**Matwej Novgorodtsev,**
**ID: 244580IV**

# 1 Exercise 1

Hopkins statistics indicates whether a dataset is suitable for clustering. The function returns a value from 0 to 1, value close to 1 suggests that the data have a strong cluster structure, and values close to 0 indicates that the data are randomly distributed, with no natural clusters present. The following plot presents the analyzed dataset in case of the Hopkin statistics.

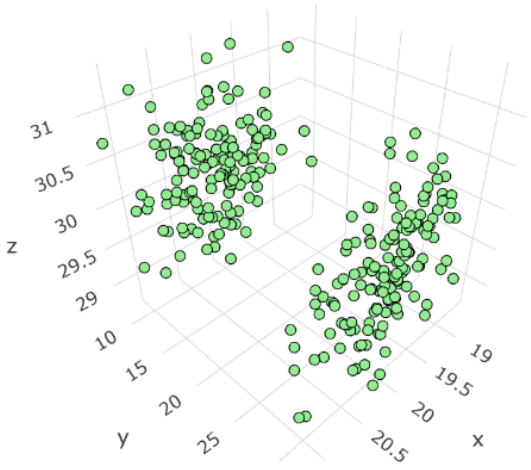


Figure 1: Presentation of the dataset.

To analyze the impact of each feature on the Hopkins Statistic value, we examined various combinations of features. First, we analyzed the relationship between the x and y values. Upon calculation, the Hopkins Statistic was found to be 0.8158806.

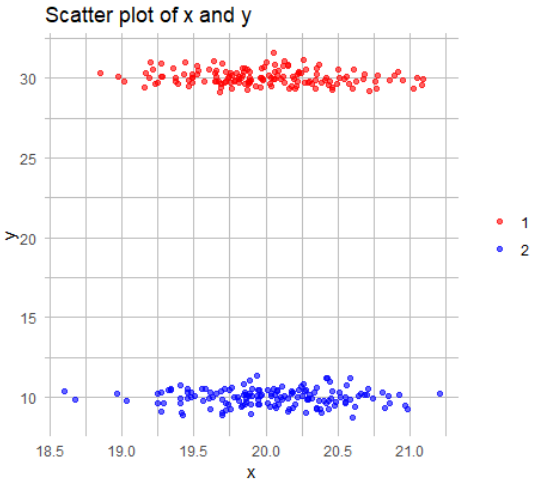


Figure 2: X-Y pair

Then, the value of x-z was checked. As we can see from the plot [3], the Hopkins Statistic is expected to be low. Upon calculation, the Hopkins Statistic was found to be 0.624331, which aligns well with our expectations.

In the final case, based on the plot [4], the Hopkins Statistic is expected to be high. After performing the calculations, this prediction was confirmed, yielding a value of 0.8922321.

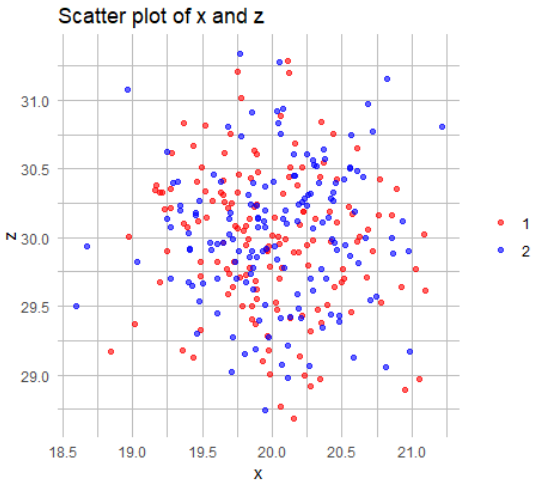


Figure 3: X-Z pair

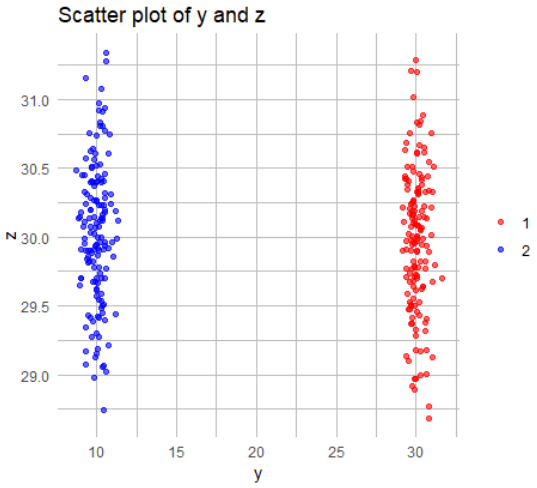


Figure 4: Y-Z pair

Analyzing the yielded values, the best feature combination is Y-Z, which results in a Hopkins Statistic value of 0.8922321.

# 2 Exercise 2

In this task, two 3D datasets were generated: one with clearly separable clusters and the other with more complex, overlapping clusters. Plot below shows those two datasets - easy separable on the left and hard separable on the right.

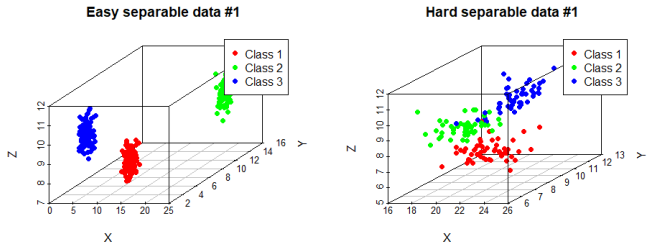


Figure 5: Presentation of two datasets

The k-means algorithm and the Expectation-Maximization (EM) algorithm were implemented and compared on both datasets, each with two different values of k.

Below we can see the result of the clustering of the K-Means algorithm with k = 3. As we can notice, the clusters are perfect.

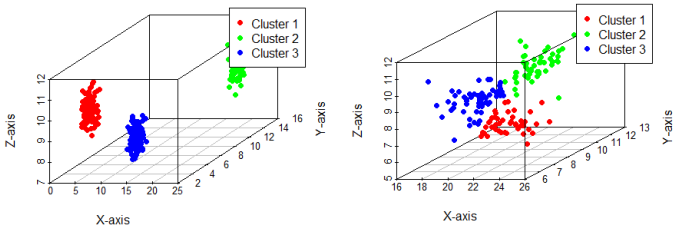


Figure 6: KMeans, k=3

Below, we can see the clustering results of the k-means algorithm with k = 4. As observed, the clusters are clearly defined as strict groups, with no distinction made for individual points.

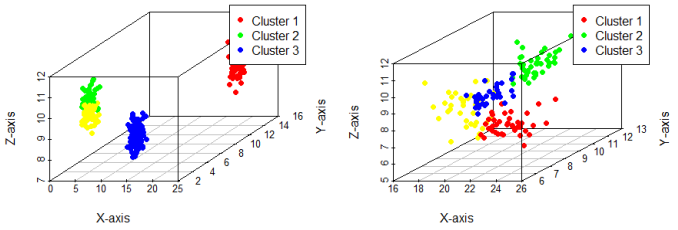


Figure 7: KMeans, k=4

Then, the EM algorithm was analyzed with the same parameters. First with k = 3.

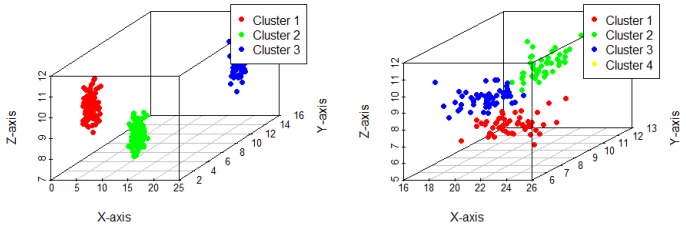


Figure 8: EM, k = 3

Then, the EM algorithm was analyzed with the k = 4. The best results were obtained with k = 3, as expected for the dataset's structure, with the EM algorithm effectively grouping the data into three distinct clusters. When k = 4, both the EM and k-means algorithms struggled, leading to overfitting and less meaningful cluster assignments, highlighting the importance of selecting the correct value for k. In general, we can observe that the EM algorithm demonstrates a greater ability to distinguish individual points, whereas the k-means algorithm strictly divides datasets into groups without accounting for individual points. The following plot shows the structures of the datasets.

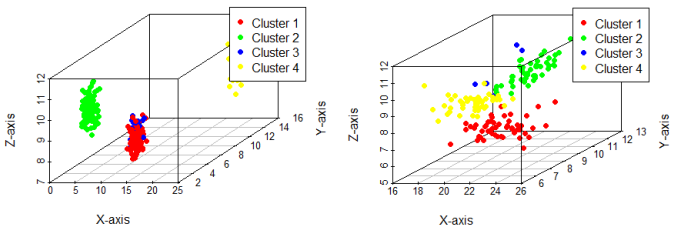


Figure 9: EM, k = 4

# 3 Exercise 3

In this task, the DBSCAN algorithm was implemented in R for density-based clustering. A 2D datasets were generated, and the algorithm's performance was demonstrated with different hyperparameter settings, such as varying the eps (radius of neighborhood) and minPts (minimum points per cluster). The best results were achieved with eps = 1 and minPts = 8, where the datasets were successfully clustered. The plot below shows the result of the DBSCAN with $\epsilon = 0.5$ and MinPoints = 2 for different datasets (less and more separable).

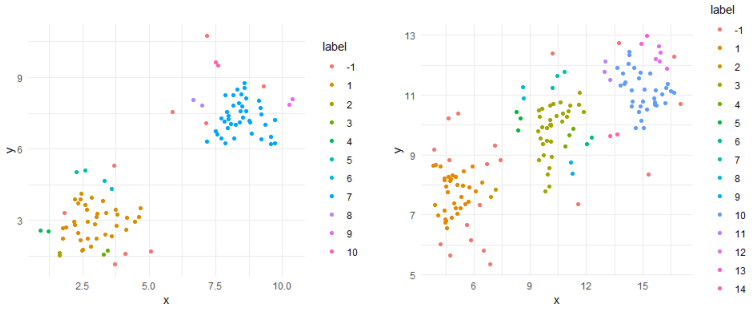


Figure 10: DBSCAN, $\epsilon = 0.5$ and MinPoints = 2

The plot below shows the result of the DBSCAN with $\epsilon = 1$ and MinPoints = 8.

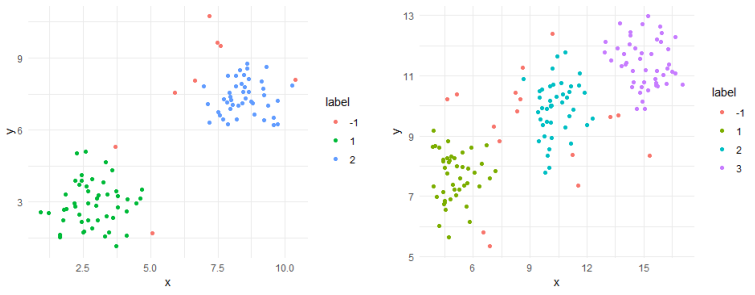


Figure 11: DBSCAN, $\epsilon = 1$ and MinPoints = 8

As we can see, the clustering improves as $\epsilon$ increases and MinPoints decreases in this particular case.

# 4 Exercise 4

The half-moon dataset was generated. A logistic regression separator was trained and evaluated, showing that the dataset was not perfectly separable in 2D using classification metrics such as accuracy and confusion matrix. A transformation was then applied to map the dataset from 2D to 3D, making the dataset perfectly separable, and a new logistic regression model

was trained and evaluated, demonstrating perfect separability in 3D. The plot below shows the half-moon dataset in the 2D space - in this case the accuracy achieved the value of 0.88.

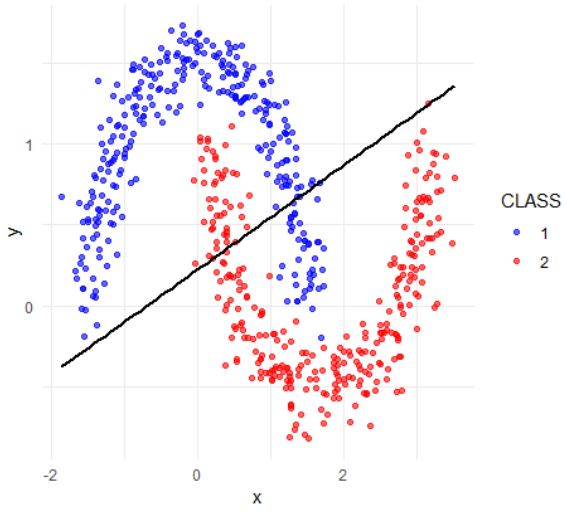


Figure 12: Halfmoons presented in 2 dimensional space

The plot below shows the half-moon transformed into 3D space. As we can see, the two clusters are now perfectly separated. The accuracy achieved the value of 1.

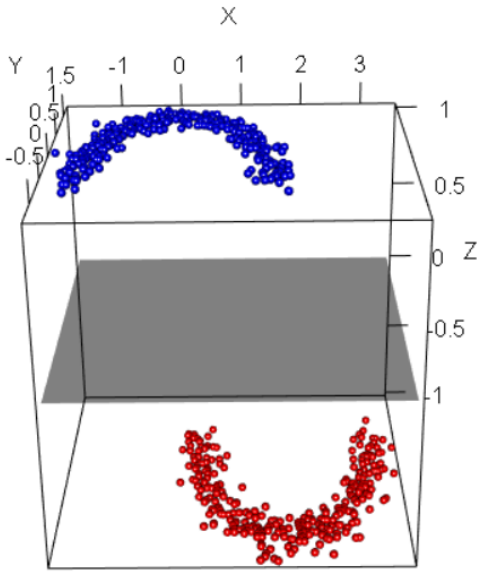


Figure 13: Halfmoons data presented in 3D