QBUS6850 Assignment 2:

Due dates: Monday 15 October 2018

Value: 10%

Notes to Students

1. The assignment MUST be submitted electronically to Turnitin through QBUS6850 Canvas site. Please do NOT submit a zipped file.
2. The assignment is due at **17:00pm on Monday, 15 October 2018**. The late penalty for the assignment is 10% of the assigned mark per day, starting after 17:00pm on the due date. The closing date Monday, 22 October 2018, 17:00pm is the last date on which an assessment will be accepted for marking.
3. Your answers shall be provided as a word-processed report giving full explanation and interpretation of any results you obtain. Output without explanation will receive **zero** marks.
4. Be warned that plagiarism between individuals is always obvious to the markers of the assignment and can be easily detected by Turnitin.
5. The data sets for this assignment can be downloaded from Canvas.
6. Presentation of the assignment is part of the assignment. Markers will reduce to 10% of the mark for poor writing in clarity and presentation. It is recommended that you should include your Python code as appendix to your report, however you may insert small section of your code into the report for better interpretation when necessary. Think about the best and most structured way to present your work, summarise the procedures implemented, support your results/findings and prove the originality of your work.
7. Numbers with decimals should be reported to the third decimal point.
8. The report should be NOT more than 10 pages, with font size no smaller than 11pt, including everything like text, figure, tables, small sections of inserted codes etc but excluding the appendix containing Python code.

Tasks

Question 1 (40 Marks)

Airbnb (**www.airbnb.com**) is a hospitality company that runs an online marketplace for renting and leasing short-term lodging. On the website, visitors have been given opportunity to review any listing on the market.

You are provided with a dataset containing review comments from visitors on a number of existing Airbnb listings in Sydney. You can download the dataset `Sydney_Reviews.csv` from Canvas. Please note this csv file is organized with a separator ";", thus it is not good to view it with Excel.

In this task, you are going to use PCA dimensionality reduction approach to visualize those comments in 3D space. You can achieve this through the following subtasks.

(a) Get a copy of template program `Assignment02_Q1_Template.py` from Canvas. The program helps you load the given csv file into your python environment. Explore (using python code) the DataFrame variable `reviews` created by the template program and report the following (1) what are the DataFrame columns? (2) How many different listings in total? (3) Find out the `listing_id` with the shortest comments (in terms of the number of chars in comments). Retain your code in your program.

(b) Build a TF-IDF representation by using the text feature extractor `Tfidfvector` from `sklearn.feature_extraction.text` to fit the review comments `reviews['comments']`. Requirements:

- max 1000 features;
- remove the top 1% of frequently occurring words;
- a word must occur at least twice to be included as a feature;
- remove common English words.

Hint: Please carefully read the documentation for `Tfidfvector` at **http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html**.

(c) The features extracted in (b) are in 1000 dimensions, which cannot be visualized. Now fit a PCA model to the features from (b) with three principal components. That is, for each listing, (b) gives a 1000 dimensional feature vector and PCA reduces the feature vector to a new vector of 3 dimensions. Finally draw a scattering graph for 3D components of all the listings. Report what you have observed and what conclusion you can draw based on the visualization.

Hint: You may use the following code to create a 3D coordinate system to draw

```
fig = plt.figure(1, figsize=(4, 3))
plt.clf()
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=134)
ax.scatter( … …. … )
```

Question 2 (60 Marks)

Given the attached "`loan.csv`" data set. Implement different classification algorithms.

General instructions:

1. Split the data into training and test set (80%, 20% split).
2. Use all the features available. Column "`y`" is the response/target variable.
3. Treating class "1" as positive.
4. 5-fold/10-fold cross validation if needed.
5. Use "`random_state=0`" whenever you need to specify random state of using Python classes
6. For all Python parameters that are not specified in the questions, use the default values.

7. You may use "`GridSearchCV`" (`from sklearn.model_selection import GridSearchCV`) attribute `best_estimator_` to select the best estimator for question (a), (b) and (c) below

(a) Run k-NN on the given data set. Select the best value of `k` between 5 and 31 using 5-fold cross-validation on the training dataset. Report the best number `k` you selected. Use your selected `k` to run k-NN on the test set, and report the prediction performance (as in the above general instructions of the model.

(b) Write Python code to run the decision tree on the given dataset. Select the best "`max_depth`" between 5 and 30 using 10-fold cross validation on the training dataset. Report your selected "`max_depth`'' of the tree. Use this depth to build the best tree model and report the prediction performance of the model.

(c) Run the Adaboost (use algorithm = "SAMME") on the given dataset. Select the best "`n_estimators`" between 5 and 50 using 5-fold cross validation on the training dataset. Report your selected "`n_estimators`"? Use the best performing Adaboost model on the test set, and report the prediction performance of the model.