**CS 536/CS 432 – Data Mining**

**Assignment 3**

**Due: April 2 (Sunday) at 12 midnight**

**Instructions:** (1) You may discuss the assignment with others. However, you MUST do and submit your OWN work. (2) Submit a soft-copy report to the submission folder on LMS. Include report and code needed to reproduce your results.

**1. Evaluating Clustering Algorithms in RapidMiner (30 points)**

In this exercise, you are required to evaluate several clustering algorithms on a dataset for which labels are available. Download the Leaf dataset from UCI Repository and go over its description.

Apply (i) k-means, (ii) k-medoids, (iii) single-link hierarchical agglomerative, and (iv) density-based clustering on this dataset, Set k = no. of classes and use the Euclidean distance. Evaluate the performance of the algorithms by comparing the clustering prediction with actual labels. For this you will have to apply the Cluster2Prediction and Performance operators after the clustering operator. Report the entrpy, precision, recall, and F-measure of each algorithm. Which algorithm is more efficient? Which one is more effective? Compare and contrast the algorithms performance. Compare with respect parameter requirements and intuitiveness of clusterings.

**2. Document Clustering in MATLAB (40 points)**

Download the Bag of Words ([NIPS full papers data](http://archive.ics.uci.edu/ml/datasets/Bag+of+Words)) dataset from UCI Repository. This dataset has two files. One file provides information on words used in documents in the (i,j, v) format (except for the first 3 rows). Here, i is the document ID, j is the word ID, and v is the count of word j in document i. The second file contains the mapping from word ID to actual word.

1. Create a document by word matrix from the first file. You can use MATLAB’s spconvert and full commands to create a full document by word matrix. You will have to remove the first 3 rows in the file before reading and converting the file.
2. Cluster the data using (i) k-means and (ii) NNMF available in MATLAB. Use k = 5, 10, 50, and 100, and use the cosine measure for similarity in the k-means algorithm. For NNMF obtain a hard clustering by taking a maxrow on the N by K matrix.
3. Plot the SSE (sum of squared error or clustering objective) against the number of clusters, k.
4. For k = 10, find and report the top 5 most frequently occurring words in each cluster.
5. Transpose the original matrix to get a word by document matrix. Use this to cluster words and perform all steps in (b) to (d) (in d report the top 5 documents). Discuss and interpret the results from the two clusterings (word clustering and document clustering).

(Note: if dataset is large and memory issues arise, use a smaller randomly sampled subset of the full data)

**3. Text Clustering using Chinese Whispers (30 points)**

Use Chinese Whispers algorithm to cluster the dataset in question 2. You can code this in the language of your choice.

1. Create the similarity matrix for the data using cosine similarity
2. Apply the deterministic CW algorithm on (a) full similarity matrix (with zeros on diagonals, of course), (b) 5 nearest neighbor defined similarity matrix, and (c) 10 nearest neighbor defined similarity matrix. For each of these, run the CW algorithm 3 times with different initial class matrices.
3. For each clustering result, report the number of clusters found and top five frequent words in each cluster.
4. Discuss and compare the performance of the algorithm with that obtained in question 2.