Stephen Reilly – 201527474

CA2 Assignment 2
Data Clustering
Implementing the k-means and k-medians clustering algorithms

Introduction

The main objective of this assignment is to implement the k-means and k-medians algorithm from scratch and apply them to a provided dataset to see which model and k value works best for given the dataset. Due to us knowing the labels of the dataset it would be expected that for all 4 models a k of 4 would yield the best results due to there being 4 classes of data.

Design Choices

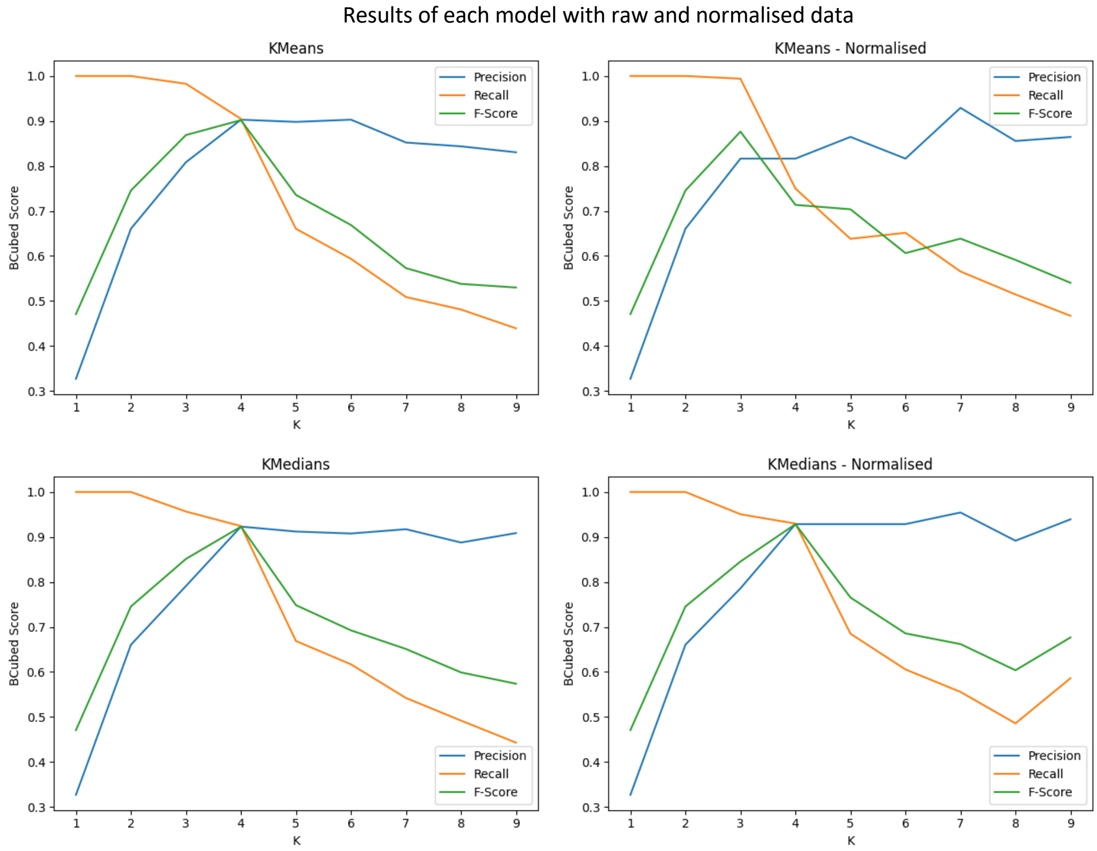
A few design choices made when developing the algorithms. One was to use random starting positions for the centroids in each model based off the min and max values of each feature. This would ensure that the starting clusters would be randomised in a way to be in some way representative of the data we are trying to cluster, this is also paired with using a random seed of 42 to ensure reproducibility. Another was to give each datapoint a classification label which would be needed for the BCUBED scores. The final design choice was to use convergence as the stopping method, this was to ensure that the most accurate model was attained and that the model was not computationally wasteful in running iterations where no change was happening. While there are many ways to achieve these choices they were made to ensure a robust and more accurate model.

Results

The data was passed to the models both raw and normalised and the results can be seen below.

As can be seen in all the models as k increases the precision will increase with it, although as k grows the amount that the precision score increases slows until it is stable and will decrease on some k values. This makes sense as precision is a measurement of the true positives within each cluster, as there are 4 classes when k is less than 4 the precision is understandably low, and as it grows the likelihood of the same classes being clustered together grows. Even when k is beyond 4 there is a high chance that the additional clusters will be smaller cluster of those 4 classes and therefore precision stays quite high. Recall is the measurement of true positives in relation to the whole dataset, this is then the inverse to the precession so recall will start high and begin to decrease as the data is becomes more clustered which will result in the same classes being potentially separated. Recall and precision are often in tension with one another such that if you increase one you decrease the other, this is when F-Score comes in useful.

The F-Score then acts as a harmonic mean between the two scores, this is used to emphasise the impact of small outliers and minimise the impact of large outliers, thus not allowing a large or small precision or recall swaying the results too much. When comparing the models, we will use the F-Score as the metric to compare them against one another. This is due to it being a good indication of an overall balanced model as it is the harmonic mean between the precision and recall. There are situations where you may want to choose the model with the best precision or the best recall depending on the use case but for this dataset the F-Score seems the best metric to use.



K-means

When the data is passed to k-means raw the best clustering occurs when k is 4 with an F-Score of 0.90 and then when normalised this score drops to 0.71 when k is 4. However, when k is 3 the normalised F-Score increases to 0.88. This shows that when the data has been normalised the best clustering is 3 clusters even though this the data itself has 4 different classes. As the dataset is a collection of words converted to vector form along 300 axis when the data is normalised, this could cause distances that were significant between datapoints to distinguish differences are now much smaller, this could then result in words that were in two separate classes now looking very similar to one another and why when normalised and the mean is used to update the centroid positions these classes are grouped closer together. This also causes precision and recall scores to behave erratically, and one could suggest that normalising this dataset if using k-means would not be advised and the raw dataset if the better data to use.

K-Medians

When the data is passed to the k-medians raw the best clustering occurs when k is 4 with an F-Score of 0.923 and when normalised the best clustering is still 4 and slightly improves with a score of 0.9285 showing that normalisation has very little effect on the data. This is likely due to while the values themselves may change the actual median is not as sensitive to these changes compare to the mean.

Conclusion

To conclude, as with any machine learning algorithms there are several factors that can affect the outputted results. These include the randomness of the starting positions of the centroids, the stopping conditions and the suitability of the data as just a few examples. In the case of this dataset and models for k-means, k-medians and k-medians normalised the best clustering is when k is 4 and for k-means normalised the best clustering is when k is 3. With a normalised k-medians being the overall best algorithm to use with the highest precision, recall and F-Score.