Data Mining Term Project

Burak Tahtacı

December 2018

1 Introduction

Data mining is the process of sorting through huge data sets to detecting patterns and discovering interesting relationships to solve problems through analyzing the data. Data mining tools allows make predictions for future. So various data mining approaches provide solutions to get information from raw data. This approaches can be divided into two parts, Supervised Methods and Unsupervised Methods. The separation criteria is simple. If our data set have class labels so it is Supervised Methods if not they are Unsupervised ones.

In this study this methods compared with well known algorithms such as K-Nearest Neighborhood Classification and K-Means Clustering. During implementation, breast cancer in Wisconsin data set used to train and test. All these algorithms implemented with Python programming language without using any machine learning framework or library.

2 Dataset

Sample data set comprised of patients who suffer from breast cancer in University of Wisconsin hospitals. Samples arrive periodically as Dr. Wolberg reports his clinical cases. The database therefore reflects this chronological grouping of the data. This grouping information appears immediately below, having been removed from the data itself:

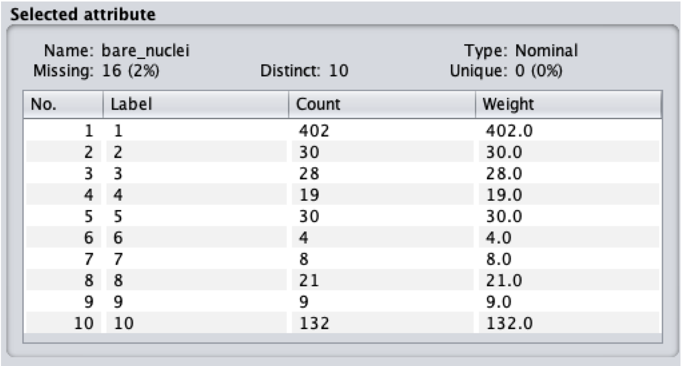
Group 1: 367 samples (January 1989)
Group 2: 70 samples (October 1989)
Group 3: 31 samples (February 1990)
Group 4: 17 samples (April 1990)
Group 5: 48 samples (August 1990)
Group 6: 49 samples (Updated January 1991)
Group 7: 31 samples (June 1991)
Group 8: 86 samples (November 1991)

This sample set have 10 attributes, the table in the below gives information about each attribute.

1. Sample code number: id number (numeric)
2. Clump Thickness: 1 - 10 (nominal)
3. Uniformity of Cell Size: 1 - 10 (nominal)
4. Uniformity of Cell Shape: 1 - 10 (nominal)
5. Marginal Adhesion: 1 - 10 (nominal)
6. Single Epithelial Cell Size: 1 - 10 (nominal)
7. Bare Nuclei: 1 - 10 (nominal)
8. Bland Chromatin: 1 - 10 (nominal)
9. Normal Nucleoli: 1 - 10 (nominal)
10. Mitoses: 1 - 10 (nominal)
11. Class: (2 for benign, 4 for malignant) (nominal)

After first experiments on WEKA have been held, 458 out of 699 clumps classified as benign, whereas the 241 left as malignant. There are 16 samples whose at least 1 attribute information is missing. All missing values are of the attribute of "Bare Nuclei". The histogram of this attribute is given below.

Figure 1: Bare Nuclei Attribute

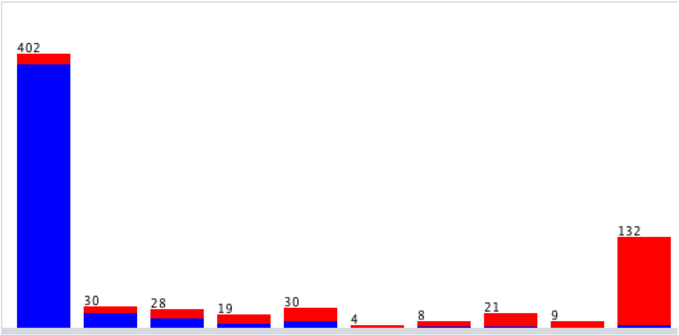


Selected attribute

Name: bare_nuclei
Missing: 16 (2%)
Distinct: 10
Type: Nominal
Unique: 0 (0%)

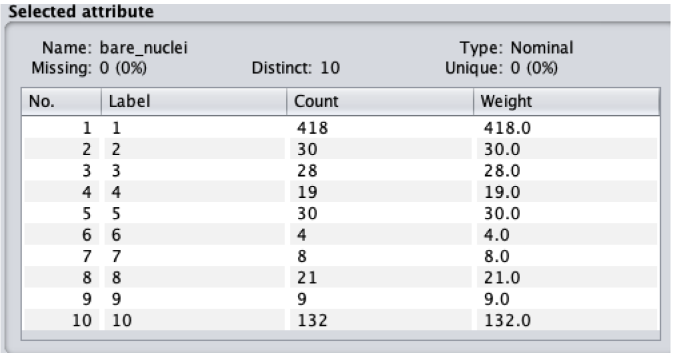
No.	Label	Count	Weight
1	1	402	402.0
2	2	30	30.0
3	3	28	28.0
4	4	19	19.0
5	5	30	30.0
6	6	4	4.0
7	7	8	8.0
8	8	21	21.0
9	9	9	9.0
10	10	132	132.0

Figure 2: Bare Nuclei Histogram



Red stands for malignant clumps whereas blue does for those are benign. As it is seen above, most of the patients have benign clumps tend to have lower rate of bare nuclei. Patients with malignant clumps do vice versa. Therefore replacing missing values of the patients whose clumps are benign with lower rates of bare nuclei and higher rates for patients with malignant clumps do make sense. After replacing missing values the distribution of values given below.

Figure 3: Bare Nuclei Histogram After Replacing Missing Values



Selected attribute

Name: bare_nuclei
Missing: 0 (0%)
Distinct: 10
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	1	418	418.0
2	2	30	30.0
3	3	28	28.0
4	4	19	19.0
5	5	30	30.0
6	6	4	4.0
7	7	8	8.0
8	8	21	21.0
9	9	9	9.0
10	10	132	132.0

3 WEKA Results

WEKA is a well known tool for implementing simple data mining algorithms. This section includes results of classification and clustering algorithms in WEKA. Given data set evaluated with different classification and clustering methods and also founded the best accuracy rates.

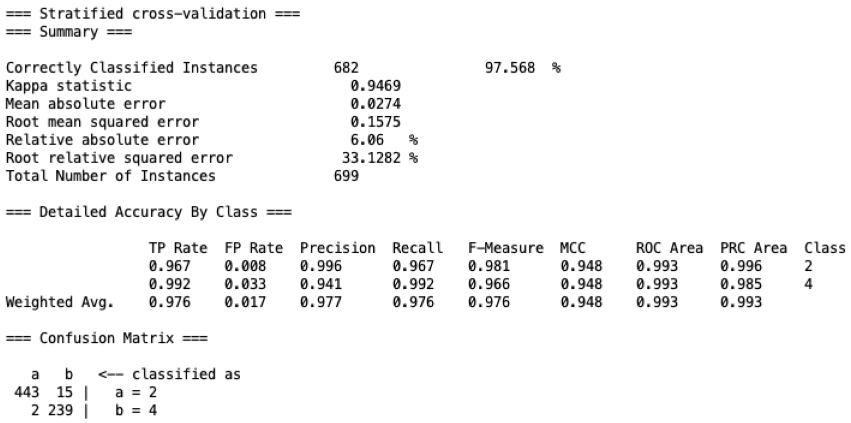
3.1 Classification

According to evaluation of classification algorithms on given dataset. Best three classification algorithms are Naive Baye, K-NN and J48 Decision Trees. Results and accuracies given the following subsections.

3.1.1 Naive Bayes

Depending on all classification methods with 10 - fold cross validation, The Naive Bayes Method gives the highest rate of classification accuracy. 97.56 of instances which makes up 681 out of 699 classified correctly. Detailed information about classification outcome and confusion matrix is as follows:

Figure 4: Naive Bayes Classification Result



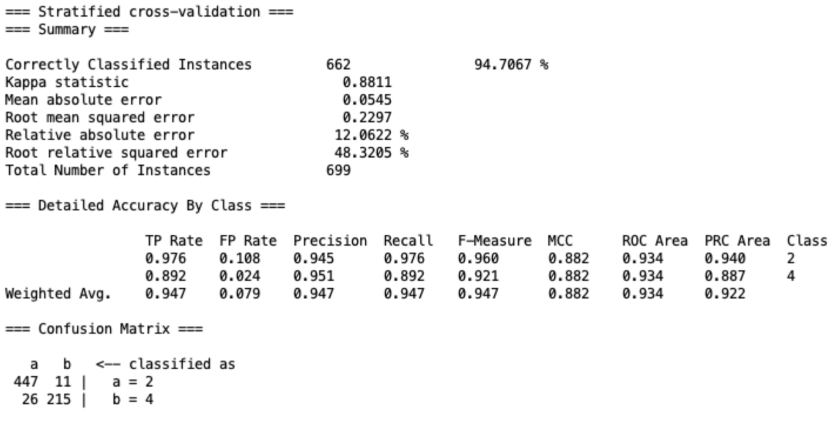
```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         682               97.568  %
Kappa statistic                          0.9469
Mean absolute error                      0.0274
Root mean squared error                  0.1575
Relative absolute error                  6.06   %
Root relative squared error             33.1282 %
Total Number of Instances              699

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.967    0.008    0.996      0.967    0.981      0.948    0.993     0.996     2
                 0.992    0.033    0.941      0.992    0.966      0.948    0.993     0.985     4
Weighted Avg.    0.976    0.017    0.977      0.976    0.976      0.948    0.993     0.993

=== Confusion Matrix ===

   a   b   <-- classified as
 443  15 |   a = 2
   2 239 |   b = 4
```

When the confusion matrix is considered, totally 18 samples out of 699 is classified incorrectly. 15 instances from class a classified as b, and the other 3 is done vice versa. It is possible to get an inference that instances of a are prone to be classified incorrectly rather than instances of b.

3.1.2 K-Nearest Neighbour

The following classification method which has the highest rate of correctly classified instances is K - Nearest Neighbor with 94.70 of correctness in terms of classification. The correctness rate changes according to the K value. Maximum correctness rate is obtained when K is set to 3. Information of the chosen classification method is given as follows

Figure 5: K-NN Classification Result

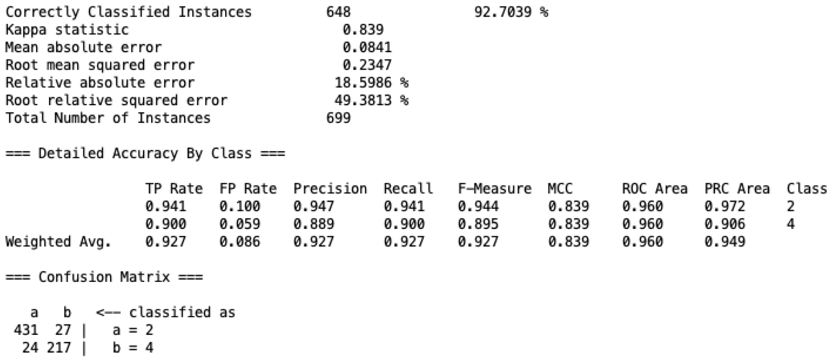


```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         662               94.7067 %
Kappa statistic                          0.8811
Mean absolute error                      0.0545
Root mean squared error                  0.2297
Relative absolute error                 12.0622 %
Root relative squared error             48.3205 %
Total Number of Instances              699

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.976    0.108    0.945      0.976    0.960      0.882    0.934     0.940     2
                 0.892    0.024    0.951      0.892    0.921      0.882    0.934     0.887     4
Weighted Avg.    0.947    0.079    0.947      0.947    0.947      0.882    0.934     0.922

=== Confusion Matrix ===

   a   b   <-- classified as
 447  11 |   a = 2
  26 215 |   b = 4
```

3.1.3 J48 Decision Tree

Third classification method with the highest rate of correctly classified instances is J48 Decision Tree. If cross validation folds set to 10, the average rate of correctly classified instances of the test set is 92.78 Detailed further information are given below.

Figure 6: J48 Decision Tree Classification Result



```
Correctly Classified Instances         648               92.7039 %
Kappa statistic                          0.839
Mean absolute error                      0.0841
Root mean squared error                  0.2347
Relative absolute error                 18.5986 %
Root relative squared error             49.3813 %
Total Number of Instances              699

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.941    0.100    0.947      0.941    0.944      0.839    0.960     0.972     2
                 0.900    0.059    0.889      0.900    0.895      0.839    0.960     0.906     4
Weighted Avg.    0.927    0.086    0.927      0.927    0.927      0.839    0.960     0.949

=== Confusion Matrix ===

   a   b   <-- classified as
 431  27 |   a = 2
  24 217 |   b = 4
```

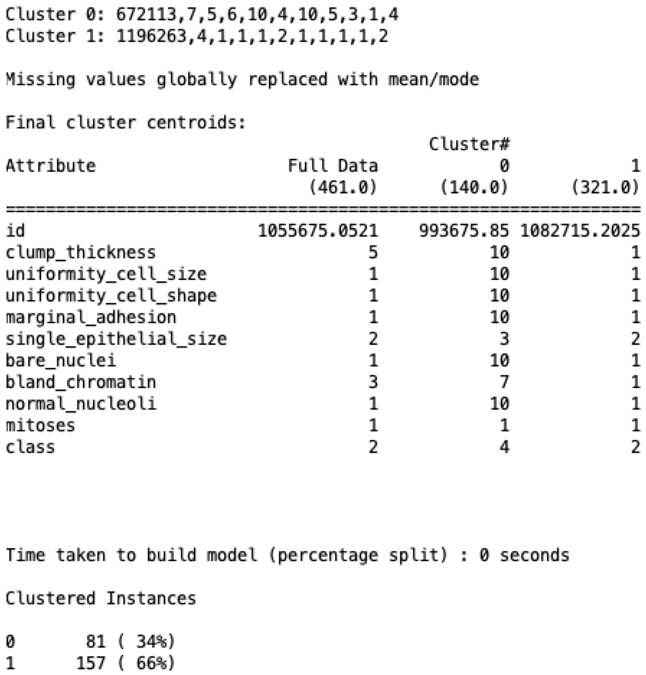
3.2 Clustering

For the given class labeled data set 34.47% of instances in the data set belong to the class label 4 which denotes malignant clumps. Other 65.53 of instances belong to class label 2. Those patients have clumps whose status is benign. The more if a clustering method gives class rates be similar to the aforementioned rates, the better and more precise that clustering method can be considered. Hence it must be regarded whether a clustering method can cluster the instances with a small difference in rates from the actual data set or not.

3.2.1 K Means Clustering

For the same data set, among some various clustering methods, simple K - Means did the best. Data set separated into two parts to create test and train values with a rate of 33 and 67 respectively. If the K is set to 2, samples are distributed over 2 clusters with a rate of 34 and 66 . These rates are quite close to the actual rates of distribution. Further information about K - Means method applied on the data set given as follows.

Figure 7: K Means Clustering Result



```
Cluster 0: 672113,7,5,6,10,4,10,5,3,1,4
Cluster 1: 1196263,4,1,1,1,2,1,1,1,1,2

Missing values globally replaced with mean/mode

Final cluster centroids:
                                          Cluster#
Attribute                     Full Data          0          1
                                (461.0)    (140.0)    (321.0)
==============================================================
id                         1055675.0521  993675.85 1082715.2025
clump_thickness                       5         10          1
uniformity_cell_size                  1         10          1
uniformity_cell_shape                 1         10          1
marginal_adhesion                     1         10          1
single_epithelial_size                2          3          2
bare_nuclei                           1         10          1
bland_chromatin                       3          7          1
normal_nucleoli                       1         10          1
mitoses                               1          1          1
class                                 2          4          2




Time taken to build model (percentage split) : 0 seconds

Clustered Instances

0       81 ( 34%)
1      157 ( 66%)
```

3.2.2 Hierarchical Clustering

Hierarchical Clustering method with complete linkage gives the second closest outcome in comparison to all other clustering methods. Distribution rates over clusters are 35 and 65.

3.2.3 Density Based Clustering

As a result of Density Based Clustering method, samples generates two different sets each of which has the amount of samples 37 and 63 respectively.

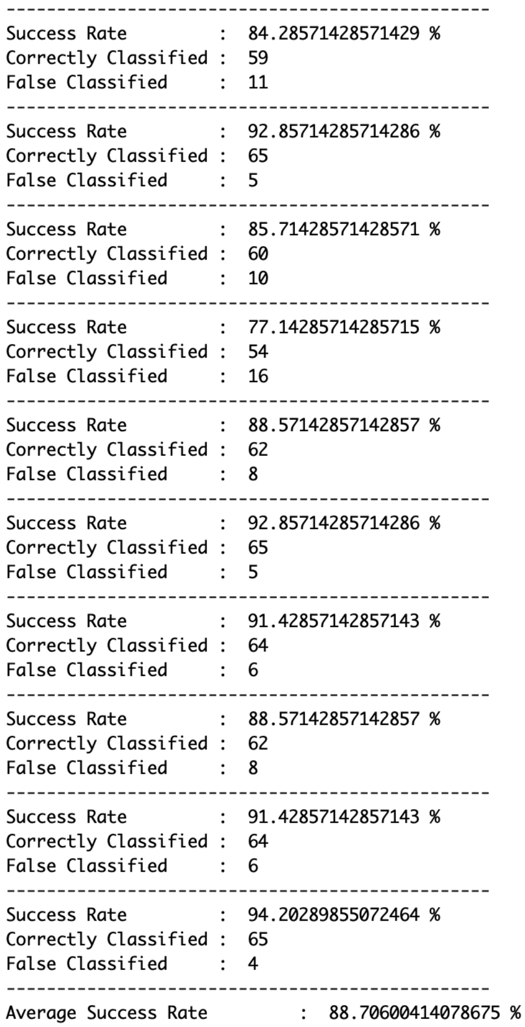
4 Python Implementation Results

As mentioned before KNN and KMeans algorithms implemented in python programming language. There were some challenges while implementing this algorithms. First one is missing values, they are handled with replacing 0. In WEKA, missing values handled replacing with class means. Second is random numbers' seed problem. Pseudo random number generators start with a seed value. To handle this issue a constant value set as initial value of seed, and then numbers are same. The results of these algorithms are given in next sections.

4.0.1 K-NN in Python

According to result of experiments, best K value observed as 3. Whole data set divided into 10 parts and 10-Fold Cross validation implemented to test average success rate of algorithm.

Figure 8: KNN Python Result

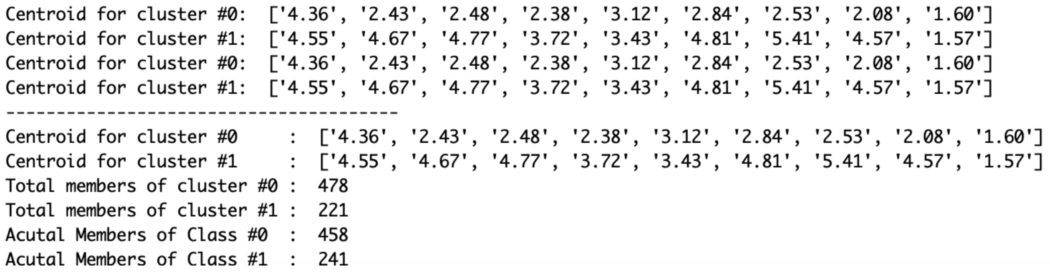


```
------------------------------------------------
Success Rate        :  84.28571428571429 %
Correctly Classified :  59
False Classified    :  11
------------------------------------------------
Success Rate        :  92.85714285714286 %
Correctly Classified :  65
False Classified    :  5
------------------------------------------------
Success Rate        :  85.71428571428571 %
Correctly Classified :  60
False Classified    :  10
------------------------------------------------
Success Rate        :  77.14285714285715 %
Correctly Classified :  54
False Classified    :  16
------------------------------------------------
Success Rate        :  88.57142857142857 %
Correctly Classified :  62
False Classified    :  8
------------------------------------------------
Success Rate        :  92.85714285714286 %
Correctly Classified :  65
False Classified    :  5
------------------------------------------------
Success Rate        :  91.42857142857143 %
Correctly Classified :  64
False Classified    :  6
------------------------------------------------
Success Rate        :  88.57142857142857 %
Correctly Classified :  62
False Classified    :  8
------------------------------------------------
Success Rate        :  91.42857142857143 %
Correctly Classified :  64
False Classified    :  6
------------------------------------------------
Success Rate        :  94.20289855072464 %
Correctly Classified :  65
False Classified    :  4
------------------------------------------------
Average Success Rate        :  88.70600414078675 %
```

4.0.2 K-Means in Python

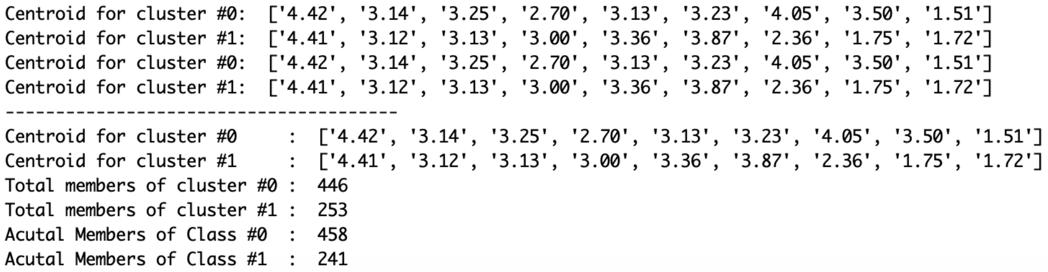
In data set there are 2 classes, benign and malicious that are represented by 2 and 4. So K value is 2. Whole data set is used for testing the performance of cluster algorithm.

Figure 9: KMeans Result Python Result - 1



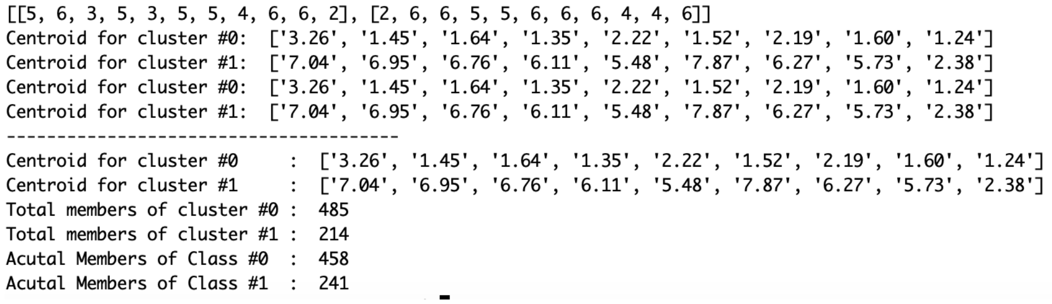
```
Centroid for cluster #0:  ['4.36', '2.43', '2.48', '2.38', '3.12', '2.84', '2.53', '2.08', '1.60']
Centroid for cluster #1:  ['4.55', '4.67', '4.77', '3.72', '3.43', '4.81', '5.41', '4.57', '1.57']
Centroid for cluster #0:  ['4.36', '2.43', '2.48', '2.38', '3.12', '2.84', '2.53', '2.08', '1.60']
Centroid for cluster #1:  ['4.55', '4.67', '4.77', '3.72', '3.43', '4.81', '5.41', '4.57', '1.57']
---------------------------------------
Centroid for cluster #0     :  ['4.36', '2.43', '2.48', '2.38', '3.12', '2.84', '2.53', '2.08', '1.60']
Centroid for cluster #1     :  ['4.55', '4.67', '4.77', '3.72', '3.43', '4.81', '5.41', '4.57', '1.57']
Total members of cluster #0 :  478
Total members of cluster #1 :  221
Acutal Members of Class #0  :  458
Acutal Members of Class #1  :  241
```

Figure 10: KMeans Result Python Result - 2



```
Centroid for cluster #0:  ['4.42', '3.14', '3.25', '2.70', '3.13', '3.23', '4.05', '3.50', '1.51']
Centroid for cluster #1:  ['4.41', '3.12', '3.13', '3.00', '3.36', '3.87', '2.36', '1.75', '1.72']
Centroid for cluster #0:  ['4.42', '3.14', '3.25', '2.70', '3.13', '3.23', '4.05', '3.50', '1.51']
Centroid for cluster #1:  ['4.41', '3.12', '3.13', '3.00', '3.36', '3.87', '2.36', '1.75', '1.72']
---------------------------------------
Centroid for cluster #0     :  ['4.42', '3.14', '3.25', '2.70', '3.13', '3.23', '4.05', '3.50', '1.51']
Centroid for cluster #1     :  ['4.41', '3.12', '3.13', '3.00', '3.36', '3.87', '2.36', '1.75', '1.72']
Total members of cluster #0 :  446
Total members of cluster #1 :  253
Acutal Members of Class #0  :  458
Acutal Members of Class #1  :  241
```

Figure 11: KMeans Result Python Result - 3



```
[[5, 6, 3, 5, 3, 5, 5, 4, 6, 6, 2], [2, 6, 6, 5, 5, 6, 6, 6, 4, 4, 6]]
Centroid for cluster #0:  ['3.26', '1.45', '1.64', '1.35', '2.22', '1.52', '2.19', '1.60', '1.24']
Centroid for cluster #1:  ['7.04', '6.95', '6.76', '6.11', '5.48', '7.87', '6.27', '5.73', '2.38']
Centroid for cluster #0:  ['3.26', '1.45', '1.64', '1.35', '2.22', '1.52', '2.19', '1.60', '1.24']
Centroid for cluster #1:  ['7.04', '6.95', '6.76', '6.11', '5.48', '7.87', '6.27', '5.73', '2.38']
---------------------------------------
Centroid for cluster #0     :  ['3.26', '1.45', '1.64', '1.35', '2.22', '1.52', '2.19', '1.60', '1.24']
Centroid for cluster #1     :  ['7.04', '6.95', '6.76', '6.11', '5.48', '7.87', '6.27', '5.73', '2.38']
Total members of cluster #0 :  485
Total members of cluster #1 :  214
Acutal Members of Class #0  :  458
Acutal Members of Class #1  :  241
```