Disclaimer: This document and the attached dataset are for the student use only. The student is **NOT** allowed to post this document or the dataset in websites like coursehero.com, studynotes.com, GitHub or any similar website. Posting the homework document or the dataset on those sites will result into a legal copyright violation.

Predictive Analytics

Assignment Three
Classifying Textual Data

Learning Outcomes:

1. Developing KNN data classification multi-class algorithm.
2. Applying KNN to classify textual data using distance metric.
3. Develop k-fold cross-validation algorithm to validate data.
4. Measuring performance of a supervised learning model.

In this assignment, you will apply a supervised learning algorithm to text data that you pre-processed in homework one and two. You will need to develop KNN (K- nearest neighbor) algorithm that finds similar documents to the document in question and predict its label (cluster/topic).

You must write your own *KNN algorithm* that takes, a raw document (not a vector): a file name or path, and the data matrix of TF/IDF along with an appropriate labels (e.g.: Predictive Analytics, Irma and Harvey, etc) you had in HW#1 and an integer K that specifies the nearest neighbors.

You may want to create a class or a method that takes care of taking a document and pre-processes that document into a vector (You can re-use code from HW#1 or 2).

See chapter data classification to learn more about K nearest neighbor algorithm.

You will need another KNN pre-built in a library ***such as*** to validate and compare you results: Java Machine Learning Library in order for your to make sure that you implemented KNN correctly.

It is highly recommended to use an Integrated Development Environment such as Eclipse to be able to import the Java library with no issues.

Tasks:

1. Split the dataset into training and test (~70% training per folder & ~30% test per folder).

2. Develop a routine (method) that takes a raw document as an input, a similarity measure, and return the K most similar/nearest documents to that document and **the label (what topics this document is about)**. The similarity measure is ***the distance you discover works better according to your HW2.***

   We will be testing with new raw documents and test the predicted labels.

   Please note that the following topics of the original documents:

   C1: Airline Safety
   C2: Amphertamine
   C3: China and Spy Plan and Captives
   C4: Hoof and Mouth Desease
   C5: Iran Nuclear
   C6: Korea and Nuclear Capability
   C7: Mortrage Rates
   C8: Ocean and Pollution
   C9: Satanic Cult
   C10: Store Irene

C11: Volcano
C12: Saddam Hussein
C13: Kim Jong-un
C14: Predictive Analytics
C15: Irma & Harvey

3. Develop a 10 fold cross-validation function which **randomly** splits the training data and train your model based on validations. (Train your models with cross-validation on your training data. Try different Ks, and select best one.)

4. You will need to use another KNN pre-built library such Java Machine Learning Library , or Weka to compare your results. **(It has cross-validation utility, performance measure)**

   Note: For selecting K, plot/compare average cross-validation errors on ranging the value of K.
   For getting the value of cross-validation errors, call the utility which you created in clustering assignment to calculate the precision/accuracy.

5. **Measuring Model Performance**

   Use the K chosen in 3 and 4, train you model on the training dataset, generate *Confusion Matrix* from clustering assignment and print out the accuracy of the model on the **Test Dataset** along with the confusion matrix. You might also use precision and recall/F-measure /confusion matrix to measure the performance of your classification algorithm.

6. (Bonus + 10pts) Apply another classification algorithm (e.g. SVM) and compare your results to KNN for two documents of your choice. Document your algorithm that you used and the comparation with KNN.

7. Ensembles **with Rapid Miner**

In this part you will be using RapidMiner as a tool where you will you should run **at least *one* of the following ensemble.** It is known in the literature that ensembles produce better prediction. An ensemble method uses multiple predictive algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone.

You will need to use the matrix or the reduce version of matrix (e.g SVD, PCA) you produced in HW#1 into one of these ensembles.

You will need to build a process that will take new document with unknown label to be processed and fed to your ensemble. Summarize your results in a word document with snapshots of the processes used in RapidMiner and the results. Save the processes and attach them in the homework folder under a subfolder (essemblesRapidMiner). **Click on the pictures bellow to learn more about the ensemble operators.**

Stacking
(RapidMiner Studio Core)

Bagging
(RapidMiner Studio Core)

Vote
(RapidMiner Studio Core)

AdaBoost
(RapidMiner Studio Core)

Random Forest
(Concurrency)

Include your code and **your report in a Zip file. Make sure that you have all the following deliverables in your code and your report:**

Deliverables	Points
ReadMe document (details how to export your project into Eclipse or JAVA IDE)	**10**
(7) Ensembles	**20**
Implement **YOUR OWN** KNN algorithm.	20
Implement cross-validation function on **Training-data**	20
Selecting the right K value in KNN. Attach snippet of plot/print Avg. Cross-validation errors vs. K value	10
Compare results with Java ML KNN library, or Weka or any other library	10
Measuring Model Performance on **Test-data (see attached test data – you can select some of the documents from test data, run your algorithms and record the results in the report)**	10
Total	**100**

The report should include and not limited to:

1. For all questions: Screenshot of each methods/classes (code) you wrote (for all question), simple explanation of each method (if you include comments in the code, that would be sufficient as a snapshot)
2. question 3 and 4: The plot of your metric with different K (let's say five different K are tried) (either 5 curves with x-axis as ten folders, or one curve with x-axis with different K)
3. question 5: K chosen, your confusion matrix and overall accuracy.
4. For the report on question 7 you will need to include all the snapshots of the processes of rapid miner and add short comments on your learning outcomes from this exercise.