REPORT

Himanshu, MDS202327 and Suneet Patil, MCS202315

Functions:

1. readDoc(): Computes the indicator matrix of whether the word appears in the document.
2. jaccardDistance(): Calculates the jaccard distance of two arrays.
3. initializedCentroids(): Initializes the centroids using kmeans++ algorithm.
4. getLabels(): Assigns labels to the document given the centroids.
5. getCentroids(): Calculates the centroids based on the labels.
6. ssdf(): Calculates the sum of square of distance using jaccardDistance().
7. kMeanspp(): Implements the min batch kMeanspp algorithm.
8. storeResults(): Store the results of the algorithm.

Workflow followed:

1. A single call is made to storeResults which in turn calls first, initializedCentroids() followed by kMeanspp().
2. kMeanspp() uses getLabels(), getCentroids() and ssdf() to implement the algorithm.

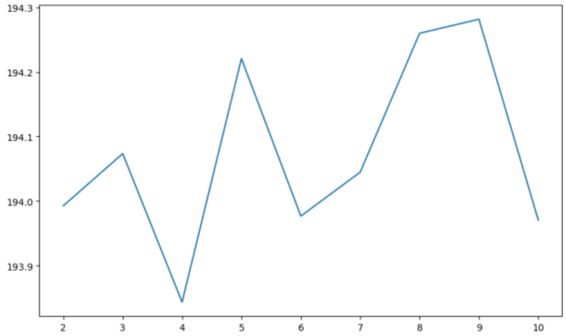
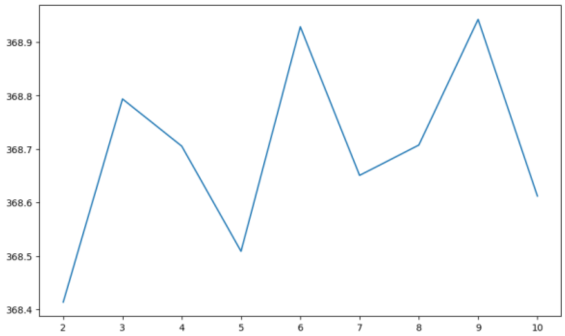
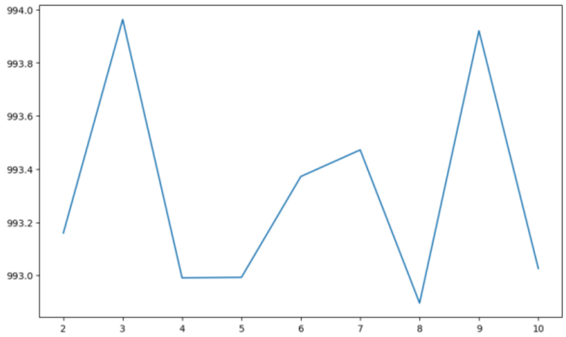
Conclusion:

Some liberty was taken while choosing k, batch size and max iteration, given the computational limitations and the datasets being insanely large.

No clear pattern was seen in the clustering, although a weak pattern of two dips, first near 4, 5 other near 6,7 can be seen. As the saying goes, your model can do as good as your data.

A report is given on the second page.

Results:

Data	Batch size (as % of no. of documents)	Plots	Possible Elbow (clear/random)
Kos	200 (5.8%)		4,6 (random)
Nips	400 (26%)		5,7 (random)
Enron	1000 (2.5%)		4,8 (random)