Group Members: Kevin Caton-Largent, Jason Delossantos

Data set: Ozone

An initial problem we encountered, when dealing with the ozone data set, was there were missing values. So we had to first convert these missing values into numeric values. We then replaced the values with the median of the good values from that row. We did this to ensure that the values had consistency with the rest of the data. We also found the “date” column to be irrelevant in predicting ground-level Ozone days so the date values were omitted by using “x [,-1]” in the code. We used the variable “xn” to store the “colnames” and the variable “nrows” to store the number of rows for further use. The libraries used were “MASS” and “leaps.”

The “MASS” library contains a function called “addterm()” that uses single-term step-wise regression where it adds one term each time from a base model up to the limit model. In order to setup the “addterm()” function we first needed to create a base formula which is “Class ~1.” The limit model used is a model containing all 72 predictor variables added to the response variable. The function then tested starting from the base to the limit model which is called “full.” It used the chi-square test for testing the model as terms were being added to it. We used the chi-square test because we found that it was the only applicable test given from the “addterm().” After the function was done analyzing the predictor variables, we stored the indexes in the variable “z” using “which(smAnalysis[5]<0.001)-1.” The number of predictors that “addterm()” found to be most predictive is 62 out of the 72 total. This is a fairly large number of predictors so to reduce the amount of predictors we used the leaps “regsubsets” function. In order to setup the leaps function we first had to make a subset of the data from the 62 predictors we found using step-modeling.

The “regsubsets” function finds the predictors through best subsets regression. In the case of our program we used “nbest = 1” and “nvmax = 5” which goes from 1-5 sized subsets and records the best subset of each size. The top choices of predictors from the 62 predictors were “WSR6, WSR10, WSR12, T4, T5, T6, T7, T15, T23, T_PK, and T_AV.” After these two models were created we used “leave-one-out cross-validation” to check the results of the step-modeling and the reduced model.

The percentage accuracy of the first model was 93.32807%. The reduced model’s percentage accuracy was 94.15713%. The increase in accuracy achieved by the reduced model is approximately 0.82906%. This increase is not much, but it is from a significantly reduced model, which was from 62 to 11 predictors, and there was also a reduction of 48 false positives to only 19 false positives.

From the results it seems that adding terms to the model did not increase its accuracy when compared to the reduced model having only 11 predictor variables and a cross-validation accuracy of about 94.15713%.