Data Science Project 2018

TEAM #45

>> APS Scania and Coloposcopy datasets <<

Rafael Belchior, ist180970
Carolina Xavier, ist181172
Duarte Galvao, ist424753

Contents

1. INTRODUCTION

On the first dataset refers to the classification of the status of Scania trucks (whether they need to go to the workshop or not).There is a description which refers a cost function, $C(x) = FN * 500 + FP * 10$, being FN the number of false negatives and FP the number of false positives. We will be aiming to build models that aim to minimize the total cost, with focus on the **False Negative rate, as it is more expensive in terms of cost**.

The second dataset refers to quality assessment of digital colposcopies, made by experts. There are several experts that reach a consensus. The consensus derives from the expert variables, with more cost put on False Positives, meaning that if three experts judge the colposcopy as good and the other three as bad, the consensus is good. **An interesting challenge is to discover which expert has more weight on the final decision (consensus)**.

2. DATA DESCRIPTION

2.1 Problem 1

The training set contains 60000 examples in total, with a proportion of 1 postive instance to 59 negative instances. Therefore, this set is unbalanced. The class attribute has two possible values, making this a binary classification problem. There are 171 attributes, being one the class attribute.

In this dataset, there are missing values that require pre-processing. The observation with the most missing values is missing 33% of its values, and the average number of missing values per observation is 8.8%. The attribute with the most missing values is **br_000**, with 82% of missing values. The average missing value rate per attribute is 8.3%.

A quick statistical analysis will be performed on the attributes that matter the most for the classification. Those are the ones with greater information gain, to guide the study conducted. *SelectKBest* (uses $\chi 2$) was used to calculate them: **bb_000, bu_000, bv_000, dq_000 and eb_000**. From the analysis of this figures, we find no common distribution such as normal, binomial or uniform.

After replacing the missing values with their attribute's mean value we proceeded to do a statistical analysis on the attributes with greater information gain of this data set. The variance values of each of these attributes are from different magnitude orders.

2.2 Problem 2

The second dataset has three modalities (Hinselmann, Green, Schiller) and has seven target variables named expert::X (X in 0,...,5) and consensus. subjective judgments of digital colposcopies as 'bad' (0) or 'good' (1). There are three separated datasets with 97, 98 and 92 instances. There are 69 attributes, seven of which correspond to the target attributes. Given that the ratio between attributes and observations is 75%(very high), the danger of the high dimensionality curse is possible. Therefore, we opted for joining the datasets, in order to achieve a healthier proportion. The join is safe, as the attributes have the same datatype and follow the same distribution whithin the three datasets. The final dataset contains 287 instances in total in which 216 of these instances have consensus as 'good' and the other 71 as 'bad', which makes this an unbalanced data set. There are no missing values. The data on each attribute does not follow any specific distribution, such as normal, binomial or uniform.

This data set contains both positive and negative values so, to do a statistical analysis of the data, the values were normalized. After this, the attributes with greater information gain were computed (attributes that are more likely to be independent of the target attribute, and therefore more relevant for classification), according to *SelectKBest* that uses the $\chi 2$ stats. We selected the 5 attributes with the most information gain and got: **rgb_cervix_b_mean, rgb_cervix_b_mean_minus_std, rgb_cervix_b_mean_plus_std, rgb_total_b_mean_plus_std and hsv_total_v_mean** as the selected features. The variance values for each of these attributes is 4184, 2700, 6553, 4727 and 3386. One can see that the attributes have the same order of magnitude, therefore we predict that normalizing the data for classification will not have a big impact with classification algorithms.

3. PRE-PROCESSING

3.1 Problem 1

Baseline for classification: As the baseline for the classification we use the Naive Bayes algorithm with the default parameters, and no pre-preprocessing. We chose the default Naive Bayes algorithm because we found no specific distribution in the data that we could take advantage of (such as a Gaussian or Binomial distribution).

Baseline dataset: Data set and missing values filled with the mean by column by class. We chose to fill the missing values in the training data set with the mean value by class, as it does not change the distribution in a significant way. It diminishes the variance, as missing values are filled with means by class. For clustering, the missing values are filled with the mean by column (the class is ignored).

Pre-processing 1: Normalization. The variance of the attributes are from different magnitude orders. Given this, there is the possibility of one attribute to dominate over others. Normalizing the attributes makes training less sensitive to their scale, so we can better solve for coefficients.

Pre-processing 2: Removing observations with more than 40% of missing values and remove attributes with more than 50% of missing values. Removing the columns with this ration removes a considerable amount of missing values, there are only 9 attributes removed (accounting only for 5% of the total).

Firstly observations with more than 20% of missing values were going to be removed, but that would remove approximately 40% of the minority class instances. dropping the observations by 5,6%. After that, it was adjusted to remove observations only with more than 40% of missing values, getting us a reduction of 1% on the dataset.

Pre-processing 3: Under sampling the majority class (random under sampling) and oversampling the minority class, using SMOTE. This will make the data set less biased to the majority class. The majority class (59000 instances) will be undersampled to 30000 instances, and the minority class (1000 instances) will be oversampled to 30 000 instances. We are keeping the original dataset length. We could have just over sampled the minority class but that would give us two problems: firstly the dataset would be too large to run algorithms like KNN. Secondly, Using SMOTE to perform an increase of 5800% does not balance properly the data.

Pre-processing 4: Removing attributes with lower χ^2. Removing attributes that are the least useful for classification or clustering. In the latter, the class is not taken into account.

Pre-processings that were not applied: We did not apply removal of outliers, as they represented 75% of the dataset (as variances are very big). Instead, we choose to apply standartization techniques that allow the reduction of variance, such as normalization.

3.2 Problem 2

Pre-processing 1: Normalization.The dataset is not normalized. With regard to the classification task, because the attributes have the same magnitude, we believe that normalizing wouldn't add much to the task. Nonetheless, a test will be conducted. Applied to KNN classification algorithm and association rules.

Pre-processing 4: Removing attributes with lower χ^2.

Pre-processing 6: Discretization. We apply discretization of the continous values in five bins of the same width, as that division performs the best (compared to the cluster technique for binarizing data).

Pre-processings that were not applied: We did not apply removal of missing values, observations or attributes, because there were not missing values. Although the dataset is unbalanced, we did not apply oversampling to the minority class, because the dataset is not very unbalanced, and that would change the distributions.

4. EXPLORATION

4.1 Problem 1

4.1.1 Methods and Parametrization

The data is already divided in training and test datasets. The large size of the data set does not require the use of K-Fold Cross Validation when running the classification models.

To evaluate the performance of the classification algorithms we consider the accuracy of classification and the cost-metric of miss-classification $[FalsePosite * 10 + FalseNegative * 500]$, this is a specific metric to be used for this dataset, as specified in the dataset description.

We established the **Naive Bayes** Classifier as our baseline model that will serve as a term of comparison for the performance of the other classification model applied to the data set (fig.1).

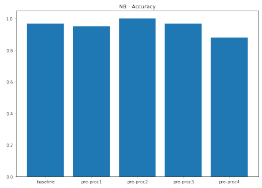

Figure 1. Naive Bayes accuracy

K-nearest neighbors algorithm (KNN): We applied the KNN algorithm to the baseline data set as well as to all the pre-processing data sets and measured the accuracies of the classifier (on the test data set) for different number of neighbors - $1 < K < 30$.

Decision Tree: We applied the Decision Tree classifier on both the baseline data set, pre-processing 2 and pre-processing 4 dataset. For pruning we started by varying the maximum depth of the tree to settle on the best value for this parameter.
After, we applied the Decision Tree classifier again on the same data sets with and without pruning and with varying number of samples required to split an internal node.

Random Forest: We applied the Random Forest classifier on both the baseline data set and the pre-processing 2 data set. As we did with the Decision Tree classifier, we started by determining the most appropriate pruning value and after we applied the Random Forest classifier again on the same data sets with and without pruning and with varying number of trees in the forest.

Clustering: We applied the K-means algorithm to the baseline and test data sets. First we compare the Inertia - measure of how internally coherent clusters are - for the k-means applied to all the before mentioned data sets. We also compared this clustering algorithm applied to the pre-processing 1 data set before and after applying Principal Component Analysis (PCA).
We applied principal component analysis before k-means to improve the clustering results (noise reduction).The intuition is that PCA represents data vectors in a linear combinations of a smaller number of eigenvectors, and does it to minimize the mean-squared error. It helps because PCA aims at compressing the features while clustering aims at compressing the data-points.

Association Rules: We decided to focus our association rules analysis on the second dataset, because it allow us to conduct more interesting experiments (as we have seven class attributes). Nonetheless, for applying this technique in this dataset in specific, there are some aspects to take into account: first, one should not remove missing values. One should look to rules in "the positive way", in the sense that we are not looking for rules such as A = Missing Value → B = Missing Value. We are looking for rules such as: A = "1" → B = "1", for a specific confidence and support. Furthermore, attributes with a large number of missing values (i.e. 80%) should be removed, because if we are looking for rules with a small support(i.e. 30%), the only rule that will appear is the one containing the missing value.

4.1.2 Results

K-nearest neighbors algorithm (KNN): The pre-processing 3 dataset is the one that obtained the lowest accuracies, however the accuracy results for all the datasets are all similar and with high values, all above 95%. In the costs however the pre-processing 3 dataset stands out for having lower costs than the other datasets (fig.2).

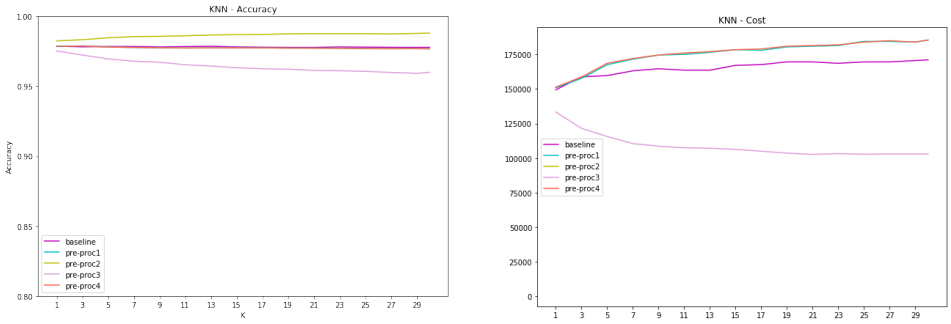

Figure 2. KNN algorithm - Accuracy and Costs (dataset 1)

Decision Tree: From the resulting graph (fig.3 (left)) we see that, although the difference is not considerable, the classifier performs better with smaller values of maximum depth, so for pruning we chose value 2 for maximum depth. When varying the number of splits the accuracies do not vary considerably between datasets (fig.3 (right)).

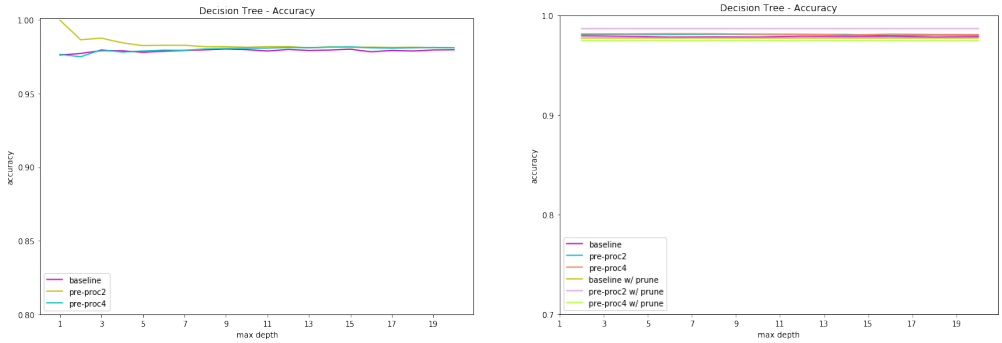

Figure 3. Decision Tree - Accuracy (dataset 1)

Random Forest: All the datasets, pruned or not, perform similarly and with considerably high values of accuracy (fig.4).

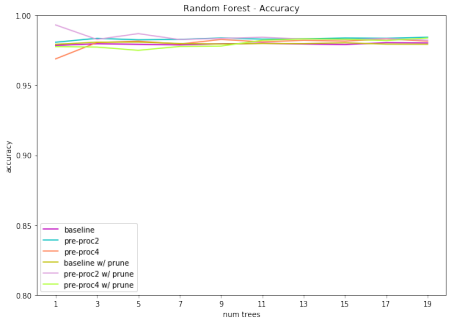

Figure 4. Random forest - Accurace (dataset 1)

Clustering: Clustering of the baseline and test datasets (fig.5 (left)).
Clusterin of the pre-processing 1 dataset (nomalized) before and after PCA (fig.5 (right)).

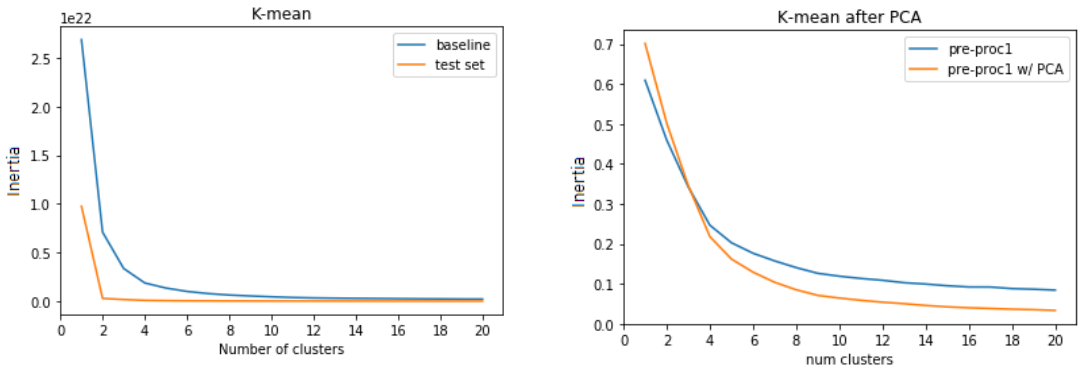

Figure 5. SSE for varying number of clusters

4.2 Problem 2

4.2.1 Methods and Parametrization

The Pre-processing 2 method was never applied to this dataset, because it is dependent on the existence of missing values, which we know do not exist in this dataset.

We established the **Naive Bayes** Classifier as our baseline model that will serve as a term of comparison for the performance of the other classification model applied to the data set, the accuracy obtained with this classifier is represented in fig.6.

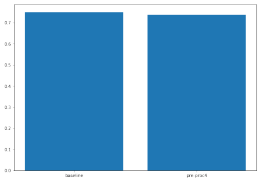

Figure 6. Naive Bayes Classifier - Accuracy (problem 2)

K-nearest neighbors algorithm (KNN): We applied the KNN algorithm to the baseline data set as well as to all the pre-processing data sets and measured the accuracies of the classifier (on the test data set) for different number of neighbors: $1 < K < 30$ (fig.7).

Decision Tree: We applied the Decision Tree classifier on the baseline and pre-processing 4 dataset. To determine the most adequate pruning parameters we started by varying the maximum depth of the tree to decide on the best value for this parameter (fig.8(left)). After, we applied the Decision Tree classifier again on the same data sets with and without pruning (maximum depth previously determined) for a varying number of samples required to split an internal node (fig.8(right)).

Random Forest: As with the Decision Tree, we applied the Random Forest classifier on the baseline and pre-processing 4 datasets (fig.9). As we did with the Decision Tree classifier, we started by determining the most appropriate pruning value and after that we applied the Random Forest classifier again on the same datasets with and without pruning and with varying number of trees in the forest.

Clustering: We applied the K-means algorithm to the baseline and pre-processing 1 dataset. We compare the Inertia - measure of how internally coherent clusters are, the Mean Squared Error (MSE) -sum of the Euclidean distances between each pattern and its cluster center, and the Silhouette score for the k-means applied to all the before mentioned data sets.

Association Rules: To analyze the association rules we used the Apriori algorithm. We also applied pre-processing 6 and 1 to this dataset. Fixed parameters: confidence (95%), minimun lenght of rules (1), maximum length of rules (3). The algorithms were applied to the baseline dataset and the normalized dataset.

An interesting experiment with Association Rules: The nature of the dataset 2 has a problem: the consensus is not necessarily truer than any of the experts since experts may have different levels of experience. We would like to verify which expert influences matters the most in regard to the consensus decision. Intuitively, there are experts that know more than others and end up influencing the consensus more than others. We would like to discover what is the expert with most weight on the consensus and also the pair of experts with the most weight on the consensus. For the first experiment, the apriori algorithm was applied with minimum confidence = 0.9, minimum support = 0.5 and length of the rules between 1 and 3.

4.2.2 Results

K-nearest neighbors algorithm (KNN): We verified that the accuracy did not considerably vary between the baseline and pre-processing datasets and, increases as the K value increases, up until K=3, after that it stabilizes (fig.7).

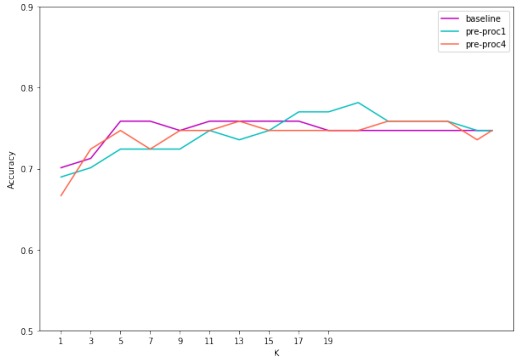

Figure 7. KNN Classifier (problem 2)

Decision Tree: (fig 8(left)). After analyzing the graph with varying maximum depth values for the tree, we chose 6 as the ideal maximum depth because it is the value with which we achieved higher accuracy of classification. When applying the Decision Tree classifier again on the same data sets with and without pruning for a varying number of samples required to split an internal node (fig 8(right)), there is not a considerable difference between the performance of this classifier with the baseline dataset or the pre-processed 4, which consists on the application of feature selection based on the feature relevance to classification.

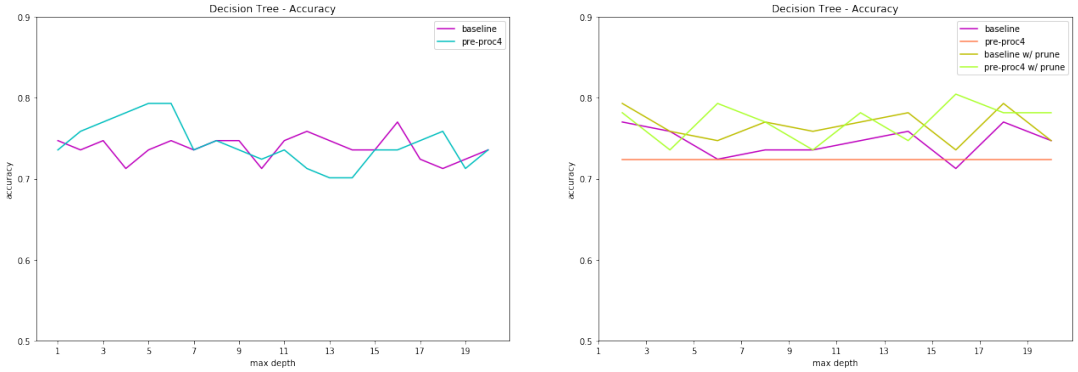

Figure 8. Decision Tree Classifier (problem 2)

Random Forest: Accuracy of the Random Forest classifier for a varying number of trees (fig.9).

Clustering: Results of inertia, mean-suqared erros (MSE) and Silhouette score for clustering of the baseline and pre-processing 1 datasets are in fig.10.

Association Rules: The figure 11 depicts the number the rules obtained in function of the minimun support, for the baseline dataset and normalized dataset.

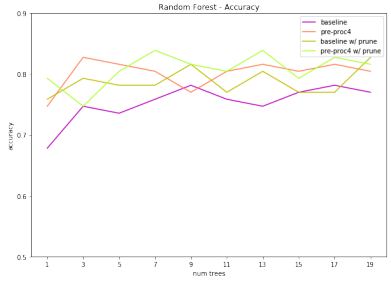

Figure 9. Random Forest Classifier (problem 2)

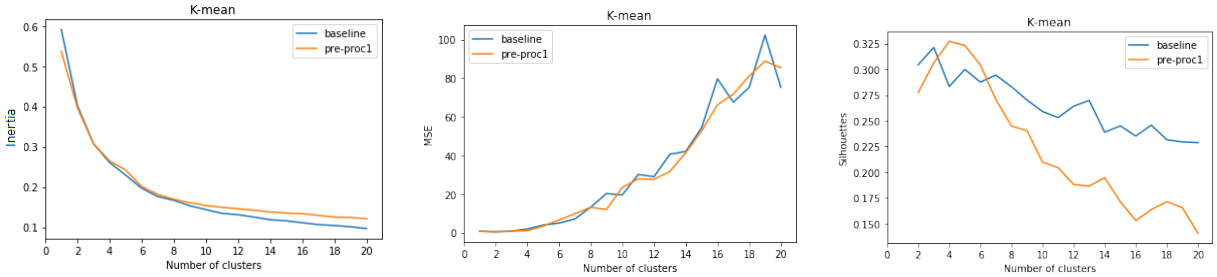

Figure 10. SSE, MSE and Silhouette for varying number of clusters

The figure 12 depicts the mean lift from the top 10% rules with higher lift, with respect to variable support.

An interesting experiment with Association Rules: We obtained 136 rules which contained the class refering to Expert 0. Adjusting the parameters (confidence = 0.95, support = 0.1), we obtained 1336 rules. These rules are more interesting, as we can be more sure of them, although they may be more rare. To make this analysis, we chose as the antecendent any expert, and as the consequent the consensus. We were interested in the rule: when expert n votes 1, consensus is 1). The figure 13 depics the top confidence and lift for each expert for the rule defined above.

The rule that includes expert 1 has the highest confidence (96%) and lift (1.28). Note that the lift is not that high, meaning that we can not blindly rely that expert 1 will always be right on his own.

With regard to obtaining the best pair of experts, we obtained 21 rules 14

The pairs experts 1 and 2, 1 and 5, 2 and 5 are equally good when it comes to influence the consensus (all have confidence = 1.33, confidence = 1). The pair with the highest support is expert 1 and 2.

5. CRITICAL ANALYSIS

Concerning unsupervised learning, we removed the class attribute

5.1 Problem 1

When applying the KNN classifier, because the pre-procesing 3 dataset has lower accuracy it may seem that for KNN this dataset represents the worst classification, but in reality the cost-metric of miss-classification is lower for this dataset which means that the performance of the classifier in this dataset is better than for the rest of the datasets.

For the Decision Tree classifier, when determining the best values for maximum depth of tree, lower values seemed to perform better, this happens because decision trees tend to do over fitting, and by reducing this value we reduce the probability of that happening

The Random Forest is a robust classifier that determines feature importance, and trains on random subsets of features, therefore the accuracy values do not vary considerably for the different datasets - baseline dataset or two datasets with different methods of feature selection - because after applying the classier feature selection is performed in all the datasets.

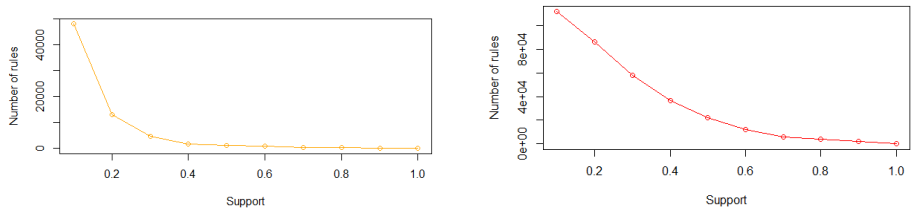

Figure 11. Variation of the number of rules discovered by support, in the baseline and normalized dataset

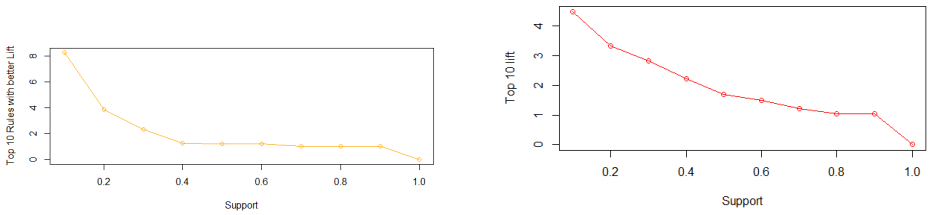

Figure 12. Variation of the quality of the mean lift of the top 10% rules by support, in the baseline and normalized dataset

In General the all the classifiers performed better than the baseline classifier chose - Naive Bayes. Pre-processing is key for this happening.

For clustering, when analyzing the graph in fig.5 (left), and using the elbow method (locating a bend in the plot), we determine that the optimal number of clusters for this data set is 2, this coincides with our knowledge of the data set that has two possible class values ('pos' and 'neg').
By applying PCA (fig.5 (right)) we obtain smaller errors for a higher number of clusters than without it, which means that this algorithm may be used to divide the data better then the attribute 'class' of the dataset.

With regard to **An interesting experiment with Association Rules**, the chosen experts were 1 and 2. Something even more interesting to do than discovering them would be creating a new variable from the two experts which are the most influent with regard to the consensus.

We would perform a consensus between them in a way that conflits are solved randomly, as depicted in fig.15.

This new consensus has, much likely, less noise and we are expecting it to perform better than the original consensus. The next steps would be benchmarking the performance of new consensus performance in classification against the attribute consensus.

5.2 Problem 2

From the application of the KNN algorithm (fig.7) we conclude that 3 is the ideal number of nearest neighbors to consider when applying the KNN algorithm to this dataset, an adequate number for this problem because the classification is binary so an odd number avoids ties.

The accuracy similarity Decision Tree classifier with the baseline dataset or the pre-processed 4 is expected because the decision tree model already determines feature importance (to keep the best performing features as close to the root of the tree). When pruning is applied, a small increase in accuracy of the model is noted because pruning is a method used to reduce the complexity of the classifier, and therefore increases accuracy.
Like the Decision Tree, the Random Forest classifier also determines feature importance, and trains on random subsets of features, therefore the accuracy similarity between datasets is also present. The overall average accuracy was higher for this classifier than for the Decision Tree classifier, because the Random Forest is a collection of decision trees whose results are aggregated into one final result, therefore it is a more robust classifer.

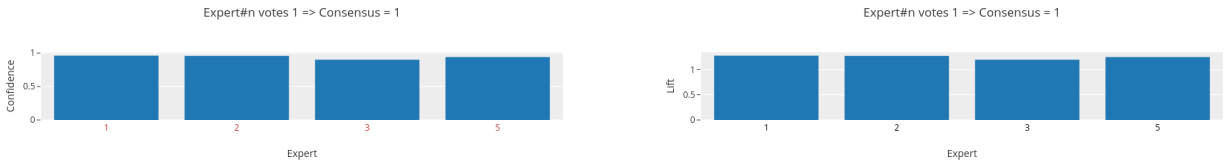

Figure 13. Variation of the quality of the mean lift of the top 10% rules by support, in the baseline and normalized dataset

Figure 14. Experiment 2 results (adapted)

Figure 15. Creation of a new attribute: New_consensus

In General the all the classifiers performed better than the baseline classifier chose - Naive Bayes. Preprocessing is key for this happening.

For clustering, when analyzing the graph referent to the inertia values in fig.10 (left), and using the elbow method (locating a bend in the plot), we determine that the optimal number of clusters for this data set is 2, this coincides with our knowledge of the data set that has two possible class values (0 and 1, or 'bad' and 'good'). MSE and Silhouette score represented in fig.10 (center) and fig.10 (right) respectively, quantify the quality of clustering achieved with k-means. The ideal number of clusters is the number that minimizes MSE and maximizes silhouette score. For the MSE values that number is 2, and the maximum silhouette score is for 4 clusters.

Relatively to the association rules, we fix the confidence at least 90%, as it is high enough (and we consider them interesting). If we compare the graphs in fig. 11, we notice something interesting: for fig. 11 (left) it was obtain 48117 rules with support %10, agianst the 94437 observations coming from the normalized dataset, also with at least 10% support. The general tendency is verifiable for all values; the more common the support is, the less rules are obtained,

Concerning the quality of the rules, represented by the lift, we can observe that in fig.13 (left) the mean lift from the ten rules with highest lift is 8.25, versus the 4.46 shown in fig.13 (right). This tendency gets inverted, though: from support = 0.3, we get superior mean values for the 10 best rules for the normalized dataset rather than the original. It seems that normalizing tends to help apriori to discover more rules.

6. CONCLUSIONS

We experienced that the format of the data is decisive to the performance of the different classifiers. Data with no missing values does not necessarily behave better. The results are highly sensitive to each small local decision that is made. Aditionally, the choice of the right metric is essential to a correct evaluation of a model or, at least, less biased evaluation.