|  |  |
| --- | --- |
| **P170M109 Computational Intelligence and Decision** | **Date: 28 February 2018** |
| TOPIC 2 | **Students: Erika Gardini – Mattia Fucili** |

Provide the results obtained following the steps given in the lab work description (txt file). Please, plan your time to meet deadlines. This document (report) should be uploaded to Moodle system.

1. *Split the given dataset (n) into two different samples: training dataset and testing data set. For example, choose randomly 10% of records from given dataset for testing (0,9n records in training dataset, 0,1n records in testing data set). Provide the code and sample sizes after splitting given dataset and check the proportion of class variable in both training dataset and testing dataset.*

To obtain the two dataset (training and test) we used this piece of code:

samp\_size <- floor(0.1 \* nrow(credit))

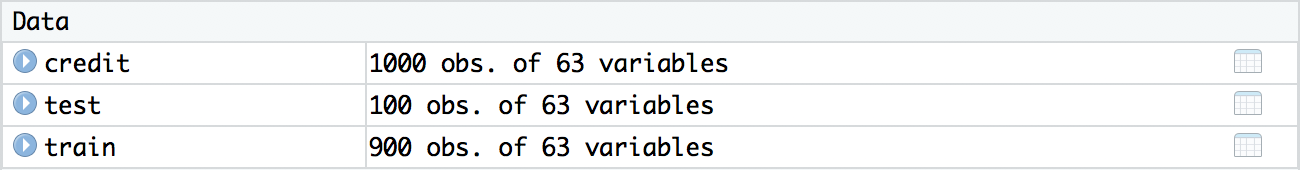
set.seed(123)

train\_ind <- sample(seq\_len(nrow(credit)), size = samp\_size)

train\_set <- credit[train\_ind, ]

test\_set <- credit[-train\_ind, ]

Here the results of this split



The proportion of the class variable in both set is quite similar, in fact the class “Good” in the training set is the 69.55% of the whole dataset and in the test set is 74% and the class “Bad” in the training set is the 30.44% of the whole dataset and in the test set is 26%.

1. *Choose the algorithm (IDR, CART, C4.5 or C5.0) for creating decision trees. Provide summary about the algorithm and clarify requirements on dataset (features) invoked by the algorithm.*
2. *Use at least two categorical features and at least two continuous features (from 1st lab) to create a decision tree. Provide what data preparation steps (if any) were performed before learning the algorithm. Provide summary of output of learning. Display decision tree graphically. Make predictions using testing data set and test the predictions on target feature (using mean values for actual values (yes/no) and predicted values (yes/no), as well as cross tabulation (confusion matrix)).*
3. *Use at least two categorical features, at least two continuous features and at least two derived features (from 1st lab) to create a decision tree. Provide what data preparation steps (if any) were performed before learning the algorithm. Provide summary of output of learning. Display decision tree graphically. Make predictions using testing data set and test the predictions on target feature (using mean values for actual values (yes/no) and predicted values (yes/no), as well as cross tabulation (confusion matrix)).*
4. *Use all features you have in your data set in order to create a classifier of decision tree. Provide what data preparation steps (if any) were performed before learning the algorithm. Provide summary of output of learning. Display decision tree graphically. Make predictions using testing data set and test the predictions on target feature (using mean values for actual values (yes/no) and predicted values (yes/no), as well as cross tabulation (confusion matrix)).*
5. *Perform a modification of learning algorithm by implementing pruning or creating model ensemble (using boosting or bagging (random forest) technique). Provide what data preparation steps (if any) were performed before learning the algorithm. Provide summary of output of learning. Display decision tree graphically. Make predictions using testing data set and test the predictions on target feature (using mean values for actual values (yes/no) and predicted values (yes/no), as well as cross tabulation (confusion matrix)).*