COMP 337/527 2020 CA Assignment 2
Data Clustering
Implementing the k-means clustering algorithm

Assessment Information

Assignment Number	2 (of 2)
Weighting	13%
Assignment Circulated	27th March 2020
Deadline	20th April 2020, 17:00 UK Time (UTC)
Submission Mode	Electronic via Departmental submission system
Learning outcome assessed	(1) A critical awareness of current problems and research issues in data mining.
Purpose of assessment	This assignment assess the understanding of k-means clustering algorithm by implementing k-means for text clustering.
Marking criteria	Marks for each question are indicated under the corresponding question.
Submission necessary in order to satisfy Module requirements?	No
Late Submission Penalty	Standard UoL Policy applies.

1 Objectives

This assignment requires you to implement the k-means clustering algorithm using the Python programming language.

> **Note that *no credit will be given for implementing any other types of clustering algorithms or using an existing library for clustering instead of implementing it by yourself. However, you are allowed to use numpy and scipy libraries for accessing data structures such as numpy.array or scipy.sparse. But it is not a requirement of the assignment to use numpy or scipy. You can use matplotlib for plotting but it is not compulsory to use matplotlib. You must provide a README file describing how to run your code to produce the results. Programs that do not run will result in a mark of zero!***

2 Word Clustering using k-means

In the assignment, you are required to cluster words belonging to four categories: *animals*, *countries*, *fruits* and *veggies*. The words are arranged into four different files. The first entry in each line is a word followed by 300 features (word embedding) describing the meaning of that word.

Questions

(1) Implement the k-means clustering algorithm with Euclidean distance to cluster the instances into k clusters. (**30 marks**)

(2) Vary the value of k from 1 to 10 and compute the precision, recall, and F-score for each set of clusters. Plot k in the horizontal axis and precision, recall and F-score in the vertical axis in the same plot. (**10 marks**)

(3) Now re-run the k-means clustering algorithm you implemented in part (1) but normalise each feature vector to unit ℓ_2 length before computing Euclidean distances. Vary the value of k from 1 to 10 and compute the precision, recall, and F-score for each set of clusters. Plot k in the horizontal axis and precision, recall and F-score in the vertical axis in the same plot. (**10 marks**)

(4) Now re-run the k-means clustering algorithm you implemented in part (1) but this time use Manhattan distance over the unnormalised feature vectors. Vary the value of k from 1 to 10 and compute the precision, recall, and F-score for each set of clusters. Plot k in the horizontal axis and precision, recall and F-score in the vertical axis in the same plot. (**10 marks**)

(5) Now re-run the k-means clustering algorithm you implemented in part (1) but this time use Manhattan distance with ℓ_2 normalised feature vectors. Vary the value of k from 1 to 10 and compute the precision, recall, and F-score for each set of clusters. Plot k in the horizontal axis and precision, recall and F-score in the vertical axis in the same plot. (**10 marks**)

(6) Now re-run the k-means clustering algorithm you implemented in part (1) but this time use cosine similarity as the distance (similarity) measure.Vary the value of k from 1 to 10 and

compute the precision, recall, and F-score for each set of clusters. Plot k in the horizontal axis and precision, recall and F-score in the vertical axis in the same plot. (**10 marks**)

(7) Comparing the different clusterings you obtained in (2)-(6) discuss what is the best setting for k-means clustering for this dataset. (**20 marks**)

3 Deadline and Submission Instructions

- Deadline for submitting this assignment is **20th April 2020, 17:00 UK time (UTC)**.
- Submit
  (a) the source code for all your programs,
  (b) a README file (plain text) describing how to compile/run your code to produce the various results required by the assignment, and
  (c) a PDF file providing the answers and graphs for the questions (2)-(8).

  Compress all of the above files into a single tar ball (tgz) file and specify the filename as *studentid.tgz*. Replace *studentid* with your student ID. It is extremely important that you provide all the files described above and not just the source code! (If you are unable to create a tgz file then create a zip file)

  Every year I get assignments that do not mention a name or a student id. Please check that your submission has these details because otherwise there is no way to find out who submitted the assignment.
- Submission is via the departmental electronic submission system accessible (from within the department) from
  `https://sam.csc.liv.ac.uk/COMP/Submissions.pl`.