IST687 – Making Predictions

The textbook's chapter on linear models ("Line Up, Please") introduces linear predictive modeling using the workhorse tool known as multiple regression. The term "multiple regression" has an odd history, dating back to an early scientific observation of a phenomenon called "regression to the mean." These days, multiple regression is just an interesting name for using a simple linear modeling technique to measuring the connection between one or more predictor variables and an outcome variable In this exercise, we are going to use an open data set to explore antelope population.

This is the first exercise of the semester where there is no sample R code to help you along. Because you have had so much practice with R by now, you can create and/or find all of the code you need to accomplish these steps:

1. Read in data from the following URL:
   http://college.cengage.com/mathematics/brase/understandable_statistics/7e/students/datasets/mlr/excel/mlr01.xls
   This URL will enable you to download the dataset into excel.

   The more general web site can be found at:
   http://college.cengage.com/mathematics/brase/understandable_statistics/7e/students/datasets/mlr/frames/frame.html

   If you view this in a spreadsheet, you will find that four columns of a small dataset. The first column shows the number of fawn in a given spring (fawn are baby Antelope). The second column shows the population of adult antelope, the third shows the annual precipitation that year, and finally, the last column shows how bad the winter was during that year.
2. You have the option of saving the file save this file to your computer and read it into R, or reading the data directly from the web into a data frame.
3. You should inspect the data using the str() command to make sure that all of the cases have been read in (n=8 years of observations) and that there are four variables.
4. Create bivariate plots of number of baby fawns versus adult antelope population, the precipitation that year, and the severity of the winter. Your code should produce three separate plots. Make sure the Y-axis and X-axis are labeled. Keeping in mind that the number of fawns is the outcome (or dependent) variable, which axis should it go on in your plots?
5. Next, create three regression models of increasing complexity using lm(). In the first model, predict the number of fawns from the severity of the winter. In the second model, predict the number of fawns from two variables (one should be the severity of the winter). In the third model predict the number of fawns from the three other variables. Which model works best? Which of the predictors are statistically significant in each model? If you wanted to create the most parsimonious model (i.e., the one that did the best job with the fewest predictors), what would it contain?

Learning Goals for this activity:

A. Develop skills for manipulating and transforming data that contains missing values.
B. Understand the application of multiple linear regression to simple situations of predicting one numeric variable from one or more other numeric variables.
C. Practice plotting skills.
D. Build debugging skills.
E. Increase familiarity with bringing external data sets into R.
F. Increase familiarity with sources of advice and ideas on R source code.

Essential Guide for All IST687 Activities (appears at the end of all activity guides)

1. All IST687 activities work on what some people call a "constructivist learning" model. By developing a product on your own, testing it to find flaws, improving it, and comparing your solution to the solutions of other people, you can obtain a deeper understanding of a problem, the tools that might solve that problem, and a range of solutions that those tools may facilitate. The constructivist model only works to the extent that the student/learner has the drive to explore a problem, be frustrated, fail, try again, possibly fail again, and finally push through to a satisfactory level of understanding.
2. Each IST687 activity builds on skills and knowledge developed in the previous activities, so your success across the span of the course depends at each stage on your investment in earlier stages. Take the time to experiment, play, try new things, practice, improve, and learn as much as possible. These investments will pay off later.
3. Using the expertise of others, the Internet, and other sources of information is not only acceptable - it is expected. You must ***always, always, always*** give credit to your sources. For example, if you find a chunk of code from r-bloggers.com that helps you with developing a solution, by all means borrow that chunk of code, but make sure to use a comment in your code to document the source of the borrowed code chunk. The discussion boards in the learning management system have been setup to encourage appropriate sharing of knowledge and wisdom among peers. Feel free to ask a question or pose a solution on these boards.
4. Building on the previous point, when submitting code as your solution to the activity, the comments matter at least as much, if not more than the code itself. A good rule of thumb is that every line of code should have a comment, and every meaningful block of code should be preceded by a comment block that is just about as long as the code itself. As noted above, you can use comments to give proper credit to your sources and you can use comments to identify your submission as your own.
5. Here's a bonus area of constructivist learning that has not appeared in previous exercises for this class: Sometimes the building process reveals unexpected results that are themselves very informative in learning. When you completed the exercise above, what did you find that was unexpected? What did you do about trying to understand what had happened? Did you do further exploration? What did that further exploration reveal?