K-means clustering with Apache Mahout

The surge in growth of Big Data usage and popularity has resulted in challenging conditions for clustering datasets. The K-means algorithm, in particular, works well when dealing with smaller datasets (Sreedhar, Kasiviswanath and Chenna Reddy, 2017).

There are 5 key steps to the K-means clustering algorithm:

1. Initialise K centroids
2. Calculate the distance from every data point to the centroid
3. Assign each data point to the nearest centroid forming clusters
4. Recalculate the centroids, by taking the mean of all the data within a cluster and use that as the new centroid
5. Repeat steps 2-4 until convergence.

Using a MapReduce methodology approach presents an alternative solution that can overcome these challenges by harnessing the numerous benefits of parallel processing (Advantages of Hadoop MapReduce Programming, 2021). There are two sub-computing tasks within the methodology of K-means clustering; assignment (Step 2 and 3) and updating (Step 4). In the MapReduce implementation of K-means, the Map task executes assignment. By calculating the distance of each data point to every centroid and outputting a key-value pair. The key and value represent the nearest centroid and data point, respectively. The reduce task executes the update step by computing the pairwise summation of the values of each key. Then, the new centroids are calculated, and the map and reduce tasks repeat until the centroid locations are stable.

The key challenge when implementing k-means clustering is maintaining commonality between the features across the clustered groups of data. Selecting the appropriate parameters based on the structure and nature of the data can improve the algorithm's performance. The number of clusters can impact how close together members of a cluster are and how dissimilar members of different clusters are. Similarly, the choice of distance measure is dependent on the nature and structure of the dataset. Cosine similarity is widely used with text-based data as it calculates distance irrespective of the length of the text used, making it less affected by magnitude (Anon, 2021; Ladd, 2021; Relationship between Cosine Similarity and Euclidean Distance., 2021). Evidently, testing and evaluating different parameters is an integral part of the evaluating a clustering algorithm.

Results

I used the Hadoop commands from the Hadoop_Commands_gh.txt to produce my results mimicked the mahout commands from Programming Activity 4.15 in Topic 4. Table 1. (see Appendix) contains the output of the k-means clustering algorithm for each value of K and both distance measures. Inter- and Intra-Cluster density were used to evaluate the performance of the algorithm. Intra-cluster density represents the distance between the data points within a cluster and Inter-cluster density represents the distance between each cluster (Figure 1. Example of inter-class and intra-class cluster similarity In..., 2021).

As illustrated in Figure1., intra-cluster density remained stable for both distance measures when increasing the number of clusters. When using the Cosine distance, the average intra-cluster density was higher (Mean = 0.58, SD = 0.016) and had a larger range (0.53 - 0.62) compared to the Euclidean distance (Mean = 0.57, SD = 0.019, Range = 0.53-0.61).

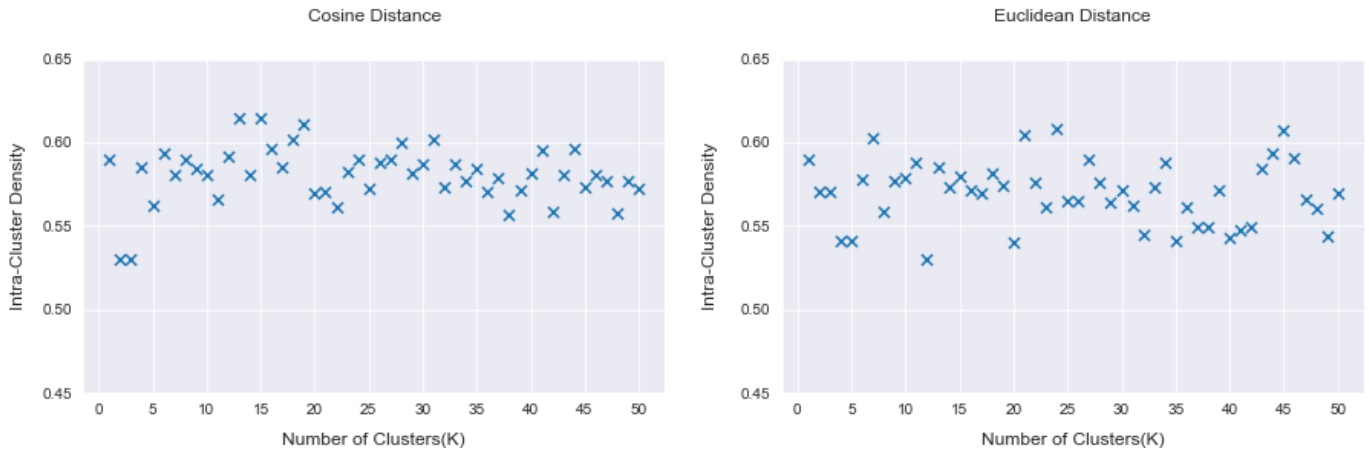

Figure 1. Effect of Varying K and Distance Measure on Intra-Cluster Density

As shown in Figure 2. below, Inter-cluster density decreased as the number of clusters increased, until reaching a point of convergence. The data was smoothed by removing any outliers and rounding the inter-cluster densities to 2 decimal places (2DP) and 1DP. The elbow method was used to determine the optimal value of K for the Cosine(K=10) and Euclidean distance(K=9). The elbow point on the graphs represents the trade-off between reduced improvements of performance and the algorithm overfitting (Weißer et al., 2020; Elbow method (clustering) - Wikipedia, 2021).

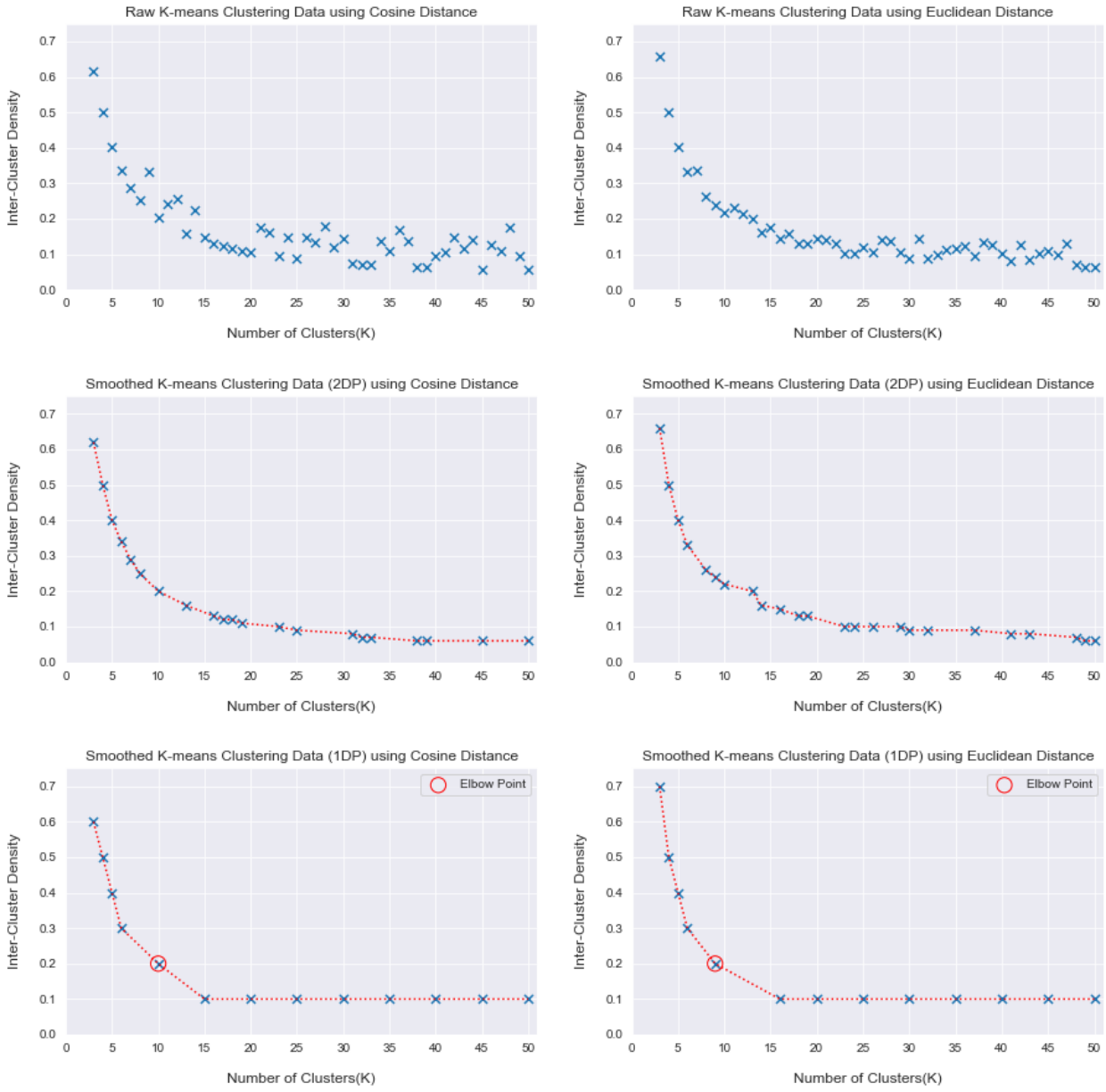

Figure 2. Effect of Varying K and Distance Measure on Intra-Cluster Density

Discussion

The elbow method determined the optimal value of K for the Cosine(K=10) and Euclidean distance(K=9). Conversely, the best distance measure is harder to determine, as there are only marginal differences between the intra- and inter-cluster density. The inter-cluster density is smaller at the optimal value of K for the Cosine distance when compared with the Euclidean distance. Similarly, the intra-cluster density is marginally greater when using the Cosine distance.

The dataset comprised a range of French and English text files including; fiction, plays, and poems. The dataset consisted of 323 French texts and 28 English texts, although the total file size of the French texts (32.1MB vs 35.6MB) being slightly smaller comparatively. Furthermore, the text files ranged in length from 172 to 8394 lines. Several articles and studies have suggested that Cosine distance is better at measuring similarity than Euclidean distance when dealing with text data. Cosine distance concerns the orientation of two points and not the exact location and distance between them, thus making it a more robust measure when dealing with several texts that vary in size (Anon, 2021; Ladd, 2021; Relationship between Cosine Similarity and Euclidean Distance., 2021). In addition to this, the Euclidean distance is more sensitive to outliers, as the distance is computed by squaring the distance between the two vectors (Nowak-Brzezińska and Horyń, 2020). Therefore, any variance experienced would be magnified. As stated in Topic 4.15 Programming Activity: "Low inter-cluster density and high intra-cluster density indicates a good solution." When considering the structure and nature of the data alongside the results, I believe the best setting for this dataset is using the Cosine distance and K=10.

MapReduce possesses certain benefits such as; high scalability, resistance to faults, and its capability of processing large volumes of data. However, both MapReduce and Hadoop possess certain limitations. For example, during each part of data processing MapReduce reads the data from the disk, and then writes it onto the disk. This repeated process leads to an exceedingly slow processing speed when dealing with large volumes of data. Although Hadoop supports batch processing (13 Big Limitations of Hadoop & Solution To Hadoop Drawbacks - DataFlair, 2021), it does not utilise streamed data, thus increasing its run-time. Similarly, MapReduce Methodology does not take advantage of all the memory available at the cluster, again limiting the execution engine's performance. MapReduce processes data through a chain of stages instead of using a cyclical flow. This characteristic reduces the efficiency of any Machine Learning algorithm that requires several iterations, such as the K-means clustering algorithm. Ultimately these limitations have only been exacerbated with the continuing increases in volume and complexity of datasets. Hence why MapReduce is rarely used anymore, with many organisations favouring Spark.

Appendix

I did run the k-means clustering algorithm for K=1 and K=2 however the clusterdump output showed NaN for the inter-cluster density as shown below the table.

Table 1. Unsmoothed Data Output from Mahout K-means

Number of Clusters (K)	*Inter-Cluster Density (Cosine Distance)*	*Intra-Cluster Density (Cosine Distance)*	*Inter-Cluster Density (Euclidean Distance)*	*Intra-Cluster Density (Euclidean Distance)*
1	*NaN*	*0.589492361*	*NaN*	*0.589492361*
2	*NaN*	*0.529860086*	*NaN*	*0.570240056*
3	*0.616406032*	*0.529860086*	*0.658225608*	*0.57023633*
4	*0.500089912*	*0.585464486*	*0.501752006*	*0.541267234*
5	*0.401291818*	*0.562346218*	*0.403744165*	*0.541260926*
6	*0.336247646*	*0.593163979*	*0.334013042*	*0.578024753*
7	*0.28775223*	*0.580168207*	*0.335879223*	*0.603085946*
8	*0.253220961*	*0.589779846*	*0.261338957*	*0.558550355*
9	*0.332952433*	*0.584588941*	*0.240004515*	*0.57736526*
10	*0.203878615*	*0.580745224*	*0.217857878*	*0.57845491*
11	*0.240265379*	*0.565448164*	*0.230821143*	*0.587857896*
12	*0.257282406*	*0.5917262*	*0.213726434*	*0.53041873*
13	*0.158561408*	*0.614365358*	*0.201340018*	*0.585340438*
14	*0.224922828*	*0.580468886*	*0.161574085*	*0.573074897*
15	*0.146938162*	*0.615050778*	*0.17605668*	*0.579524664*
16	*0.130304209*	*0.595993968*	*0.145195849*	*0.571639417*
17	*0.123700868*	*0.585464307*	*0.158859833*	*0.569585634*
18	*0.116953506*	*0.601849971*	*0.128397109*	*0.581185719*
19	*0.10957551*	*0.610892573*	*0.130355951*	*0.574429551*
20	*0.105119745*	*0.569483604*	*0.143999822*	*0.540428798*
21	*0.174040368*	*0.570295541*	*0.13925432*	*0.604775791*
22	*0.162257698*	*0.560962459*	*0.13047383*	*0.576159158*
23	*0.095105884*	*0.582063103*	*0.102464374*	*0.561698546*
24	*0.147644668*	*0.589876332*	*0.103223918*	*0.608354735*
25	*0.08881451*	*0.572556811*	*0.117996825*	*0.564526973*
26	*0.147674406*	*0.587666848*	*0.104643435*	*0.565196551*
27	*0.134949763*	*0.589594995*	*0.141680603*	*0.589649015*
28	*0.17722872*	*0.599627601*	*0.135739103*	*0.575732356*
29	*0.119385781*	*0.581801553*	*0.104918543*	*0.564448029*
30	*0.142234239*	*0.586793108*	*0.088621559*	*0.571706543*
31	*0.07569264*	*0.601322637*	*0.143633599*	*0.5624414*
32	*0.071685828*	*0.573478921*	*0.08847707*	*0.545087052*
33	*0.069214653*	*0.586649805*	*0.098786235*	*0.5732263*
34	*0.135981525*	*0.576745545*	*0.112430878*	*0.587710172*
35	*0.109320952*	*0.584108955*	*0.116730733*	*0.540733618*
36	*0.170064324*	*0.570617225*	*0.122283749*	*0.560923099*
37	*0.138002294*	*0.578385618*	*0.093949878*	*0.548933857*

38	0.064453267	0.556270579	0.135089827	0.549520109
39	0.063984002	0.571218022	0.127673421	0.571380272
40	0.095417287	0.581985546	0.102109776	0.543139842
41	0.104284342	0.595252834	0.080332672	0.547482645
42	0.145858944	0.558136213	0.125100987	0.549287728
43	0.114799871	0.5810428	0.084533105	0.584232086
44	0.141423138	0.596288577	0.102247514	0.593700061
45	0.057202288	0.57351899	0.10814179	0.607669637
46	0.125478471	0.580354344	0.099298111	0.590448603
47	0.110207532	0.577322661	0.128340902	0.56598595
48	0.176064618	0.557953561	0.069558336	0.560509947
49	0.093849955	0.576736026	0.064169069	0.544101686
50	0.057117748	0.572201512	0.062645907	0.569621744

Output of NaN Results for K=1,2 for both distances:

Results for Euclidean Distance at K=1

```
Inter-Cluster Density: NaN
Intra-Cluster Density: 0.5894923607413921
CDbw Inter-Cluster Density: NaN
CDbw Intra-Cluster Density: 0.09336182204611321
CDbw Separation: NaN
```

Results for Euclidean Distance at K=2

```
Inter-Cluster Density: NaN
Intra-Cluster Density: 0.5702400563253324
CDbw Inter-Cluster Density: 0.0
CDbw Intra-Cluster Density: 0.16906644922322223
CDbw Separation: 621787.0743478383
```

Results for Cosine Distance at K=1

```
Inter-Cluster Density: NaN
Intra-Cluster Density: 0.5894923607413921
CDbw Inter-Cluster Density: NaN
CDbw Intra-Cluster Density: 0.09336182204611321
CDbw Separation: NaN
```

Results for Cosine Distance at K=2

```
Inter-Cluster Density: NaN
Intra-Cluster Density: 0.5298600863328939
CDbw Inter-Cluster Density: 0.0
CDbw Intra-Cluster Density: 0.17858210197268684
CDbw Separation: 691445.8794719757
```

References

DataFlair. 2021. *13 Big Limitations of Hadoop & Solution To Hadoop Drawbacks - DataFlair*. [online] Available at: <https://data-flair.training/blogs/13-limitations-of-hadoop/> [Accessed 6 July 2021].

Dean, J. and Ghemawat, S., 2004. MapReduce: Simplified data processing on large clusters.[online] Available at: https://static.usenix.org/publications/library/proceedings/osdi04/tech/full_papers/dean/dean.pdf [Accessed 21 June 2021].

En.wikipedia.org. 2021. *Elbow method (clustering) - Wikipedia*. [online] Available at: https://en.wikipedia.org/wiki/Elbow_method_(clustering) [Accessed 9 July 2021].

Ladd, J., 2021. *Understanding and Using Common Similarity Measures for Text Analysis*. [online] Programminghistorian.org. Available at: https://programminghistorian.org/en/lessons/common-similarity-measures [Accessed 5 July 2021].

Leskovec, J., Rajaraman, A. and Ullman, J., 2014. *Mining of Massive Datasets - Stanford InfoLab*. 3rd ed. Cambridge: Cambridge University Press, pp.25-31. [online] Available at: http://infolab.stanford.edu/~ullman/mmds/ch2.pdf [Accessed 21 June 2021].

Machinelearningplus.com. 2021. [online] Available at: https://www.machinelearningplus.com/nlp/cosine-similarity/ [Accessed 4 July 2021].

Medium. 2021. *Relationship between Cosine Similarity and Euclidean Distance.*. [online] Available at: https://medium.com/ai-for-real/relationship-between-cosine-similarity-and-euclidean-distance-7e283a277dff [Accessed 4 July 2021].

Nowak-Brzezińska, A. and Horyń, C., 2020. Exploration of Outliers in If-Then Rule-Based Knowledge Bases. *Entropy*, [online] 22(10), p.1096. Available at: https://pubmed.ncbi.nlm.nih.gov/33286864/ [Accessed 9 July 2021].

ResearchGate. 2021. *Figure 1. Example of inter-class and intra-class cluster similarity In...*. [online] Available at: https://www.researchgate.net/figure/Example-of-inter-class-and-intra-class-cluster-similarity-In-this-paper-we-emphasis-only_fig1_280627665 [Accessed 9 July 2021].

Sreedhar, C., Kasiviswanath, N. and Chenna Reddy, P., 2017. Clustering large datasets using K-means modified inter and intra clustering (KM-I2C) in Hadoop. *Journal of Big Data*, [online] 4(1). Available at: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-017-0087-2 [July 2021].