**Reminder**: All homework solutions must be written up independently, even though you are allowed to discuss with other students. You need to save your homework assignment in a pdf/html format and upload it with the R code (.R or .rmd) into the Canvas before 11:59pm CT on the due day. No late homework assignment will be graded in any circumstance.

**Problem 1(25 points)**: In Homework 1, Problem 3, we described a data set which contained 96 oil samples each from one of seven types of oils (pumpkin, sunflower, peanut, olive, soybean, rapeseed, and corn). Gas chromatography was performed on each sample and the percentage of each type of 7 fatty acids was determined. We would like to use these data to build a model that predicts the type of oil based on a sample’s fatty acid percentages. These data can be found in the **caret** package using data(oil). The oil types are contained in a factor variable called oilType. The types are pumpkin (coded as A), sunflower (B), peanut (C), olive (D), soybean (E), rapeseed (F) and corn (G). In R,

|  |
| --- |
| >library(caret)  > data(oil)  > str(oilType)  Factor w/ 7 levels "A","B","C","D",..: 1 1 1 1 1 1 1 1 1 1 ...  > table(oilType)  oilType  A B C D E F G  37 26 3 7 11 10 2  > |

1. Given the classification imbalance in oil Type, describe how you would create a training and testing set.
2. Which classification statistic would you choose to optimize for this problem and why?
3. Split the data into a training and a testing set, pre-process the data, and build models and tune them via resampling described in Chapter 12. Cleary list the models under consideration and the corresponding tuning parameters of the models.
4. Of the models presented in this chapter, which performs best on these data? Which oil type does the model most accurately predict? Least accurately predict?

**Problem 2 (25 points)**: Use the fatty acid data from **Problem 1** above.

1. Use the same data splitting approach (if any) and pre-processing steps that you did Problem 1. Using the same classification statistic as before, build models described in Chapter 13: Nonlinear Classification Models for these data. Which model has the best predictive ability? How does this optimal model’s performance compare to the best linear model’s performance?
2. Would you infer that the data have nonlinear separation boundaries based on this comparison?
3. Which oil type does the optimal model most accurately predict? Least accurately predict?

**Problem 3 (25 points)**: The “churn” data set was developed to predict telecom customer churn based on information about their account. The data files state that the data are “artificial based on claims similar to real world.” The data consist of 19 predictors related to the customer account, such as the number of customer service calls, the area code, and the number of minutes. The outcome is whether the customer churned:

1. Start R and use these commands to load the data

|  |
| --- |
| > library(modeldata)  > data(mlc\_churn)  > str(mlc\_churn)  tibble [5,000 × 20] (S3: tbl\_df/tbl/data.frame)  $ state : Factor w/ 51 levels "AK","AL","AR",..: 17 36 32 36 37 2 20 25 19 50 ...  $ account\_length : int [1:5000] 128 107 137 84 75 118 121 147 117 141 ...  $ area\_code : Factor w/ 3 levels "area\_code\_408",..: 2 2 2 1 2 3 3 2 1 2 ...  $ international\_plan : Factor w/ 2 levels "no","yes": 1 1 1 2 2 2 1 2 1 2 ...  $ voice\_mail\_plan : Factor w/ 2 levels "no","yes": 2 2 1 1 1 1 2 1 1 2 ...  $ number\_vmail\_messages : int [1:5000] 25 26 0 0 0 0 24 0 0 37 ...  $ total\_day\_minutes : num [1:5000] 265 162 243 299 167 ...  $ total\_day\_calls : int [1:5000] 110 123 114 71 113 98 88 79 97 84 ...  $ total\_day\_charge : num [1:5000] 45.1 27.5 41.4 50.9 28.3 ...  $ total\_eve\_minutes : num [1:5000] 197.4 195.5 121.2 61.9 148.3 ...  $ total\_eve\_calls : int [1:5000] 99 103 110 88 122 101 108 94 80 111 ...  $ total\_eve\_charge : num [1:5000] 16.78 16.62 10.3 5.26 12.61 ...  $ total\_night\_minutes : num [1:5000] 245 254 163 197 187 ...  $ total\_night\_calls : int [1:5000] 91 103 104 89 121 118 118 96 90 97 ...  $ total\_night\_charge : num [1:5000] 11.01 11.45 7.32 8.86 8.41 ...  $ total\_intl\_minutes : num [1:5000] 10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...  $ total\_intl\_calls : int [1:5000] 3 3 5 7 3 6 7 6 4 5 ...  $ total\_intl\_charge : num [1:5000] 2.7 3.7 3.29 1.78 2.73 1.7 2.03 1.92 2.35 3.02 ...  $ number\_customer\_service\_calls: int [1:5000] 1 1 0 2 3 0 3 0 1 0 ...  $ churn  *>* |

1. Explore the data by visualizing the relationship between the predictors and the outcome. Are there important features of the predictor data themselves, such as between-predictor correlations or degenerate distributions? Can functions of more than one predictor be used to model the data more effectively?
2. Split the data into a training and a testing set, pre-process the data if appropriate.
3. Try building other models discussed in this chapter. Do any have better predictive performance?

**Problem 4 (25 points)**: Use the fatty acid data from **Problem 3** above.

1. Use the same data splitting approach (if any) and pre-processing steps that you did in Problem 3.
2. Fit a few basic trees to the training set.
3. Does bagging improve the performance of the trees? What about boosting?
4. Which model has better performance, and what are the corresponding tuning parameters?