rest of the 25 variables are similar, and can be interpreted from their names in the data.

First split the data into training and testing samples (size and sampling scheme is up to you).

(a) Using the training data, build a model for log crime rate by using forward stepwise selection guided by both AIC and BIC to search through all main effects.

(b) Redo **(a)** allowing for all possible interactions. What has changed?

(c) Still within the training data, use the LASSO to select a model from all main effects and interactions.

(d) Compute the BIC-based model probabilities for all the models found thus far, and the model which includes all main effects. Plot the fit for each model against the true value of log crime rate.

(e) Use the test data to compare out-of-sample MSE performance. Compare your results with what you found in **(d)**.

(f) Both BIC and LASSO rely on an assumption called *sparsity.* A regression is said to be "sparse " if a few variables matter a lot and the rest of the variables don't matter at all (for predicting Y). Formally, if we assume that we have p X variables (these p variables already include interactions, powers, etc) and that

$$\mathbb{E}[Y|X] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p,$$

then sparsity requires that most of the $\beta_k = 0$. The model selection problem occurs because we don't know which variables have coefficients of zero. If we knew that, we'd just delete all the useless variables from our data and proceed with the analysis.

But is sparsity always a reasonable assumption? This is kind of up to the user and the application. Sometimes, it makes more sense to assume that all the variables matter, but just a little bit.[1] We'll explore one technique for dealing with this situation called Ridge regression, another type of penalized regression. Ridge regression solves

$$\min \left\{ n^{-1} \sum (Y_i - \boldsymbol{X}_i' \boldsymbol{\beta})^2 + \lambda \sqrt{\sum_{j=1}^{p} |\beta_j|^2} \right\}.$$

(Compare to LASSO and the generic idea of penalized regression from the slides.) Without getting into many details, Ridge shares some of the good aspects of LASSO, but does not perform variable selection at all, it keeps all the variables in the model, but compared to the full model, the coefficients will be different.[2]

Use `cv.glmnet` to fit Ridge regression on all main effects and interactions. Compute the out-of-sample MSE and compare to **(e)**. *(You may want to consult* `?predict.cv.glmnet`*.)*

3 Inference After Model Selection

This question illustrates conceptual material, and thus it has a lot of exposition.

In class we discussed model selection tools in the context of building a high-quality prediction model, but cautioned that statistical inference (testing, confidence intervals, etc) were unreliable following model selection. Let us review why. In week 2 we showed that our uncertainty regarding b_1 as an estimator of β_1 comes from the fact that if the data were to change, so would our estimate. The standard errors we derived, and those reported by the computer, capture this uncertainty. However, when doing variable selection, an *additional* layer of uncertainty is introduced: the fact that as the data changes, the very model we select may change, above and beyond to the coefficient estimates within the model changing. This second layer of uncertainty is not at all reflected in the standard errors.

This question will explore inference after model selection and highlight some of the potential problems using Monte Carlo simulation, just like we did in week 2. The set up will be the simplest possible, just to make things easy and to illustrate the issues.

We have an outcome Y and two predictors, X_1 and X_2, in the standard multiple linear regression model:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon, \qquad \varepsilon \overset{iid}{\sim} \mathcal{N}(0, \sigma^2). \tag{1}$$

Our goal is to do inference on β_1, the coefficient on X_1, for example forming a confidence interval, *after* deciding if X_2 belongs in the model. If X_2 belongs in the model and we run the *full* model, i.e. `lm(Y ~ X_1 + X_2)`, we get an estimator b_1 that is unbiased and Normally distributed with standard error $s_{b_1,F}$ (the "F" is for "Full"); that is, in math

$$\frac{b_1 - \beta_1}{s_{b_1,F}} \approx \mathcal{N}(0,1). \tag{2}$$

[1] Another common idea is that the variables matter in groups called "factors", where each contributes just a little bit to its factor, and there are a small number of factors. Any time you hear about factor analysis, principle component analysis, linear discriminant analysis, or singular value decomposition, this is usually what's going on.

[2] If you want to build some intuition on Ridge regression, you can redo the intuitive explanation of LASSO from class. In class, we drew the analogy between the LASSO penalty and a linear budget set, we drew the "diamonds" representing the budget set. To adapt this from ridge regression, just change these diamonds to circles around the point $(0,0)$. See `https://maxhfarrell.com/bus41100/lasso_vs_ridge_regression.png` for a picture.

On the other hand, if X_2 does not belong in the model, then we will get a better estimate of β_1 by running the *restricted* model that only includes X_1, i.e. `lm(Y ~ X_1)`, and

$$\frac{b_1 - \beta_1}{s_{b_1,R}} \approx \mathcal{N}(0,1), \tag{3}$$

where, typically $s_{b_1,R} < s_{b_1,F}$, reflecting that the new estimate b_1 is more precise.

To form a confidence interval for β_1, we will perform the following steps.

(1) Somehow decide which model is better, F or R.

(2) Run the selected model, and obtain estimates b_1 and its standard error.

(3) Form the confidence interval $[b_1 \pm 2 \times s_{b_1,M}]$, where b_1 is the estimate of β_1 coming from the chosen model and $s_{b_1,M}$ is the standard error from that model (i.e. either $M = F$ or $M = R$).

We aim to answer questions like: What is the sampling distribution of the estimate b_1 from the algorithm above? Is it the same or different as the one we would expect from week 2? Why? In particular, how does the confidence interval behave?

To answer theses, you will use perform simulation studies from the model in Equation (1), using the code file `homework5-ModelSelectionMonteCarlo.R` from the course website. At the top of the code file, you will see a section where you can change the various parameters of the model:

```
## Sample size
  n <- 100
## set regression coefficients and other parameters
  beta.1 <- 2
  beta.2 <- 2
  x.cov <- 0  #the covariance between the two X variables
  sigmaSquared <- 1  #the variance of the epsilons
```

The code chooses a model based on the partial F test.

(a) Set $\beta_1 = \beta_2 = 2$ and $\mathbb{COV}(X_1, X_2) = 0.5$ and verify numerically that inference for β_1 works well even after model selection. Explain why the sampling distribution of b_1 is not affected by model selection in this case. Is the sampling distribution of b_1 given by Equation (2) or (3)?

(b) Set $\beta_1 = 2$, $\beta_2 = 1/4$, and $\mathbb{COV}(X_1, X_2) = 0$ and verify numerically that inference for β_1 works well even after model selection. Explain why the sampling distribution of b_1 is not affected by model selection in this case. Is the sampling distribution of b_1 given by Equation (2) or (3)?

(c) Set $\beta_1 = 2$, $\beta_2 = 0$, and $\mathbb{COV}(X_1, X_2) = 0.5$ and verify numerically that inference for β_1 works well even after model selection. Explain why the sampling distribution of b_1 is not affected by model selection in this case. Is the sampling distribution of b_1 given by Equation (2) or (3)?

Parts (a), (b), and (c) cover all the extreme cases of when model selection does NOT matter. In most other cases, it will affect inference to some degree. So the lesson from (a), (b), and (c) should NOT be "eh, it's mostly not an issue", but instead, "only in very particular cases can we ignore the problem."

(d) Set $\beta_1 = 2$ and $\beta_2 = 1/4$ like in **(b)**. Now run the code for $\mathbb{COV}(X_1, X_2) = 0.2, 0.5, 0.8$. What is going on? Can you explain what exactly is the distribution you see with $\mathbb{COV}(X_1, X_2) = 0.8$?

(e) Repeat part **(d)** with $n = 500$.

(f) Repeat part **(d)** with $n = 500$ and $\sigma^2 = 10$.

Part (e) makes it seem like this is just a "small sample size" problem. But that is NOT correct. It is true that the problem is a lack of information, and that a bigger n represents more information, but this neglects the "noise" part of the signal-to-noise ratio: the epsilons. For any sample size, the variance could be large enough to make this a problem, i.e. to mask the signal.

The small the signal is, the harder it is to detect, as we show next.

(g) Set $\beta_1 = 2$, $\beta_2 = 0$, and $\mathbb{COV}(X_1, X_2) = 0.5$ as in part **(c)**. Then try increase β_2 in steps of 0.05 and see what happens. Explain the pattern you find.

So far, we have only used the partial F test to select the model. The problem of inference after model selection is more general. To see this, let's try again with AIC and BIC focusing on one of the problematic cases.

(h) Set $\beta_1 = 2$, $\beta_2 = 1/4$, and $\mathbb{COV}(X_1, X_2) = 0.8$ and change the code to pick the model based on **B**IC. Using our discussion of BIC from class, explain what you find relative to part **(d)**.

(i) Set $\beta_1 = 2$, $\beta_2 = 1/4$, and $\mathbb{COV}(X_1, X_2) = 0.8$ and change the code to pick the model based on **A**IC. Explain what you find relative to parts **(d)** and **(h)**.

In this whole problem we just had two X variables. This isn't meant to be realistic; with two variables, you would just include both of them, i.e. not do model selection. The idea is to just illustrate the potential problems. You can imagine how bad the problem can be when you have hundreds or thousands of variables.

4 Furniture Sales

The data set `furniture.csv` on the class website contains data from 1992–2001 on monthly furniture `sales` (in millions of dollars). If you didn't know any context this would look like an ordinary data set (rows, columns, numbers, etc), but two things make it different. First, the *order* matters. These rows are in time order, and have time labels. What we care about, and are going to try to capture, is how sales itself evolves over time.

(a) Create a variable called `time` that simply counts the months starting with the first, consecutively to the end (i.e. January, 1992 = 1; February, 1992 = 1; . . . , December, 2001 = 120).

Plot the time series data for monthly furniture sales. This is just our usual plot with the outcome, `sales`, on the y axis and `time` on the x axis, but now `time` has extra meaning. Comment on what pattern(s) you see in the context of linear regression.

(b) Run a regression of `sales` against `time`, and add the regression line to the plot you made in part **(a)**. That is, fit the linear model

$$\texttt{sales}_i = \beta_0 + \beta_1 \times \texttt{time}_i + \varepsilon_i.$$

(c) Plot the residuals of this regression against time. Does this plot look how you expect? Explain.

(d) Again plot the residuals against time, but this time make all the points for December a different color. Also, plot the data and the regression line, but this time make all the points for December a different color. What do these two plots tell you? What would you change about the regression after seeing these plots?

(e) Let's turn to prediction. Remember that the idea behind prediction is to try to form a good guess for an outcome that you have not seen, Y_f, based on the data you have and a newly observed X_f. (In our favorite example: Y_f is the price of a house that has not yet sold, X_f would be the square footage, which can measure right now.) But here we don't have an X variable, just time. What we really want to do, then, is try to predict a future outcome.

Pretend you did not have the data for 2001. Re-run the regression above using only the data up to, and including, December 2000. Use this regression to predict furniture sales for each month of 2001. Comment on your predictions.

(f) Create an ACF plot of the series of `sales`. Given this plot, is the regression specification from part **(b)** the appropriate first step? Why or why not? What regression model do you recommend to capture this time series, and why?