Homework 7

Your Name: \_Vikas Sanil\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student ID:\_\_A20511552\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1. (40 points) Clustering by the K-Means approach**

Consider the following document-term matrix, where each entry represents the raw frequency of a term Ti in document Dj. We would like to apply clustering to automatically group these documents into 3 classes (clusters). Note: you are encouraged to use a spreadsheet program such as Microsoft Excel to facilitate computation in intermediate steps.



Suppose we initially assign D2 to Cluster 1, D4 and D6 to Cluster 2, and D5 and D7 to Cluster 3. Using the K-means clustering method discussed in class, compute the final contents of the 3 Clusters. Use the Cosine similarity of two vectors (NOT only the dot product) as your similarity measure. Show the details of your computation, including intermediate steps in each iteration of the algorithm.

Note: Recall that the Cosine similarity of two vectors is their dot product divided by the product of their norms. For example, Consider the two vectors X and Y:

X = <3, 0, 1, 2, 0, 3>

Y = <2, 0, 0, 3, 8, 4>

The dot product is given by sum of the coordinate-wise multiples:

dot-product(X, Y) = 3\*2 + 0\*0 + 1\*0 + 2\*3 + 0\*8 + 3\*4

= 6 + 0 + 0 + 6 + 0 + 12

= 24.

The norm of each vector is the square-root of the sum of the squares of its dimension values. So, the norms of X and Y are:

     

and the Cosine similarity of X and Y is given by:



Your answer:



After two iterations we can see the clusters are remaining the same. Hence we can conclude Document D2 and D4 belongs to Cluster1, Documents D3 and D6 belongs to Cluster 2, and Documents D1, D5 and D7 belong to Cluster 3.

Please refer spreadsheet "K Means Cosine Similarity" to check the work done.

**2. (40 points) Practice with hierarchical clustering**

Perform a hierarchical clustering of the following data points:

D1: <1, -2>

D2: <-1, 9>

D3: <-6, 4>

D4: <0, -5>

D5: <1, -9>

clusters are represented by their centroid (use means), and at each step the clusters with the closest centroid are merged. And you should use the “Bottom-to-Top” approach to construct your hierarchical tree. Use the Euclidean distance to measure the distance between instances or clusters.

You must draw a hierarchical tree structure and give the sequence (such as, ①②③…) to each merging operation. For example, number ① means this merge is the 1st merge operation. In addition, you must use the centroid value or vector to represent the new value or vector for each group of points. Write down the centroid value or vector to the side of the ① notations.

Your answer:











Please refer spreadsheet "Hierarchical Clustering" to check the work done.

**3. (20 points) How to find the best value of K in K-Means clustering**

1). Read the paper X-Means.pdf, and introduce how you can find the best value of K in K-Means clustering [10]

**According to this paper's authors, K-means suffers three major shortcomings despite its popularity for general clustering:**

**It scales poorly computationally,**

**The number of clusters K has to be supplied by the user, and**

**The search is prone to local minima.**

**In this paper, the authors propose solutions for the first two problems and a partial remedy for the third.**

**They are proposing a new algorithm X-means, which consists of the following two operations repeated until complete.**

**X-means:**

**Improve Params**

**Improve Structure**

**If K > Kmax stop and report the best scoring model found during the search,**

**Else go to 1.**

**The Improve Params operation is running conventional K-means to convergence.**

**The Improve Structure operations find out if and where new centroids should appear.**

**The authors explain further Improve Structure operation by initially discussing the pros and cons of 2 Cluster splitting ideas:**

**Picking a centroid and running a k-means on a nearby new centroid and choosing the one with a better model**

**Halving centroids and verifying whether the original centroid or resulting centroids have the best model. And retain the centroids which have the best model.**

**Then they combined both ideas and introduced Bayesian Information Criterion(BIC) as the criterion to choose among the models.**

**Following is the final formula they have derived from the BIC formula(Kass and Wasserman, 1995).**



**Later they discussed how the X-means algorithm can be accelerated by “blacklisting” the centroid clusters which showed better models. Also, they have suggested further acceleration can be done to the algorithm using other criteria like MDL, to evaluate the new centroids created.**

**Then they discuss the experiments they have run using the X-means algorithm and K-means algorithm in parallel for similar datasets. They have observed for lower K values the performance of both algorithms are at par but as the number of classes increases in the dataset, the X-means algorithm has provided better performance in finding the K value.**

2). Search online, and find at least one extra method to define the best value of K in K-Means clustering [10]

**Scalable K-means++: As is well-known, proper initialization of k-means is crucial for obtaining a good final solution. The recently proposed k-means++ initialization algorithm achieves this, obtaining an initial set of centers that is provably close to the optimum solution. A major downside of the k-means++ is its inherent sequential nature, which limits its applicability to massive data: one must make k passes over**

**the data to find a good initial set of centers.**

**Algorithm 1 k-means++(k) initialization.**

**1: C ← sample a point uniformly at random from X**

**2: while |C| < k do**

**3: Sample x ∈ X with probability d2(x,C)**

**φX (C)**

**4: C ← C ∪ {x}**

**5: end while**

**Algorithm 2 k-means||(k, `) initialization.**

**1: C ← sample a point uniformly at random from X**

**2: ψ ← φX (C)**

**3: for O(log ψ) times do**

**4: C′ ← sample each point x ∈ X independently with**

**probability px = `·d2(x,C)**

**φX (C)**

**5: C ← C ∪ C′**

**6: end for**

**7: For x ∈ C, set wx to be the number of points in X closer**

**to x than any other point in C**

**8: Recluster the weighted points in C into k clusters**

**Source:** [**https://theory.stanford.edu/~sergei/papers/vldb12-kmpar.pdf**](https://theory.stanford.edu/~sergei/papers/vldb12-kmpar.pdf)