Please complete the assigned problems to the best of your abilities. Ensure that the work you do is entirely your own, external resources are only used as permitted by the instructor, and all allowed sources are given proper credit for non-original content.

1 Recitation Exercises

These exercises are to be found in: **Introduction to Statistical Learning, 2nd Edition (Online Edition)** by *Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani.*

1.1 Chapter 8

Exercises: 1,3,4,5

1.2 Chapter 9

Exercises: 1,2,3

2 Practicum Problems

These problems will primarily reference the *lecture materials and the examples given in class* using **R** and **CRAN**. It is suggested that a *RStudio* session be used for the programmatic components.

2.1 Problem 1

Simulate a binary classification dataset with a single feature via a mixture of normal distributions using **R** (**Hint**: Generate two data frames with the random number and a class label, and combine them together). The normal distribution parameters (using the function **rnorm**) should be (5,2) and (-5,2) for the pair of samples - you can determine an appropriate number of samples. Induce a binary decision tree (using **rpart**), and obtain the threshold value for the feature in the first split. How does this value compare to the empirical distribution of the feature? How many nodes does this tree have? What is the entropy and Gini at each? Repeat with normal distributions of (1,2) and (-1,2). How many nodes does this tree have? Why? Prune this tree (using **rpart.prune**) with a complexity parameter of *0.1*. Describe the resulting tree.

2.2 Problem 2

Load the *Wine Quality* sample dataset from the UCI Machine Learning Repository (**winequality-red.csv** and **winequality-white.csv**) into **R** using a dataframe. Create an *80/20* test-train split of each wine dataframe, and use the **rpart** package to induce a decision tree of both the red and white wines, targeting

the *quality* output variable. Visualize the tree using the *rpart.plot* library, and use the **caret** package *confusionMatrix* method to determine the decision tree accuracy on the test set. Compare the decision trees for red and white wine - what differences in terms of tree structure and variables of interest can be noted? Use the **randomForest** package to repeat the fit with a random forest tree model, and compare the resulting test accuracy against the original single tree model.

2.3 Problem 3

Load the *SMS Spam Collection* sample dataset from the UCI Machine Learning Repository (**smsspamcollection.zip**) into **R** using a dataframe (**Note**: The column names will need to be set manually). Use the **tm** package to create a *Corpus* of documents (**Hint**: Construct the corpus using a *VectorSource* of the text column). Apply the following transformations from the **tm** package to the corpus in order prepare the data: a) Convert lowercase, b) Remove stopwords, c) Strip whitespace, and d) Remove punctuation. Use *findFreqTerms* to contruct features from words occuring more than 10 times and proceed to split the data into a training and test set - for each create a *DocumentTermMatrix*. Finally convert the *DocumentTermMatrix* train/test matrices to a Boolean representation (counts greater than zero are converted to a 1) and fit a SVM using the **e1071** package. Report your training and test set accuracy.