**Data Science for Social Scientists**

Psyc 546, Spring 2023

Homework Assignment 10 (LAST ONE OF THE SEMESTER!!!)

**Due Date**: April 19th (by 8:15 PM)

**Reminder**: See the assigned readings, resources on Canvas, and the lecture slides for a tutorial on how to use R to perform the various functions included in the homework assignment below. **Once completed, you should submit a completed version of this document and your final R script file to the Homework Assignment 10 – Submission Portal on Canvas**.

Your submitted R script file should contain code to answer the questions below. Please use comments (e.g., #Question 1) to label the code for each question.

1. Q1 uses the **loan.csv** from Canvas. You will use the holdout method of cross validation. First, set the seed of the R environment to some integer between 500 and 1000. Then, split the loan data file into 60%/40%. Have the 60% applied to a train object; have the 40% applied to a test object.

Next, perform a logistic regression model on the training data with previous default (default) as the DV and the following predictors: income (income), debt-to-income ratio (debtinc), credit card debt (creddebt), and other debt (othdebt). Then, use the fitted model to provide predicted default (rounded to 0 and 1) for the participants in the testing data. Finally, use confusionMatrix() to calculate the following values on the test data: [2 points overall]

1. The accuracy of the model: 0.7484
2. The sensitivity of the model: 0.7818
3. The specificity of the model: 0.5349
4. The kappa of the model: 0.2275
5. Q2 will use repeated k-fold cross validation with the model from Q1. First, set the seed of the R environment to an integer above 1000. Then, use the trainControl() function to choose repeated k-fold CV with 5 folds and 10 repetitions. Finally, use the train() function to run the model from Q1. Report the following values from the results of the CV: [2 points overall]
6. The accuracy: 0.7591507
7. The kappa: 0.2397547
8. Q3 will use leave-one-out cross validation with a predictive/regression model. It will use the **patient\_hw10.sav** file on Canvas.First, set the seed of the R environment to an integer above 5000. Then, use the trainControl() function to choose leave-one-out CV. Finally, use the train() function to run a model with treatment cost (cost) as the DV and the following predictors: age of the patient (age), history of diabetes (diabetes), and cholesterol level (choles). Because it is a predictive/regression model, make sure the DV is treated numerically. Report the following values from the results of the CV: [2 points overall]
9. Root Mean Squared Error: 16.0263
10. R2: 0.1149686
11. Q4 uses our favorite **survey.csv** data file from Canvas. This question focuses on the difference in ability of traditional linear regression vs. local regression (LOESS) in predicting an outcome in a single sample of data. The model will consist of predicting life satisfaction (Mlifesat) with the following predictors: positive affect (Mposaff), negative affect (Mnegaff), stress (Mpstress), and age in years (age).

First, run the model as a traditional linear regression model (i.e., using the lm() function). Then, run the model using a loess function. Next, calculate predicted scores for these two models (i.e., one column of predicted scores using the linear regression model and one column of predicted scores using the loess model). Finally, calculate the R2 for these two models. Remember, one way to calculate R2 is to square the Pearson correlation between a model’s predicted scores on the DV and the actual scores on the DV. Provide the answers below. [2 points]

1. R2 for linear regression model: 0.2944
2. R2 for loess model: 0.3674529
3. Q5 uses the holdout method split that you performed for Q1 on the loan.csv data file. It utilizes the k-nearest neighbors method. Specifically, you will use the knn3() function to run the classification model from Q1—where previous default (default) is the DV and income (income), debt-to-income ratio (debtinc), credit card debt (creddebt), and other debt (othdebt) are the predictors.

Imagine that you are interested in manually tuning the hyperparameter k to see the impact on the generalizability of the model. As a result, you decide to run the model on the training data from Q1 three times: one time with the k parameter set to 1, then set to 10, and finally set to 20. In each instance, use the model to predict classes on the test data. Then use the confusionMatrix() function to report the following classification metrics for each of the three model runs. [2 points overall]

|  |  |  |  |
| --- | --- | --- | --- |
|  | k = 1 | k = 10 | k = 20 |
| Accuracy | 0.7264 | 0.7358 | 0.7327 |
| Kappa | 0.288 | 0.1736 | 0.1093 |