Course Project N.19: Fruits

Ariola Leka
Francisco Amoros Cubells

June 1, 2023

Abstract

The main goal of this project, is to explore the dataset containing observations of three different fruits, aiming to cluster them based on their characteristics. Fruit clustering involves grouping fruits based on their shared characteristics or attributes. The objective is to develop an effective fruit clustering system using clustering algorithms. This analysis will help in understanding the similarities and differences amongst fruits, providing insights into their unique characteristics.

First, we extensively explore and describe the dataset by employing standard descriptive statistics and visualizations. We examine the variables through graphs, distributions, and histograms, identifying any outliers and correlations. Through this analysis, we gain insights into the unique characteristics of each fruit and their potential discriminative features.

Next, we preprocess the data, handling the few missing cells that were present in the dataset. Also, we normalise the values per each feature, and remove the outliers. By applying these transformations, we ensure the robustness and reliability of our subsequent clustering analyses.

To cluster the fruits, we employ three distinct clustering algorithms. We use **Agglomerative** algorithm, **Gaussian Mixture Model (GMM)** and **K-Means**. We use the accuracy, recall, precision as well as the confusion matrix to evaluate each model.

We concluded that due to the overlapping of the features for two of the fruits, regardless of the feature combination, the optimal number of clusters was **2**.

1 Introduction

Fruit clustering can be a very important task, specifically in the field of Agriculture, food industry and nutrition. By separating fruits in different clusters, it can help: **Product Classification**, **Quality Control**, **Nutritional Studies** and many other sectors. The goal of this project is to built at least 2 models able to cluster the different fruits. The dataset we have to reach this goal has 4 features plus the labels (which are going to be explained in details in the section below).

Before applying any clustering algorithm, we explore the dataset with different visualisation methods. We get an in detail understanding of the correlations, missing values, outliers and overlapping features which will help the clustering.

Once we finish with data exploration and data pre-processing we fit into three different models and get the optimal number of clustering.

2 Explore and describe the data

2.1 Dataset

Our **Fruits** dataset contains 150 values. This dataset contains information about three different fruits : **cherry**, **apple** and **peach**. It includes several features for each fruit, such as the amount of **vitamin A** and **vitamin C**, the **weight**, **diameter**, and the type of fruit. The latest is our target variable. In the Table 1 below, we can see each of the features that our dataset contains, its type and as well a short description.

Nr.	Variable	Type	Description
1	vitaminA	numeric	A nutrient that the body needs in small amounts to function and stay healthy.
2	vitminC	numeric	An antioxidant that helps protect your cells against the effects of free radicals.
3	weight	numeric	The measure of how heavy or light the fruit is.
4	diameter	numeric	The measurement of the distance across the widest part of the fruit, passing through its center.
5	fruit	categorical	A mature ovary of a flowering plant that typically contains seeds.

Table 1: Dataset description

2.2 Data Exploration

To begin with, we check the **csv file** of the dataset without feeding into any data visualisation method, and we see that the dataset contains 150 data/rows.
Next step, we check if we have null values. We get to discover that we only have a few data points missing per feature.

Dataset null values:
vitaminA: 1, vitminC: 1, weight: 2, diameter: 1, fruit: 0.

Having the fact that our dataset is quite small, we checked the file to see for which of the fruits, the respective missing values belongs to.
Null values per fruit:
cherry: weight, peach: vitminC, apple: vitaminA, apple: diameter, apple: weight.

On the next step, we find the standard descriptive statistic for the Dataset as is shown in Table 2

Table 2: Standard descriptive statistic

Metric	Vitamin A	Vitamin C	Weights	Diameter
count	149.0	149.0	148.0	149.0
mean	5838.92617	3059.060403	3.763514	11.953020
std	829.091933	436.821044	1.763467	7.632023
min	4300.0	2000.0	1.0	1.0
25%	5100.0	2799.0	1.6	3.0
50%	5799.0	3000.0	4.35	13.0
75%	6400.0	3300.0	5.1	18.0
max	7900.0	4400.0	6.9	25.0

From table 2 we can see the missing values. From 150 we have 149 for Vitamin A,C and diameter and weights has two missing values. From the min and max row, we observe the maximum and the minimum weight and diameter that the fruits have.

2.3 Variables Relationships

We start exploring the relationship between the variables as shown in Figure 1. As it can be seen, on the **weight** and **diameter** features, there is a clear gap between two groups of fruits based on these two characteristics. More specifically there is a fruit **(small)** which can be easily separated/clustered from the others. Also, we observe that for this small fruit, the majority of the data points have a weight between 1 and 2 and very rarely reaches 2. Same thing for the diameter, most of the points are between 1 and 4 and rarely passes this number. What is certain, is the fact that there are going to be two subgroups of fruits.

On the other hand, if we see vitamin C, in overall we have a balanced distribution of the data across the possible fruits. Also, three of the most appearing vitamin C values in the data-points(**three peaks**), reach the value 20 and above.

Finally, to talk a little bit about **vitamin A**. The distribution seems to be a **Multimodal Distribution** but not quite distinguishable. This might be telling us two things:

- We might have 3 groups (because of 3 peaks)
- There exists an underlying phenomena.

For the moment, these observations are not very clear up to further explorations.

A continuous distribution of the features is shown as well in Figure 3. It can be as well observed the existence of two distinct groups of fruits.

We then continue further our data exploration on searching more for patterns between the data. To give an insight of which features are correlated with which one, we show in Figure 5 the **Pearson Correlation**. Observations made from the correlation plots:

- **weight** and **diameter** have a very high correlation between each other **(96%)**. This comes from the fact that when a fruit has a bigger diameter, deterministically is going to weight more. So practically we can remove one of these variables in our next steps, since together they do not add extra information.
- Similar remark we can do for **vitamin A**, which has a considerable high correlation with **diameter** and **weight** respectively 82% and 87%.
- Finally, we can observe that **vitamin C** has barely a negative correlation (almost no correlation) with any of the three other variables.

The observations we had at the Pearson Correlation Figure 5 we can also see them at Scatter plot in Figure 4.

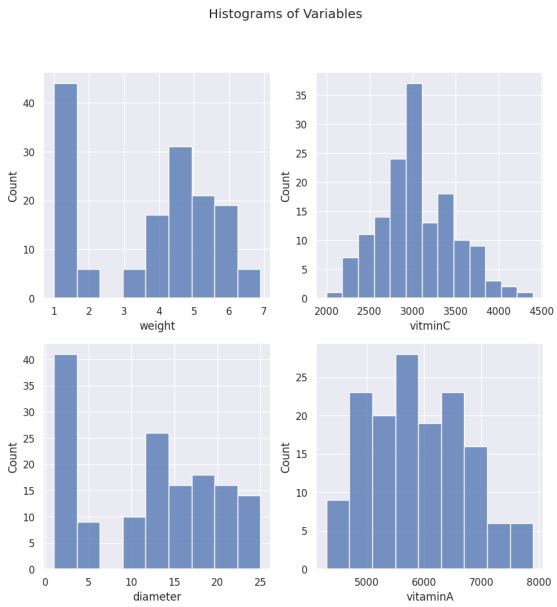

Figure 1: Histogram per variable

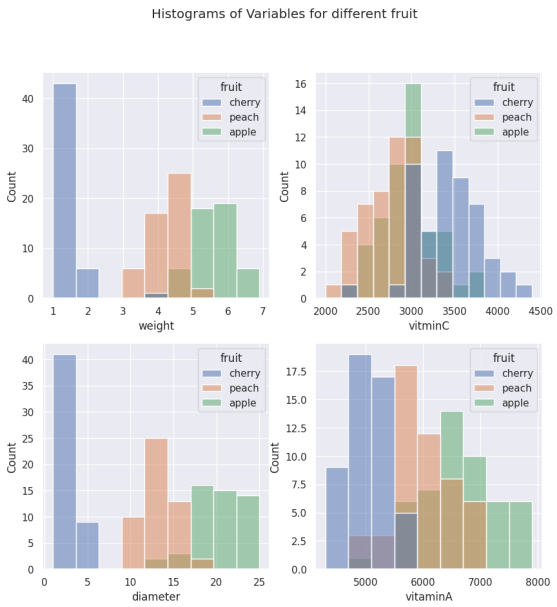

Figure 2: Histogram by fruit

- In the subplot of weight-diameter/diameter-weight we see the high correlation these two features have between each other. This due to the fact that we can linearly fit them in a line.
- **Vitamin A** has high correlation with diameter and slightly higher with weight.
- **Vitamin C** has shown again to have no correlation with any of the other features.

Finally, to show once more the correlation between features we present the barplot on Figure 6.
Observations:

- As the size of the diameter increases, **diameter** so does the **weight** (**positive**, high correlation).
- No pattern between the increase of **weight** and **vitamin C**, but we can say that in overall is slightly a negative correlation.
- No pattern between **vitamin A** and **vitamin C**. In overall shows to be a zero correlation maybe slightly negative, and even more neutral than the previous case.
- High correlation between **vitamin A** and **diameter**, but because of the fluctuations, we see that is going to be less correlated than the first case (diameter-weight).

3 Data pre-processing

In the preprocessing step, we take care of 5 important things:

- Fill/remove missing data.
- Visualise and remove outliers.
- Normalise the data.
- Label encoding.
- Split the data.

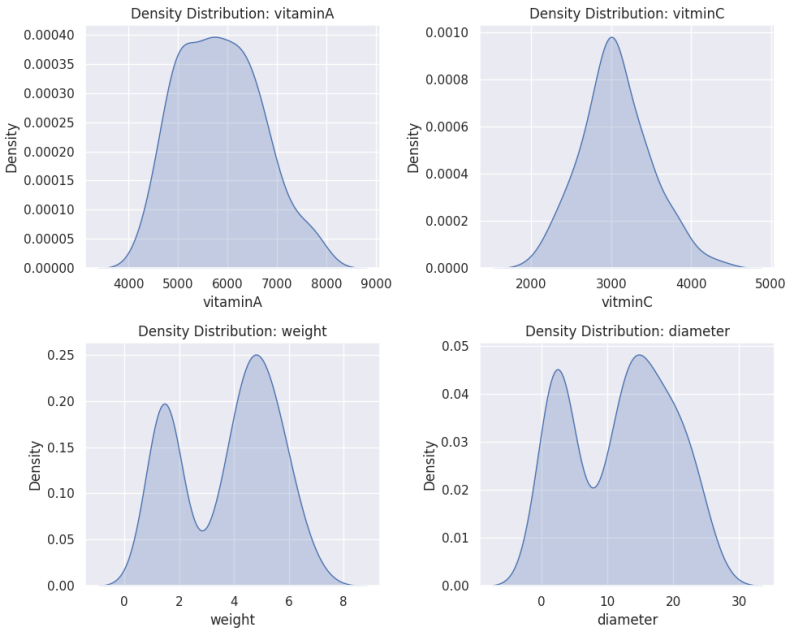

Figure 3: Density distribution for each variable

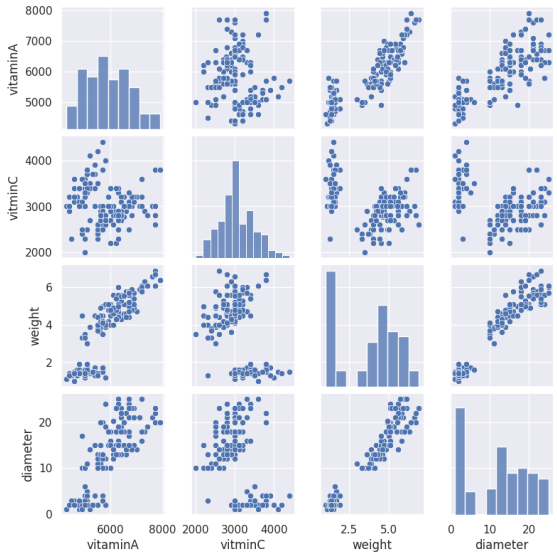

Figure 4: Scatter plot

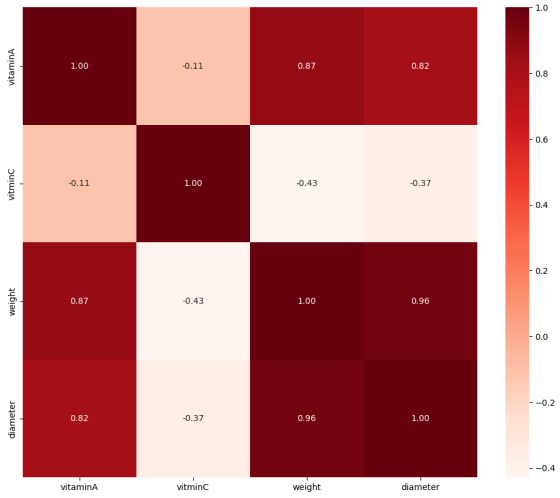

Figure 5: Pearson Correlation between variables

Firstly, we start the preprocessing by **dropping** the null values.

Afterward , we plot a **boxplot**, Figure 7, to easily see the **outliers** and the quartiles of the different features. We observe that in the **'Vitamin C'** feature, exists an outlier. Taking into account that some of the clustering methods are sensitive to outliers, like **K-means**[1], we decided to remove it. To remove the outlier, we drop the row in witch the z-score (how this data point is separated from the mean in terms of standard deviation) is bigger than 3 standard deviations.

We continue with the normalisation of all the numerical features of the dataset. To do so, we use the **MinMaxScaler** function of Sklearn to scale the data in the **(-1, 1)** range.

As a penultimate step, we drop the fruit column (label). We encode it into a 0,1,2 **vector** where 1 is **cherry**, 2 is **peach**, and 0 is the **apple**.

As a final step, we only split the data into **X** being the four features and **y** being the vectorised labels which are going to be used in the evaluation step.

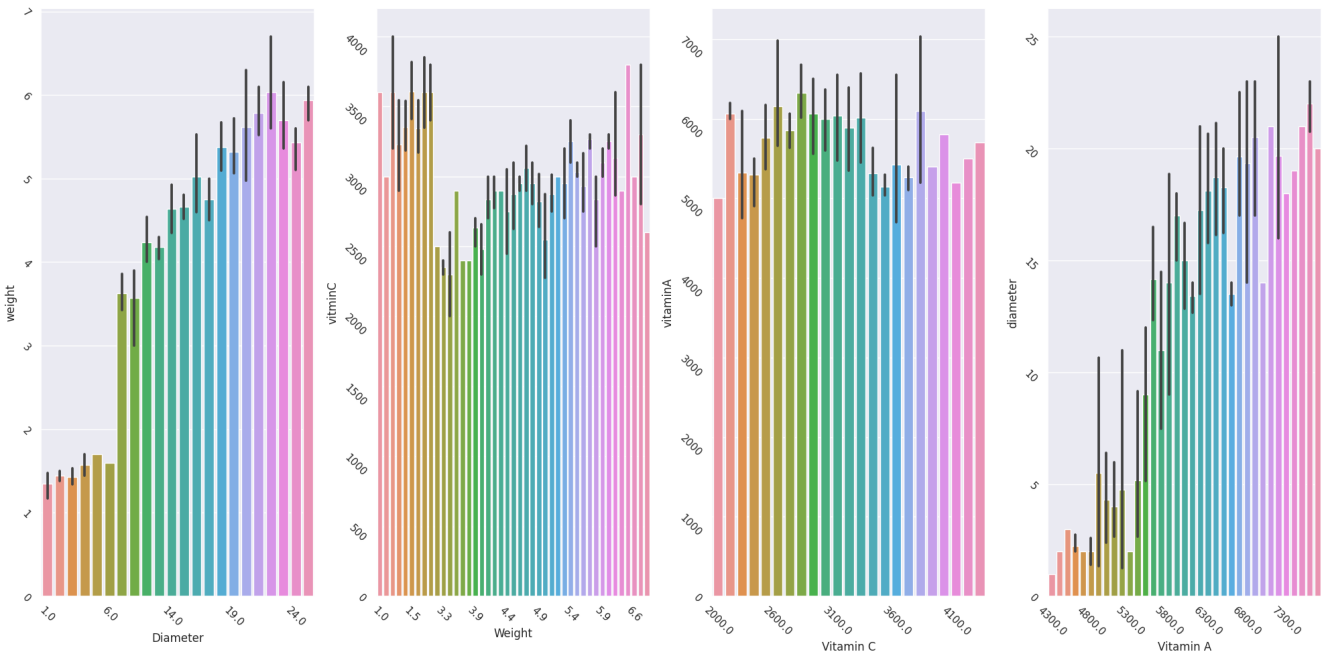

Figure 6: Barplot of a) Diameter vs Weight, b) Weight vs Vitamin C, Vitamin C vs Vitamin A and Vitamin A vs Diameter

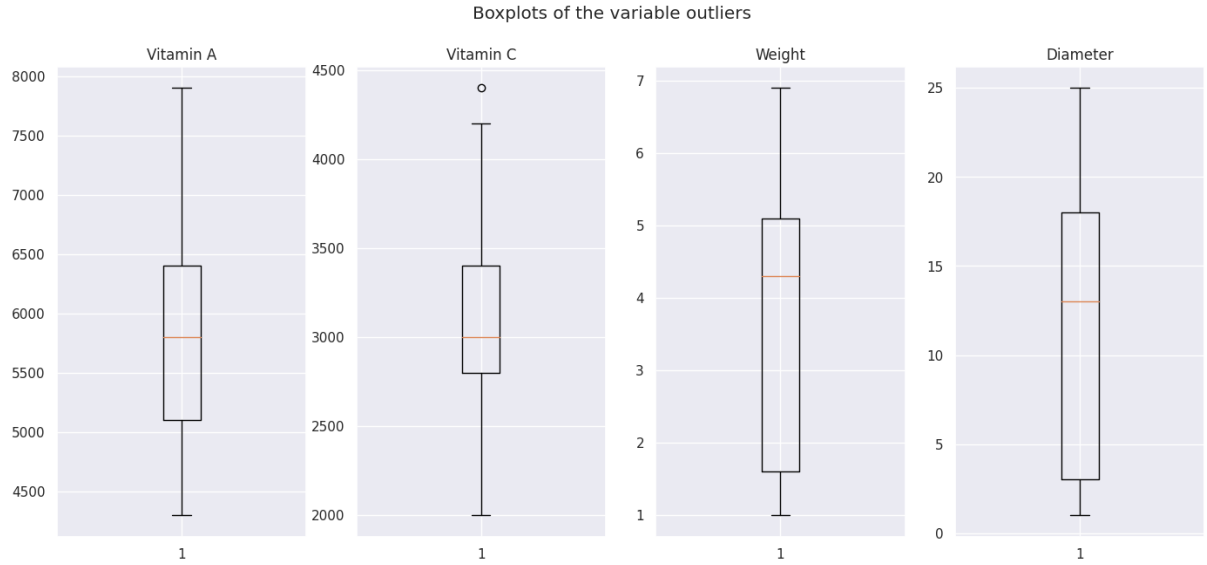

Figure 7: Box plot analysis for outliers

4 Methodology

In order to do our fruit clustering, we consider the following three clustering algorithms:

- **Kmeans**: Is an iterative algorithm that tries to partition the dataset into K pre-defined distinct non-overlapping subgroups (clusters), where each data point belongs to only one group. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible. It assigns data points to a cluster such that the sum of the squared distance between the data points and the cluster's centroid (arithmetic mean of all the data points that belong to that cluster) is at the minimum. The less variation we have within clusters, the more homogeneous (similar) the data points are within the same cluster.[1]
- **Agglomerative**: The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. It's also known as AGNES (Agglomerative Nesting). The algorithm starts by treating each object as a singleton cluster. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects. The result is a tree-based representation of the objects, named dendrogram. [2]
- **GMM:** A Gaussian Mixture is a function that is comprised of several Gaussians, each identified by $k \in \{1, ..., K\}$, where K is the number of clusters of our dataset.[3] Each Gaussian k in the mixture is comprised of the following parameters:
  - A mean μ that defines its centre.
  - A covariance Σ that defines its width. This would be equivalent to the dimensions of an ellipsoid in a multivariate scenario.
  - A mixing probability π that defines how big or small the Gaussian function will be.

Also we consider the **Siluette Score** in order to calculate the optimal number of possible clusters. In figure 8 we can see the siluette score for our three different models. Siluette score evaluates how close are the points of one cluster with respect to the rest of the other clusters[4]. Siluette score per cluster varies in a range from $[-1, 1]$. -1 : the points are further away and should be in another cluster. 1: the points are well defined in the cluster. The average of all the clusters gives us the Siluette score of the model with k clusters.

We run these algorithms for two different scenarios :

- All features included. We do not remove any of the features even why we have a high correlation between three of them. We do so taking into account the fact that we have small amount data and few features.
- Remove **diameter**. Since in the data exploration so far we have said that weight and diameter have a very very high correlation, we try removing one of them, particularly diameter.

5 Results

From Experiment 1 with all features included, we get from the three models that the optimal number of cluster prediction is **2**. From the Silhouette scores of the models, the highest value is reached for number of clusters two as seen in Figure 8. On the other hand, in Figure 9, we have a scatter-plot showing us the two clusters that each model predicts while having **diameter** and **vitamin C** in the axis. We can clearly see that the data are grouped in two well-separated clusters. For visual representation purposes, in Figure 10 we have the 3D plot of this clustering while taking as axis **weight**, **vitamin A** and **vitamin C**. In both the just-mentioned figures, the star in the K-Means model are the **centroids** of the clusters. To conclude with experiment 1, in Table3, we show for each model, the values of each metric used to calculate the evaluation of the models. The three of them predict the clusters with the same accuracy. This being 67%. We can observe as well that the **precision** and the **recall** for one row of each algorithm is **zero**, stating that one of the fruits is being totally misclassified. The confusion matrix proves this statement.

For Experiment 2, where we exclude the diameter feature, we get nearly the same results as Experiment 1. We can see in Figure 11 that still the algorithms are predicting 2 clusters, with a minor change of 2 data-points being clustered on the opposite clusters for the **K-Means** and **Agglomerative**. This detail can be seen in Table 4, where in the confusion matrix for these 2 algorithms, we have 2 data points clustered wrongly. In overall, this didn't change the performance of the algorithms by still having a 67% accuracy and a row with zero recall and precision.

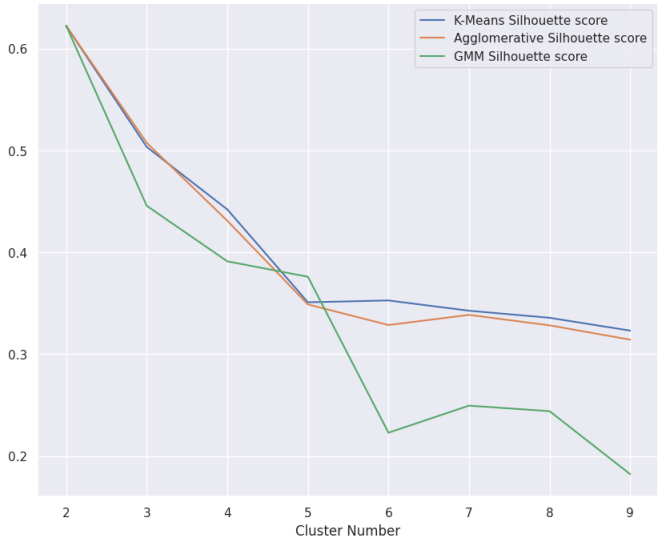

Figure 8: Optimal cluster prediction, all features included

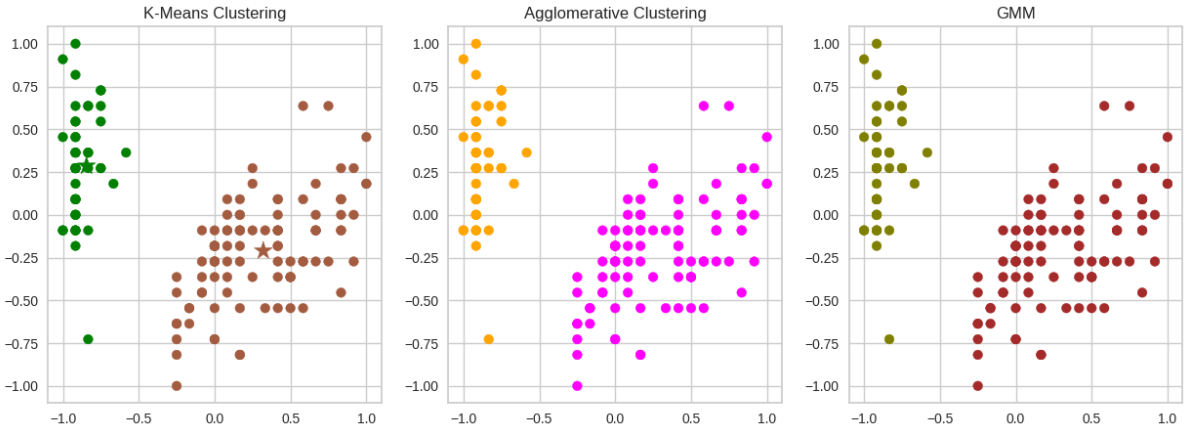

Figure 9: Cluster prediction while having all the features

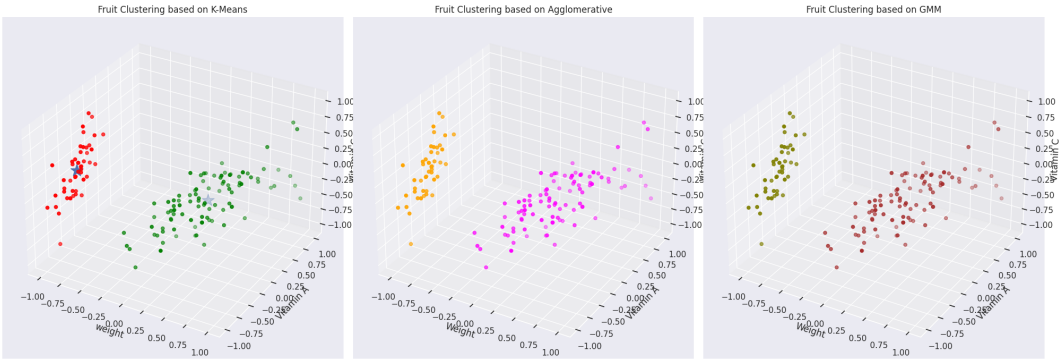

Figure 10: 3D Cluster prediction while having all the features

Figure 11: Cluster prediction while removing diameter

Table 3: Clustering evaluation with all the features

Algorithm	**Confusion Matrix**	**Metric**			
Agglomerative	0 0 47 0 48 0 0 0 49	*Precision*	*Recall*	$F1-Score$	*Support*
		0	0	0	47
		1	1	1	48
		0.51	1	0.68	49
		Accuracy : 0.67	**Macro Avg** : 0.50	**Weighted Avg** : 0.51	
GMM	0 0 47 0 48 0 0 0 49	*Precision*	*Recall*	$F1-Score$	*Support*
		0	0	0	47
		1	1	1	48
		0.51	1	0.68	49
		Accuracy : 0.67	**Macro Avg** : 0.50	**Weighted Avg** : 0.51	
K-Means	47 0 0 0 48 0 49 0 0	*Precision*	*Recall*	$F1-Score$	*Support*
		0.49	1	0.66	47
		1	1	1	48
		0	0	0	49
		Accuracy : 0.66	**Macro Avg** : 0.50	**Weighted Avg** : 0.49	

Table 4: Clustering evaluation without **diameter** feature

Algorithm	**Confusion Matrix**	**Metrics**			
		Precision	*Recall*	$F1-Score$	*Support*
	0 0 47	0	0	0	47
Agglomerative	0 47 1	1	0.98	0.99	48
	0 0 49	0.51	1	0.67	49
		Accuracy : 0.67	**Macro Avg** : 0.50	**Weighted Avg** : 0.51	
		Precision	*Recall*	$F1-Score$	*Support*
	0 0 47	0	0	0	47
GMM	0 48 0	1	1	1	48
	0 0 49	0.51	1	0.68	49
		Accuracy : 0.67	**Macro Avg** : 0.50	**Weighted Avg** : 0.51	
		Precision	*Recall*	$F1-Score$	*Support*
	47 0 0	0.49	1	0.66	47
K-means	0 48 0	0.98	1	0.99	48
	48 1 0	0	0	0	49
		Accuracy : 0.66	**Macro Avg** : 0.49	**Weighted Avg** : 0.49	

6 Conclusions

In this project we aimed to cluster the fruit dataset given the 4 features mentioned so far. After trying different experiments, we reached the results described in the section above. In all of them we observed that the accuracy of the models does not reaches more than 67% and we have a miss-classified fruit which sends the precision and recall of a row to zero. This happens because our model only predicts 2 clusters. We obtain this information from the **siluette** score. It leads us to think that the best way to correctly classify our dataset is with **2** clusters. The reason is the fact that in overall the distribution is over two main groups of data-points.

This happens due to the overlapping that two fruits have on the features. Firstly, here is an overlap when **Peach** is in its max weight/diameter and **apple** on the smallest weight/diameter. Secondly, all the fruits have a huge overlapping in **vitamin C** and **vitamin A**. This leads us to the conclusion that is quite fair for the algorithms to predict only two clusters from the information's they have. For illustration purpose, these conclusions can be seen in Figure 2.

From Experiment 2 we reached the same results. They hold for the same reasons we just explained for Experiment 1. To conclude, we think that for the given dataset, it is needed a scoring algorithm that permits us to calculate the optimal number of clusters which takes into account the overlapping between features.

References

[1] K-means. `https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and`

[2] Agglomerative clustering. `https://www.datanovia.com/en/lessons/agglomerative-hierarchical-clustering/`.

[3] Gmm clustering. `https://towardsdatascience.com/gaussian-mixture-models-explained-6986aaf5a95`.

[4] siluette score. `https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html`.