TO: Professor Jeff Chen, Dan Hammer
FROM: Justin Goss, Yixuan Huang, Nghia-Piotr Le
RE: Data Science Final Project Proposal
DATE: 3/29/2017

Summary: Create "regressr" an R library dedicated to helping user build regression models, providing text output to ease interpretation of the model's results, and optimizing model specifications. The purpose of this library is to make the process of running regression modeling and coefficient interpretation easier for those unfamiliar with regressions.

Descriptions of Functions

Model Builder

Function 1:

Users input a dataset. The function returns with a table of variable description and summary statistics, with instructions on choosing a dependent variable and a set of independent variable(s). The function also returns instruction on choosing dependent and independent variables for regression, using accessible language for those unfamiliar with regressions.

Function 2:

Users input a dataset and dependent variable. The function checks dependent variable for class and recommends an initial regression technique. For instance, if the dependent variable is binary, the function recommends a logit or probit regression as opposed to recommending OLS for a continuous dependent variable.

Model Interpretation

Function 3:

Users input a data set and a formula of regression. The function runs the regression and outputs the coefficients of predictors and the model's diagnostics, along with text interpretation for each parameters and explanation of the model's various diagnostics. The interpretation will be based on model specifications -- log-log, lin-log, quadratics, VARs, etc. For instance, the function could describe what a coefficient means in terms of the user's independent and dependent variables, for example: "a one unit change in x correlates with a B unit change in y." Other text outputs could explain the meaning of a p-value and statistical significance, the adjusted R-squared term or the cumulative significance of the model.

Model Optimization

Function 4:

Users input a dependent variable and a set of potential independent variables, and specify a model diagnostic parameter. Function checks combinations of independent variables looking for those with the best model fit, such as the lowest error rate or highest adjusted R-squared, and outputs the top N number of model (user specified or default). The function also allows users to input options including what variables should be included in all specifications, whether to include quadratic terms, etc.

[Additional Thoughts for the Library]

A ShinyApp that completes the tasks above:

1. Users upload .csv files.
2. With instructions given, users choose a dependent variable and a set of independent variables, using drop-down menu/ check boxes that are generated from the data file.
3. The app outputs variable coefficients and model diagnostics, with interpretations.
4. Users choose a set of potential independent variables, and a model diagnostic parameter, the app checks combinations of independent variables looking for those with the best model fit and return the model.