Classification Trees

In this assignment you will complete a variety of tasks related to binary classification with classification trees. The dataset that we will be using is related to criminal justice and deals specifically with parole violations.

Deliverable: All of your work for this assignment should be done in an R Markdown document. Knit your document into a Word file and submit the Word file as the deliverable for this assignment.

Libraries: For this assignment you will need the following libraries: tidyverse, caret, rpart, rattle, and RColorBrewer.

Before beginning the assignment tasks, you should read-in the data for the assignment into a data frame called parole. **Carefully** convert the male, race, state, crime, multiple.offenses, and violator variables to factors. Recode (rename) the factor levels of each of these variables according to the description of the variables provided in the ParoleData.txt file (located with the assignment on Canvas).
Note: You did this in a previous assignment. I would encourage you to re-use your code.

Task 1: Split the data into training and testing sets. Your training set should have 70% of the data. Use a random number (set.seed) of 12345.

Task 2: Create a classification tree using all of the predictor variables to predict "violator" in the training set. Plot the tree.

Task 3: For the tree created in Task 2, how would you classify a 40 year-old parolee from Louisiana who served a 5 year prison sentence? Describe how you "walk through" the classification tree to arrive at your answer.

Task 4: Use the printcp function to evaluate tree performance as a function of the complexity parameter (cp). What cp value should be selected? Note that the printcp table tends to be a more reliable tool than the plot of cp.

Task 5: Prune the tree from Task 2 back to the cp value that you selected in Task 4. **Do not attempt to plot the tree.** You will find that the resulting tree is known as a "root". A tree that takes the form of a root is essentially a naive model that assumes that the prediction for all observations is the majority class. Which class (category) in the training set is the majority class (i.e., has the most observations)?

Task 6: Use the unpruned tree from Task 2 to develop predictions for the training data. Use caret's confusionMatrix function to calculate the accuracy, specificity, and sensitivty of this tree on the training data. Note that we would not, in practice, use an unpruned tree as such a tree is very likely to overfit on new data.

Task 7: Use the unpruned tree from Task 2 to develop predictions for the testing data. Use caret's confusionMatrix function to calculate the accuracy, specificity, and sensitivty of this tree on the testing data. Comment on the quality of the model.

Task 8: Read in the "Blood.csv" dataset. The dataset contains five variables:
Mnths_Since_Last: Months since last donation
TotalDonations: Total number of donation
Total_Donated: Total amount of blood donated
Mnths_Since_First: Months since first donation
DonatedMarch: Binary variable representing whether he/she donated blood in March (1 = Yes, 0 = No)

Convert the DonatedMarch variable to a factor and recode the variable so 0 = "No" and 1 = "Yes".

Task 9: Split the dataset into training (70%) and testing (30%) sets. **You may wish to name your training and testing sets "train2" and "test2" so as to not confuse them with the parole datsets** Use set.seed of 1234. Then develop a classification tree on the training set to predict "DonatedMarch". Evaluate the complexity parameter (cp) selection for this model.

Task 10: Prune the tree back to the optimal cp value, make predictions, and use the confusionMatrix function on the both training and testing sets. Comment on the quality of the predictions.