3η Εργασία στο μάθημα Big Data Mining

“Mining Over Datasets”

Γρηγόρης Μπουζιωτόπουλος

Βαγγέλης Κορμέντζας

Νικηφόρος Αλυγιζάκης

Διδάσκων: Ερευνητής Δρ. Η. Ζαβιτσάνος

Διϊδρυματικό πρόγραμμα μεταπτυχιακών σπουδών MSc in Data Science

Πανεπιστήμιο Πελοποννήσου και Εθνικό Κέντρο Έρευνας Φυσικών Επιστημών "ΔΗΜΟΚΡΙΤΟΣ"

Ημερομηνία κατάθεσης εργασίας: 14 Φεβρουαρίου 2021

Dataset 1

Tasks

- Cluster the types of crimes based on the success of the police in facing/solving them.

- Cluster the types of crimes and explain what each cluster represents.

- Identify outliers in crime types and explain what they represent/why they are outliers.

- Try to predict the super-category (e.g. ΕΠΙΚΡΑΤΕΙΑ/ΚΛΟΠΕΣ-ΔΙΑΡΡΗΞΕΙΣ, ...) of a record given only its numeric fields (τελ/να, απόπειρες, εξιχνιάσεις, ημεδαποί, αλλοδαποί), providing an explanation of the main factors for the decision and report the performance on a cross-validation evaluation.

Report

Before we could begin working on the tasks, the data on the excel had to be transformed. Therefore, columns which belonged to crime super categories, such as "ΚΛΟΠΕΣ - ΔΙΑΡΡΗΞΕΙΣ" etc., were removed from the initial data and "ΕΠΙΚΡΑΤΕΙΑ" was replaced by "ΟΝΟΜΑ ΕΓΚΛΗΜΑΤΟΣ". The super categories were then moved in a new tab named "ΥΠΕΡΚΛΑΣΕΙΣ". Furthermore, all the data around 2015 was also dropped from the dataset.

For the first task we have to cluster the types of crimes based on the success of the police in facing/solving them.

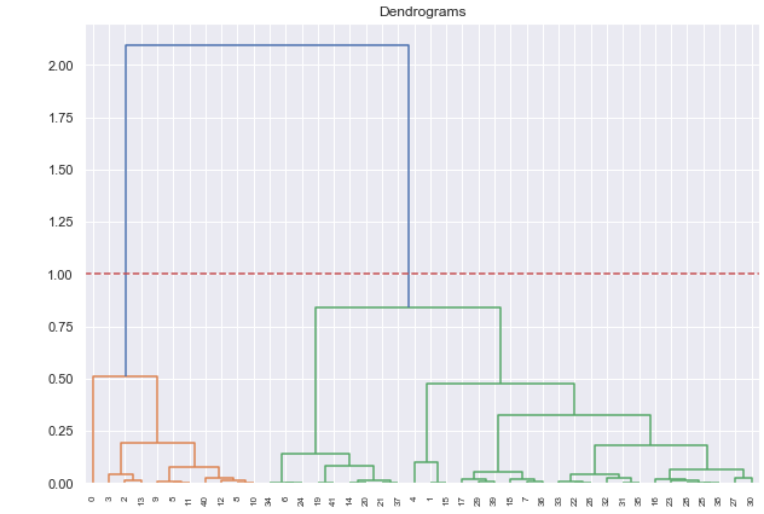
In order to achieve this, we have to keep the columns "τελ/να" and "απόπειρες". The type of clustering that was chosen is Hierarchical clustering since we have few data, and we do not know the number of clusters beforehand.

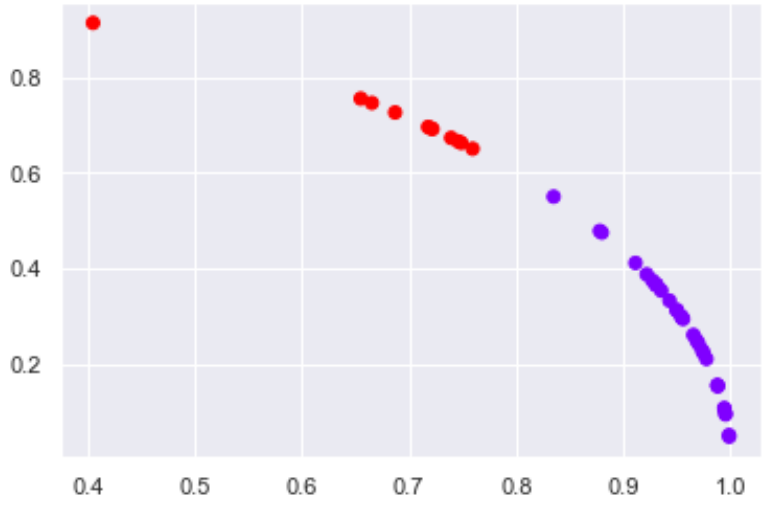
However, the decision of the number of clusters that can best depict different groups can be chosen by observing the dendrogram. The best choice of the number of clusters is the no. of vertical lines in the dendrogram cut by a horizontal line that can transverse the maximum distance vertically without intersecting a cluster.

For k- means clustering we would need to have prior knowledge about the clusters.

Before applying clustering, we have to normalize the data so that the scale of each variable is the same. Why is this important? If the scale of the variables is not the same, the model might become biased towards the variables with a higher magnitude like ΑΠΑΤΕΣ or ΕΠΑΙΤΕΙΑ.

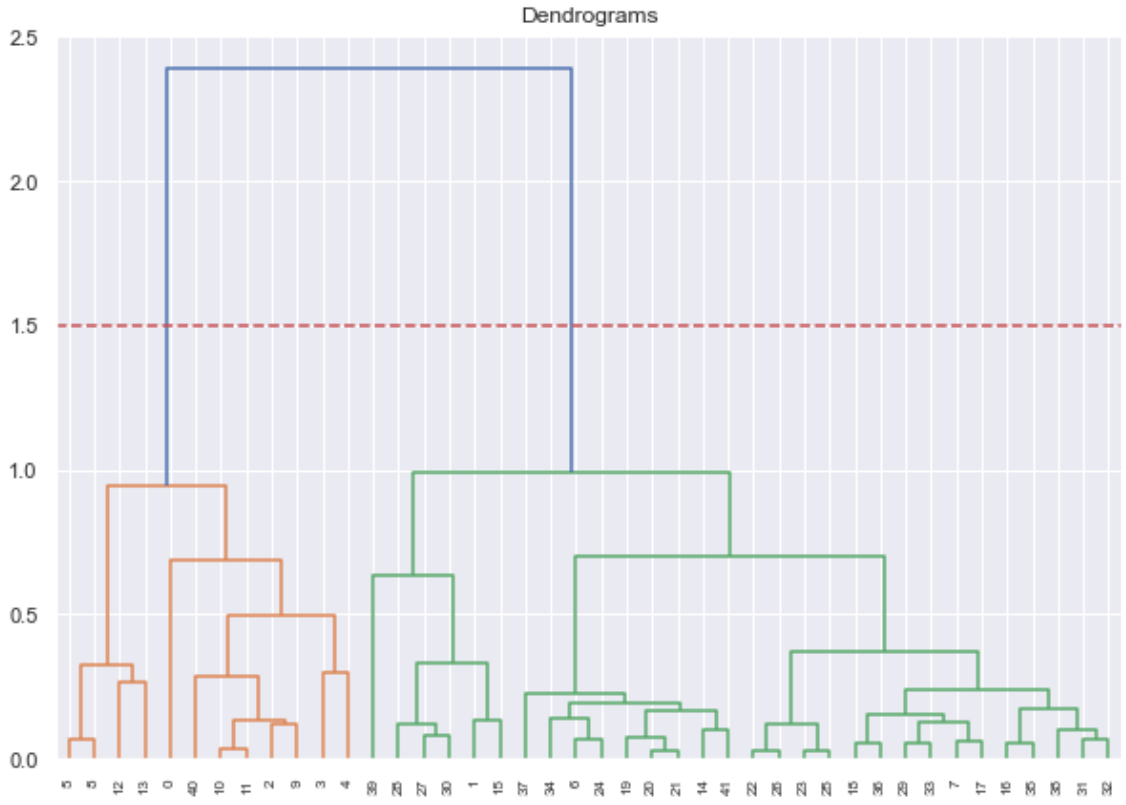
By observing the notebook – crime.ipynb, we can see that the number of clusters is 2 and that most of the data belonged to the first cluster.


Dendrograms
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00


0.8
0.6
0.4
0.2
0.4
0.5
0.6
0.7
0.8
0.9
1.0

```
('ΑΝΘΡΩΠΟΚΤΟΝΙΕΣ', 1),
('ΑΠΑΤΕΣ', 0),
('ΑΡΧΑΙΟΚΑΠΗΛΕΙΑ', 1),
('ΒΙΑΣΜΟΙ', 1),
('ΕΚΒΙΑΣΕΙΣ', 0),
('ΕΠΑΙΤΕΙΑ', 1),
('ΖΩΟΚΛΟΠΗ', 0),
('ΚΥΚΛΟΦΟΡΙΑ ΠΑΡΑΧΑΡΑΓΜΕΝΩΝ', 0),
('ΛΑΘΡΕΜΠΟΡΙΟ', 1),
('Ν περί ΝΑΡΚΩΤΙΚΩΝ', 1),
('Ν περί ΟΠΛΩΝ', 1),
('Ν περί ΠΝΕΥΜΑΤΙΚΗΣ ΙΔΙΟΚΤΗΣΙΑΣ', 1),
('ΠΛΑΣΤΟΓΡΑΦΙΑ', 1),
('ΣΕΞΟΥΑΛΙΚΗ ΕΚΜΕΤΑΛΛΕΥΣΗ', 1),
('Κλοπές - Διαρρήξεις από ιχε αυτ/τα', 0),
('Κλοπές - Διαρρήξεις ιερών ναών', 0),
('Κλοπές - Διαρρήξεις καταστημάτων', 0),
('Κλοπές - Διαρρήξεις λοιπές', 0),
('Κλοπές - Διαρρήξεις οικιών', 0),
('Κλοπές - Διαρρήξεις σε συγκοινωνιακά μέσα', 0)
('Κλοπές με αρπαγές τσαντών', 0),
('Κλοπές σε δημόσιο χώρο-μικροκλοπες', 0),
('Κλοπές Τροχοφόρων ΙΧΕ αυτ/των', 0),
('Κλοπές Τροχοφόρων ΙΧΦ-Λεωφορείων', 0),
('Κλοπές Τροχοφόρων Λοιπών οχημάτων', 0),
('Κλοπές Τροχοφόρων Μοτοποδηλάτων', 0),
('Κλοπές Τροχοφόρων Μοτοσυκλετών', 0),
('Ληστείες εντός καταστημάτων', 0),
('Ληστείες εντός οικιών', 0),
('Ληστείες κινητών τηλεφώνων-μικροποσών', 0),
('Ληστείες λοιπές', 0),
('Ληστείες με αρπαγή τσάντας', 0),
('Ληστείες οδηγών ταξί', 0),
('Ληστείες πρατηρίων υγρών καυσίμων', 0),
('Ληστείες σε ΕΛ.ΤΑ.', 0),
('Ληστείες σε Μίνι Μάρκετ-κατ/τα ψιλικών', 0),
('Ληστείες σε περίπτερα', 0),
('Ληστείες σε πρακτορεία ΟΠΑΠ', 0),
('Ληστείες σούπερ μάρκετ', 0),
('Ληστείες ταχυδρομικών διανομέων', 0),
('Ληστείες τραπεζών,ταχ/κών ταμιευτηρίων', 1),
('Ληστείες χρηματαποστολών', 0)])
```

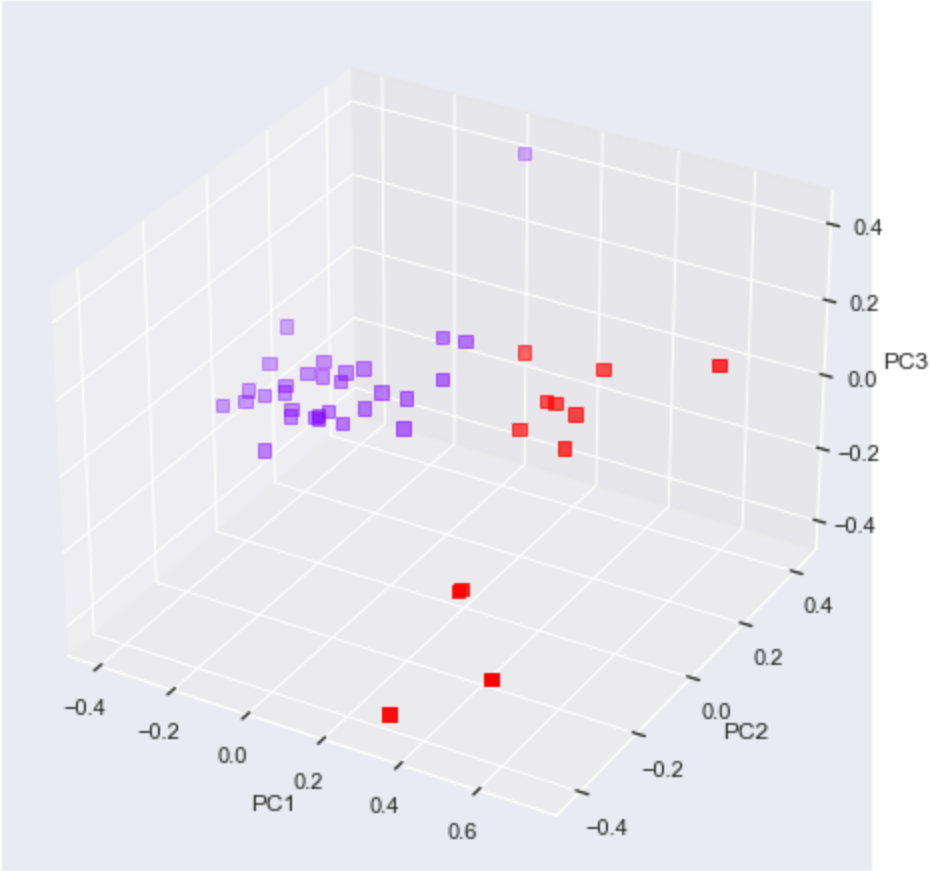
For the second task we have to cluster the types of crimes and explain what each cluster represents. We will use the same algorithm as before but this time we will keep all the columns except for the "ΟΝΟΜΑ ΕΓΚΛΗΜΑΤΟΣ"


Dendrograms
2.5
2.0
1.5
1.0
0.5
0.0
5 8 12 13 0 40 10 11 2 9 3 4 39 28 27 30 1 15 37 34 6 24 19 20 21 14 41 22 26 23 25 18 36 29 33 7 17 16 35 38 31 32

```
[('ΑΝΘΡΩΠΟΚΤΟΝΙΕΣ', 1),
 ('ΑΠΑΤΕΣ', 0),
 ('ΑΡΧΑΙΟΚΑΠΗΛΕΙΑ', 1),
 ('ΒΙΑΣΜΟΙ', 1),
 ('ΕΚΒΙΑΣΕΙΣ', 1),
 ('ΕΠΑΙΤΕΙΑ', 1),
 ('ΖΩΟΚΛΟΠΗ', 0),
 ('ΚΥΚΛΟΦΟΡΙΑ ΠΑΡΑΧΑΡΑΓΜΕΝΩΝ', 0),
 ('ΛΑΘΡΕΜΠΟΡΙΟ', 1),
 ('Ν περί ΝΑΡΚΩΤΙΚΩΝ', 1),
 ('Ν περί ΟΠΛΩΝ', 1),
 ('Ν περί ΠΝΕΥΜΑΤΙΚΗΣ ΙΔΙΟΚΤΗΣΙΑΣ', 1),
 ('ΠΛΑΣΤΟΓΡΑΦΙΑ', 1),
 ('ΣΕΞΟΥΑΛΙΚΗ ΕΚΜΕΤΑΛΛΕΥΣΗ', 1),
 ('Κλοπές - Διαρρήξεις από ιχε αυτ/τα', 0),
 ('Κλοπές - Διαρρήξεις ιερών ναών', 0),
 ('Κλοπές - Διαρρήξεις καταστημάτων', 0),
 ('Κλοπές - Διαρρήξεις λοιπές', 0),
 ('Κλοπές - Διαρρήξεις οικιών', 0),
 ('Κλοπές - Διαρρήξεις σε συγκοινωνιακά μέσα', 0),
 ('Κλοπές με αρπαγές τσαντών', 0),
 ('Κλοπές σε δημόσιο χώρο-μικροκλοπες', 0),
 ('Κλοπές Τροχοφόρων ΙΧΕ αυτ/των', 0),
 ('Κλοπές Τροχοφόρων ΙΧΦ-Λεωφορείων', 0),
 ('Κλοπές Τροχοφόρων Λοιπών οχημάτων', 0),
 ('Κλοπές Τροχοφόρων Μοτοποδηλάτων', 0),
 ('Κλοπές Τροχοφόρων Μοτοσυκλετών', 0),
 ('Ληστείες εντός καταστημάτων', 0),
 ('Ληστείες εντός οικιών', 0),
 ('Ληστείες κινητών τηλεφώνων-μικροποσών', 0),
 ('Ληστείες λοιπές', 0),
 ('Ληστείες με αρπαγή τσάντας', 0),
 ('Ληστείες οδηγών ταξί', 0),
 ('Ληστείες πρατηρίων υγρών καυσίμων', 0),
 ('Ληστείες σε ΕΛ.ΤΑ.', 0),
 ('Ληστείες σε Μίνι Μάρκετ-κατ/τα ψιλικών', 0),
 ('Ληστείες σε περίπτερα', 0),
 ('Ληστείες σε πρακτορεία ΟΠΑΠ', 0),
 ('Ληστείες σούπερ μάρκετ', 0),
 ('Ληστείες ταχυδρομικών διανομέων', 0),
 ('Ληστείες τραπεζών,ταχ/κών ταμιευτηρίων', 1),
 ('Ληστείες χρηματαποστολών', 0)])
```

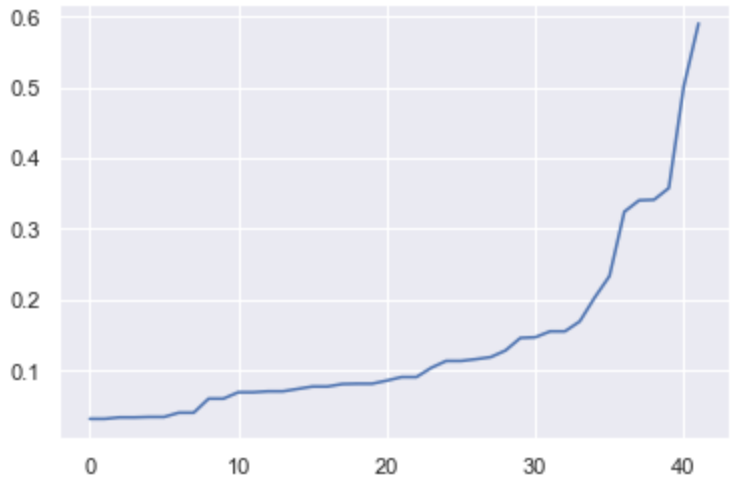
We can now see that the crimes are again divided into 2 clusters. In the one cluster we have all crimes that belong to 'Ληστείες' and 'Κλοπές` as well as some other general crimes while on the other cluster we have the rest of the crimes, with some exceptions.

We used the PCA algorithm, for 3 dimensions, in order to visualize the clusters.



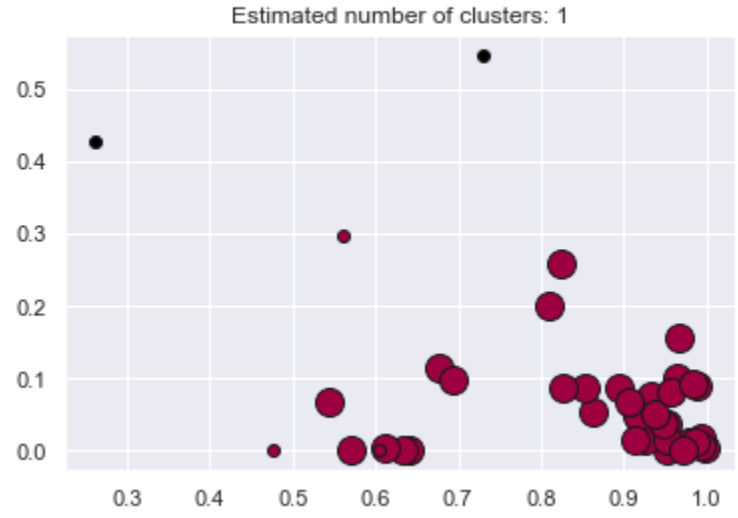
From the schema above we can see that the purple cluster is denser and has few outliers, while the red cluster is sparser with more outliers.

For the third task – the outlier detection, the DBSCAN algorithm was used. In order to find a suitable value for DBSCAN epsilon value did calculate the distance to the nearest n points for each point, sorting and plotting the results. Then we looked to see where the change is most pronounced and select that as epsilon. The epsilon that was chosen was 0.35.



DBSCAN Results

```
Estimated number of clusters: 1
Estimated number of noise points: 2
Silhouette Coefficient: 0.422
```



We can see that DBSCAN detects only one possible cluster with a minimum of 5 samples. By observing the cluster, we can also see that it is a bit dense, however some samples are farther than others. Finally, we can only spot 2 outliers in the grid.

Outliers:

['ΑΝΘΡΩΠΟΚΤΟΝΙΕΣ', 'Ληστείες ταχυδρομικών διανομέων']

The above data points are outliers. One possible explanation for this could be that the police keep data from previous years as well. For example, a crime could have been committed on a previous year, for example 2014, and it may have been solved on 2016. Therefore, it will be included in the data.

For the last task - predicting the super category of the crimes we will use 4 algorithms: RandomForest, Decision Tree, SVM and KNN, with GridSearch and cross validation, in order to find the best hyperparameters. These algorithms were chosen because they are suitable for classification, especially the ensemble algorithm - RandomForest.

The dataset was split into a train (70%) and a test set (30%).

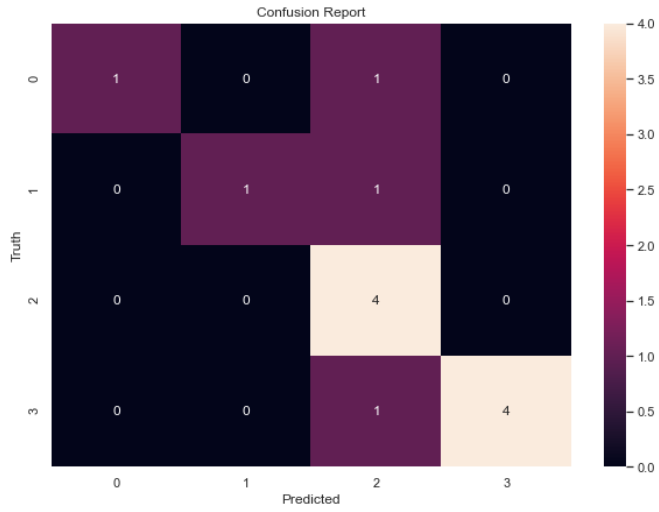
Classification using Random Forest Classifier

```
Best Parameters:  {'criterion': 'gini', 'max_depth': 4, 'max_features': 'auto', 'n_estimators': 100}
Best Accuracy Score Achieved in Grid Search:  0.7333333333333333
Classification Report
                    precision    recall  f1-score   support

ΚΛΟΠΕΣ - ΔΙΑΡΡΗΞΕΙΣ       1.00      0.50      0.67         2
  ΚΛΟΠΕΣ ΤΡΟΧΟΦΟΡΩΝ       1.00      0.50      0.67         2
           ΛΗΣΤΕΙΕΣ       0.57      1.00      0.73         4
    ΛΟΙΠΑ ΕΓΚΛΗΜΑΤΑ       1.00      0.80      0.89         5

          accuracy                           0.77        13
         macro avg       0.89      0.70      0.74        13
      weighted avg       0.87      0.77      0.77        13

Accuracy per class
{0: 0.5, 1: 0.5, 2: 1.0, 3: 0.8}
```



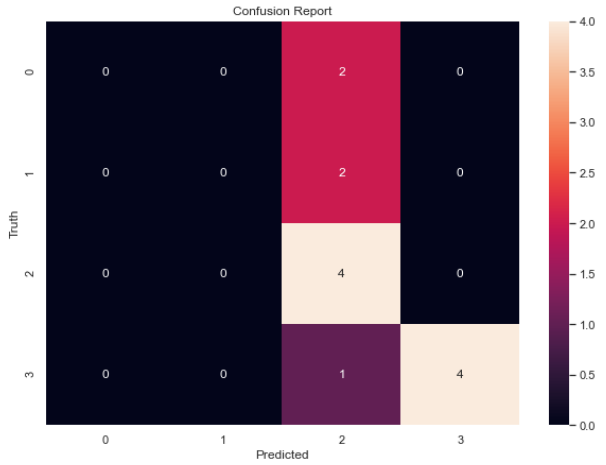
Classification using SVM Classifier

```
Best Parameters:  {'C': 1000, 'gamma': 0.001, 'kernel': 'rbf'}
Best Accuracy Score Achieved in Grid Search:  0.5833333333333333
Classification Report
                     precision    recall  f1-score   support

ΚΛΟΠΕΣ - ΔΙΑΡΡΗΞΕΙΣ       0.00      0.00      0.00         2
  ΚΛΟΠΕΣ ΤΡΟΧΟΦΟΡΩΝ       0.00      0.00      0.00         2
           ΛΗΣΤΕΙΕΣ       0.44      1.00      0.62         4
    ΛΟΙΠΑ ΕΓΚΛΗΜΑΤΑ       1.00      0.80      0.89         5

           accuracy                           0.62        13
          macro avg       0.36      0.45      0.38        13
       weighted avg       0.52      0.62      0.53        13

Accuracy per class
{0: 0.0, 1: 0.0, 2: 1.0, 3: 0.8}
```



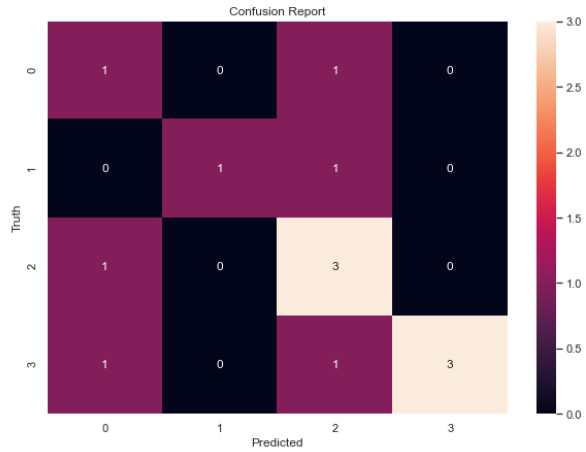
Classification using kNN Classifier

```
Best Parameters:  {'n_neighbors': 5}
Best Accuracy Score Achieved in Grid Search:  0.6333333333333333
Classification Report
                     precision    recall  f1-score   support

ΚΛΟΠΕΣ - ΔΙΑΡΡΗΞΕΙΣ       0.33      0.50      0.40         2
  ΚΛΟΠΕΣ ΤΡΟΧΟΦΟΡΩΝ       1.00      0.50      0.67         2
           ΛΗΣΤΕΙΕΣ       0.50      0.75      0.60         4
    ΛΟΙΠΑ ΕΓΚΛΗΜΑΤΑ       1.00      0.60      0.75         5

           accuracy                           0.62        13
          macro avg       0.71      0.59      0.60        13
       weighted avg       0.74      0.62      0.64        13

Accuracy per class
{0: 0.5, 1: 0.5, 2: 0.75, 3: 0.6}
```



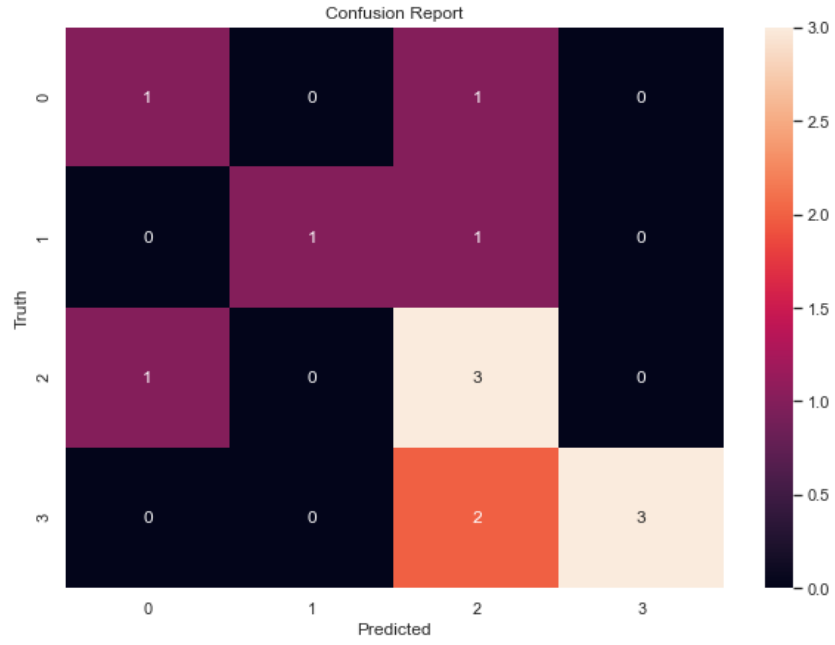
Classification using Decision Tree Classifier

```
Best Parameters:  {'criterion': 'gini', 'max_depth': 4, 'min_samples_leaf': 2, 'min_samples_split': 2}
Best Accuracy Score Achieved in Grid Search:  0.7
Classification Report
                    precision    recall  f1-score   support

ΚΛΟΠΕΣ - ΔΙΑΡΡΗΞΕΙΣ       0.50      0.50      0.50         2
  ΚΛΟΠΕΣ ΤΡΟΧΟΦΟΡΩΝ       1.00      0.50      0.67         2
          ΛΗΣΤΕΙΕΣ       0.43      0.75      0.55         4
    ΛΟΙΠΑ ΕΓΚΛΗΜΑΤΑ       1.00      0.60      0.75         5

          accuracy                           0.62        13
         macro avg       0.73      0.59      0.62        13
      weighted avg       0.75      0.62      0.64        13

Accuracy per class
{0: 0.5, 1: 0.5, 2: 0.75, 3: 0.6}
```



From the above results and the confusion matrices, we can observe that the Random Forest performs better than the rest algorithms.

However, the overall scores are not too high. That makes sense, since the dataset is quite small and contains cumulative information from the past crime years as well.

Dataset 2

One of the challenges of the second dataset was to properly parse the dataset. The python packagee *arff* was not able to read the dataset, because one of the airline carriers was marked as "PA (1)". For this reason, we read line by line and parsed the dataset with a for loop. The list was transformed in a panda's data frame. However, the columns of the data frame were identified as object type, which is a non-suitable data type to apply the desirable functions. For this reason, we wrote the data frame in a csv file and read again the csv setting proper data type. We used pandas *scatter_matrix* function to plot the distribution of the features. Moreover, we printed the dataset size and the features as requested by the exercise.

For the second task of the exercise ("Describe the average delays per airport/airline"), we used pandas and *groupby* function to investigate the average delay per airline and per airport (destination airport and origin airport). We noticed that the column ArrDelay has positive and negative values. Therefore, we filtered flights that arrived without delay or earlier.

The third task ("Identify and report the most prominent rules of association between delays and point of origin AND/OR point of arrival.") was dealt with a similar manner. We grouped by both origin and destination airport and used *count* function to investigate the routes with the higher number of delayed flights. Moreover, we counted the delayed flights by origin and destination airport.

For the forth task ("Try to predict the delay given all other features and report the appropriate performance on cross-validation."), we detected and removed outlier instances using the function *stats.zscore*. 1857 flights with extreme delays were removed. Moreover, we plotted the outlier detection in a scatter plot. Afterwards, we converted the categorical columns in multiple columns with zero and one using the function *get_dummies* from pandas package. The dataset was split in training set and test set using the function *train_test_split* from sklearn package. We performed model selection by trying four different models (linear regression, ridge regression, elastic net, K neighbors regressor) and 10-fold cross validation. Ridge regression showed the best performance. We used grid search to tune the ridge regression hyperparameters and trained the model. We evaluated the performance on the test set using as performance criteria the mean squared error and R^2.

For the firth task ("Identify patterns/rules regarding delays and try to explain when delays should be expected, based on these patterns."), we used descriptive statistics to highlight that IAD airport presents the highest delays. Afterwards, we applied apriori algorithm to detect i) the most frequently delayed flight routes, ii) the most frequently delayed airlines based on the departure time and iii) the most frequently delayed airlines based on the day of the week. We extracted the respective rules in all cases.

Dataset 3

The first step was to load the dataset and explore it in order to familiarize ourselves with the dataset. The attributes of the dataset which were used to answer the questions were:

- CNAME
- TOTCHUR
- Religions= (234 - 8)/2= 113
- ARAPO_M, GRKAD_M, ACROC_M, BEOC_M for the Orthodox Members question

In order to summarize the data, we used the function *describe*, which shows us the needed information.

As for the question “Which are the countries with the highest per person ratio of Orthodox Christian members?”, we found out that there were four Orthodox churches which create the Orthodox dogma. Those attributes were ARAPO, ACROM, GRKAD and BEOC. In order to answer the question, we summarized the members of those religions and we divided the result with the TOTMEMB (Total number of members). To get the ratio, we multiplied that number with 100 and sorted the dataset with descending order.

For the question “Can you find the 3 most extreme (outlier) counties with respect to the distribution of their churches across religions?”, we used two methods to find the extreme values. In the first method, we found the number of churches for every county and sorted the dataset with descending order. Moreover, as a second method, we visualized the result using boxplot to show the extreme values.

For the question “Where would you create a cross-religion center of discussion between religions to maximize its impact? Support the proposal based on data analysis results.”, we sorted the dataset according to the Total number of churches and we concluded that the best place to build a cross-religion center of discussion was the province with the highest number of churches.