FUNDAMENTALS OF DATA SCIENCE – DSC 441

HOMEWORK 4

Turn your assignment in as a PDF file to the D2L submission folder by the due date associated with that folder (unless otherwise specified explicitly by the Instructor). Late assignments will incur a penalty. Please ask if you need an extension and come to office hours if you need assistance.

Academic integrity is key to maintaining the impact of your degree. You may discuss the problems and methods with others, but everything you submit must be your own work. For violating, you may incur significant sanctions including failing the course or penalties from the University. If in doubt, ask first.

To get credit, you must clearly label your responses with the questions they answer. Include results and visualizations that you generate, and clearly written answers to any questions included in the problem. Your code file must be submitted alongside your document in D2L.

This homework assignment covers material from Modules 7 and 8. You will use kNN classification and compare it to other techniques, and then move on to clustering. In applying k-means and hierarchical agglomerative clustering (HAC), you will practice choosing the best parameters, including number of clusters, distance functions and linkages. You will also use what you have learned to compare performance of these clustering algorithms.

Problem 1 (25 points):

For this problem, you will tune and apply kNN and compare it to other classifiers. We will use the wine quality data, which has a number of measurements about chemical components in wine, plus a quality rating. There are separate files for red and white wines, so the first step is some data preparation.

a. Load the two provided wine quality datasets and prepare them by (1) ensuring that all the variables have the right type (e.g., what is numeric vs. factor), (2) adding a *type* column to each that indicates if it is red or white wine and (2) merging the two tables together into one table (hint: try *full_join()*). You now have one table that contains the data on red and white wine, with a column that tells if the wine was from the *red* or *white* set (the *type* column you made).
b. Use PCA to create a projection of the data to 2D and show a scatterplot with color showing the wine *type.*

c. We are going to try kNN, SVM and decision trees on this data. Based on the ‘shape’ of the data in the visualization from (b), which do you think will do best and why?

d. Use kNN (tune k), use decision trees (basic *rpart* method is fine), and SVM (tune C) to predict *type* from the rest of the variables. Compare the accuracy values – is this what you expected? Can you explain it?

   Note: you will need to fix the columns names for rpart because it is not able to handle the underscores. This code will do the trick (assuming you called your data wine_quality):

   ```
   colnames(wine_quality) <- make.names(colnames(wine_quality))
   ```

e. Use the same already computed PCA again to show a scatter plot of the data and to visualize the labels for kNN, decision tree and SVM. Note that you do not need to recreate the PCA projection, you have already done this in 1b. Here, you just make a new visualization for each classifier using its labels for color (same points but change the color). Map the color results to the classifier, that is use the “predict” function to predict the class of your data, add it to your data frame and use it as a color. This is done for KNN in the tutorial, it should be similar for the others. Consider and explain the differences in how these classifiers performed.

Problem 2 (15 points):

In this question we will use the *Sacramento* data, which covers available housing in the region of that city. The variables include numerical information about the size of the housing and its price, as well as categorical information like zip code (there are a large but limited number in the area), and the type of unit (condo vs house (coded as residential)).

a. Load the data from the *tidyverse* library with the `data(“Sacramento”)` command and you should have a variable *Sacramento*. Because we have categoricals, convert them to dummy variables.

b. With kNN, because of the high dimensionality, which might be a good choice for the distance function?

c. Use kNN to classify this data with *type* as the label. Tune the choice of *k* plus the type of distance function. Report your results – what values for these parameters were tried, which were chosen, and how did they perform with accuracy?

Problem 3 (25 points):

In this problem we will continue with the wine quality data from Problem 1, but this time we will use clustering. Do not forget to remove the *type* variable before clustering because that would be cheating by using the label to perform clustering.

a. Use k-means to cluster the data. Show your usage of silhouette and the elbow method to pick the best number of clusters. Make sure it is using multiple restarts.

b. Use hierarchical agglomerative clustering (HAC) to cluster the data. Try at least 2 distance functions and at least 2 linkage functions (cluster distance functions), for a total of 4 parameter combinations. For each parameter combination, perform the clustering.

c. Compare the k-means and HAC clusterings by creating a crosstabulation between their labels.

d. For comparison – use PCA to visualize the data in a scatterplot. Create 3 separate plots: use the color of the points to show (1) the *type* label, (2) the k-means cluster labels and (3) the HAC cluster labels.

e. Consider the results of C and D and explain the differences between the clustering results in terms of how the algorithms work.

Problem 4 (20 points)

Back to the Starwars data from a previous assignment! Remember that the variable that lists the actual names and the variables that are actually lists will be a problem, so remove them (*name, films, vehicles, starships*). Make sure to double check the types of the variables, i.e., that they are numerical or factors as you expect.

a. Use hierarchical agglomerative clustering to cluster the Starwars data. This time we can leave the categorical variables in place, because we will use the *gower* metric from *daisy* in the *cluster* library to get the distances. Use *average* linkage. Determine the best number of clusters.

b. Produce the dendogram for (a). How might an anomaly show up in a dendogram? Do you see a Starwars character who does not seem to fit in easily? What is the advantage of considering anomalies this way as opposed to looking for unusual values relative to the mean and standard deviations, as we considered earlier in the course? Disadvantages?

c. Use dummy variables to make this data fully numeric and then use k-means to cluster. Choose the best number of clusters.

d. Compare the HAC and k-means clusterings with a crosstabulation.