MACHINE LEARNING ASSIGNMENT 1

Classification problem for school student’s academic performance

Wladyslaw Eysymontt

INDEX

1. Introduction

In this work we are going to present the results of the machine learning analysis of a dataset based on the information about the students of two Portuguese schools, Gabriel Pereira and Mousinho da Silveira.

The goal of this work is to create a machine learning technique able to predict student’s academic performance as binary classification problem with “Passed the course” and “Had not passed the course” as classes.

2. Problem description

The dataset used during the work is called Student Performance Data Set and it is publicly available in the UCI Machine Learning Repository (see the link in references). This dataset is formed of two separate csv files, one with data of Portuguese language subject and another one with mathematics one. Only the second one was used.

This file contains 31 variables, such as student’s sex, previous academic results, alcohol consumption, family status and so on (see the full description of the variables in references) and 395 observations, each representing one student.

3. Methodology

3.1 DATA TRANSFORMATIONS

To perform the analysis R programming language and RStudio software were used. In order to be able to use all the machine learning models, the variables were all transformed to numerical scale. After this transformation three data frames were created:

- One with all numerical variables, to be used, for example, for correlation matrix.
- One with some numerical continuous and other numerical treated as factors variables, to be used in all the algorithms which allow both types of variables.
- And another one with all variables treated as factors to be used in algorithms which require this type of data, for example rule induction or classification trees.

And each of the three was divided in training 300 and testing 95 observations subsets.

The output variable “AcadPerformance” were created as the mean value of the three columns representing student’s academic performance during the year (G1,G2,G3).

3.2 FEATURE SUBSET SELECTIONS

Once the data frames were prepared, we performed two feature subset selection methods in order to identify best predictor variables. First method consisted in a correlation matrix (see Figure 1 below). Highly correlated variables were detected and eliminated:

- Walc (weekend alcohol consumption) was highly correlated to Dalc (labour days alcohol consumption) and goout (going out with friends).
- Fedu (father’s education) was highly correlated to Medu (mother’s education)
- Address was highly correlated to traveltime (travel time).
- traveltime was highly correlated to Address and none of them were influential on them were influential on the output variable.
- school was highly correlated to Address and to traveltime.

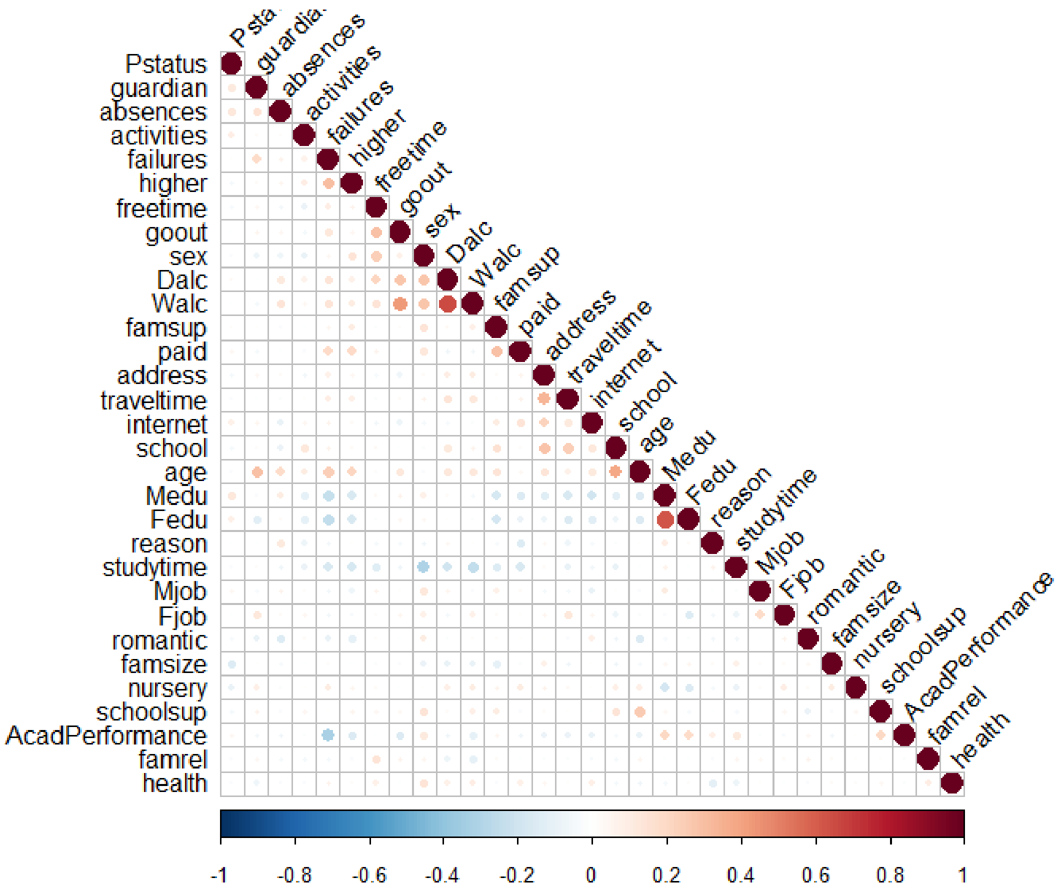


Figure 1. Correlation matrix

So, six variables were dropped, but the overall correlation between features is low and another method should be applied to get further features reduction.

As the second method, binomial generalized linear model was used, and the results showed P-values < 0.05 for the following features, then used in multivariate feature subset:

- sex
- failures
- schoolsup
- famsup
- internet
- health

Being failures best predictor variable used in univariate feature subset.

3.3 APPLICATION OF THE MODELS

The work was divided in three identical R scripts, one with the all original variables used, one with the multivariate feature subset and another one with the univariate feature subset. Then all the models in each of the R scripts were trained on the corresponding training subsets and tested on the testing subset. The corresponding accuracy was calculated as a classification quality measure.

4. Results

4.1 SUPERVISED CLASSIFICATION

In this chapter we are going to review the results of the application of machine learning supervised classification models, both probabilistic and non-probabilistic each applied to three data subsets:

- With all original variables used as a predictor variables (AOV).
- Selected multivariate feature subset (MFS).
- Selected univariate feature subset (UFS).

Note that the apriori probability of correct classification based on the frequency of the most common value is 54,68%.

4.1.1 K NEAREST NEIGHBOURS

For each data subset the hyper parameter K was adjusted as the one with the best results among 1 to 20 nearest neighbours tested each 20 times with 5-fold cross-validation. The best result for AOV were 0.6736842 with one nearest neighbour:

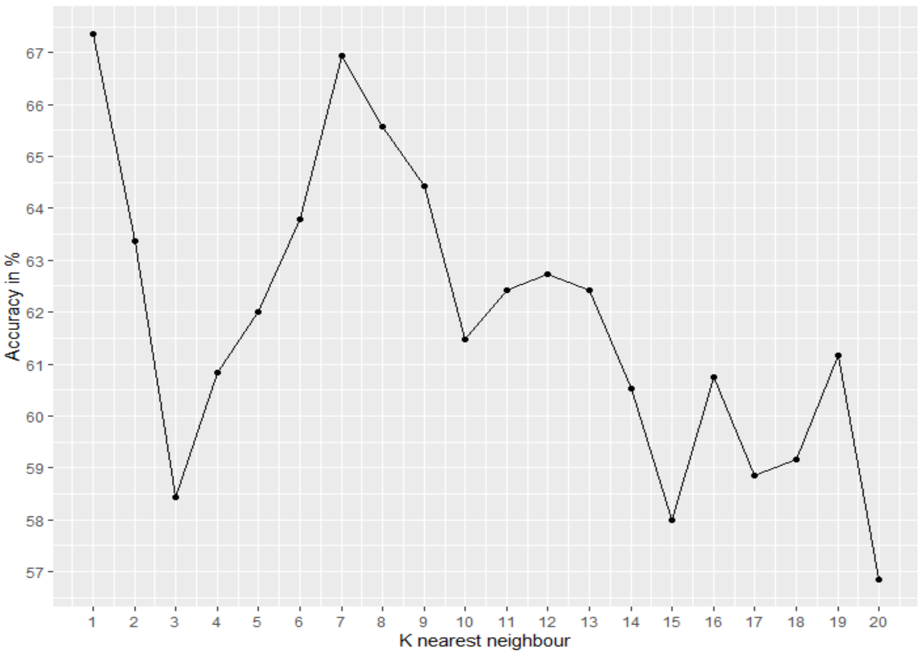
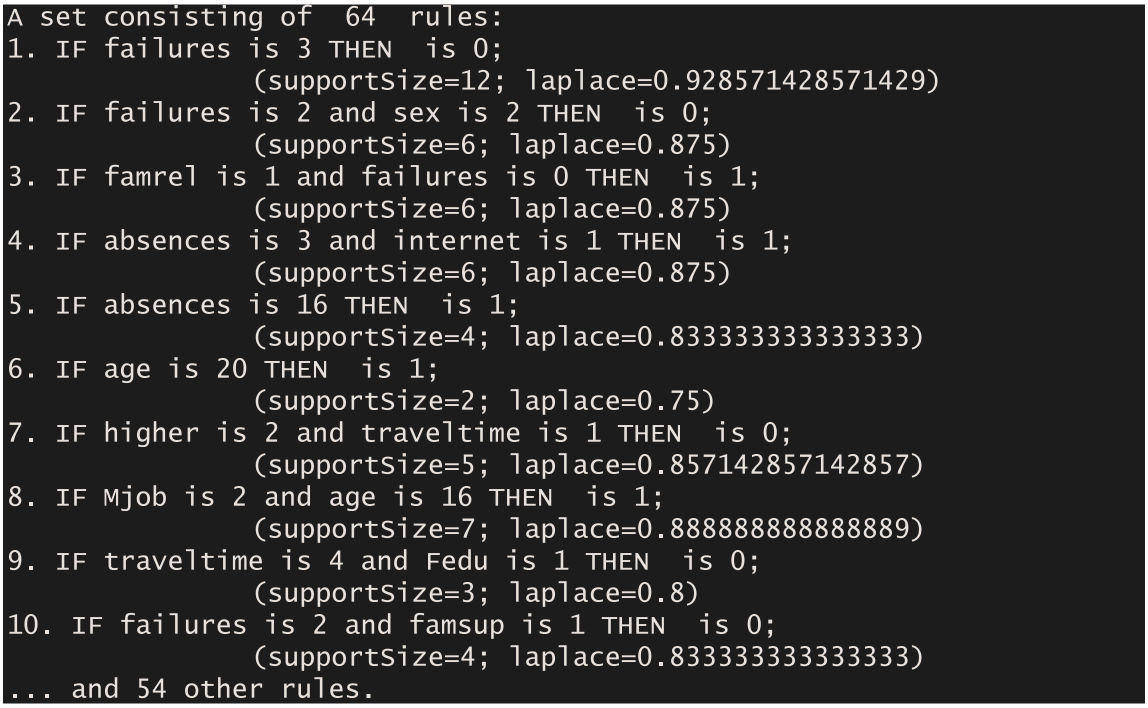


Figure 2. AOV KNN

The output for MFS and UFS was 1 for both, so we assume that we cannot trust these results

4.1.2 RULE INDUCTION

The rule induction algorithm's hyperparameter complexity was manually adjusted, being 5 the one which was usually giving the most accurate predictions. In the following you can see the example of decision rules for AOV with the accuracy of 0.6526316:



```
A set consisting of  64  rules:
1. IF failures is 3 THEN  is 0;
               (supportSize=12; laplace=0.928571428571429)
2. IF failures is 2 and sex is 2 THEN  is 0;
               (supportSize=6; laplace=0.875)
3. IF famrel is 1 and failures is 0 THEN  is 1;
               (supportSize=6; laplace=0.875)
4. IF absences is 3 and internet is 1 THEN  is 1;
               (supportSize=6; laplace=0.875)
5. IF absences is 16 THEN  is 1;
               (supportSize=4; laplace=0.833333333333333)
6. IF age is 20 THEN  is 1;
               (supportSize=2; laplace=0.75)
7. IF higher is 2 and traveltime is 1 THEN  is 0;
               (supportSize=5; laplace=0.857142857142857)
8. IF Mjob is 2 and age is 16 THEN  is 1;
               (supportSize=7; laplace=0.888888888888889)
9. IF traveltime is 4 and Fedu is 1 THEN  is 0;
               (supportSize=3; laplace=0.8)
10. IF failures is 2 and famsup is 1 THEN  is 0;
               (supportSize=4; laplace=0.833333333333333)
... and 54 other rules.
```

The corresponding accuracies for MFS was 0.6421053.

As for UFS, it was not possible to perform this algorithm due to the fact of not having enough explanatory variables.

4.1.3 CLASSIFICATION TREE

First, we tried to manually adjust the hyperparameters such as minimal number of observations needed to split the node, maximum number of levels and so one, but no result better then automatic adjustment was achieved. Below in the figures 3, 4 and 5, you can observe decision trees for AOV, MFS and UFS. Here are the corresponding accuracies:

- AOV = 0.6631579 (Figure 3)
- MFS = 0.6842105 (Figure 4)
- UFS = 0.7263158 (Figure 5)

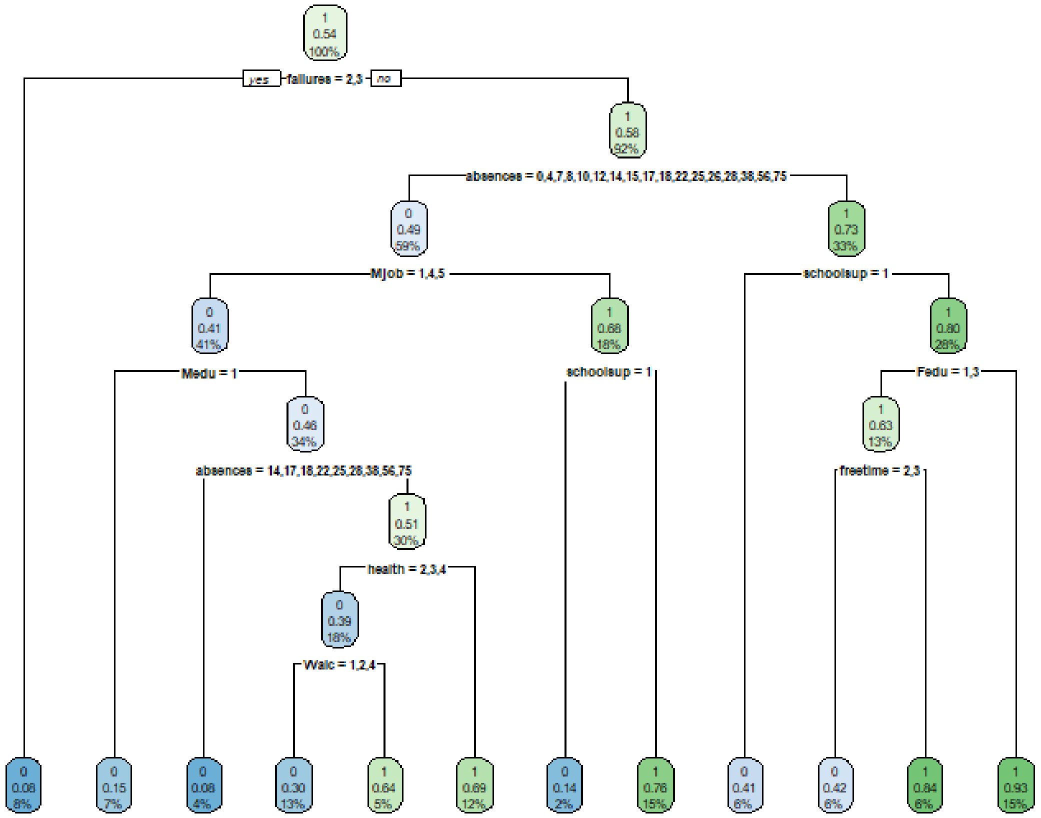


Figure 3. AOV Classification Tree

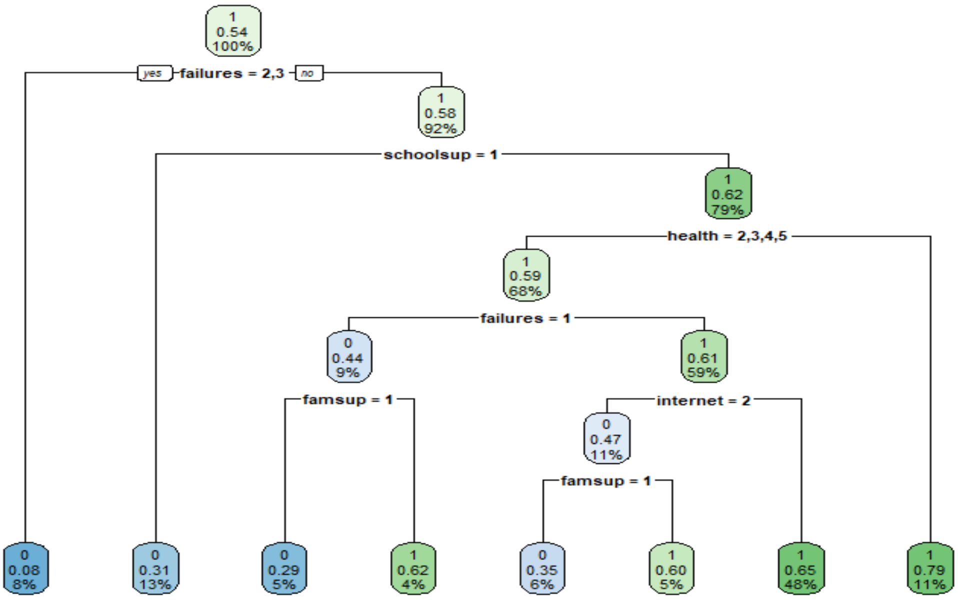


Figure 4. MFS Classification Tree

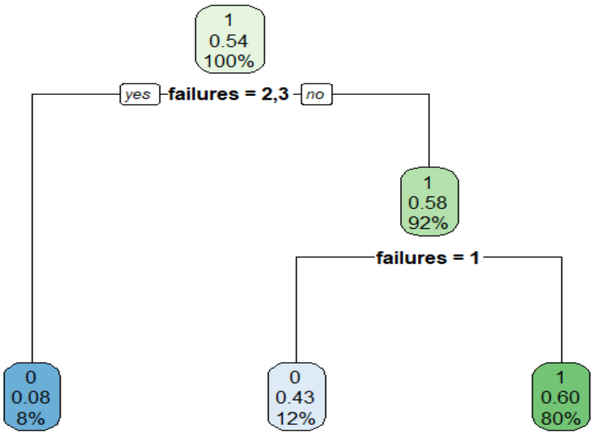


Figure 5. UFS Classification Tree

4.1.4 SUPPORT VECTOR MACHINES

As hyperparameters for this method were used the linear kernel and 0.15 misclassification cost, which showed the best results in most cases and was manually adjusted. In general terms this was an algorithm with the one of best overall accuracies, being this:

- AOV = 0.7052632
- MFS = 0.7157895
- UFS = 0.7263158

4.1.5 ARTIFICIAL NEURAL NETWORK

The Artificial Neural Network were trained for each feature subset with 3 hidden layers and logistic activation functions on them and on the output layer. The resulted accuracies were:

- AOV = 0.6421053 (Figure 6)
- MFS = 0.7263158 (Figure 7)
- UFS = 0.7263158 (Figure 8)

The weights and the biases were adjusted by the back-propagation algorithm. Please find the plots of the Artificial Neural Networks corresponding to each feature subset below.

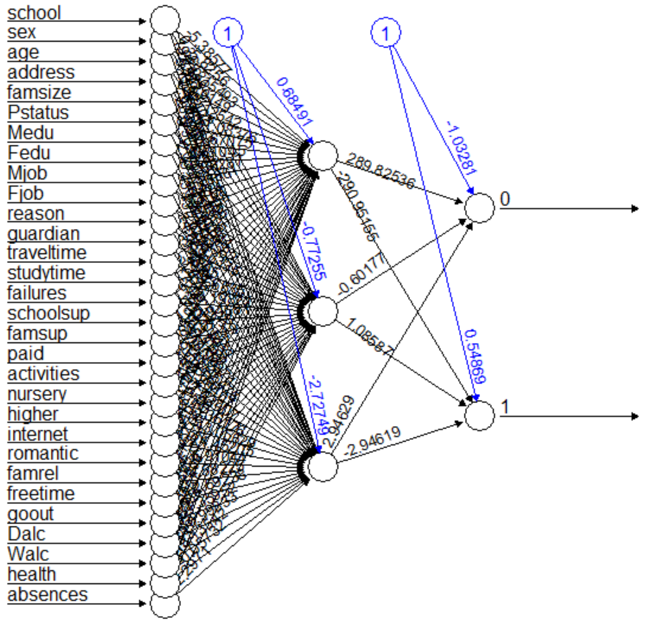


Figure 6. AOV Artificial Neural Network

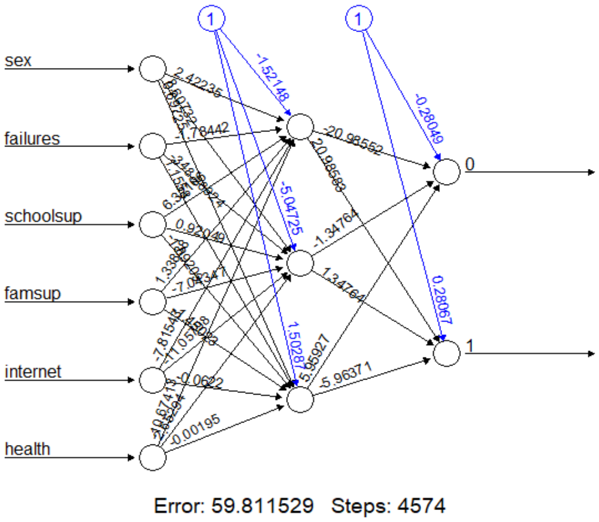


Figure 7. AOV Artificial Neural Network

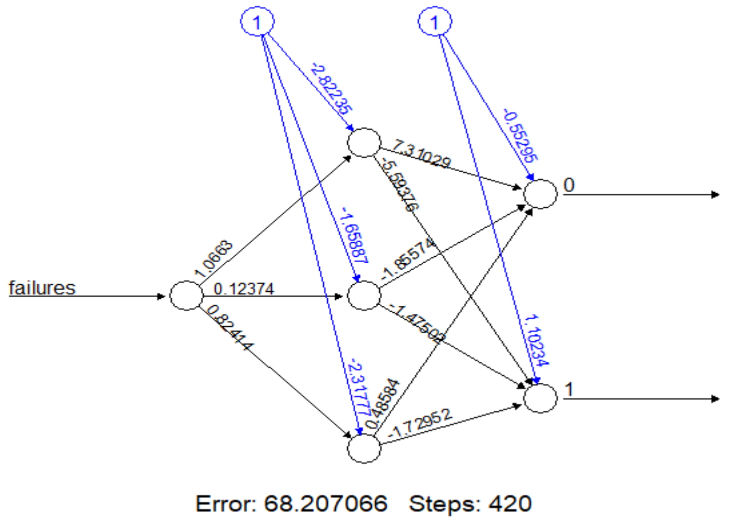


Figure 8. UFS Artificial Neural Network

4.1.6 LOGISTIC REGRESSION

The overall performance of the logistic regression was not impressive. In the following you can observe the AIC and the accuracy for each subset:

- AOV AIC = 403.55; AOV accuracy = 0.6315789
- MFS AIC = 380.95; MFS accuracy = 0.7052632
- UFS AIC = 390; UFS accuracy = 0.7263158

We can see that even though MFS has more variables and less accuracy than UFS, its AIC is lower, suggesting it being better model

4.1.7 BAYESIAN CLASSIFIERS

Naïve Bayes model was used and this algorithm, together with support vector machines one, gave one of the most accurate results, being these:

- AOV = 0.7368421
- MFS = 0.7157895
- UFS = 0.7263158

4.1.8 DISCRIMINANT ANALYSIS

Four types of discriminant analysis algorithms were applied (Linear, Mixture, Flexible and Regularized), the best results for AOV were achieved with Regularized one, for MFS, with the mixture model and the three equal best results for UFS were achieved with linear, flexible and regularized discriminant analysis models.

- AOV = 0.6842105
- MFS = 0.7263158
- UFS = 0.7263158

4.1. METACLASSIFIERS AND GLOBAL RESULTS

As metaclassifiers simple and weighted majority vote algorithms were used. Due to non-significance or non-possibility of implementation of Rule Induction model for UFS and of KNN model for MFS and UFS, those models had not been considered in metaclassifier selections for the corresponding feature subsets.

The results of the simple majority vote metaclassifiers:

- AOV = 0.5789474
- MFS = 0.5789474
- UFS = 0.5789474

The results of the weighted majority vote metaclassifier:

- AOV = 0.7263158
- MFS = 0.7052632
- UFS = 0.7263158

4.2 UNSUPERVISED LEARNING

In this chapter we are going to review the results of the application of machine learning unsupervised classification models. The main problem faced applying these algorithms was the difficulty to calculate the accuracy, so for each model the special function was created for this propose. We will go through these functions in the corresponding subchapters.

4.2.1 HIERARCHICAL CLUSTERING

After applying the method, the cluster dendrogram was drowned to select the optimal number of clusters for every feature subset. The function invented to calculate the accuracy was assigning the most frequent class from each cluster to correct results, and another class to incorrect results. After that, the accuracy was calculated by a standard formula, as now we had got correct and incorrect classifications clearly separated.

- The AOV accuracy for 45 clusters was 0.6202532 (Figure 9)
- The MFS accuracy for 5 clusters was 0.6202532 (Figure 10)
- The UFS accuracy for 2 clusters was 0.6202532 (Figure 11)

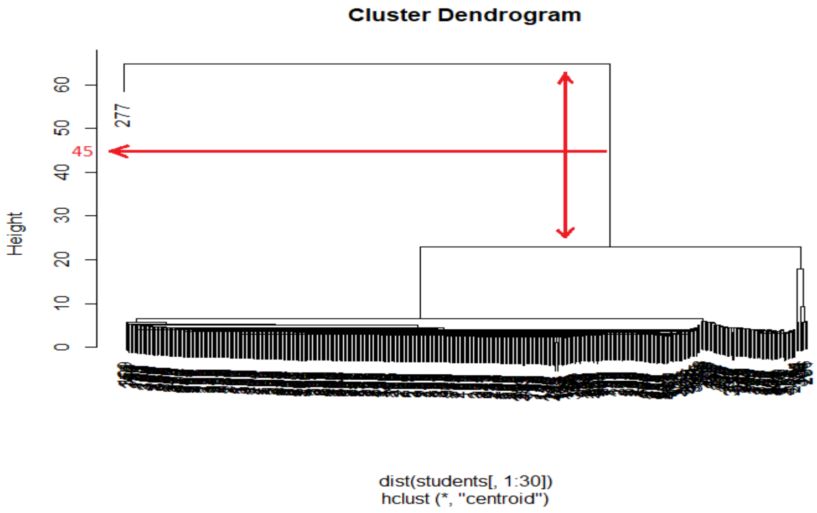


Figure 9. AOV Cluster Dendrogram

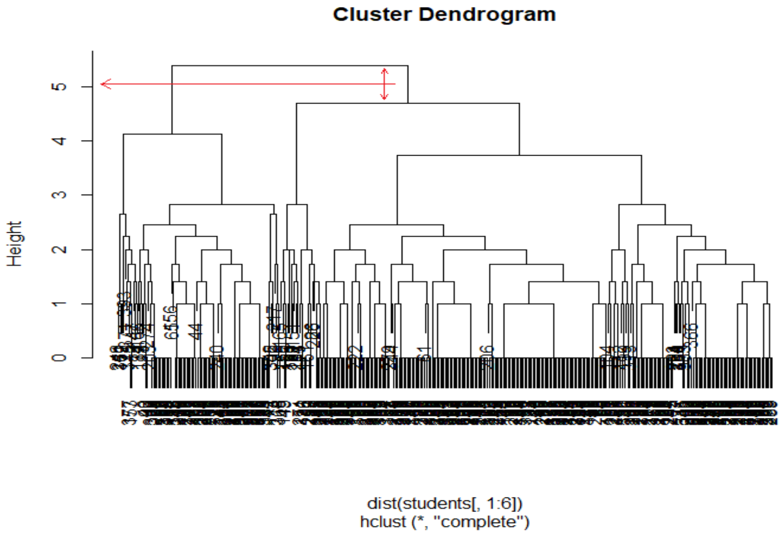


Figure 10. MFS Cluster Dendrogram

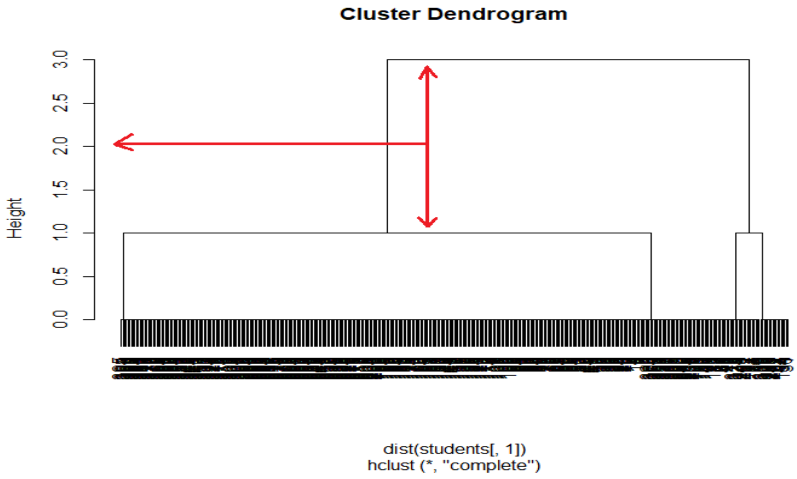


Figure 11. UFS Cluster Dendrogram

4.2.1 PARTITIONAL CLUSTERING

In order to now how many clusters we may use in this case, we applied "fviz_nbclust" function, the optimal number of clusters was the same for all three feature subsets:

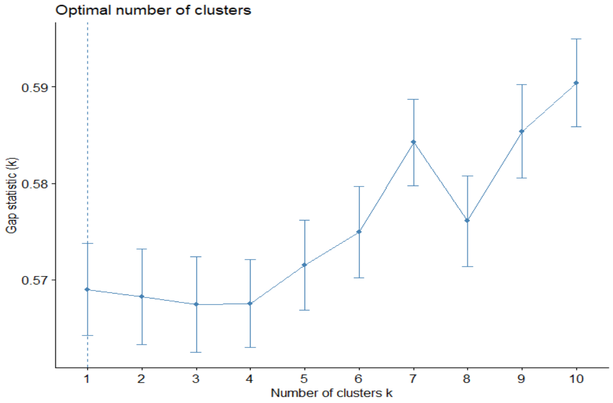


Figure 9. AOV Optimal number of clusters

So, in AOV case 10 clusters were implemented with the resulted accuracy of 0.685404:

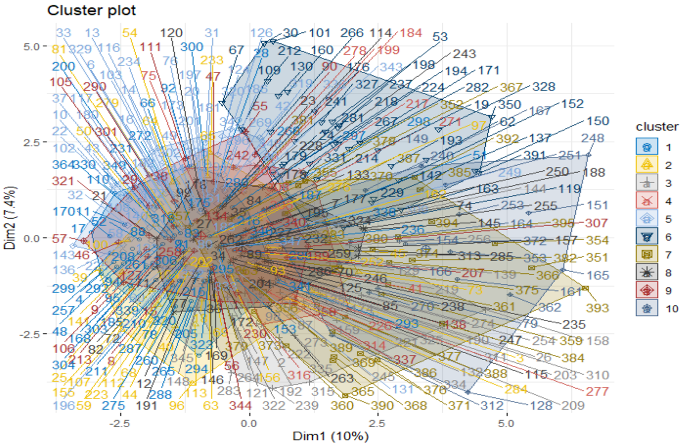


Figure 10. AOV Cluster representation

In MFS case, the results were 0.7850625:

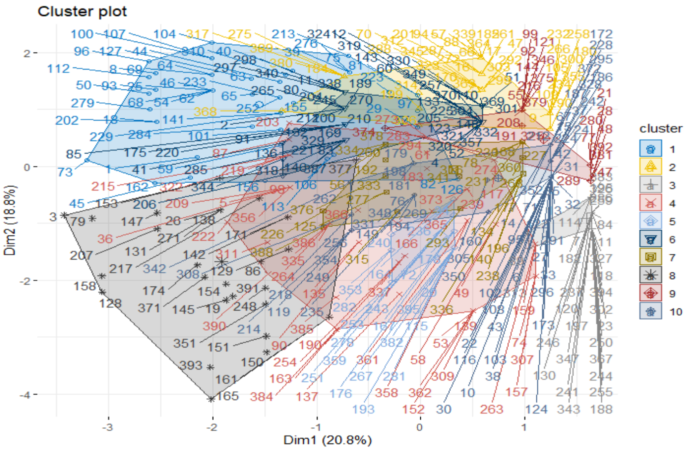


Figure 10. MFS Cluster representation

For UFS there were not enough variables to perform the clustering.

As the prediction output of this method is continuous, the accuracy was calculated as the mean of the (1-clustering result if clustering result is < 0,5 and clustering result otherwise).

4.2.1 PROBABILISTIC CLUSTERING

With “Mclust” function for probabilistic clustering the accuracy was 1 in each case, so we cannot rely on its results.

5. Conclusions

We can observe that for most of the models, less predictive variables meant better classification results, but first, it may be due to random, as the amount of data is relatively small and second, there are still some models which performed better with more predictor variables, as Naïve Bayes one, for example. This overall better performance with less selected features can be explained by the fact of having many non-influential or almost non-influential features which were confusing the model training process. The impact of this issue would be least if we had more data.

Among unsupervised classification models the most precise ones for our dataset were:

1) Naïve Bayes model with prediction accuracies between 71,6(MFS) and 73,7%(AOV) among the tree feature subsets.
2) And the one based on the Support Vector Machines algorithm with prediction accuracies between 70,5(AOV) and 72,6%(UFS) among the tree feature subsets.

Metaclassifiers had not showed better performance then Naïve Bayes algorithm, so this one is the one we will use to predict student’s future academic performance as a supervised classification problem. This model would be used with all available features as a predictor variables.

Among unsupervised classification models the most precise one was partitional clustering, which had not performed well for AOV (68,5%) but showed the best result among all the models implemented during this study, 78,5% for MFS. Due to that fact, this would be the model implemented for the future predictions and the Multivariate Feature Subset previously selected would be used.

Based on these results, we can say, that some models are more sensitive to the number of predictor variables, than others.

No model was able to achieve high accuracy (90% or more) because student’s academic performance is a very complex parameter that depends, aside of a high randomness, on many different things, some of them used as predictors and some even not included in the studied dataset.

5. References

Dataset - http://archive.ics.uci.edu/ml/datasets/Student+Performance#

Columns description -
https://github.com/weysymontt/ML_Assignment_1/blob/master/Dataset/student.txt

Git Hub Repository -
https://github.com/weysymontt/ML_Assignment_1/blob/master/Wladyslaw_Eysymontt_ML_ASSIGNMENT_1.docx

R Script with all variables -
https://github.com/weysymontt/ML_Assignment_1/blob/master/script.R

R Script with multivariate feature subset -
https://github.com/weysymontt/ML_Assignment_1/blob/master/scriptFS.R

R Script with univariate feature subset -
https://github.com/weysymontt/ML_Assignment_1/blob/master/scriptOFS.R

Other references:

https://www.datacamp.com/community/tutorials/neural-network-models-r

https://dataaspirant.com/2018/01/15/feature-selection-techniques-r/

http://www.sthda.com/english/wiki/correlation-matrix-an-r-function-to-do-all-you-need

https://stats.stackexchange.com/questions/111145/how-to-fit-mixture-model-for-clustering

https://cran.rstudio.com/web/packages/GMCM/vignettes/GMCM-JStatSoft.pdf

https://www.intechopen.com/books/recent-applications-in-data-clustering/partitional-clustering

https://www.rdocumentation.org/packages/FNN/versions/1.1.3/topics/knn

http://www.sthda.com/english/articles/36-classification-methods-essentials/146-discriminant-analysis-essentials-in-r/

https://stats.stackexchange.com/questions/215146/using-the-naive-bayes-classifier-in-r-with-continuous-variables

https://www.datacamp.com/community/tutorials/logistic-regression-R

https://www.datacamp.com/community/tutorials/support-vector-machines-r

https://www.rdocumentation.org/packages/rpart/versions/4.1-15/topics/rpart.control

https://www.statmethods.net/advstats/cart.html

https://rdrr.io/cran/RoughSets/man/RI.CN2Rules.RST.html

http://www.milanor.net/blog/cross-validation-for-predictive-analytics-using-r/

https://www.edureka.co/blog/knn-algorithm-in-r/#KNN%20Algorithm%20Use%20Case