Project Assignment 2 ANLY 501 Fall 2018 (COSC 587 for CS Students)
Due: November 2 at 11:59 pm

PROCEDURES AND LATE POLICY REMINDER

- **Turn-in:** Please turn in your project through Box. Use one zip folder that contains all project elements. A link will be provided. Please make sure the zip file name is netids of all the students.
- **Deadline**: The on-time deadline for all students is 11:59 pm on the due date.
- **Late policy**: All written work is to be turned in at 11:59 pm on the day that it is due. Written work turned in after the deadline will be accepted but penalized 50% per day. Once an assignment has been returned, a late assignment will not be accepted.

Overview

This project asks you to use the data you collected and cleaned in Project1, and analyze it. You will explore it, and will begin developing support for hypotheses. Five analyses requirements are listed below. In addition to conducting each analysis, you should also explain what each analysis means (provide an interpretation). The interpretation will be in the Project report. All the analyses must be conducted in Python3.

Code examples for different parts are part of your class notes/slides, etc. Please refer to those to get additional help on this project.

Exploratory Analysis

Basic Statistical Analysis and data cleaning insight (15%)

- Determine the mean (mode if categorical), median, and standard deviation of at least 10 attributes in your data sets. Use Python to generate these results and use the project report to show and explain each.

- In the last assignment you took several steps to clean your data. Here you need to check to make sure that the cleaning decisions you made make sense for the analysis you will do in this assignment. To do this, consider your raw data and consider your current cleaned data. Next, do the following:
    - Identify any attributes that may contain outliers. If you did not deal with outliers in Project 1, do this now by writing Python3 code to locate and potentially clean outliers. In your report, note the attributes that contained potential outliers (you do not have to list all the outliers themselves)
    - Explain how you detected the outliers, and how you made the decision to keep or remove them.
    - From the cleaning phase of Project 1, also discuss which attributes had missing values and explain your strategy for handling them.
    - If you find that you data needs to be further cleaned or differently cleaned based on analyses, include explanations here. Be specific about what you did and why.

- For at least one of the numeric variables in one of the datasets, write code to bin the data. This will create a new column. Use the binning strategy that is most intuitive for your data. Explain your decision. Include why you chose to bin the specific attribute selected, the binning method used, and why that method makes sense for your data.

Additional Part for CS Students (5%)

- Use LOF to identify outliers in your data set. Try 3 different values for k. Do you have multi-dimensional outliers - explain your findings

Histograms and Correlations (10%)

- Use a histogram to plot at least three (3) of the variables (attributes) in either dataset. Discuss the insight generated by the histograms. What do they show or suggest?
- Identify three (3) quantitative variables from either data set. Find the correlation between all the pairs of these quantity variables. Include a table of the output in your report, and explain your findings – what does this indicate about your data? Use scatterplots to display the results. Ideally, create a set of scatterplot subplots.

Cluster Analysis (20%)

- Conduct three (3) cluster analyses on your data. Include a hierarchical clustering method (such as Ward), a partition clustering method (use k-means), and third, use the dbscan clustering analysis on your data. Explain your findings.
- Use the Calinski-Harabaz procedure to assess the quality of the clusters. Ref: http://scikit-learn.org/stable/modules/clustering.html
- Plot the clusters or if the dimensionality is too high, plot a PCA projection of the clusters. Does the plot give you additional insight about the clustering – explain.

REF: http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html
REF: http://scikit-learn.org/stable/auto_examples/decomposition/plot_pca_iris.html

Association Rules / Frequent Itemset Mining Analysis (15%)

- Use Python3 to code and run association rule mining on a subset of your data. Use at least 3 different support levels. Use the Apriori algorithm. Recall that the support (sup) and the confidence (conf) can be calculated. What patterns are most frequent? Is this surprising? Explain your findings.

Resources: Review Week 5 slides and the following resource.

REF: Kumar Chapter 6 Association Analysis

http://www-users.cs.umn.edu/~kumar/dmbook/ch6.pdf

Add the python package and run Apriori –
Two packages that are available: https://pypi.python.org/pypi/apyori/1.1.1 and http://www.borgelt.net/pyfim.html

Predictive Analysis – Part 1

Hypothesis Testing (30%)

- This component contains both the traditional statistical hypothesis testing, and the beginning of machine learning predictive analytics. Here you will write three (3) hypotheses and see whether or not they are supported by your data. **You must use all of the methods listed below (at least once) on your data.**
- You do not need to try all the methods for each hypothesis. For example, you might use ANOVA for one of your hypotheses, and you might use a t-test and linear regression for another, etc. It will be the case, that some of the hypotheses will not be well supported.
- When trying methods like a decision tree, you should use cross-validation and show your ROC curve and a confusion matrix. For each method, explain the method in one paragraph.
- Explain how and why you will apply your selected method(s) to each hypothesis, and discuss the results.
- Therefore, you will have at least three (3) hypothesis tests and will apply all seven (7) of the following methods to one or more of your hypotheses.

Required Methods

Parametric statistical tests (assume data is normal)

- t-test or Anova (choose one)
- Linear Regression or Logistical Regression (multivariate or multinomial) (choose one)

Data driven predictive models (no assumption of normality) (choose 5 from this group)

- Decision tree
  - http://scikit-learn.org/stable/modules/tree.html
- A Lazy Learner Method (such as kNN)
  - http://scikit-learn.org/stable/modules/neighbors.html
  - http://scikitlearn.org/stable/auto_examples/neighbors/plot_classification.html#sphx-glr-auto-examples-neighbors-plot-classification-py
- Naïve Bayes
  - http://scikit-learn.org/stable/modules/naive_bayes.html
- SVM
  - http://scikit-learn.org/stable/modules/svm.html
- Random Forest
  - http://scikitlearn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

Writeup (10%)
You should begin to have an overall story coming together. Please integrate this with Project 1 to begin to pull together the overall story.

A Note About Your Data:
You should not add more data without discussing it with me. Data collection was the goal of Project 1.

A few final notes:

- All your python code should be well commented, well-structured and easy to read and understand visually; with reasonable variables names, well organized functions, sufficient comments, etc.
- All code must run. Once you submit, download your submission, and re-run it. This will assure that your submission was successful and what you intended it to be.
- You will submit your project through Box using a zip folder that contains: (1) All Code, (2) The Project Report Document that address and discusses all noted requirements and elements, (3) Any needed files and/data that your code will read from, (4) a README.txt to explain basic code use if you feel this is needed.

Extra Credit – This project is eligible for up to 3% extra credit.