CHAPTER 5

IMPLEMENTATION AND TESTING

5.2 Implementation

This program is made in Jupiter Notebook using the Python programming language. Below will explain the program code.

```
8.  import numpy as np
9.  import pandas as pd
10. import matplotlib.pyplot as plt
11. from sklearn.metrics import silhouette_score
12. from sklearn.cluster import Birch
13. from sklearn import linear_model
14. import missingno as mno
```

Line 1-8 is the library that will be used for the program for pre-processing until the clustering stage.

```
1. DataFile = pd.read_csv
   (r"C:/Users/acer/Desktop/CodeDataMin/dataOlimpiade/athlete_e
   vents.csv")
2. DataOL =
   DataFile.loc[(DataFile['Sport'].isin(['Football','Beach
   Volleyball','Volleyball']))]
3. DataOL =
   DataFile.loc[(DataFile['Sport'].isin(['Basketball']))]
4. DataFilter = DataOL[['Height', 'Medal']]
5. isikosong = DataFilter.fillna(0)
6. isikosong.loc[isikosong.Medal == 'Gold', 'Medal'] = '3'
7. isikosong.loc[isikosong.Medal == 'Silver', 'Medal'] = '2'
8. isikosong.loc[isikosong.Medal == 'Bronze', 'Medal'] = '1'
9. isikosong.loc[isikosong.Height == 0.0, 'Height'] = np.NAN
```

Line 1 is used to read the CSV file and input it into the parameters, line 2 selects the sports data to be processed, line 2-3 is used to select data for basketball, football, volleyball, and beach volleyball. If want to select all sports data then don't need to use lines 2-3 directly use line 4. Line 4-8 the data that is still in the form of letters are changed to numbers first to be processed in the

BIRCH clustering. The data that needs to be converted to numbers is medals. Not getting a medal = 0, getting a bronze medal = 1, getting a silver medal = 2, getting a gold medal = 3 and return the height data back to NaN values, to be processed in deterministic regression.

```
1. missing_columns = ['Height']
2. def random_imputation(df, feature):
3. 
4.     number_missing = isikosong[feature].isnull().sum()
5.     observed_values = isikosong.loc[df[feature].notnull(), 
   feature]
6.     isikosong.loc[isikosong[feature].isnull(), feature + 
   '_imp'] = np.random.choice(observed_values, number_missing, 
   replace = True)
7.     return isikosong

8. for feature in missing_columns:
9.     isikosong[feature +'_imp'] = isikosong[feature]
10.    isikosong = random_imputation(isikosong, feature)
11. 
12. deter_data = pd.DataFrame(columns = ['Det' + name for name 
   in missing_columns])
13. 
14. for feature in missing_columns:
15.    deter_data['Det' + feature] = isikosong[feature + 
   '_imp']
16.    parameters = list(set(isikosong.columns) - 
   set(missing_columns) - {feature + '_imp'})
17. 
18.    model = linear_model.LinearRegression()
19.    model.fit(X = isikosong[parameters], y = 
   isikosong[feature + '_imp'])
20. 
21.    deter_data.loc[isikosong[feature].isnull(), 'Det' + 
   feature] = model.predict(isikosong[parameters])
   [isikosong[feature].isnull()]
22.    mno.matrix(deter_data, figsize = (20, 10))
```

Line 1 - 22 is a code to fill in the missing athlete's height data from the Olympic CSV using the deterministic regression method. and in lines, 2-7 is the code to account for missing data using regression. After doing the calculations for missing data, in code 8 - 12, it is continued with deterministic regression

imputation, after imputing deterministic regression for missing data, it is continued by checking the missing data with linear regression as inline code 14 - 22.

```
1. for k in range(2, 19+1):
2.     model = KMeans(n_clusters=k)
3.     model.fit(df)
4.     pred = model.predict(df)
5.     score = silhouette_score(df, pred)
6.     print('Silhouette Score for k = {}: {:<.3f}'.format(k,
score))
```

Line 1-6 is a silhouette scoring method, used to find a good number of clusters to use when running the BIRCH algorithm. By calculating the silhouette score for each cluster that has been assigned to line 1. using code line 6, each cluster will display the silhouette score value.

5.3 Testing

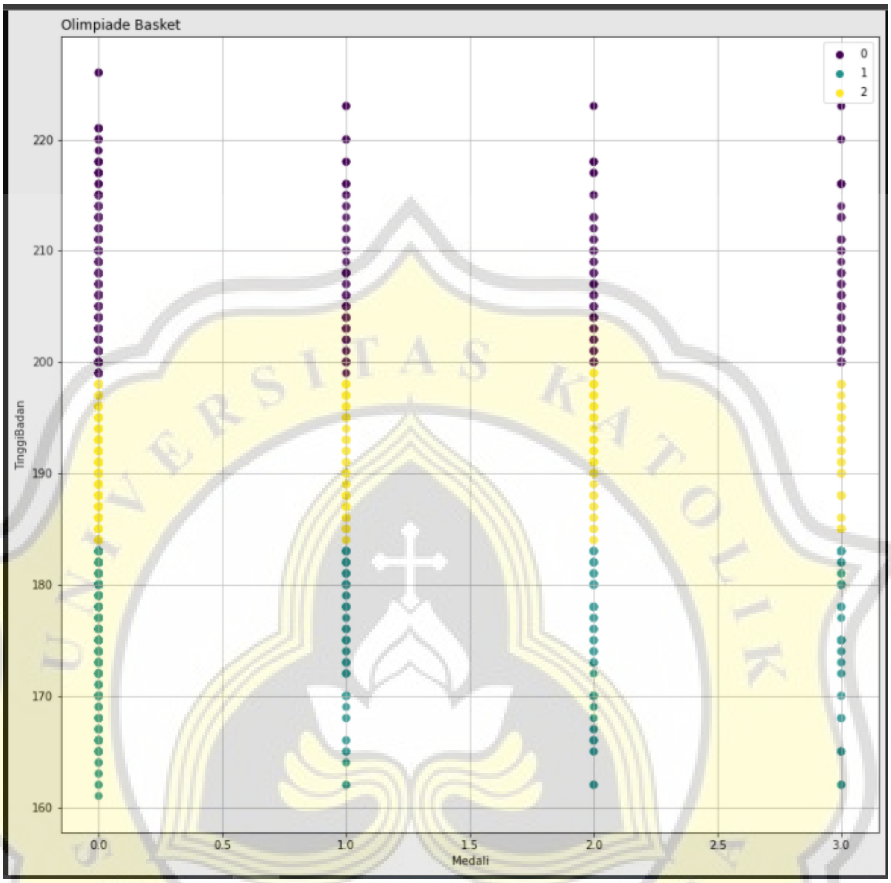

Figure 4: Cluster Basketball

For the results of the basketball olympiad cluster, it can be seen in the picture above, that the cluster is divided into 3 and from the results it can be seen that basketball athletes with a height above 200cm can also not win medals at the olympics, and other results show that basketball athletes with a height of 180cm and below also can win a gold medal. and other athletes with a height of 200cm and over or athletes with a height of 180cm and below also won bronze and silver medals.

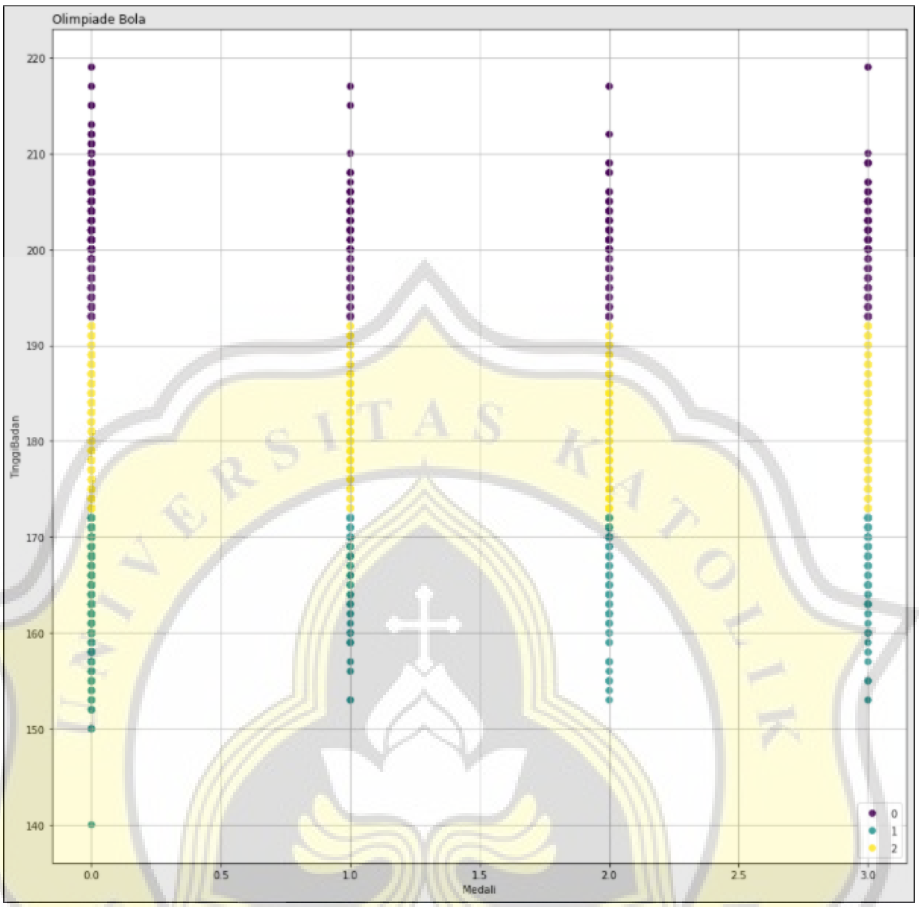

Figure 5: Cluster Football, Volleyball, Beach Volleyball

From the image of the cluster results above, it can be concluded that the results of the soccer, volleyball, and beach volleyball clusters can be drawn. It is divided into 3 clusters, and shows that those with a height of 193cm and above and 172cm and below are more likely to not win a medal at all, and in terms of winning medals, they win more silver medals compared to gold and bronze when viewed from the results of the Olympic sports cluster. soccer, volleyball and beach volleyball.

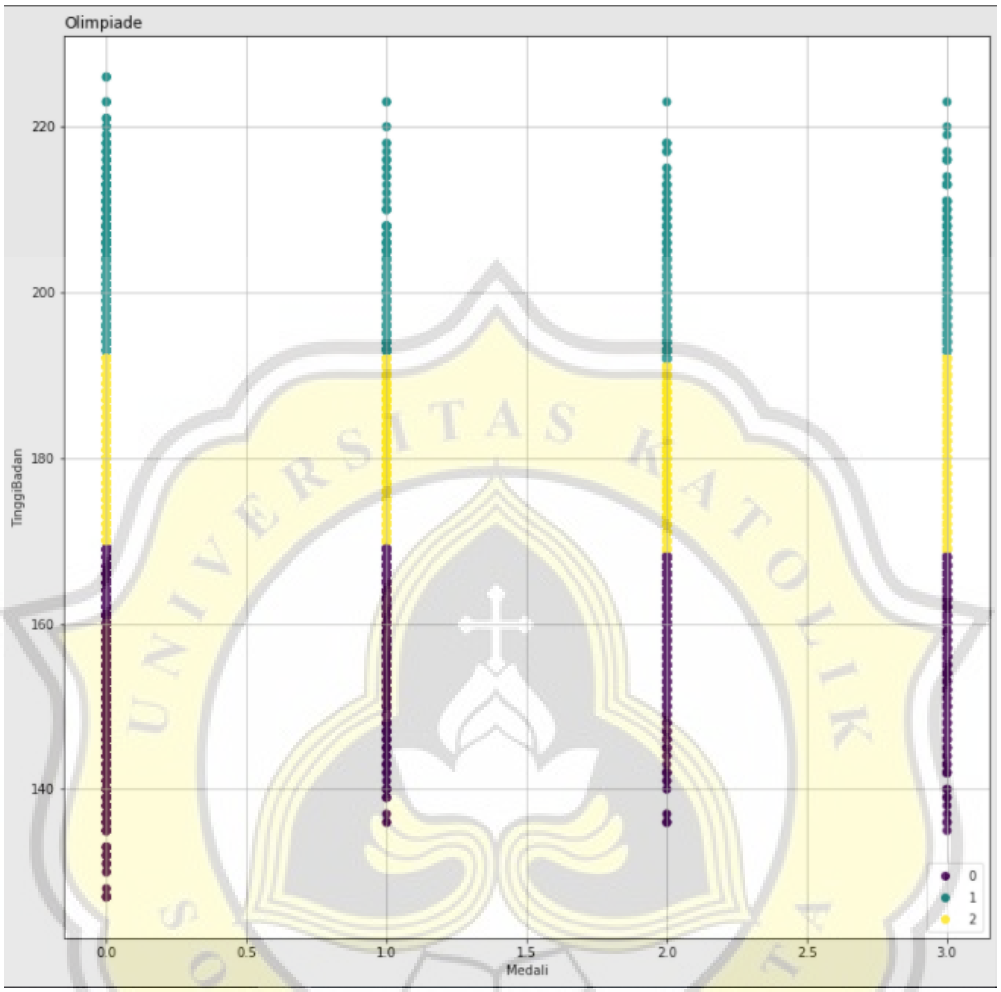

Figure 6: Cluster All Sport Olympic

From the picture of the results of the cluster above, it can be concluded that the results of the cluster of all sports at the Olympics can be concluded. By looking at the results of the clusters, many clusters did not win medals for athletes with a height of 193 and above and also 169cm and below. but the results are different for athletes with height 193 and above and also 169cm and under who won fewer medals than those who did not win any medals at all. for the comparison of athletes who won gold medals only, more in height 169cm and below.

5.4 Evaluation

In this evaluation discussion, we will discuss the results of the homogeneity, completeness, and V-Measure calculations from the basketball cluster, Beach Volleyball Football cluster, and all sports clusters in the Olympics, to see whether the cluster performance produced by the BIRCH algorithm is good or not. By looking at the results of the V-Measure. The results of homogeneity, completeness are used to calculate the V-Measure. The closer the V-Measure value is to the number 1, the better the resulting cluster

The table below describes the results of the homogeneity, completeness and measure of the basketball cluster. It shows that the completeness result is 1, and the homogeneity is 0.28, completeness 0.99, and the result of V-Measure is 0.44. The 0.44 V-Measure result means that the basketball cluster's performance results are not good.

Table 5.1: Result Evaluation Basketball

Homogeneity	0.28341135525242406
Completeness	0.997462750812483
V-Measure	0.4414052383181702

After seeing the results of the basketball cluster, then looking at the results of the Football, Volleyball, and Beach Volleyball clusters The table below describes the results of the homogeneity, completeness, and measure of the Football, Volleyball, and Beach Volleyball cluster. It shows that the result is, homogeneity is 0.24, completeness 0.99, and the result of V-Measure is 0.39. The 0.39 V-Measure result means that the Football, Volleyball, and Beach Volleyball cluster's performance results are not good.

Table 5.2: Result Evaluation Football,Volleyball, and Beach Volleyball

Homogeneity	0.24744240114897698

Completeness	0.9999999999999997
V-Measure	0.39671956143396464

The table below describes the results of the homogeneity, completeness and cluster size of the overall sports in the Olympics. This shows that the homogeneity result is 0.20, the completeness is 0.99, and the V-Measure result is 0.34. The measurement result of 0.34 means that the performance results of the Football, Volleyball, and Beach Volleyball clusters are not good. From the results of all clusters that have been calculated using homogeneity, completeness and V-Measure, the V-Measure is still very far from approaching number 1 so the performance results of the BIRCH algorithm are still not good enough.

Table 5.3: Result All Sport Olimpic

Homogeneity	0.2080736632245153
Completeness	0.9900922803821292
V-Measure	0.34387912427104816