Data Science Project 2018

Group #47

>> 16 de Novembro de 2018 <<

Inês Leite, ist181328
Gonçalo Rodrigo, ist181543
André Xiang, ist181577

Contents

1. INTRODUCTION

In this project our goal is to apply data science techniques to discover information in two distinct problems (datasets), we will create models about the data we have, and then understand and relate those models to conclude new information. Additionally, we will also criticize the results achieved, and discuss the difficulties faced on mining the different datasets.

1.1 First Problem - APS Dataset

The dataset consists of data collected from heavy Scania trucks in everyday usage. The system in focus is the Air Pressure system (APS) which generates pressurised air that are utilized in various functions in a truck, such as braking and gear changes. The datasets' positive class consists of component failures for a specific component of the APS system. The negative class consists of trucks with failures for components not related to the APS. The data consists of a subset of all available data, selected by experts.

1.1.1 First look at the Dataset

Number of Instances: The training set contains 60000 observations in total in which 59000 belong to the negative class and 1000 positive class. The test set contains 16000 observations.

Number of Attributes: 171 variables 7 of which are categorical variables. Missing values are denoted by 'na'.

Attribute Information: The attribute names of the data have been anonymized for privacy reasons. It consists of both single numerical counters and histograms consisting of bins with different conditions. Typically the histograms have open-ended conditions at each end.

1.2 Second Problem - Colposcopies Dataset

The dataset consists of data acquired and annotated by professional physicians from Colposcopies they performed at 'Hospital Universitario de Caracas'. The dataset has three modalities (i.e. Hinselmann, Green, Schiller), which have included original images and the manual segmentations.

1.2.1 First look at the Dataset

The first problem is composed by 3 datasets (Green, Hinselmann and Schiller), one for each modality. of the three, datasets, Hinselmann is described by 97 subjects, Green with 98 subjects and Schiller with 98 subjects, and then each of these 3 datasets all have the 69 different attributes, the first 62 attributes are predictive attributes, floating numbers equal or above zero, and the last 7 attributes are target variables which can be 0 or 1.

The low amount of subjects in each of these 3 datasets(97, 98 and 98) meant that if we merged the 3 datasets to one with total of 287 subjects, we would get a more consistent dataset, but we decided in end to not do that because we analyzed the means and values of each of the three dataset and found that they were quite different.

Given that these 3 datasets were part of the same experience, but acquired with different image modalities, the question whether to merge the datasets or perform individual analysis rose. Below, we will explain our reasoning to not merge the three datasets, but suffice is to say that we conducted extensive and in depth analysis of three different statistical indicatiors:

- Distribution of features
- Correlogrammic analysis of attributes across the board
- Chi-Square tests to check for the difference of the distributions of the categorical variables (i.e., experts annotations and consensus annotations.

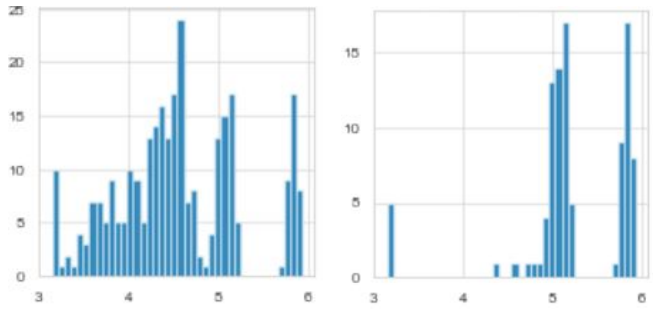

Figure 1. Attribute "hsv_cervix_h_mean" distribution for Merged Dataset(Left) and Green Dataset(Right)

After this, and being a surprise to us (the feedback for the teaching staff was contraditory), we decided to keep the 3 datasets separate so that we could get extra conclusions in the end in comparing the different modalities since they had different behaviours in distribution.

With these two different distribution graphs we show our reasoning for keeping the 3 datasets separate, the first graph is the attribute distributions after merging together the 3 datasets and the second graph is the attribute distributions of the Green Dataset individually, comparing these two graphs it should be obvious that there are key differences in the Distributions, most of these attributes have different properties after being merged into one from the three datasets, so we thought that merging together the datasets we would lose information.

2. PRE-PROCESSING

2.1 Problem 1 - APS Dataset

- **Pre-processing 1:** Apply **Feature Selection** method to first find pairs with a certain correlation to then remove one of the value of each of these pairs.
- **Pre-processing 2:** Dataset balancing with **Oversampling** method.
- **Pre-processing 3:** Filter our dataset to get a better selective dataset to work with, therefore choosing **Principal Component Analysis(PCA)** method to parametrize the dataset, which is an unsupervised linear dimensionality reduction algorithm that helps us find a more meaningful basis and coordinate system for our data by leaving only the strongest features of the dataset.
- **Pre-processing 4:** We apply **SMOTE**, which is a data balancing technique.
- **Pre-processing 5: Normalize** the dataset, with two types of normalization: L1-norm and L2-norm. L1-norm, known as least absolute deviations is basically minimizing the sum of the absolute differences between the target value and the estimated values. L2-norm, known as least squares, is basically minimizing the sum of the square of the differences between the target value and the estimated values.
- **Pre-processing 6: Discretization** to allow for more frequent itemsets to be found by concentrating the values in bins.
- **Pre-processing 7:** See if there were any **missing values**, which we did find and there were many, so to process the missing values by replacing them with the mean value of the respective column.

Baseline pre-processing: Out of the steps above, the baseline of pre-processing will be composed by *Pre-processing 1 + Pre-processing 2 + Pre-processing 8*

What we didn't apply:

- **Cross-Validation:** It was unnecessary for this dataset since it was already such a big sample of data with so many instances, even if we divided the dataset into separate sections for cross-validation, even the smaller section would still be big enough to not gain enough new information with this pre-processing.

2.2 Problem 2 - Colposcopies Dataset

- **Pre-processing 1:** Apply **Feature Selection** method to first find pairs with a certain correlation to then remove one of the value of each of these pairs.
- **Pre-processing 2:** Dataset balancing with **Oversampling** method.
- **Pre-processing 3:** Apply the **Pearson's chi-squared test** to this dataset to find independent variables between the consensus and the experts.
- **Pre-processing 4:** Filter our dataset to get a better selective dataset to work with, therefore choosing **Principal Component Analysis(PCA)** method to parametrize the dataset, which is an unsupervised linear dimensionality reduction algorithm that helps us find a more meaningful basis and coordinate system for our data by leaving only the strongest features of the dataset.
- **Pre-processing 5: Normalize** the dataset, with two types of normalization: L1-norm and L2-norm.
- **Pre-processing 6:** Give an external estimator(logistic regression) that assigns weights to features (e.g., the coefficients of a linear model), the goal of **Recursive feature elimination(RFE)** is to select features by recursively considering smaller and smaller sets of features.
- **Pre-processing 7: Discretization** to allow for more frequent itemsets to be found by concentrating the values in bins.

Baseline pre-processing: Out of the steps above, the baseline of pre-processing will be composed by *Pre-processing 1 + Pre-processing 2 + Pre-processing 3*

What we didn't apply:

- **Missing Values:** Here in the problem 2 we looked aswell at the dataset to see if there were any **missing values**, but this time we didn't find any missing values, so we just didn't have to process any missing values.
- **Undersampling:** In this dataset we already had very few instances, so applying undersampling here would be a overkill and balancing with undersampling is just not necessary and causes loss of information.

3. EXPLORATION

3.1 Problem 1 - APS Dataset

3.1.1 Methods and Parametrization

- **Association Rules:** Baseline pre-processing + Pre-processing 7

  To find patterns in Association rules we applied here the Apriori algorithm method over the **discretized** APS Dataset to try and find association rules.

- **Clustering:** Baseline pre-processing + Pre-processing 4

  The method of our choice here was K-means clustering, a popular cluster analysis method of vector quantification.

- **Classification:**

  KNN: Baseline + Pre-processing 5

  Naive Bayes: Baseline pre-processing

  Decision Trees: Pre-processing 5 + Pre-processing 6

  Random Forest: Pre-processing 5 + Pre-processing 6

3.1.2 Results

So first regarding pre-processing:

- For the correlation value, we ended up considering the value of 0.94, because as seen from the **Figure 2** below it was the best balance for the situation, when using higher values we would be losing too many pairs and using lower values we would be finding too many pairs. (removing one of the values in each pair)

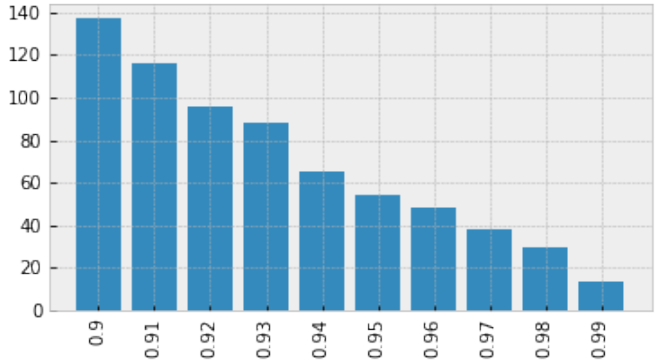

Figure 2. Pair Correlation Values of Problem 1 Dataset

- In the chi-squared Statistical test the resulted p-value was close to zero, and therefore with a p-value $<$ 0.05 we can conclude that the variables are independent of one another, rejecting then the null hypothesis and conclude that the variables are correlated.
- Our application of the PCA method, as seen from the **Figure 3** below, led us to conclude the value of 10 for the number of components and thevalue of 2 for the numbers of clusters, given the percentage of variance explained.

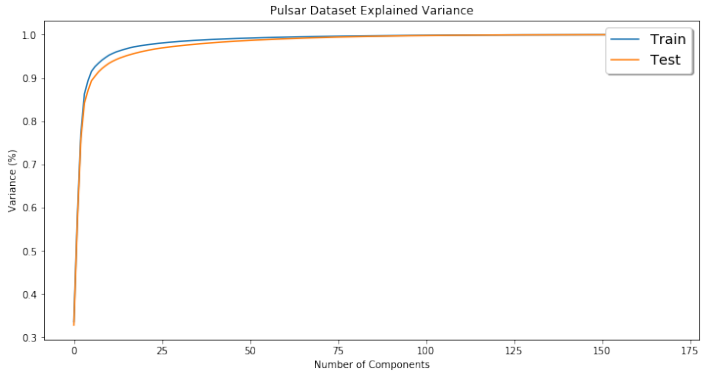

Figure 3. Pulsar Dataset Explained Variance for PCA

- The initial results from K-means clustering that we ran was not as expected, because the distance between the resulting clusters was too big, but we have then applied the PCA method and did again after the K-means clustering analysis the second time to achieve a better result, and the comparison of before and after can be seen below in the **Figure 4**.
- As seen from the comparison in **Figure ??** below, which shows the five most used types of balancing, we can conclude that the type of balancing with highest true positive rate compared to false positive rate, in other words, the one with highest accuracy, is SMOTE.

Now regarding the results from the Association Rules, Clustering and Classification:

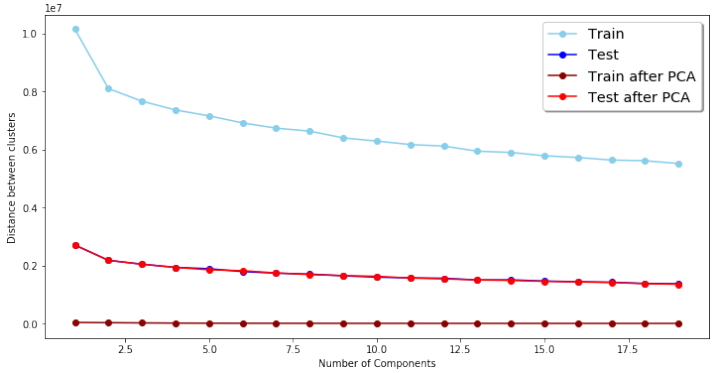

Figure 4. Normalized distance between clusters compared to number of components of the the training and test datasets before and after applying PCA

- In the Apriori algorithm method we had to find the most appropriate parametrization for our situation dataset to find association rules. Because of the nature of this dataset(way too many instances of class negative and very little amount of class positive), we had to do many iterations of different minimum confidence level and many more of different minimum support level, specially having to decrease the minimum support level to a very low value until we finally found the first frequent itemsets as seen in **Figure 5**, and in the end we arrived at the parametrization with minimum confidence value of 0.8 together with a minimum support level of 0.001 where we found 12 association rules.

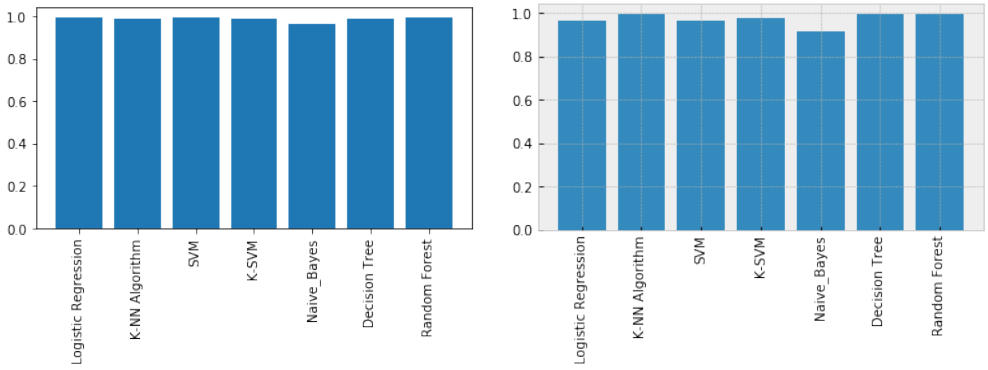

Figure 5. Avaliation of the methods of pre-processing (SMOTE))

3.2 Problem 2 - Colposcopies Dataset

3.2.1 Methods and Parametrization

Knowing that our Dataset had only 98 instances (not many instances), and also to prevent *overfitting*, we chose the strategy of **training and cross-validation test** with 10 folds(normally) for the following methods and parametrizations:

- **Association Rules:** Baseline pre-processing + Pre-processing 7

  To find patterns in Association rules we applied here the Apriori algorithm method over the **discretized** APS Dataset to try and find association rules.

- **Clustering:** Baseline pre-processing + Pre-processing 4

  The method of our choice here was K-means clustering, a popular cluster analysis method of vector quantification.

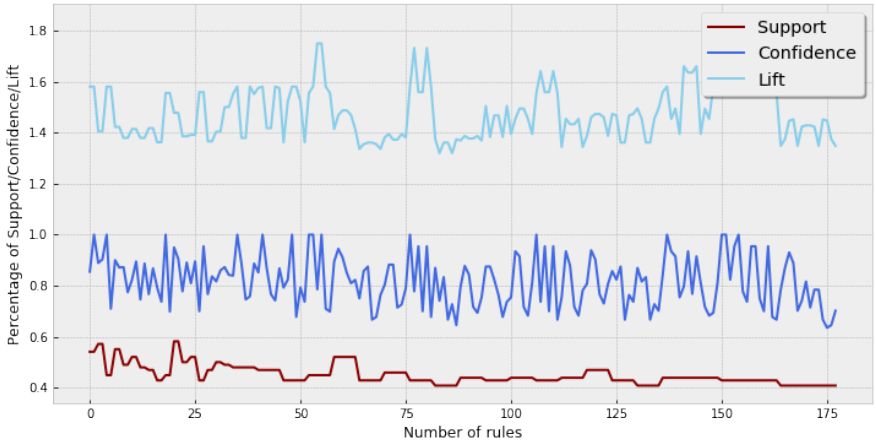

Figure 6. Number of rules with different values of Support/Confidence/Lift

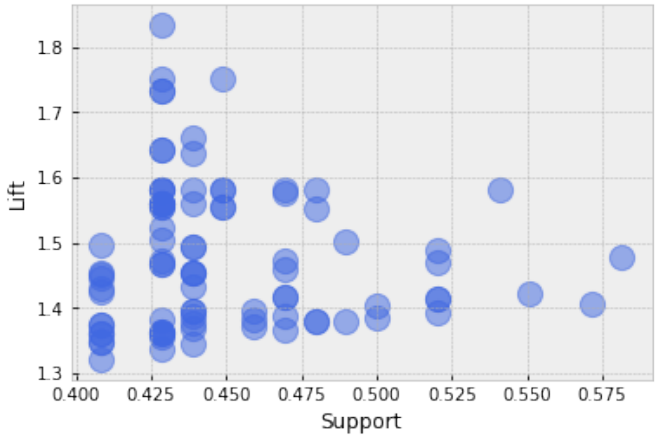

Figure 7. How value of lift behaves when changing Support level

- **Classification:**

  KNN: Baseline pre-processing + Pre-processing 5

  Naive Bayes: Baseline pre-processing

  Decision Trees: Pre-processing 5 + Pre-processing 6

  Random Forest: Pre-processing 5 + Pre-processing 6

3.2.2 Results

So first regarding pre-processing:

- For the correlation value, we ended up considering the value of 0.95, when using higher values we would be losing too many pairs and using lower values we would be finding too many pairs. (removing one of the values in each pair)
- In the chi-squared Statistical test the resulted p-value was close to zero, and therefore with a p-value $<$ 0.05 we can conclude that the variables are independent of one another, rejecting then the null hypothesis and conclude that the variables are correlated.
- Our application of the PCA method, as seen from the 8 below, led us to conclude the value of 25 for the number of components and the value of 4 for the numbers of clusters.

- The initial results from K-means clustering that we ran was not as expected, because the distance between the resulting clusters was too big, but we have then applied the PCA method and did again after the K-means clustering analysis the second time to achieve a better result, and the comparison of before and after can be seen below in the **Figure 8**.

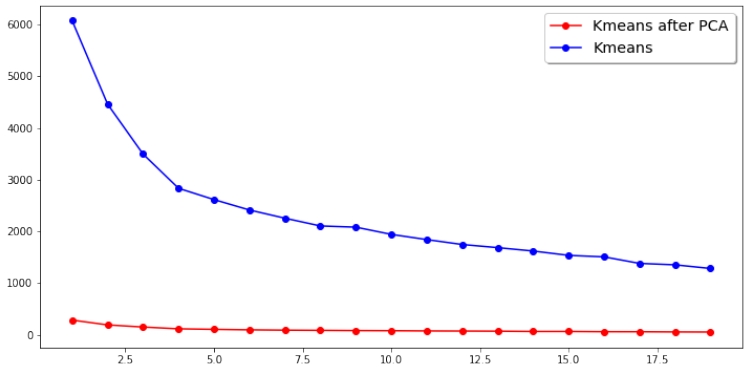

Figure 8. K-mean analysis before and after PCA

- To find the maximum number of features, we used the recursive feature elimination method and computed a cross-validated score. The accuracy scoring is proportional to the number of correct classifications. Therefore as seen from the graph below in **Figure 9**, we can conclude that the optimal number of features to use in the dataset is 3.

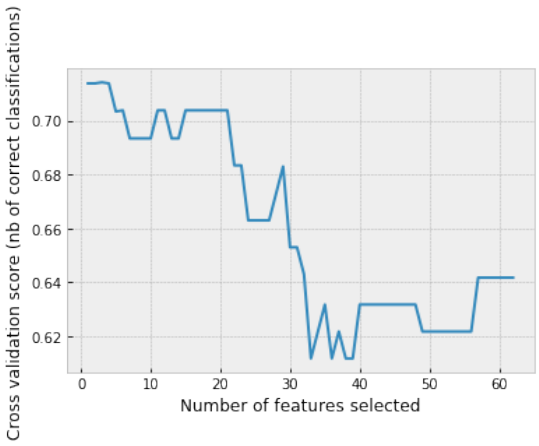

Figure 9. Cross-validation score depending on number of features selected

- After applying the two classification methods Naive Bayes and K-NN on same data sets to find the optimal result, we saw that the K-NN classification method gives more accuracy when compared to the Naive Bayes classification method.

Now regarding the results from the Association Rules, Clustering and Classification:

- In the Apriori algorithm method we had to find the most appropriate parametrization for our situation dataset, we did many iterations and ended up choosing a minimum confidence of 80% and minimum support of 6%. But before arriving at this parametrization we did test several variations with different confidence levels and different support levels, and we found out that it was when increasing the support level further from 25% to 40% that the resulting association rules found went from 4768 to 178, and for the confidence level the variations were all over 50%.

- When comparing the number of components with the cumulative explained variance, we observed that the accumulated variance up to 95% was present with a number of components up to the first 20, and with this we could conclude that for the PCA method to apply here in the Problem 2 Dataset, the best number of components to use was therefore 20 as seen in **Figure 10**.

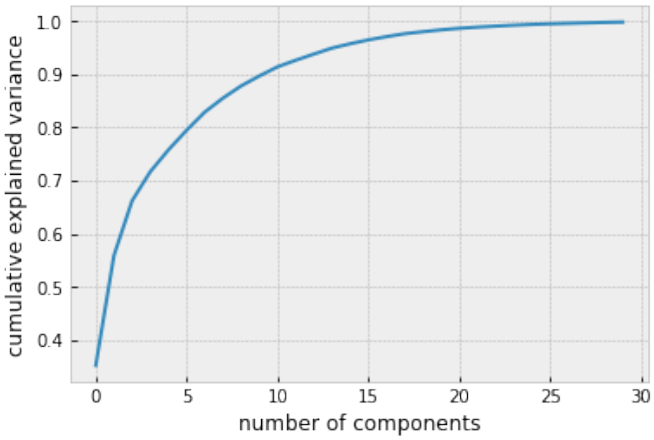

Figure 10. dsaNcomponents

- To better evaluate the classifications of Random Forest and Decision Tree for our datasets, we applied 3 different aproaches of pre-processing, respectively Normalization L1, Normalization L2, and Feature Selection. And to also add to the comparison we did also the classification for accuracy without pre-processing. So to analyze now the results which we can see from the two respective graphs below in **Figure 11**, we noted that in the Random Forest classification(left graph) the Pre-processing with normalization L2 is the one with the best accuracy, this can be justified with the non-sparseness of normalizations and not having many outliers in the dataset. While in the case of Decision Tree classification, the pre-processing with best accuracy result was Feature Selection because this classification allow us to select the most prominent features in each dataset, to later classify the test of the dataset.

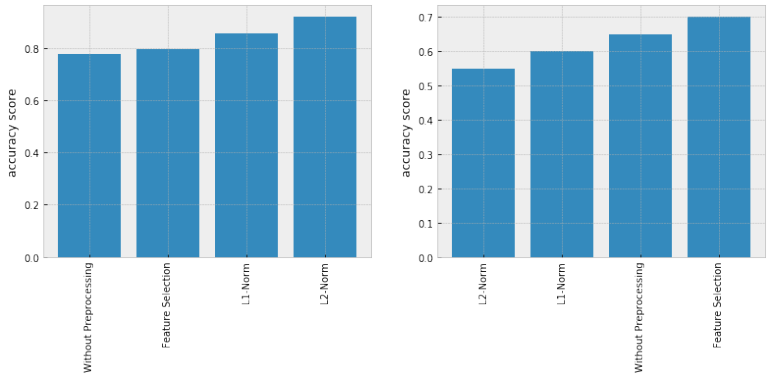

Figure 11. Avaliation of the methods of pre-processing using the classifications Random Forest(left) and Decision Tree(right)

- In **Figure 12**, we show 7 different ways to perform classification on the different datasets. In order to explain the accuracy score of each one of the classification methods, we will now go through the different methods in details. Using decision tree learning, we looked for the most relevant feature while in a node dividing situation, whereas in decision tree learning, we used the most relevant feature mechanism available through sklearn.

  Talking now about the second dataset, the data is much more noisy. The number of missing values was enourmous, and we tried several imputations methods, like average-imputing, median-imputing, intra class

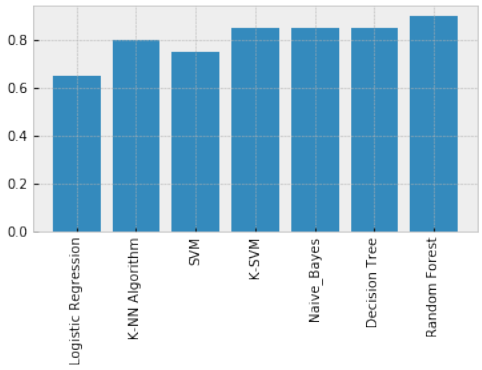

Figure 12. classificationMethods

average imputing and intra-class median imputation. In 12, we can see that instance learning is not the most robust.

4. CRITICAL ANALYSIS

In the Problem 1 Dataset, despite the disadvantage of replacing missing values with the mean reducing the variance, we did also consider replacement with median or mode, but the resulting dataset obtained had a very different distribution. We think because there were no big outliers and the result variance is acceptable, it made using replacement with the mean is the best option to keep the distribution mostly the same after treatment of missing values.

In the exploration part, regarding association rules, the results obtained were much better and significant after using discretization on the dataset by bins of equal-height, in comparison the results obtained with the base dataset were hard to conclude valuable information. We think using discretization in bins of equal-height in the dataset allowed for the attribute values to be much closer together and make it easier to find frequent itemsets when you have the values concentrated.

In order to obtain better result we used here an advanced method, Gaussian Mixture Model, which is a advanced clustering algorithm that is lot more flexible in terms of cluster co-variance, and k-means is actually a special case of GMM in which each one of the clusters' co-variance along all dimensions have values close to zero. The results of this Gaussian mixture model are seen below in the **Figure 13**, where we can see 4 clusters with much better definition utilizing only PCA and K-means.

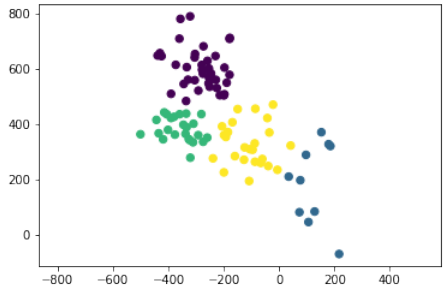

Figure 13. Gaussian Mixture Model clustering algorithm for the 2nd DataSet

Another learning method that we could have tried was neural network learning. However, neural network learning does not perform well in an imbalanced classification, therefore, we deciede not to continue this course of action. Another thing that we could have done, was doing our combination of learning algorithms. Using sklearn this is possible to do. Another aspect that we could have used was to use the F1-Score for measuring precision and recall for each class. Classification accuracy, in an imbalanced learning environment gives results that area normally misleading, given that the algorithms only learns how to classify properly the majority class.