A Technical Review of Clustering

Ang Shu Hui U1922145K
Bachhas Nikita U1921630F
Srinivas Shruthi U1923611G
Unnikrishnan Malavika U1923322E

Abstract- Clustering is the task of categorizing a set of data points into groups so that data points in the same group are more similar to each other than data points in other groups. In a nutshell, the goal is to divide groups with similar characteristics into clusters. This study presents a review of several common clustering algorithms - *K* Means, Agglomerative clustering and DBSCAN. First, the report explains the concept of clustering, following which the three methods are discussed, and their properties are analysed. To conclude, the report summarises the strengths and weaknesses of the different methods.

Keywords- Clustering, Clustering Techniques, *K*-Means, Hierarchical, Agglomerative, DBSCAN

1 INTRODUCTION

In a machine learning system, in order to understand and learn the data better, we try to group the unlabelled data based on certain similarities. This process, which relies on unsupervised machine learning, is called clustering. The process of clustering allows the segregation of data into different groups, where points within a group are more similar to each other than points in different groups.[1]

In this study, we will be analysing three classes of clustering models.

First, the connectivity model is based on the distance between the data points and can follow two approaches. In the first method, data points are segregated into clusters and are then combined based on decreasing distance and in the second method, the data points are in a single cluster and are then segregated based on increasing distance. The type of connectivity model that we are implementing is agglomerative hierarchical clustering, which is an example of the first method.

The second class of centroid models follows the concept of finding the closeness of the data point to the centroid of the predetermined clusters. This means that the number of clusters must be known beforehand and so should all the information of the dataset. The simplest, most common model that follows this is K-means clustering. This iterative clustering algorithm seeks the local maxima in each iteration.

The third class consists of density models such as DBSCAN which associates data points in regions with the same density to a cluster and mainly differentiates between clusters based on varying densities in the data space. DBSCAN is the spatial clustering of applications with noise.

2 LITERATURE REVIEW

The exploration and analysis of large data sets in order to discover meaningful rules and patterns is known as data mining. Data mining is a multistep method that entails accessing and preparing data for mining, evaluating the results, and taking appropriate action. Clustering is a common

data mining task which involves discovering groups and structures in data that are "similar" in some way or another, without the use of previously known structures in the data.[2]

According to the paper, "Survey on Clustering Techniques in Data Mining" , published in the International Journal of Computer Science and Information Technologies, a successful clustering approach will yield high-quality clusters with high intra-cluster similarity but low inter cluster similarity. The quality of a clustering result is determined by both the similarity measure employed by the method and its implementation. The capacity of a cluster created by a clustering method to uncover some or all of the hidden patterns also contributes to its quality.

Other needs for clustering algorithms include scalability, the capacity to deal with insensitivity to input record order, and the ability to deal with noisy data.

Clustering plays an important role in information retrieval, data mining and document classification. The most common clustering methods include hierarchical and partitional clustering.[3] Both of these are explored within this report. Clustering techniques have been proven to be useful in many disciplines. The pattern recognition framework, for example, has been mentioned in Duda and Hart (1973) and clustering algorithms suitable for image segmentation and computer vision can be found in Jain and Flynn (1966). In data mining, clustering was popularised by the growth in research in information retrieval and text mining (Dhillon et al., 2001). Over the years, various papers have produced a plethora of algorithms and visualisations for efficient clustering which greatly aided the data mining and analysis process in a myriad of industries. The contributions to the data analytics industry are vast and inspiring. In this report, a simpler analysis of the common clustering methods are performed.

3 METHODS

3.1 K Means Clustering

K Means clustering is a centroid-based algorithm or a distance-based algorithm, where we calculate the distances between different data points and compare it with the other distances. The data point will be grouped in the cluster which has the smallest distance between the point and its centroid.

Each cluster has one centroid, an “artificial” point within a cluster representing the average of all the data points within it.

The main objective of this method is to minimise the sum of distances between the points and their respective cluster centroid.

The algorithm consists of the following steps:

Initialisation: The parameter *K*, the number of clusters, has to be chosen first.

Randomisation: Data points are chosen at random to be centroid. K data points have to be selected.

Clustering: Calculating the distance between a data point and the randomly selected *K* centroids and assigning it to the cluster with the smallest distance.

Recomputation: Re-calculate the mean value of the cluster and reassign the centroid value.

Repetition: The clustering and recomputation steps are repeated again

Termination: The program is only terminated when any of the following three criteria have been achieved:

1) Centroids of newly formed clusters do not change

2) Points remain in the same cluster
3) Maximum number of iterations are reached (This has to be preset beforehand.)

3.1.1 Calculation of Distance between Data Points and Centroids

There are many different methods to calculate the distance between the different data points and centroids such as the Euclidean distance, Cosine similarity, Jaccard Similarity, e.t.c.
In our calculations, we have used Euclidean distance to measure these distances.
Euclidean distance between data points can be found out through the following equation:

$$D(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2)}$$

Euclidean distance

Figure 1: Calculation of Euclidean Distance between two data points

The data points is represented as a 2-Dimensional vector with the coordinates being *(x,y)*

3.1.2 Calculation of Mean Value of a Cluster

If data is in a numerical state and represented as a 2-Dimensional vector, we simply get the mean X-Coordinate and mean Y-Coordinate of all the data points present in the cluster to get the coordinates of the centroid.
However, for categorical data, we have to use a different way to calculate mean. These can be some of the methods implemented.

1) Replace it with a clusteroid
2) Replace categorical data such as red and blue in the colour wheel with suitable similarity metrics, such as using the Hamming distance
3) Categorical values can be transformed to real-valued embeddings.

3.1.3 Setting parameter *K* and Initialisation Criteria

1) Choosing *K:* *K* represents the number of clusters that has to be chosen before this algorithm can be carried out. There are some heuristic approaches that have been developed in an attempt to get the most accurate *K* as possible:
    a) Minimum Message Length (MML)
    b) Minimum Description Length (MDL)
    c) Bayes Information Criterion (BIC)
    d) Akaike Information Criterion (AIC)
    e) Dirichlet Process
    f) Gap Statistics

    In our calculations, we have plotted a graph with the number of clusters against cluster Inertia, and we use the Elbow Method, the point where the gradient of the graph changes drastically to select the most optimal *K* value. This is an example of what the graph looks like:

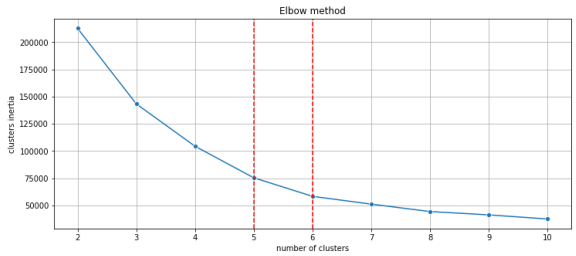


Figure 2: Elbow Method to find and set parameter *K*

2) The Initialisation Criteria is used to select the initial *k* centroids. *K* Means randomly selects these few centroids. However, this can greatly

affect the end results and hence, we can instead use *K* Means++, proposed by Arthur and Vssilvitskii, which generates initial centroids to be as distant from each other as possible so that better results can be obtained.

3.2 Hierarchical Clustering: Agglomerative Clustering

Agglomerative clustering is a type of hierarchical clustering and is a bottom-up approach.
It consists of the following three steps:
Initialization: Each data point is initialised as a single cluster. This means that one data point forms one cluster.
Computation: The similarities of clusters is calculated and
Calibration: The two most similar clusters/data points will be merged into a parent cluster. (i.e. the two "nearest" clusters are combined into to form one bigger cluster)
Repetition: The previous step is continuously repeated by combining smaller clusters into parent clusters.
Termination: The program is only terminated when either of the following two criteria have been achieved:

1) Inputted target number, K, of clusters have been merged (must be decided beforehand or must be known beforehand)
2) When a certain set criterion is met:
    a) If diameter of the cluster exceeds a preset threshold
    b) If density of the cluster is below a preset threshold
    c) If merging the clusters results in a bad cluster*
3) Formation of clusters stops when only one cluster remains.

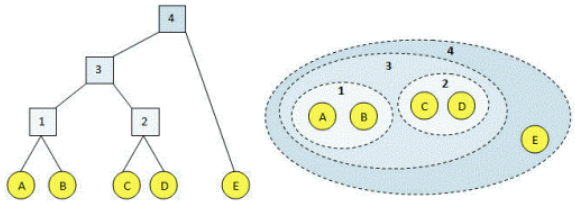


Figure 3: Cluster Formation in Hierarchical: Agglomerative Clustering

*Bad Cluster: When addition of a data points cause the diameter of the cluster to change drastically (i.e. the data point is too far away to be counted as part of the cluster)

3.2.1 Calculation of Similarity

Similarities of the clusters can be computed through various methods such as Single Linkage, Complete Linkage, Average Linkage, Centroid Method and Ward's Method.
For our calculations we have used, we have used Ward's method to find out the similarities between clusters.
Ward's method states that we combine clusters where the increase within cluster variance is to the smallest degree. This allows the cluster variance to be minimised as much as possible.

3.2.2 Setting parameter K, Diameter Threshold, Density Threshold

1) Choosing K: K represents the target number of clusters that need to have merged before the program is terminated
2) Diameter threshold is another alternative variable that helps decide when to terminate the program. The diameter of the threshold must be greater or equal to this threshold for the program to be terminated.

3) Density threshold is another variable that can determine if merging of the clusters should be stopped. If the density of the clusters reaches below a certain threshold then the program is terminated.

In our calculations, we have decided to set the number of clusters merged, K, manually depending on the dendrogram that it forms. (refer to section 6.2)

3.3 DBScan

Before we discuss the algorithm itself, we must first look at some key parameters and definitions-

- Epsilon_radius (eps) - the maximum distance between two points to be considered neighbours
- Minimum number of points (minPts) - minimum number of data points required to form a cluster
- Neighbourhood - Area around a point determined by a circle of radius eps
- Core Point - A point that has at least minPts number of points (including itself) in its neighbourhood
- Border Point - A point that has less than minPts number of points but has at least one core point in its neighbourhood
- Outlier - A point that has less than minPts number of points and no core points in its neighbourhood

3.3.1 Algorithm

In this section, we present the algorithm DBSCAN (Density Based Spatial Clustering of Applications with Noise). DBSCAN (Density Based Spatial Clustering of Applications with Noise) is a density based clustering algorithm meaning that it works best for data points with varying densities. We require parameters Eps and MinPts of each cluster to be set before we begin clustering using this algorithm. These parameters will remain the same across all clusters.

To find a cluster, the algorithm chooses an arbitrary point P. If P is a core point, a new cluster C(P) is created. C(P) is then expanded using the Expand(P) function (explained below)
Otherwise P is marked as an Outlier (Noise).
This is repeated until every point is assigned

Expand(P) - using the Breadth-First-Search algorithm (BFS)
For every point in the neighbourhood of P - P', if P' is a core point, all neighbourhood points of P' are added to the FIFO queue of BFS. Otherwise, if P' is not a core point, it is treated as a leaf node (no further expansion). This is repeated until the FIFO queue of BFS is empty.

3.3.2 Setting parameters Epsilon (eps) and Minimum Number of Points (minPts)

The parameter minPts is chosen based on knowledge of the dataset.

- The larger the dataset, the larger the value of minPts
- The noisier the dataset, the larger the value of minPts
- minPts should also be greater than the dimensionality of the dataset

Once a suitable value for minPts is found, epsilon is chosen using a K-distance graph where k = minPts chosen previously.

1. The average distance between each point in the dataset and its nearest k neighbours is calculated.

2. These distances are then sorted in ascending order.
3. A graph between the data points and the calculated (sorted) distances is plotted
4. The elbow point in the graph is chosen and its corresponding distance value is chosen as epsilon

4 ADVANTAGES AND DISADVANTAGES OF METHODS

Method	Advantages	Disadvantages
K-Means	● It is simple, fast and easy to implement. ● It minimises within-cluster variances. ● If groups are truly well isolated from each other, there is a high likelihood that the data points may converge to the global minimum and not just a local minimum.	● Requires number of clusters to be specified beforehand when setting the parameter *K*. This limits flexibility while clustering and is also highly dependent on the initialisation of the centroids. The most ideal way to select initial centroids that are far away from each other. This is done is an improved version of the *K* Means algorithm known as the *K* Means++ ● Selecting too few data points as centroids for clusters can lead to inefficiency and longer run times as average distance to centroids increases. ● Selecting too many data points as centroids reduces accuracy of clustering. ● It is extremely sensitive to outliers. ● The algorithm does not work well on data with a non-spherical shape.
Agglomerative Hierarchical Clustering	● High Simplicity: It is a simple and easy to understand and use algorithm ● Agglomerative clustering saves the effort of tuning the k parameter since it is optional for the user to set the value of k for agglomerative clustering. ● Once the clustering has finished, the user can obtain the number of clusters simply by looking at the dendrogram. ● There is no assumption about the shape of the cluster and hence, the algorithm has the flexibility to find the best shape for the clusters in the	● It is important to take care of outliers as they could cause the issue of cluster merging, also known as chaining. ● There is a high potential for a bad cluster forming.

	data points.	
DBSCAN	• Can discover arbitrarily shaped clusters (eg. clusters completely surrounded by other clusters) • Robust toward outlier/noise detection. • Does not require the number of clusters to be specified beforehand	• Sensitive to clustering parameters minPts and epsilon. Choosing a meaningful eps value can be difficult if the data isn't well understood. • Fails to identify clusters if density varies and if the dataset is too sparse. • Not entirely deterministic. This is because the algorithm starts with a random point. Therefore border points that are reachable from more than one cluster can be part of either cluster.

Table 1: Advantages and Disadvantages of various clustering methods

5 DATASET ANALYSIS

3 datasets - Frequent Flyer Program, Wine, and Mall Customer - are selected to perform clustering algorithms on.

5.1 Frequent Flyer Program

This dataset contains information about the behavior of NZ Airline's FFP customers.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3999 entries, 0 to 3998
Data columns (total 8 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   FFP#               3999 non-null   int64
 1   AwardMiles         3999 non-null   int64
 2   EliteMiles         3999 non-null   int64
 3   PartnerMiles       3999 non-null   int64
 4   PartnerTrans       3999 non-null   int64
 5   FlyingReturnsMiles 3999 non-null   int64
 6   FlightTrans        3999 non-null   int64
 7   EnrollDuration     3999 non-null   int64
dtypes: int64(8)
memory usage: 250.1 KB
```

Figure 7: Variables in Frequent Flyer Program dataset

These are the variables information. There are a total of 8 variables - FFP#, which is a unique key value, AwardMiles, EliteMiles, PartnerMiles, PartnerTrans, FlyingReturnsMiles, FlightTrans and EnrollDuration.

In Figure 8, we can see that EnrollDuration shows near normal distribution and All other variables show skewed distribution. There isn't any clear cluster segregation visible from the pairplot. There is a positive correlation between FlightTrans and FlyingRetursMiles, and between PartnerTrans and PartnerMiles.

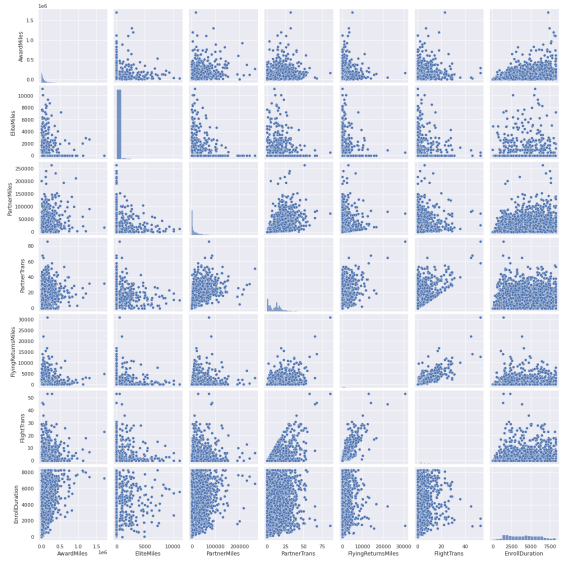


Figure 8: Pairplot of the variables in Frequent Flyer Program dataset

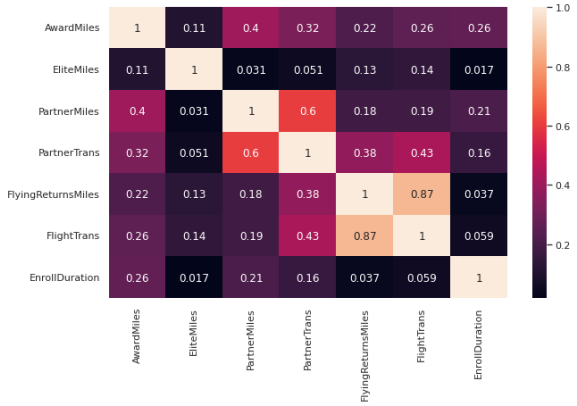


Figure 9: Heatmap of variables in Frequent Flyer Program dataset

From Figure 9, it is evident that there is a positive correlation between PartnerTrans and PartnerMiles, and between FlightTrans and FlyingReturnsMiles.

In conclusion, when we are building our clustering models, we will drop PartnerTrans and FlightTrans. The models will be build with the following variables:
AwardMiles, EliteMiles, PartnerMiles, FlyingReturnsMiles, and EnrollDuration.

5.2 Mall Customer

This dataset contains the basic information about the customers.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   CustomerID              200 non-null    int64
 1   Gender                  200 non-null    object
 2   Age                     200 non-null    int64
 3   Annual Income (k$)      200 non-null    int64
 4   Spending Score (1-100)  200 non-null    int64
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
```

Figure 10: Variables in Mall Customer dataset

These are the attribute information. There are a total of 5 variables: CustomerID, which is a unique key, Gender, Age, Annual Income and Spending Score. The datatypes of the variables are also indicated in Figure 10.

```
df.isnull().sum()

Class_Label              0
Alcohol                  0
Malic_Acid               0
Ash                      0
Ash_Alcanity             0
Magnesium                0
Total_Phenols            0
Flavanoids               0
Nonflavanoid_Phenols     0
Proanthocyanins          0
Color_Intensity          0
Hue                      0
OD280                    0
Proline                  0
dtype: int64
```

Figure 11: Null Values in Mall Customer Dataset.

Output in Figure11 shows that there are no null values in this dataset.

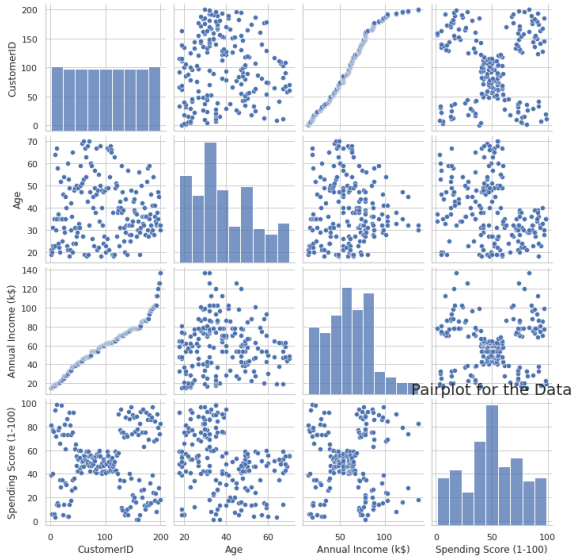


Figure 12: Pairplot of Variables in Mall Customer dataset

It is worth noting that customers between the ages between 20 and 40 have a higher spending score.

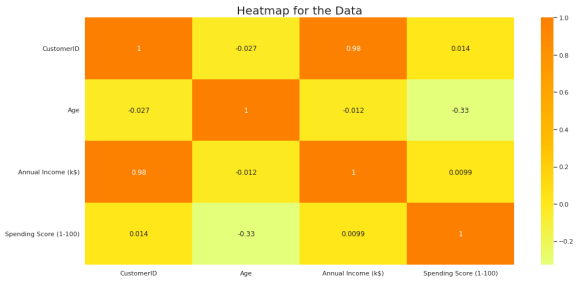


Figure 13: Heatmap of variables in Mall Customer dataset

Figure 13 shows the correlation between the different attributes of the Mall Customer dataset. These attributes do not have good correlation among them, hence we can proceed with all numeric variables when building our clustering models.

5.3 Wine

This dataset contains information about different types of wines

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 178 entries, 0 to 177
Data columns (total 14 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Class_Label           178 non-null    int64
 1   Alcohol               178 non-null    float64
 2   Malic_Acid            178 non-null    float64
 3   Ash                   178 non-null    float64
 4   Ash_Alcanity          178 non-null    float64
 5   Magnesium             178 non-null    int64
 6   Total_Phenols         178 non-null    float64
 7   Flavanoids            178 non-null    float64
 8   Nonflavanoid_Phenols  178 non-null    float64
 9   Proanthocyanins       178 non-null    float64
 10  Color_Intensity       178 non-null    float64
 11  Hue                   178 non-null    float64
 12  OD280                 178 non-null    float64
 13  Proline               178 non-null    int64
dtypes: float64(11), int64(3)
memory usage: 19.6 KB
```

Figure 14: Variables in Wine dataset

These are the variables information. There are a total of 13 variables: Class_label (1, 2, and 3 respectively), Alcohol, Malic Acid, Ash, Alcanity of ash, Magnesium, Total phenols, Flavanoids, Nonflavanoid phenols, Proanthocyanins, Color intensity, Hue, OD280 and Proline.

The datatypes of the variables are also indicated in Figure 14. There are no null values in this dataset.

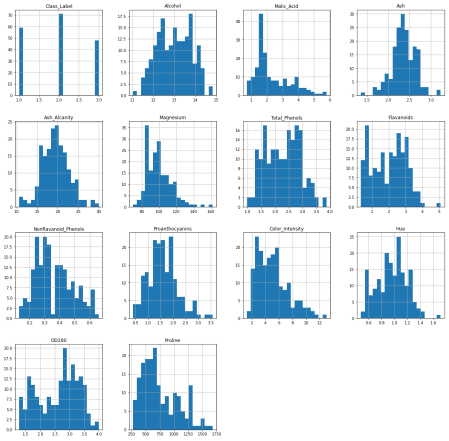


Figure 15: Histograms of the variables

The distributions of the variables are shown in Figure 15. As the class label is given, it is predictable that there will be 3 clusters available.

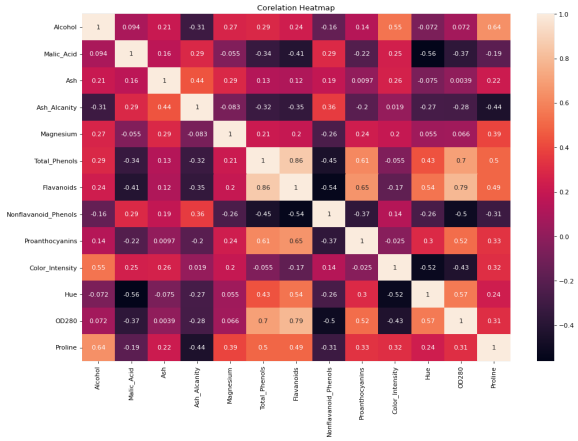


Figure 16: Heatmap of the variables for Wine dataset

Total Phenols, Ravanoids, Hue, OD280 and Proline show a strong negative correlation with the class label.

It is also important to note the following for us to select the variables for model building later.

- Ash_Alcanity has a positive correlation with Ash.

- OD280 has a strong positive correlation with Total_Phenols and with Flavanoids
- Proanthocyanins has a strong positive correlation with Total_Phenols and with Flavanoids

Therefore, we will drop the variables - Ash_Alcanity, OD280, and Proanthocyanins- when we build the clustering models.

6 EXPERIMENTS

We have used 3 clustering algorithms - K Means Clustering, Hierarchical Clustering and DBScan for each dataset that we have chosen.
Here is a summary of the variables that we have included in the clustering models for each dataset:

- **Frequent Flyer Program**
  AwardMiles, EliteMiles, PartnerMiles, FlyingReturnsMiles, and EnrollDuration
- **Mall Customer**
  Age, Annual Income and Spending Score
- **Wine**
  Alcohol, Malic Acid, Ash, Magnesium, Total phenols, Flavanoids, Nonflavanoid phenols, Color intensity, Hue, and Proline.

6.1 K Means Clustering

The table below shows the elbow plot of K Means clustering for each dataset.

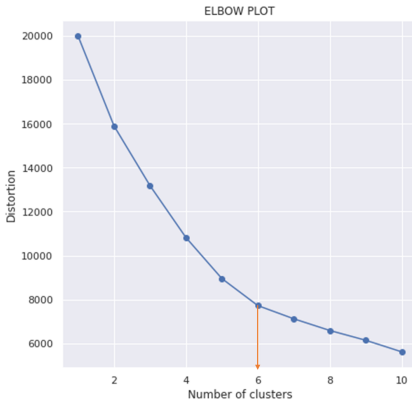
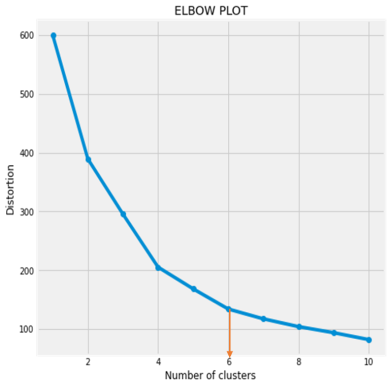
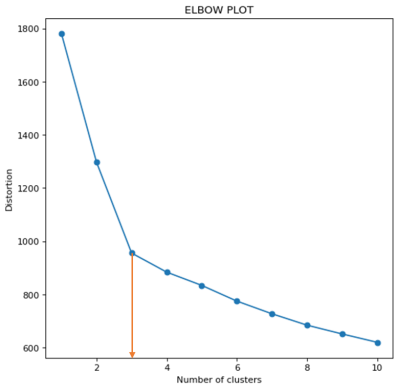
Frequent Flyer Program	Mall Customer	Wine
 Figure 17:Elbow Method for Frequent Flyer Program	 Figure 18:Elbow Method for Mall Customer	 Figure 19:Elbow Method for Wine
k = 6	k = 6	k = 3

Table 2: Elbow plot - K-Means Clustering

After obtaining the k value from the graphs, we proceeded to execute K Means clustering on each dataset.

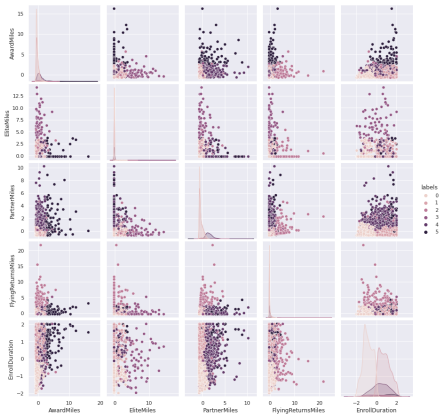
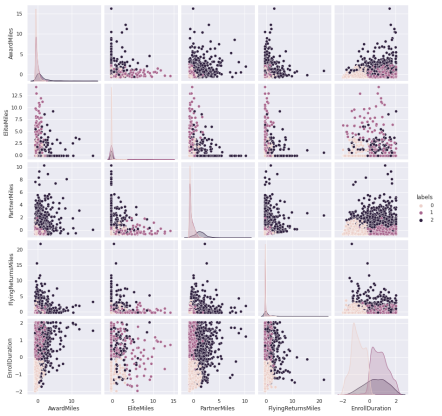
Frequent Flyer Program	
K = 6 (from elbow plot)	K = 3
 Figure 20	 Figure 21

Table 3: K-Means Clustering Results for Frequent Flyer Program

From this result, we can see that the k value we selected from the elbow plot is not the optimal k value as the clusters when k = 6 are less distinct than the clusters when k = 3. It is also worth noting that K Means clustering is not a good option for this dataset as most clusters overlap.
The table below shows the result of K Means clustering for the Mall Customer dataset

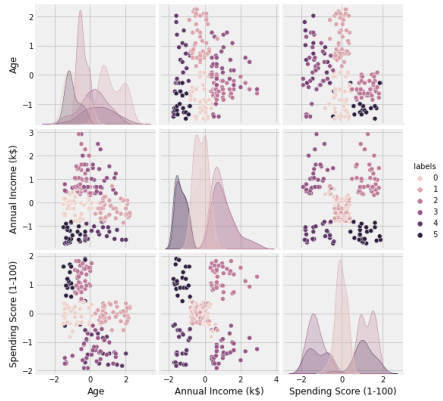
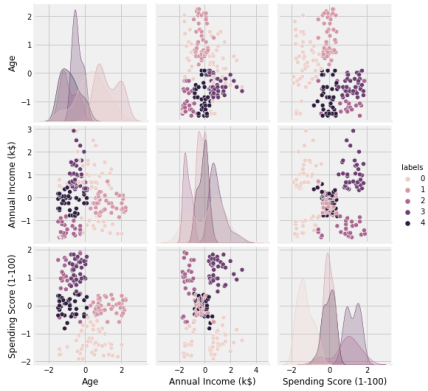
Mall Customer	
K = 6 (from elbow plot)	K = 5
 Figure 22	 Figure 23

Table 4: K-Means Clustering Results for Mall Customer

The clusters formed when K = 6 are more distinct than the clusters formed when K = 5. The clusters do overlap in the middle range of values.

Wine
K = 3 (from elbow plot)

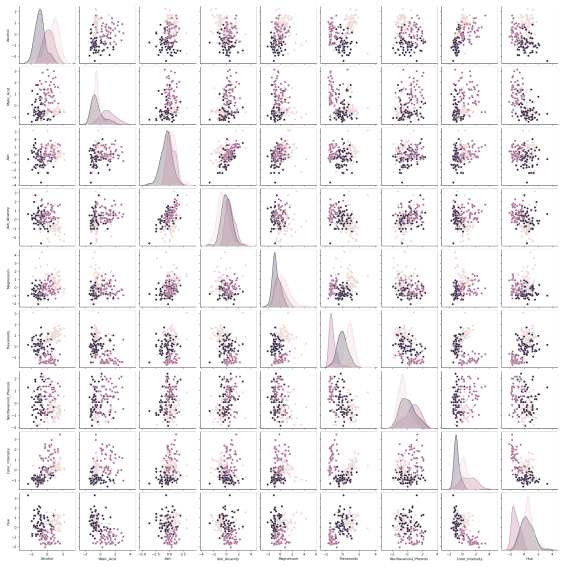



Figure 24

Table 5: K-Means Clustering Results for Wine

For the Wine dataset, there are no distinct clusters formed with K Means clustering.

6.2 Hierarchical Clustering

Next, we executed hierarchical clustering on the datasets. For hierarchical clustering, we would either need to declare n_clusters or the distance_threshold. In our experiments, we declare the n_clusters based on the number of clusters we found by plotting the dendrogram. The result is as shown below. We can see that the clusters overlap each other as well.

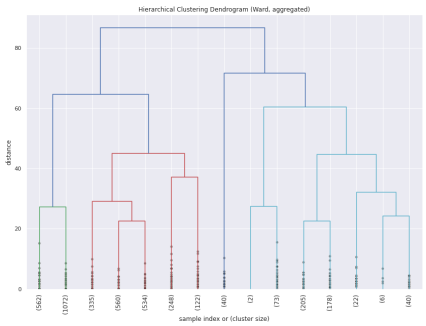
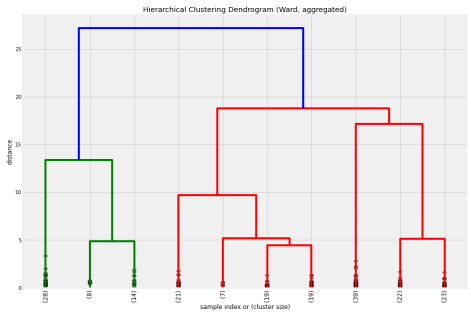
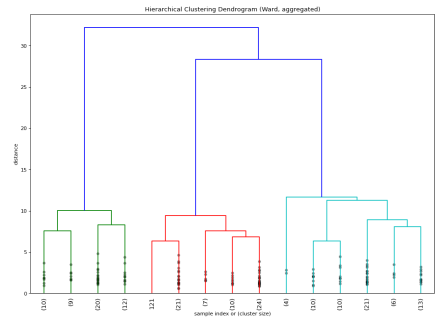
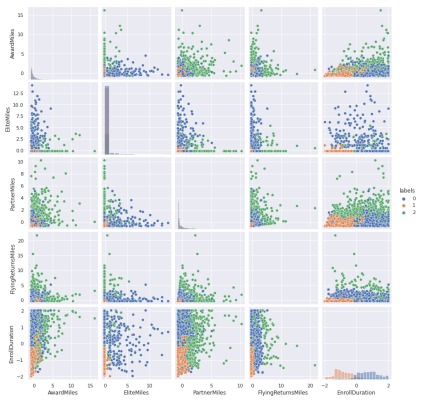
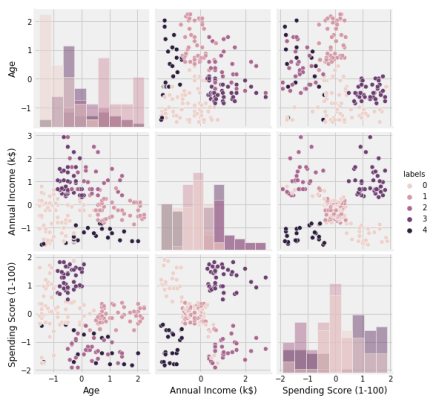
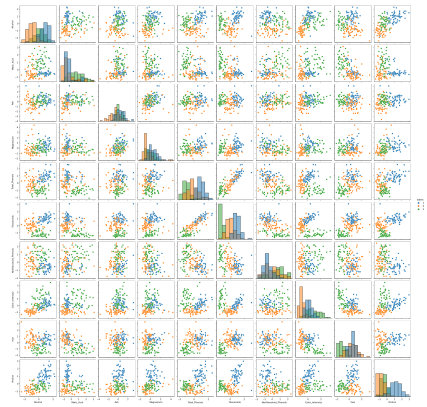
Frequent Flyer Program (k = 3)	Mall Customer (k = 5)	Wine (k = 3)
		
		

Table 6: Results of Hierarchical Clustering

6.3 DBScan

Finally we executed DBScan clustering on the datasets. We had to determine the optimal minPts value based on our analysis of the datasets and its corresponding optimal epsilon radius from the k-distance graph plotted for each dataset for a particular value of k = minPts. The results are tabulated below.

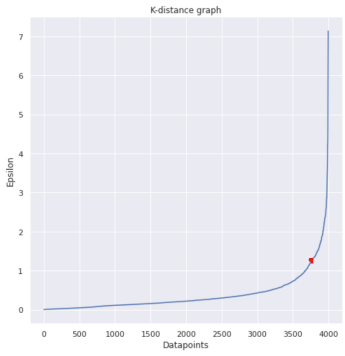
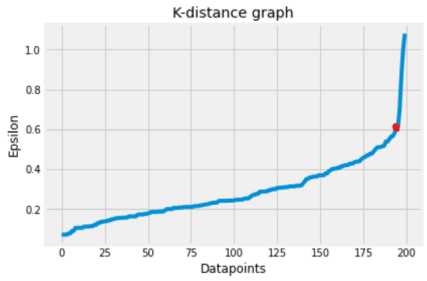
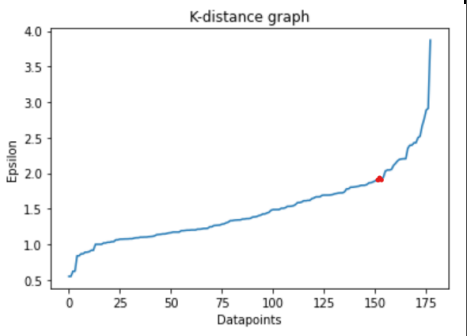
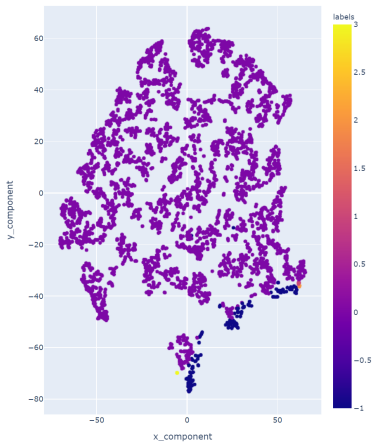
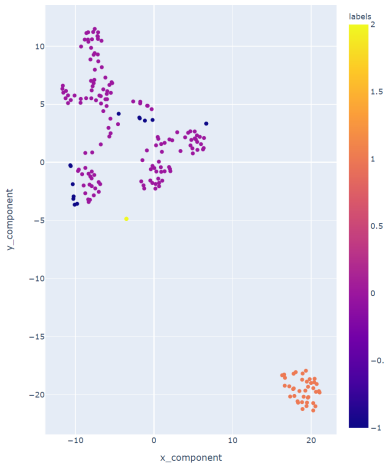
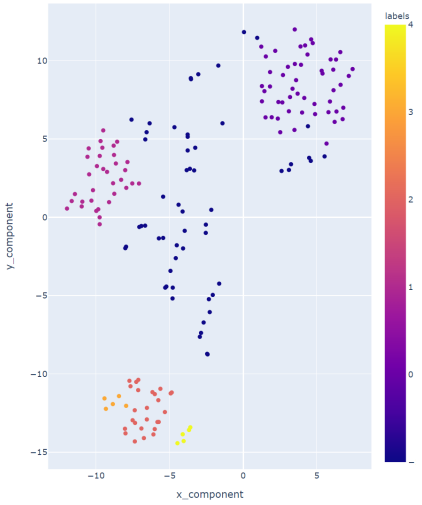
Frequent Flyer Program minPts = 5, eps = 1.3 No. of clusters = 4 Noise = 153/3999	Mall Customer minPts = 3, from the graph eps = 0.6 No. of clusters = 3 Noise = 14/200	Wine minPts = 5, from the graph eps = 1.9 No. of clusters = 5 Noise = 57/178
		
		

Table 7: Results of DBScan Clustering

7 COMPARISON

The following table shows the difference between the three different types of method for clustering:

Comparison Factor	Dataset	*K* Means	Agglomerative	DBScan
Number of Clusters	Flyer	6	3	4
	Mall	6	5	3
	Wine	3	3	5

Table 8: Methods Comparison

As seen in the above comparison, all three methods result in a different number of clusters formed or merged. No dataset has had the same results from all three methods. The wine dataset has returned results with the highest accuracy with *K* Means and Agglomerative returning 3 clusters and DBScan returning 5 clusters.
It would be unfair to merely compare numbers of the different methods against each other as each method depends on different parameters and *K* Means and Agglomerative methods are also highly sensitive to abnormal data.

8 CONCLUSION

By means of this analysis and documentation, the various methods of clustering - their working and effectiveness have been explored. It has been found that different datasets behave differently to different clustering methods. Thus the correct algorithm to be chosen can be decided only upon in depth analysis of the database and its variables. The clustering methods respond differently to the way the data is arranged within each database and also depend upon different sets of parameters. While some clustering methods work on a divisive approach, others work on combining clusters of data together. In short, one cannot blindly apply any clustering algorithm to any dataset and expect to get useful results. As the field of data analysis continues to grow, we are on a constant search to find patterns and connect the dots to relate something new to something familiar. The concept of clustering has played a vital role in numerous industrial advancements and will continue to play a pivotal role in helping us to understand data. Clustering is a very powerful tool and a good understanding of its principles are important for it to be used effectively.

9 REFERENCES

[1]S. Kaushik, "Clustering | Types Of Clustering | Clustering Applications", Analytics Vidhya, 2016. [Online]. Available: https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/#h2_9. [Accessed: 18- Nov- 2021].
[2]K. K and M. K, "Survey on Clustering Techniques in Data Mining", 2014. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.440.2001&rep=rep1&type=pdf. [Accessed: 18- Nov- 2021].
[3]P. K and N. AM, "A Literature Review on Document Clustering", 2010. [Online]. Available: https://www.researchgate.net/profile/Kandasamy-Premalatha/publication/47690250_A_Literature_Review_on_Document_Clustering/links/0deec534284eee4950000000/A-Literature-Review-on-Document-Clustering.pdf. [Accessed: 19- Nov- 2021].

[4]Duda, R. and P.Hart, Pattern Classification and Scene Analysis, 1973
[5]Jain, A.K. and P.J. Flynn, Image Segmentation using Clustering, 1996
[6]Dhillon, I., J.Fan and Y. Guan, Efficient Clustering of Very Large Document collections, 2001