Stat 5630 Mini Project: Spam Email Detection

This Project is due on 9pm, Monday, November 4th, 2019. A paper copy of the well-written report should be submitted to my mailbox (Xiwei Tang), which can be found in the first floor of Halsey Hall, the corner where you make a left turn after entering the front door. An electronic copy of your code should be submitted on Collab. In the paper report, the results should be clearly stated with some reasonable explanations (including necessary plots). **Please also copy your codes at the end of the paper report.** Please DO NOT just copy and paste the raw outputs obtained from your software. Also please write down both your full name and the computing ID at the first page.

1. Please analyze the email spam dataset using different classification approaches. The dataset consists of two parts : a training dataset with 3065 obs and 58 variables, a testing data set with 1536 obs and 58 variables. In each dataset, the first 57 columns store the predictors, and the last column stores the binary response variable (spam=1, not spam=0). The datasets are in .txt files attached as well. You might use following codes to read the data into R.

```r
setwd("path of the folder where you put the data files")
train=read.table("traindata.txt")
test=read.table("testdata.txt")
```

Exploratory Data Analysis

(a) Briefly summarize the response variable (e.g., proportion of each class of observations, spam vs not spam)

(b) Briefly summarize the predictors (e.g., how many of them are continuous/categorical).

Modeling and Data Analysis

For a classification problem, we usually have following measurements to evaluate the model performance:

classification accuracy =num of obs that are correctly classified / total num of obs;

sensitivity =num of spam obs which are correctly classified as spam/ num of totoal spam obs

specificity =num of non-spam obs which are correctly classified as non-spam/ num of totoal non-spam obs

(a) Perform LDA and QDA with the predictors V55-V57 (columns 55-57), and apply your trained model on the testing set. Report the corresponding **accuracy, sensitivity and specificity** for testing set predictions.

(b) Perform LDA and QDA with all predictors V1-V57 (columns1-57), and apply your trained model on the testing set. Report the corresponding **accuracy, sensitivity and specificity** for testing set predictions. Comparing your results with those obtained in part (a).

(c) Perform a logistic regression model and a linear SVM model with all predictors V1-V57 (columns1-57), and apply your trained model on the testing set. Please report the corresponding **accuracy, sensitivity and specificity**, respectively, for testing set predictions.

(d) Try a non-linear SVM model with predictors V55-V57 (columns 55-57), and apply your trained model on the testing set. Please report the corresponding **accuracy, sensitivity and specificity**, respectively, for testing set predictions. Compare your results to those based on a linear SVM model above.

(e) Suppose we just make a random prediction on the testing set, that is, for each obs in the testing set, we randomly toss a coin, and assign the label 1 to this obs if we get a ”head” while assign 0 if get a ”tail”. Calculate the corresponding classification accuracy, sensitivity and specificity. You might generate the prediction on the testing set by using the following codes to generate random binary (Bernoulli) outcomes:

```
# n is the sample size of the testing set, and let prob=0.5
y.test.prediction=rbinom(n, 1, prob)
```

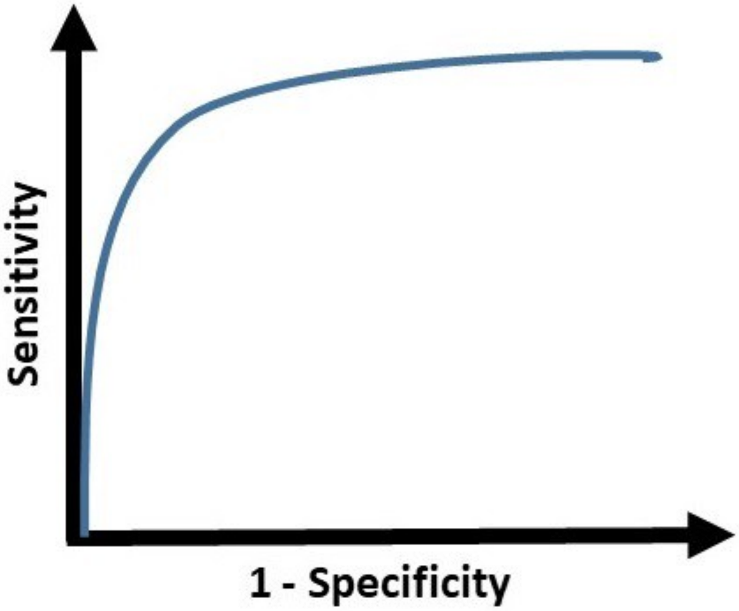
(f) Following the procedure in part (d), we try a sequence of values of ”prob”: $(0, 0.1, 0.2, \dots, 0.9, 1)$. Given each ”prob” value, you should obtain a set of (accuracy, sensitivity, specificity) values. Please generate a plot showing the values of (1-specifcity) (as X axes) v.s. sensitivity (Y axes), which usually refers to the ROC curve. Based on this exploration, can you select the best ”prob” in a sense that it optimizes the value of (sensitivity + specificity)?

(g) Run a random-forest model on the spam data and report the overall accuracy, sensitivity and specificity on the testing predictions. Use a five-folder cross-validation to perform the tuning regarding the notesize. For the other two important parameters, we set total number of trees to be 500 and set the number of selected variables at each split as $mtry = \sqrt{p} \approx 7$.

(h) In the above random-forest model, we also want to check the importance of each covariate regarding the classification. Please take advantage of the ”importance” argument in the ”randomForest” function provided in the sample code and make a barplot of the MeanDecreaseGini values with respect to all covariates. (See Sample Code!)

(i) Run a boosting model with the exponential loss (using "adaboost" for the argument "distribution" in the "gbm" function). Tune the number of trees and the shrinkage factor by five-folder cross-validation. The number of trees is tuned over (100, 200, 500, 1000, 2000), and the shrinkage factor is tuned over 0.01 to 0.1 with a step size 0.01. Report the selected tuning parameter values and the corresponding prediction accuracy, sensitivity and specificity on the testing set.

2. **Bonus**: Suppose there is a ROC curve (1-specificity v.s. sensitivity) for a binary classifier.



(a) At which point of the curve, it maximizes sensitivity + specificity?

(b) What is the lower bound of the area under the curve (AUC)?