Assignment Two

Clustering Textual Data

Prepared by Dr. Bari

Learning Outcomes:

1. Developing KMeans data clustering algorithm.
2. Applying KMeans to cluster textual data using distance metric.
3. Converting features in lower dimension using PCA/SVD.
In this assignment, you will apply an unsupervised learning algorithm to text data that you pre-processed in homework one. You will need to develop a clustering algorithm that will group similar documents vectors you produced in homework one in order to check the quality of your document vector representation (TF/IDF matrix).

**Pre-requisites:** Review the notes from Chapter 7 in the course website and read chapter 6 from the PA for D book.

You must write your own *K-means clustering algorithm* that takes as inputs: the data matrix of TF/IDF you had in HW#1 and an integer K that specifies the number of clusters.

You will need another K-means pre-built in a library in [Java Machine Learning Library](https://www.cs.colostate.edu/~donny/software.html) in order to compare your results. It is preferable to use an Integrated Development Environment such as Eclipse to be able to import the Java library with no issues.

**Tasks:**

1. Develop a routine (method) that takes a document as an input, a similarity measure, and return the K most similar documents to that document. The similarity measures. Later in your experiment you will need to implement the *Cosine similarity and the Euclidian distance*. See 7 for how you will apply your routine.

   Apply the routine to three documents of your choice from one of the 15 folders and register your analysis on which distance is more accurate. You might need to call the routines you developed from HW#1 to convert the document into a vector of n terms. Report your results.

2. Implement a routine K-means clustering algorithm that takes as input the data matrix of word-documents matrix, a similarity measure and a K (number of desired clusters) the algorithm will output the K clusters. It is up to you to design this routine in terms of inputs and outputs, as long as it performs K-means and outputs the results.

3. You will need to use another K-means pre-built in a library such [Java Machine Learning Library](https://www.cs.colostate.edu/~donny/software.html) To compare your results.
4. **Plot the Clusters**

Visualize the final clusters. Give different color to each cluster. You can use R/Python/Java/Rapidminer/Tableau/ or any visualization tool including Microsoft Excel.

**Hint:** How do we convert N-dimensions to 2D plot (you may want to apply PCA or SVD)

5. **Measuring Model Performance**

In this case, we know which documents belongs to which group, you will need to measure the performance of the KMeans model using *Confusion Matrix* and print out the accuracy of the model along with the confusion matrix. You might use *precision and recall* or F-measure to measure the performance of your clustering algorithm.

6. (Bonus – 20pts) The choice of the initial centroids will have a major effect on the performance Kmeans. Research and implement an algorithm that will enhance your K-means. **You might consider Kmeans++**. Report the same performance analysis as stated in 7.

7. Here are the scenarios you would apply Kmeans to and measure the performance for:

   - Apply SVD to your matrix from HW#1. Register extracted insights.
   - Apply Kmeans with K=3 for documents from three different folders of your choice with Cosine Distance.
   - Apply Kmeans K=3 for documents from three different folders of your choice with Euclidean Distance.
   - Apply Kmeans to all the documents (full matrix) K=15 with Euclidean Distance.
   - Apply Kmeans to all the documents (full matrix) K=15 with Cosine Distance.

If the results are not very good, you will have a chance to review the matrix you generated in HW1.

You must use Java. You can use library for PCA and SVD.
<table>
<thead>
<tr>
<th>Deliverables</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implement YOUR OWN KMeans algorithm.</td>
<td>50</td>
</tr>
<tr>
<td>Compare results with Java ML KMeans library</td>
<td>30</td>
</tr>
<tr>
<td>Apply SVD to your matrix from HW#1. Register extracted insights.</td>
<td>20</td>
</tr>
<tr>
<td>Distance matrix: Euclidean for K=3, K=15</td>
<td>20</td>
</tr>
<tr>
<td>Distance matrix: Cosine for K=3,K=15</td>
<td>20</td>
</tr>
<tr>
<td>Plot the clusters</td>
<td>30</td>
</tr>
<tr>
<td>Measuring Model Performance</td>
<td>40</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>210</strong></td>
</tr>
</tbody>
</table>

**Bonus: 20pts.**