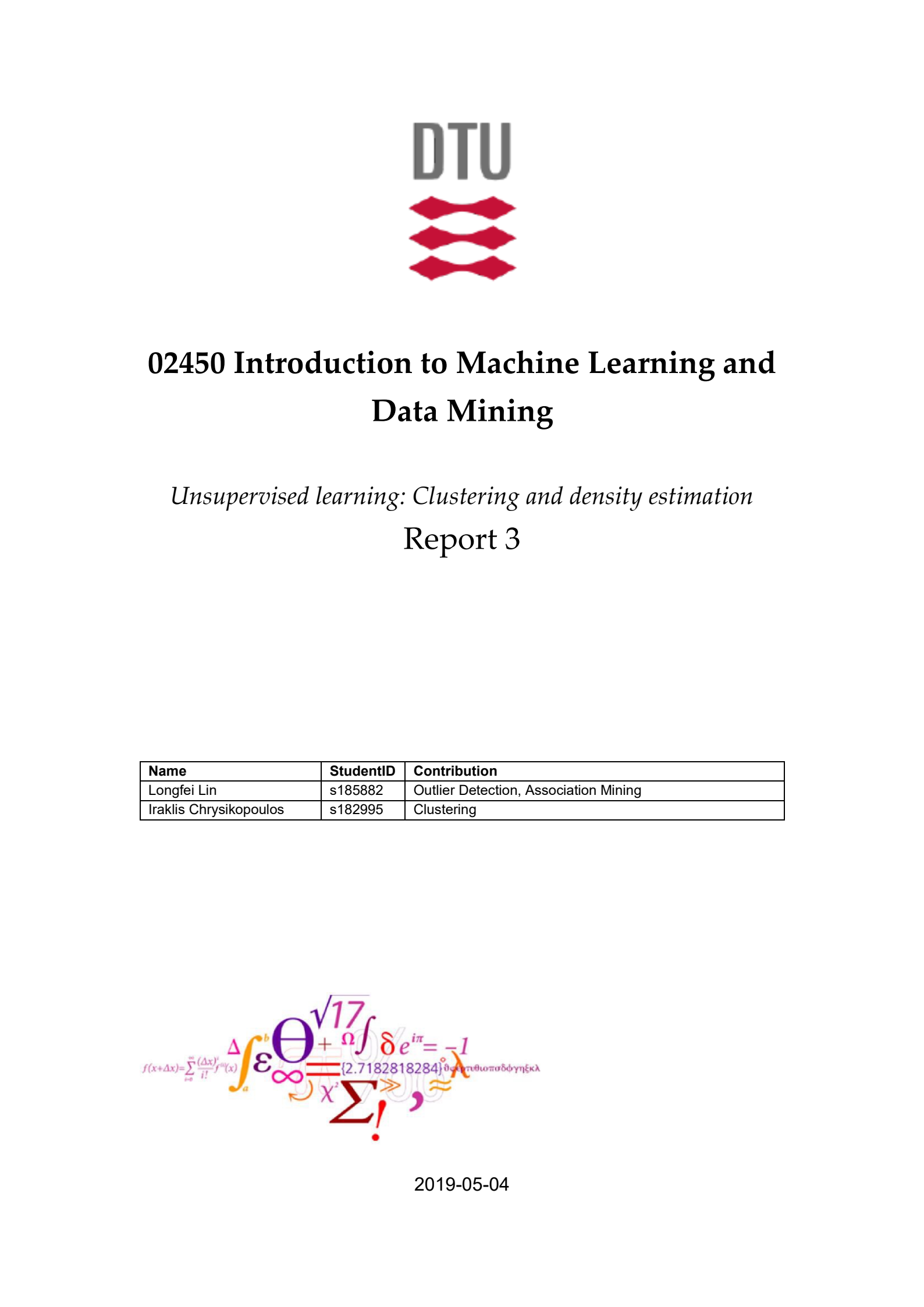

02450 Introduction to Machine Learning and Data Mining

Unsupervised learning: Clustering and density estimation

Report 3

Name	StudentID	Contribution
Longfei Lin	s185882	Outlier Detection, Association Mining
Iraklis Chrysikopoulos	s182995	Clustering

2019-05-04

1. Clustering

In unsupervised machine learning, a method named clustering of data is used and in this part it will be shown and discussed how our dataset about heart disease can be analyzed in this way. A known method in clustering is Gaussian Mixture Model (GMM) and with cross-validation we will find the optimum number of clusters for our dataset. After that another method of clustering called hierarchical clustering (HC) will be applied in the dataset using different linkage functions. Finally, a comparison will be made between GMM and HC methods in order to extract conclusions about the quality of the clustering that has been made.

1.1 Gaussian Mixture Model (GMM)

Gaussian Mixture Model is a method of clustering and in other words is a set of multivariate normal distribution and is used to make a more flexible distribution to our data. In the first step we try to find a good number of cluster to our dataset. Thus we use the BIC (Bayesian information criterion), AIC (Akaike information criterion) and 10-fold cross validation.

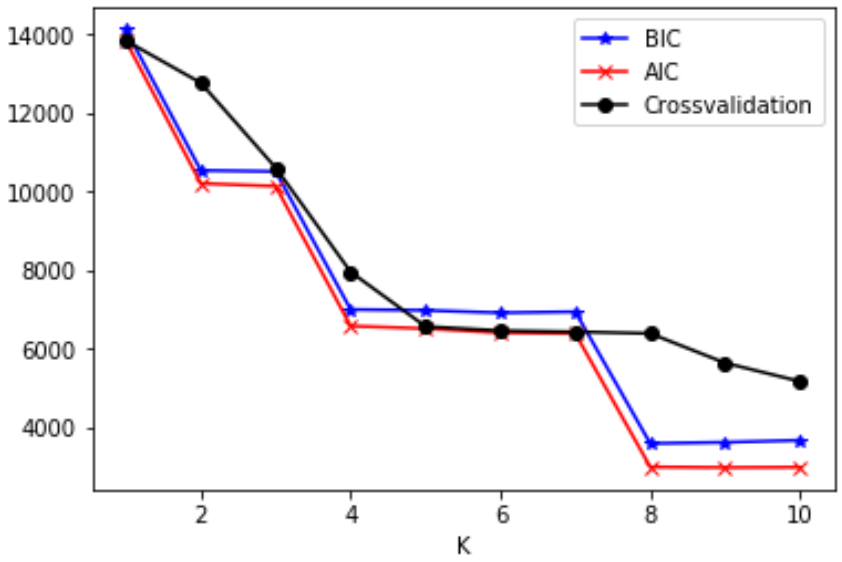

Fig. 1 - BIC, AIC, 10-fold cross validation – score vs number of clusters

From Fig. 1, we can extract information about the optimal number of clusters/components we should choose to proceed. As the score gets lower then that's the optimal number of cluster we should choose. In our case we can see that the cross-validation curve stabilizes after K=4 so the best number of clusters is 4. Also, it's not useful to choose a large number of cluster i.e. K=10. With the python scripts there are many options concerning covariance matrix constraints such as diagonal, full, tied etc.

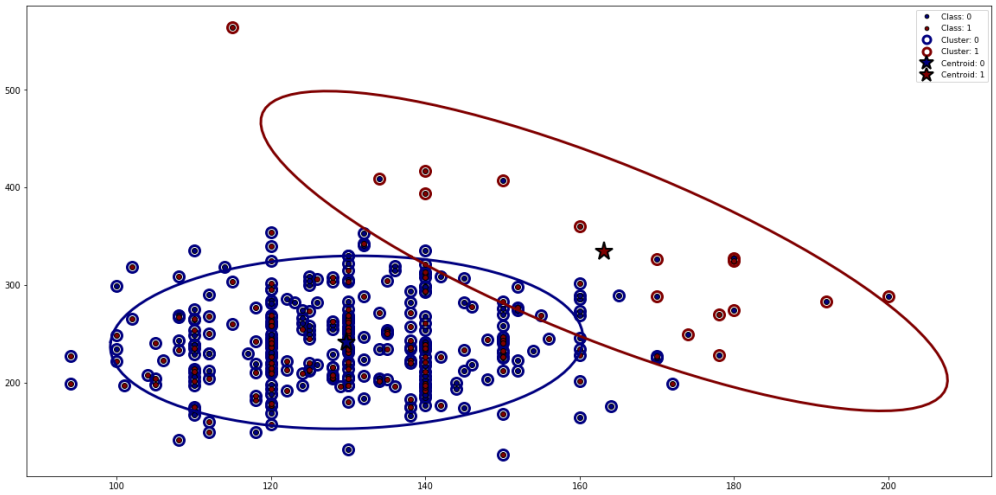

Fig. 2 - Cluster representation and centroids

From Fig. 2 it is shown the 2 different clusters that occur and their centroids. The attributes that are used are TRESTBPS (resting blood pressure) and CHOL (serum cholesterol).

1.2 Hierarchical Clustering

Hierarchical clustering arranges the data in a nested sequence of partitions organized as a hierarchy and as a result overcomes the limitation by instead of finding a single K. The low point of the hierarchy correspond to each observation in a unique cluster and the top point in the hierarchy is a unique cluster of all observations. This kind of clustering needs a specific method called linkage function. In this way, this function can find the closest neighbor and then merge the two clusters in a tree shape (dendrograms). The three function that we use in this project are:

- Minimum (single)
- Maximum (complete)
- Average

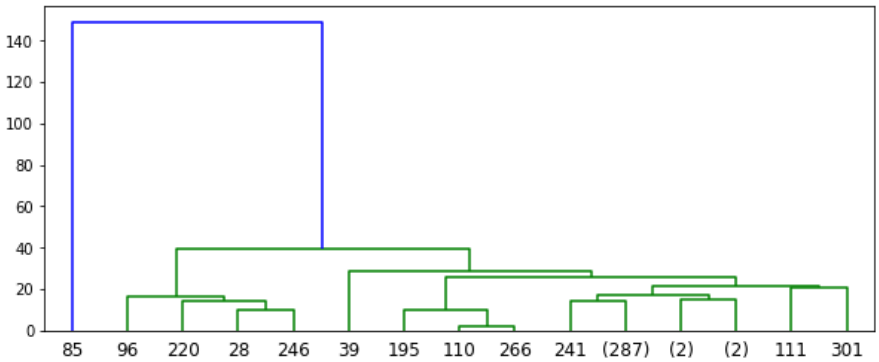

Fig. 3 - Dendrogram of Single Linkage Function

In this case of the minimum linkage the distance between the groups is the distance between the closest pair of observations. We can see from the colors that our data are separated in 2 clusters.

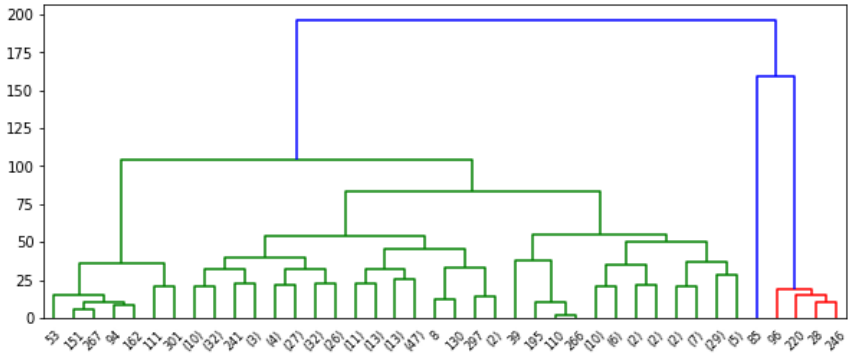

Fig. 4. - Dendrogram of Average Linkage Function

In this case of the average linkage, the distance between the groups is the average distance between all pairs in the groups. We can see a separation of our data which gives us 3 clusters.

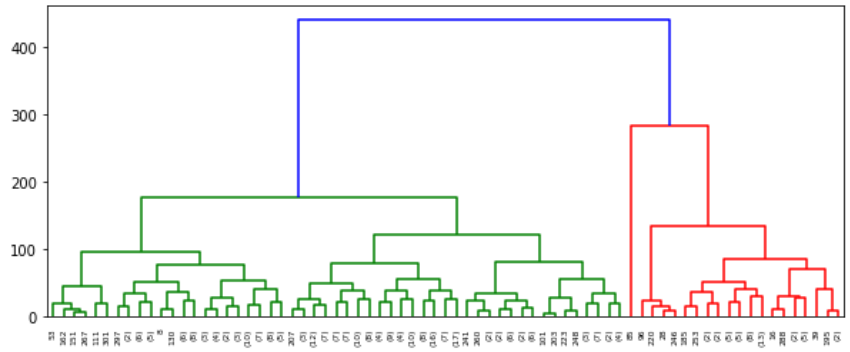

Fig. 5. - Dendrogram of Complete Linkage Function

In the last case of the maximum linkage the distance between the groups is the distance between the most distant pair of observations. We can see from the colors that our data are separated in 3 clusters. The complete linkage function seems to have better results.

1.3 Hierarchical Clustering vs GMM – Quality Evaluation

In order to compare and evaluate the two different methods of clustering we use 3 different indices for cluster validity: Rand statistic, Jaccard coefficient and normalized mutual information (NMI).

GMM	
Rand	0.12
Jaccard	0.12
NMI	0

Table 1 - Indices for GMM evaluation

Hierarchical Clustering	
Rand	0.50
Jaccard	0.50
NMI	0

Table 2 - Indices for hierarchical clustering evaluation

From table 1 and 2, it is easy to compare and decide which method gives us better clusters. In our case Hierarchical clustering is better as we see from Rand and Jaccard indices. For some reason NMI in both cases is 0, so we don't take it into account.

2. Outlier Detection

In this part, we used three different density estimators including Gaussian Kernel Density, KNN density and KNN average relative density to evaluate and rank the outlier score of our data set, and then we talk about whether there may be outliers in our data set based on three different scoring methods.

2.1 Gaussian Kernel Density Estimator

At first, we applied a gaussian kernel density Estimator on our data set which has 303 obersevations. A kernel density estimator is a deterministic approximation to the Gaussian Mixture Model (GMM) which tries to overcome some limitation of GMM, and in this report, we get the optimal kernel width λ by using leave-one-out cross validation which can calculate the maximum log of the likelihood for the whole data set. The following table is the values of the log of the likelihood in different λ :

Fold	Kernel Width	Log of the Likelihood
1	0.000977	-inf
2	0.001953	-inf
3	0.003906	-inf
4	0.007813	-inf
5	0.015625	-inf
6	0.031250	-24745.546978
7	0.062500	-12963.483128
8	0.012500	-7790.083359
9	0.250000	-5901.747565
10	0.500000	-5612.601211
11	1.000000	-6044.790314
12	2.000000	-6770.159939
13	4.000000	-7681.069602

Table 3 - The log of the likelihood

We selected 13 different λ from 2^{-10} to 2^{2}, and we found that we got the highest log likelihood when kernel width $\lambda = 0.5$, then we calculated the density using kernel width $\lambda = 0.5$. The results of the outlier scores as follows:

\

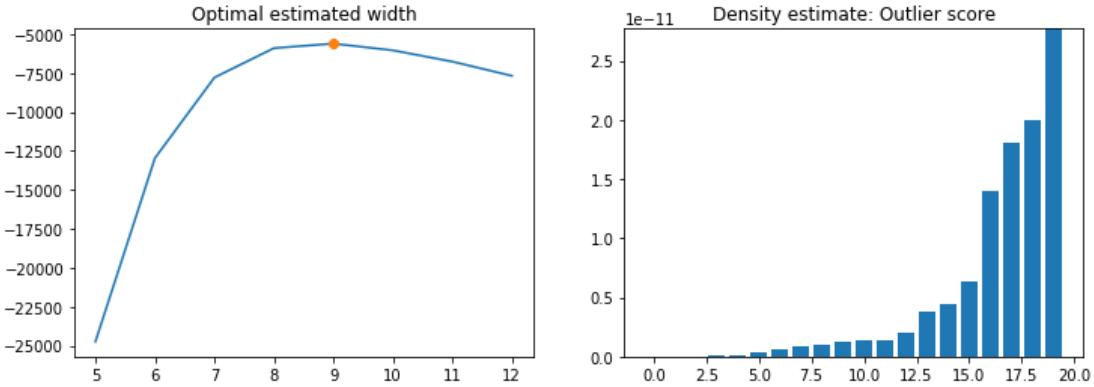

Fig. 6 – Gaussian kernel density estimation

We show the 20 worst (lowest) scores in our data set in the right-hand pane of the Figure. It's clear that there is a big difference between them, and the index of the first 5 observations are 291, 85, 223, 292 and 260.

2.2 K-Nearest Neighbor Density

Second, we used a KNN density estimator to measure the density of the data set. The KNN density tries to overcome the problem that KDE and GMM are unable to handle clusters of different densities well and it uses the K-nearest neighbor algorithm to calculate the average distance in Euclidian space. The core expression as follows:

$$density_X(x,k) = \frac{1}{\frac{1}{K}\sum_{x' \epsilon N_x(x,K)} d(x,x')}$$

In this report, we chose K=4 because our data set is relatively small. The results of the outlier scores are plotted below:

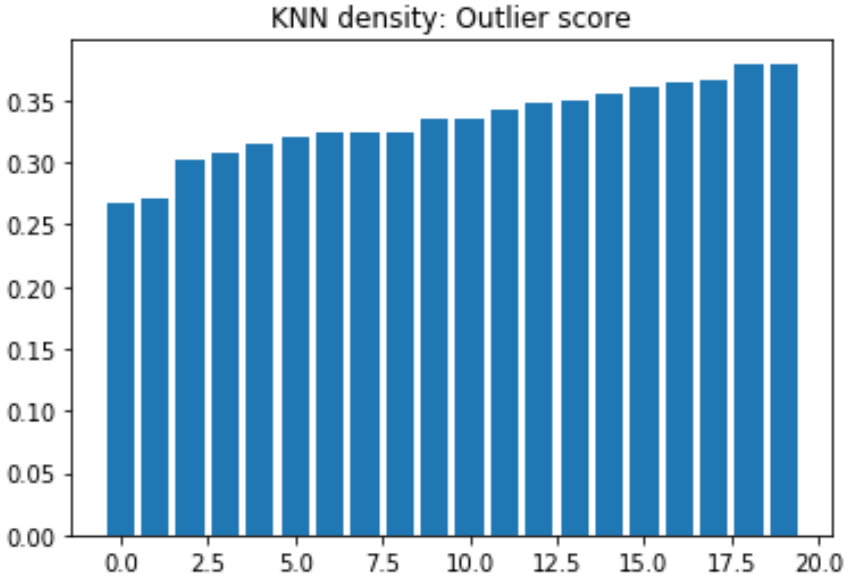

Fig. 7 – KNN density estimation

In this figure, we showed the 20 worst scores of all observations, but they don't have an obvious difference base on this histogram. The indices of the worst 5 observations are 291, 85, 223, 204 and 292.

2.3 K-Nearest Neighbor Average Relative Density

Finally, we used KNN average relative density because we also wish to find some points where the density is lower than what it typically is for surrounding points. The KNN average relative density is one method which can be useful to define a notion of density that is relative to the neighborhood of the object. The expression as follows:

$$ard_X(x,k) = \frac{1}{\frac{1}{K}\sum_{x' \epsilon N_x(x,K)} d(x,x')}$$

So in this way, we could get the results that the density of a given observation x relative to the average of its K nearest neighbor.

In this estimator, we used the same K as what we did in KNN density, and the results as follows:

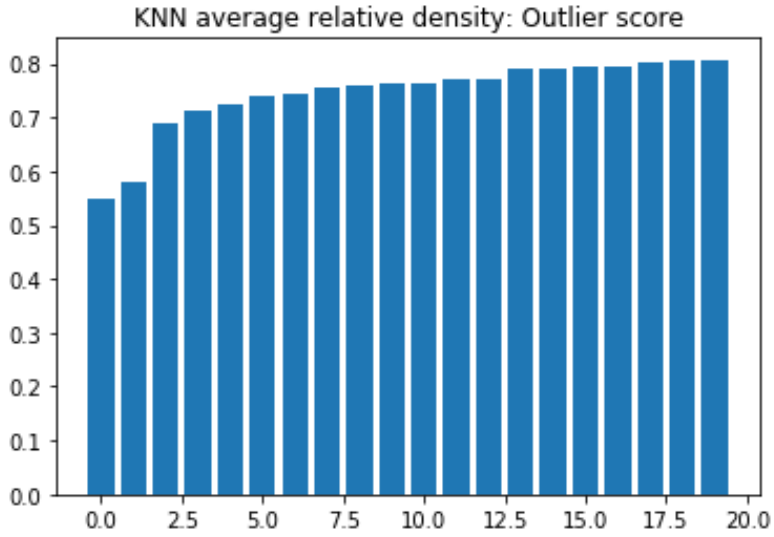

Fig. 8 – KNN average relative density estimation

This figure also plotted the 20 worst observations of the data set, the indices of the worst 5 observations are 48, 24, 195, 122 and 141, which are relatively different from the previous two estimators, but we also found that index of the 10th observation in this figure is 291.

2.4 Outlier Detection based on Scoring Methods

Based on previous experiments and results from three different scoring methods, we found that there are some observations which have relatively low scores based on density. There are 2 common definitions of the outlier:

1) *Hawkin's definition*: An outlier is an observation that differs so much from other observations as to arouse suspicion that it was generated by a different mechanism.

2) *Probabilistic definition*: An outlier is an object that has a low probability with respect to a probability distribution model of the data.

Based on the Probabilistic definition, we can say that there are some outliers in our data set because we used some scoring methods to estimate the density distribution of our data set, and the results showed there are some observations which low density, which means that these observations have a low probability compared with the probability distribution model of the whole data. However, when we checked the real data of these observations such as the 291st observation:

55	*0*	*0*	*128*	*205*	*0*	*2*	*130*	*1*	*2*	*1*	*0*

We found that there was no outliers according to the Hawkin's definition because the values of the all attributes are in a normal scope.

So, generally, there may be outliers based on the scoring methods (Probabilistic definition) and there is no outlier according to the real meaning and values of the observations and attributes (Hawkin's definition).

3. Association Mining

3.1 Apriori Algorithm

In this part, we will investigate the associations in our data set using the Apriori Algorithm. In order to using the algorithm, first, we converted our data set into a format suitable for association mining by using the one-out-of-k encoding technique because there are some continual and categorical attributes in our data set. The result is as follows:

X_og	float64	(303, 13)	[[63. 1. 3. ... 0. 0. 1.] [37. 1. 2. ... 0. 0. 2.]
X_trans	float64	(303, 30)	[[1. 1. 0. ... 0. 0. 1.] [0. 0. 1. ... 1. 0. 1.]

Fig. 7 - Comparing the size of the data set

From this figure, we can find that the size of the data set has been transformed from 303×13 to 303×30, and all the attributes are binary values. Then, we generate many transactions based on the observations and attributes name.

Size	Value
303	[['age 50th-100th percentile', 'trestbps 50th-100th percentile', 'oldp ...

Fig. 8 - Transactions

Finally, we can apply the Apriori Algorithm on our data set. For easier analyzing, we set a threshold for support and confidence, the algorithm will find association rules with $\geqslant$ 20% and $\geqslant$80%. Because there are too many associations so we just show part of results below:

Association	Support	Confidence
{cp_0, thal_3} -> {target_0}	0.234	0.910
{sex_0, target_1} -> {thal_2}	0.228	0.958
{target_0, trestbps 0th-50th percentile} -> {sex_1}	0.211	0.901
{thal_2, oldpeak 0th-50th percentile, age 0th-50th percentile} -> {target_1}	0.221	0.918
{thal_2, age 0th-50th percentile, slope_2} -> {target_1}	0.201	0.924

Table 4 – The associations in our data set

3.2 Analysis of the results

The reason why we set support $\geqslant$20% is we set the maximal k is 4 that when we applied the one-out-of-k encoding techniques on part of the attributes, so we cannot have support $\geqslant$25%, that's one of the limitations of Apriror algorithm which mentioned in the textbook [1].

Our data set is for predicting the heart disease, so all attributes except target are the health conditions of a person, i.e. the values of an observation. The 'sex_0' means male, 'sex_1' means female, the 'oldpeak' means ST depression induced by exercise relative to rest, the 'trestbps means resting blood pressure, the 'slope' means slope of the peak exercise ST segment, the 'cp' means chest pain type and the 'thalach' means maximum heart rate achieved.

So, according to the results table, we can say that people who have a chest pain type 3 and thal type 3 are more likely don't have a heart disease; people who are male and have heart disease are more likely have a thal type 2;people who don't have heart disease and the resting blood pressure lower than 130 are more likely a female; people who have a thal type 2, the ST depression induced by exercise relative to rest lower than 0.8 and the age lower than 55 are more likely to have heart disease; people who have a thal type 2, the age lower than 55 and the slope of the peak exercise ST segment is 2 are more likely to have a heart disease.

References

[1] Introduction to Machine Learning and Data Mining, Technical University of Denmark. Tue Herlau, Mikkel N. Schmidt and Morten Morup

[2] https://archive.ics.uci.edu/ml/datasets/Heart+Disease