1 Data Manipulation

The data which contains 1000 rows and 21 columns has been used in this report. This data has been generated from 1000 bank customers' information and target output for this data is to define customer's being good or bad. The data description is in following figure (see Fig. 1).

	X01	X02	X03	X04	X05	X06	X07	X08	X09	X10	...	X12	X13	X14	X15	X16	X17	X18	X19	X20	Y
0	A11	6	A34	A43	1169	A65	A75	4	A93	A101	...	A121	67	A143	A152	2	A173	1	A192	A201	1
1	A12	48	A32	A43	5951	A61	A73	2	A92	A101	...	A121	22	A143	A152	1	A173	1	A191	A201	2
2	A14	12	A34	A46	2096	A61	A74	2	A93	A101	...	A121	49	A143	A152	1	A172	2	A191	A201	1
3	A11	42	A32	A42	7882	A61	A74	2	A93	A103	...	A122	45	A143	A153	1	A173	2	A191	A201	1
4	A11	24	A33	A40	4870	A61	A73	3	A93	A101	...	A124	53	A143	A153	2	A173	2	A191	A201	2
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
995	A14	12	A32	A42	1736	A61	A74	3	A92	A101	...	A121	31	A143	A152	1	A172	1	A191	A201	1
996	A11	30	A32	A41	3857	A61	A73	4	A91	A101	...	A122	40	A143	A152	1	A174	1	A192	A201	1
997	A14	12	A32	A43	804	A61	A75	4	A93	A101	...	A123	38	A143	A152	1	A173	1	A191	A201	1
998	A11	45	A32	A43	1845	A61	A73	4	A93	A101	...	A124	23	A143	A153	1	A173	1	A192	A201	2
999	A12	45	A34	A41	4576	A62	A71	3	A93	A101	...	A123	27	A143	A152	1	A173	1	A191	A201	1

1000 rows × 21 columns

Fig. 1. Original data set.

As seen from data it needs to be manipulated to implement algorithms. First step done here was to call columns with clear name which explains feature. Second step was to change data completely to the numeric format. These steps was done by the help of data description and as a result the following data (see Fig. 2) was obtained.

	Status of check	Duration of the credit (Month)	Credit history	Purpose of the credit	Credit amount (EUR)	Savings	Current Job	Installment rate(%)	Personal status	Guarantors	...	Property	Age(year)	Other installment	Housing situation	Credit Count
0	0	6	4	3	1169	4	4	4	2	0	...	0	67	2	1	2
1	1	48	2	3	5951	0	2	2	1	0	...	0	22	2	1	1
2	3	12	4	6	2096	0	3	2	2	0	...	0	49	2	1	1
3	0	42	2	2	7882	0	3	2	2	2	...	1	45	2	2	1
4	0	24	3	0	4870	0	2	3	2	0	...	3	53	2	2	2

5 rows × 21 columns

Fig. 2. Manipulated data set

2 Data Cleaning

In this section, the data has been checked whether it has a incomplete, irrelevant value or not and following picture (see Fig. 3) shows that the data is clean.

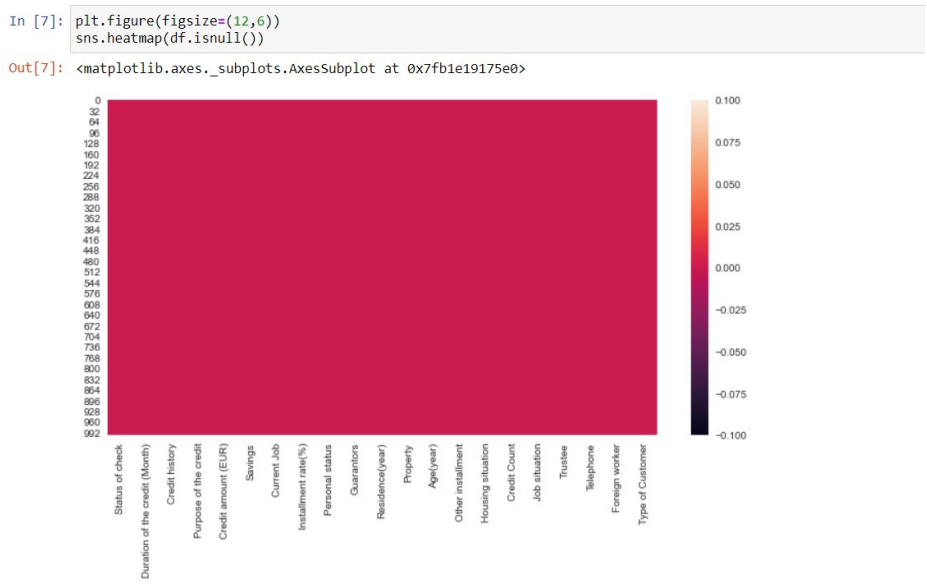

Fig. 3. Data cleaning checking

3 Exploratory Data Analysis

In this section, the distribution and correlation of data has been explored.The data distribution was conducted for each column and column related to another column (see Fig. 4). Generally, it can be conclude that the distribution for the age and credit amount feature is skewed-right, for the current job column is normal,for the trusty column which shows number of people the customer being liable to provide maintenance. Furthermore, the correlation results show that there is high correlation compared to others between credit amount and duration of the credit (positively 0.62) (see Fig. 5). Additionally, the data can be considered as an unbalanced date as seen from following picture (see Fig. 6).

4 Data Pre-Processing

In this section, the standardization was applied to the data with a purpose of converting the structure of disparate data set into a Common Data Format.

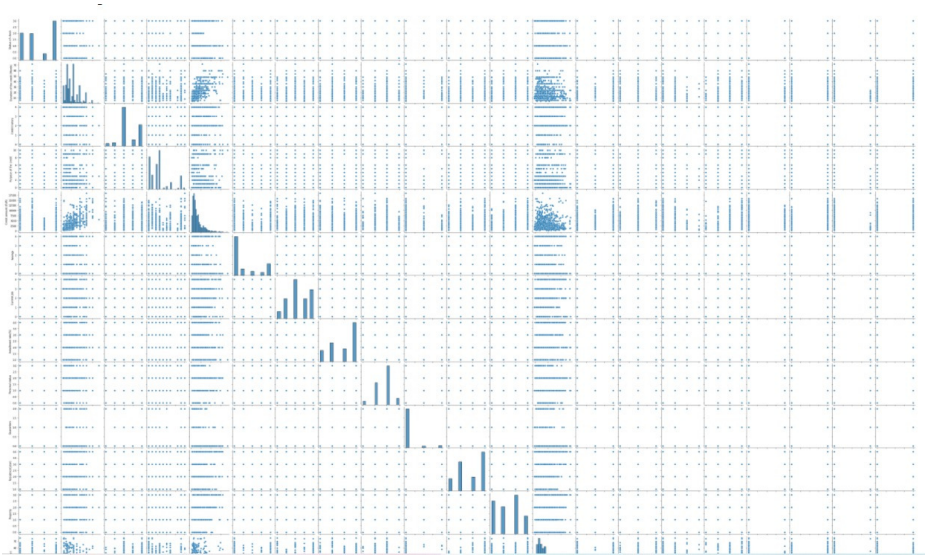

Fig. 4. Distribution

	Status of check	Duration of the credit (Month)	Credit history	Purpose of the credit	Credit amount (EUR)	Savings	Current Job	Installment rate(%)	Personal status	Guarantors	Residence(year)	Property
Status of check	1.000000	-0.072013	0.192191	0.028783	-0.042705	0.222867	0.106339	-0.005280	0.043261	-0.127737	-0.042234	-0.032260
Duration of the credit (Month)	-0.072013	1.000000	-0.077186	0.147492	0.624984	0.047661	0.057381	0.074749	0.014789	-0.024490	0.034067	0.303971
Credit history	0.192191	-0.077186	1.000000	-0.090336	-0.059905	0.039058	0.138225	0.044375	0.042171	-0.040676	0.063198	-0.053777
Purpose of the credit	0.028783	0.147492	-0.090336	1.000000	0.068474	-0.018684	0.016013	0.048369	0.000157	-0.017607	-0.038221	0.010966
Credit amount (EUR)	-0.042705	0.624984	-0.059905	0.068474	1.000000	0.064630	-0.008367	-0.271316	-0.016091	-0.027832	0.028926	0.311599
Savings	0.222867	0.047661	0.039058	-0.018684	0.064630	1.000000	0.120950	0.021993	0.017349	-0.105069	0.091424	0.018948
Current Job	0.106339	0.057381	0.138225	0.016013	-0.008367	0.120950	1.000000	0.126161	0.111278	-0.008116	0.245081	0.087187
Installment rate(%)	-0.005280	0.074749	0.044375	0.048369	-0.271316	0.021993	0.126161	1.000000	0.119308	-0.011398	0.049302	0.053391
Personal status	0.043261	0.014789	0.042171	0.000157	-0.016091	0.017349	0.111278	0.119308	1.000000	0.050634	-0.027269	-0.006940
Guarantors	-0.127737	-0.024490	-0.040676	-0.017607	-0.027832	-0.105069	-0.008116	-0.011398	0.050634	1.000000	-0.025678	-0.155450
Residence(year)	-0.042234	0.034067	0.063198	-0.038221	0.028926	0.091424	0.245081	0.049302	-0.027269	-0.025678	1.000000	0.147231
Property	-0.032260	0.303971	-0.053777	0.010966	0.311599	0.018948	0.087187	0.053391	-0.006940	-0.155450	0.147231	1.000000
Age(year)	0.059751	-0.036136	0.147086	0.001312	0.032716	0.084245	0.256227	0.058266	0.007783	-0.029873	0.266419	0.072606
Other installment	0.046841	-0.054884	0.121973	-0.096612	-0.046008	0.001908	-0.040154	-0.000983	-0.036765	-0.059023	0.002089	-0.090033
Housing situation	0.022424	0.157049	0.062095	0.018391	0.135632	0.006505	0.111126	0.089405	0.099579	-0.065889	0.011941	0.345219
Credit Count	0.076005	-0.011284	0.437066	0.054935	0.020795	-0.021644	0.125791	0.021669	0.064672	-0.025447	0.089625	-0.007765
Job situation	0.040663	0.210910	0.010350	0.008085	0.285385	0.011709	0.101225	0.097755	-0.011956	-0.057963	0.012655	0.276149
Trustee	-0.014145	-0.023834	0.011550	-0.032577	0.017142	0.027514	0.097192	-0.071207	0.122165	0.020400	0.042643	0.011872
Telephone	0.066296	0.164718	0.052370	0.078371	0.276995	0.087208	0.060518	0.014413	0.027275	-0.075035	0.095359	0.196802

Fig. 5. Correlation

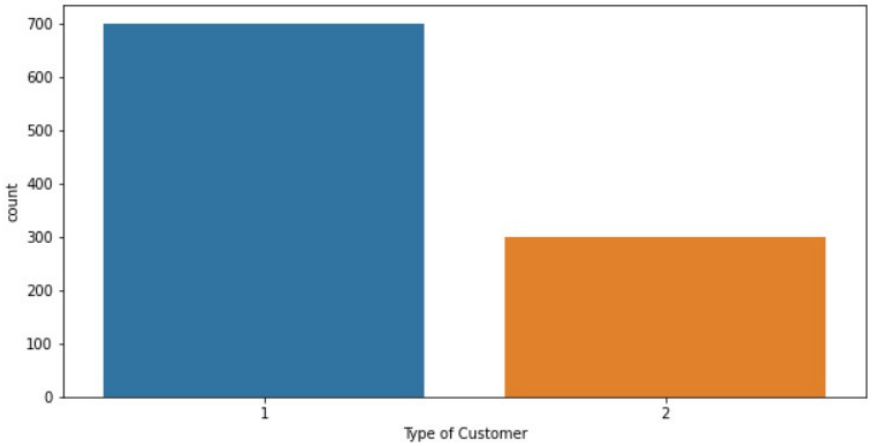

Fig. 6. Data exploratory for checking unbalance and balance

Therefore, as a tool sklearn.preprocessing.StandardScaler has been used. Also, the data set was separated as a training and test data.Here, 30 percentage of data for test, remaining for the training data was used.

5 Model Building

Here, to determine bad and good customer firstly, Logistic Regression has been applied to the data. The results obtaining from training data is in following picture (see Fig. 7).

If we observe result,it can be told that the accuracy is acceptable and there is no under-fitting.After, The next step was applying training model on test data and following results were obtained (see Fig. 8).

These results show that there is no over-fitting. The second algorithm applied to the data is Random Forest. Also, this algorithm was implemented on the same training and test data.The training results are in following picture (see Fig. 9) and it can be seen that the result is extraordinarily perfect.

The test data results are in following picture (see Fig. 10) and we can see that testing accuracy is almost the same with logistic regression accuracy. Also, it is obvious that there is over -fitting in random forest model.

5.1 Cross Validation

Cross Validation method was applied to get more accurate result. Because with this method you can separate data to different folds and expose each fold to train and test concepts and get average results of folds We applied cross validation only for the logistic regression and as a result 0.75 accuracy was obtained.

```
confusion_matrix
[[441  50]
 [111  98]]
classification_report
              precision    recall  f1-score   support

           1       0.80      0.90      0.85       491
           2       0.66      0.47      0.55       209

    accuracy                           0.77       700
   macro avg       0.73      0.68      0.70       700
weighted avg       0.76      0.77      0.76       700

Accuracy = 0.77
F1_score = 0.8456375838926175
```

Fig. 7. Logistic Regression train results.

```
confusion_matrix
[[189  20]
 [ 47  44]]
classification_report
              precision    recall  f1-score   support

           1       0.80      0.90      0.85       209
           2       0.69      0.48      0.57        91

    accuracy                           0.78       300
   macro avg       0.74      0.69      0.71       300
weighted avg       0.77      0.78      0.76       300

Accuracy = 0.7766666666666666
F1_score = 0.849438202247191
```

Fig. 8. Logistic Regression test results.

```
confusion_matrix
[[491   0]
 [  0 209]]
classification_report
              precision    recall  f1-score   support

           1       1.00      1.00      1.00       491
           2       1.00      1.00      1.00       209

    accuracy                           1.00       700
   macro avg       1.00      1.00      1.00       700
weighted avg       1.00      1.00      1.00       700

Accuracy = 1.0
F1_score = 1.0
```

Fig. 9. Random Forest train result.

```
confusion_matrix
[[194  15]
 [ 58  33]]
classification_report
              precision    recall  f1-score   support

           1       0.77      0.93      0.84       209
           2       0.69      0.36      0.47        91

    accuracy                           0.76       300
   macro avg       0.73      0.65      0.66       300
weighted avg       0.74      0.76      0.73       300

Accuracy = 0.7566666666666667
F1_score = 0.841648590021692
```

Fig. 10. Random Forest test result.

5.2 Recursive Feature Elimination

İn this subsection, it has been intended to select best features for applying Logistic Regression and Random Forest algorithms. For that reason, Recursive Feature Elimination algorithm has been used and following result was obtained (see Fig. 11). The result shows that 13 features have been selected and others have been eliminated as a week features for the Logistic Regression and for the Random forest best features number is 13 (see Fig. 12).

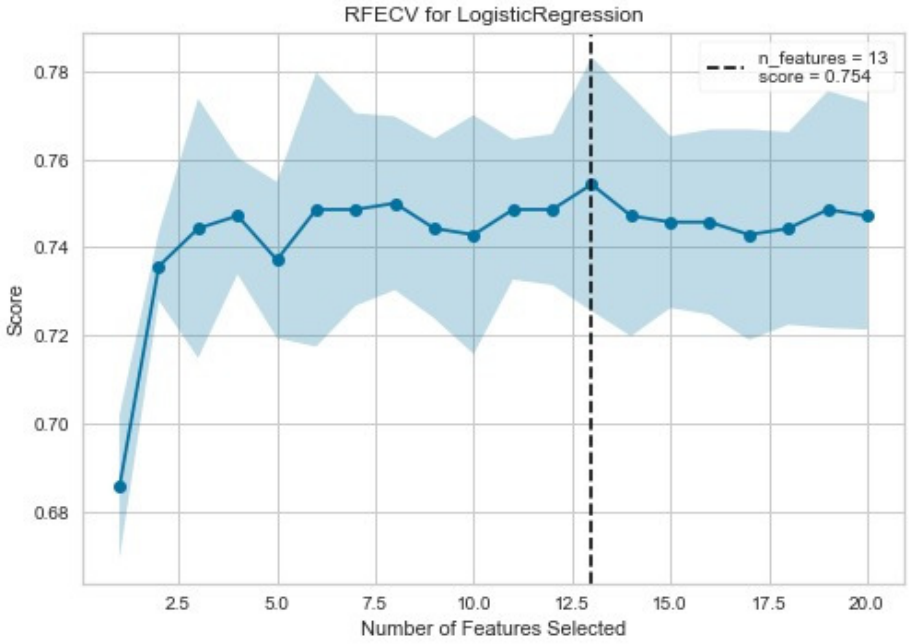

Fig. 11. Recursive Feature Elimination on Logistic Regression

5.3 Clustering

For the clustering problem, K-means algorithm has been applied. As a first step, one of the dimension reduction techniques Principal Component Analysis (PCA) has been implemented and two principal components selected (see Fig. 13). After implementing PCA, clustering was applied with 1 to 10 clusters and these clusters have been determined by elbow method in order to find which cluster is better (see Fig. 18).

From the elbow result,we can see that the best cluster number is 3. For the demonstration, we have been used Silhoutte Coefficient which is the metric used to calculate the goodness of a clustering technique. The clustering results are in following figures.

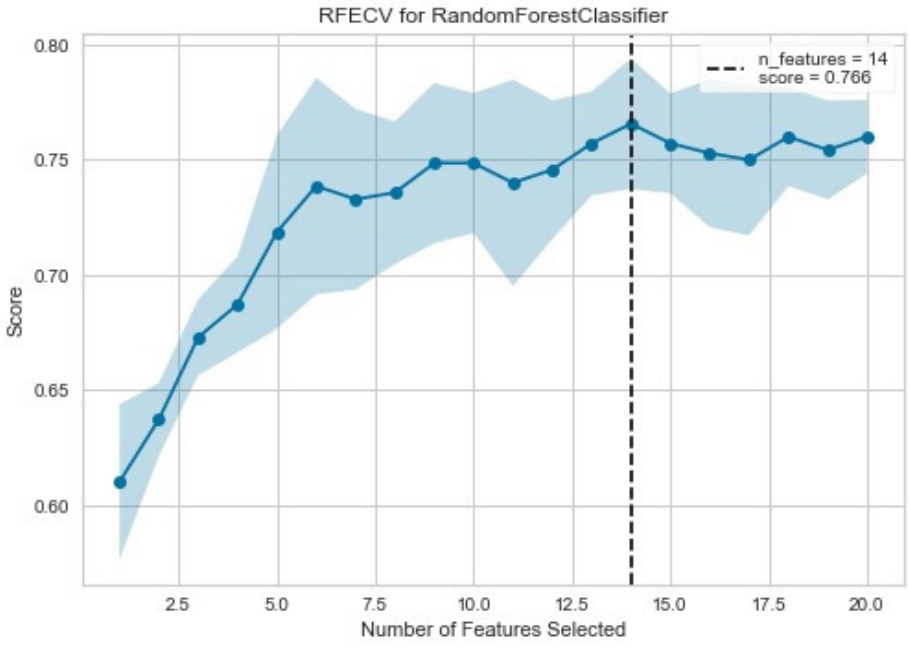

Fig. 12. Recursive Feature Elimination on Random Forest.

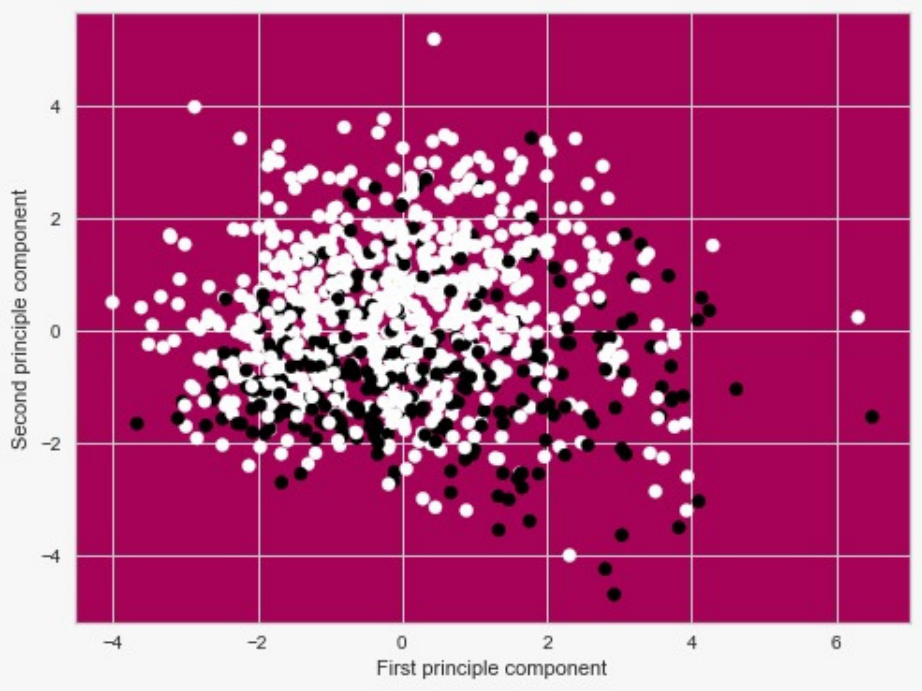

Fig. 13. PCA result.

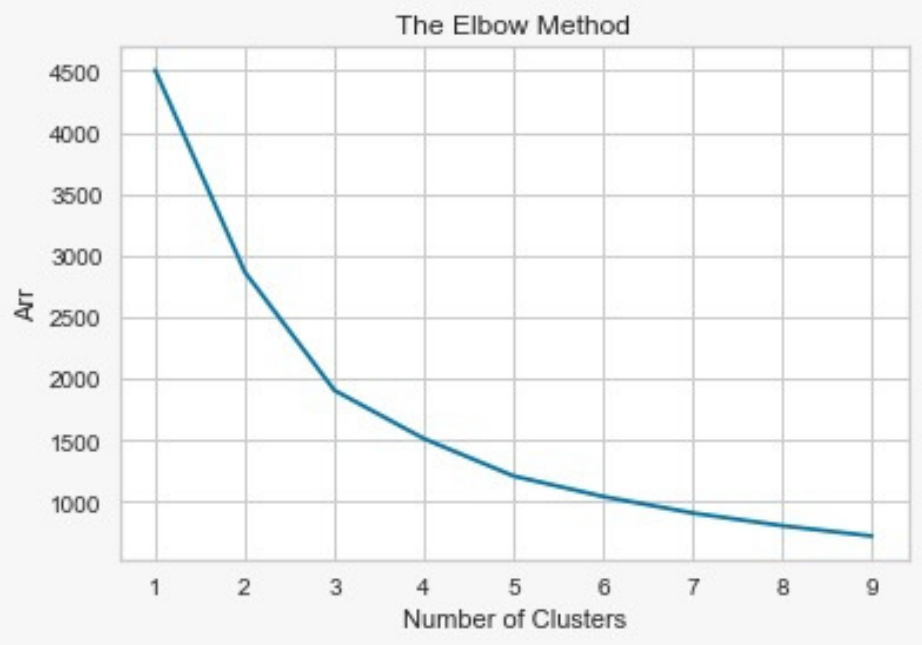

Fig. 14. Elbow result.

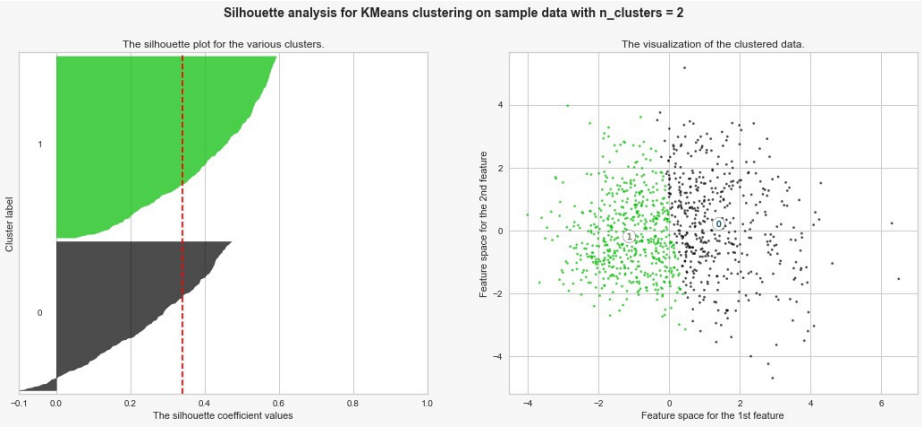

Fig. 15. Cluster 2

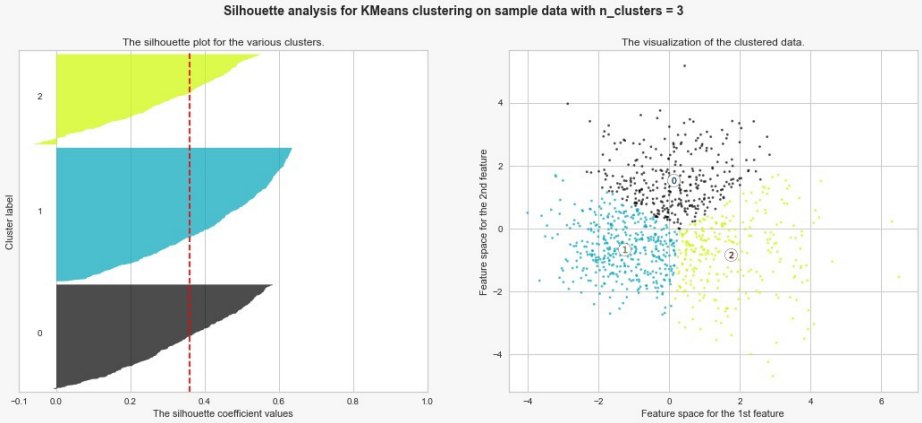

Fig. 16. Cluster 3

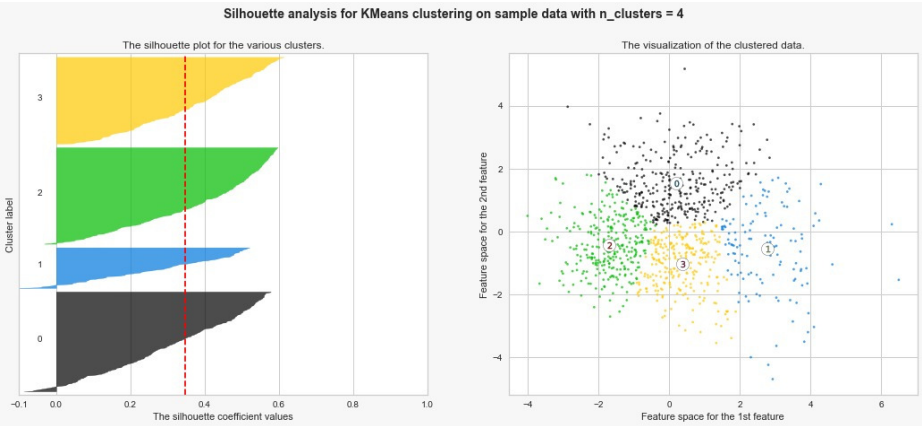

Fig. 17. Cluster 4

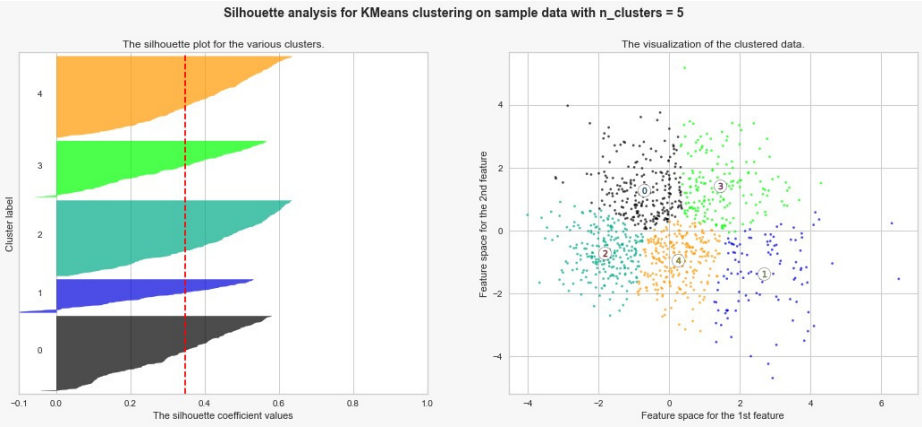

Fig. 18. Cluster 5