Statistical & Machine Learning

Individual Project

Mario Cortez

Credit Card Default Individual Assignment

Task 1:

For the explanation of the models, we selected 6, this is the list:

1. Logistic regression
2. Decision tree
3. Random forest
4. Support Vector Classifier
5. Bagging Classifier (ensemble)
6. Ada Boost Classifier (ensemble)

1. Logistic Regression: is a statistical model that models conditional probability using the Logistic function. It can predict a binary outcome based on the observations of the data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables. It also can take multiple inputs. The assumptions for Logistic Regression are that there are no outliers in the data and there is no correlation between the independent variables. It helps transform complex calculations of probabilities into a straightforward arithmetic problem. Its very easy to set up and train than other machine learning algorithms, its also very efficient when comparing, it can help reveal interrelationships between different variables and their impact on outcomes, but it has a limitation, as we previously stated, it assumes linearity between the dependent variables and the independent variables.

2. Decision Tree: is comes from a tree structure constituting nodes and branches. Data is spitted based on any of the input features at each node, generating two or more branches as outputs. This iterative process increases the number of generated branches and partitions the original data. This continues until a node is generated where almost all the data belong to the same class are assigned and there are no further splits possible. They come handy when there are several choices involved to arrive at any decision and you must choose accordingly to get the favorable outcome. Their pros are that they are easy to interpret, handles categorical and continuous data well, it can work on large datasets, it's not sensitive to outliers, but the cons are that they are prone to overfitting, it can't guarantee optimal trees, it gives low prediction accuracy for a dataset compared to other algorithms, and calculations can become complex when there are many class variables, it also has high variance which tells us that it the outcome can change from test and train data.

3. Random Forest: This is a supervised ML algorithm which builds from decision tress on different samples and takes their majority vote for classification. The difference between Decision Tree and Random Forest are that RG are created from subsets of data and the final output is based on the average or majority ranking and the problem of overfitting is taken care of, its also slower, it selects observations and builds decision trees, and the average result is taken, it doesn't use any set of formulas. The random forests are much successful than decision tress only if the threes are diverse

and acceptable. The pros are that its robust to outliers, work well with nonlinear data, has a lower risk of overfitting, runs efficiently on a large data set, but the cons are that it might be biased when working with categorical variables, the train is slow, and it is not suitable for linear methods with lots of sparse features.

4. Support Vector Machine: is a supervised algorithm and can be used in nonlinear classification, regression and outlier detection, it's a non-probabilistic linear classifier that uses geometrical approach to distinguish the different classes in a dataset. Kernel is a mathematical function used in SVM to transform nonlinear data to a higher dimensional data set so that, SVm can separate the classes of the data by using a hyperplane. There are various types of kernel like linear, rbf and sigmoid. This kernel can be adapted to the type of dataset we encounter. The pros are that SVM is effective in high dimensional space, is memory efficient, is still effective where a number of dimensions are greater than the number of samples. But for the cons we encounter that if the number of features is much bigger than the number of samples then it avoids over fitting in choosing the kernel function, and it does not directly provide probability estimation like Logistic Regression.

5. Bagging Classifier (ensemble): The term comes from bootstrap and aggregating, each weak learner is trained on random subsample of data sample with replacement and then the model's predictions are aggregated. Bootstrapping guarantees independent and diversity, because each subsample of data is sampled separately with replacement, and we are left with different subsets to train our base estimators. The base estimators are weak learners that perform a little bit better than random guessing, like a decision tree of only depth of three. Then the predictions from these models are combined through averaging. It reduces variance given that sampling is done randomly this usually help reduce variance and avoid overfitting, they have robustness as the aggregation help provide more stability to the predictions, but they are difficult to interpret as they are based on mean predictions from base estimators, but this improves accuracy.

6. Ada Boost Classifier (ensemble): This method uses an iterative approach to lean from the mistakes of weak classifiers and turn them into strong ones. Just as kids learn from the mistakes, boosting algorithm try to build a strong learner from mistakes of the weaker learner, then we create another model by trying to reduce the errors from the previous model. Models are added sequentially, each one correcting its predecessor, until the training data Is predicted, or the maximum number of models have been added. Ada Boost gives for the first stump the same weight and then it starts distributing them by accuracy, until each point has been correctly classified. The pros of Ada Boost is that it's easier to use with less need for optimization of parameters unlike SVM, it can also be used to improve accuracy of the weak classifiers making it more flexible, but for the cons we have that as Boosting learns progressively the quality of the data is important, and also they are sensitive to noisy data and outliers.

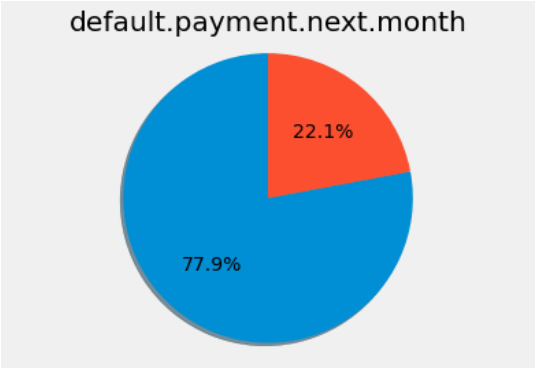
Task 2:

Introduction:

Our task is to first select 5 machine learning predictive algorithms, explain their mechanism and then apply these five models to our Bastable, once cleaned, discretized, binned, and remapped, we will proceed with cross out validation, then fit our models, perform hyper parameter tunning with cross validation, and then evaluating our model with proposed metrics. But first we need to start with a simple exploratory analysis:

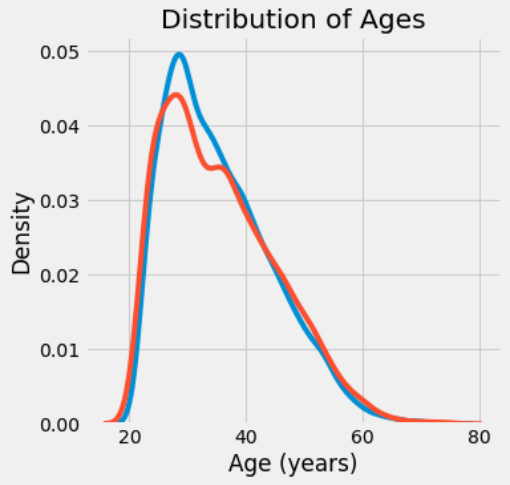
Exploratory Data Analysis (EDA):

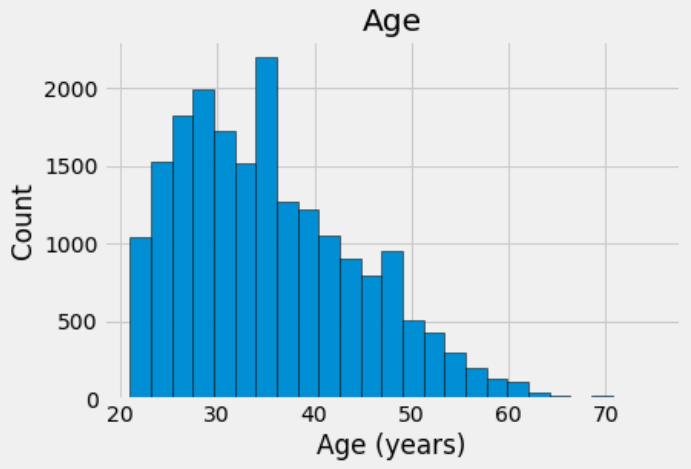
First, the analysis will be centered in the prediction for the default of clients for next month.



For the target value, in this case if the person if going to default in the next month, we observe that 77% of the observed subjects did not default in the studied month but 22% did.

For the Age category we can observe that this feature follows a normal distribution skewed to the left, this is nothing new as it follows the same distribution, we would expect in the bank service industry. Also, for each target value there is no difference in age distribution.

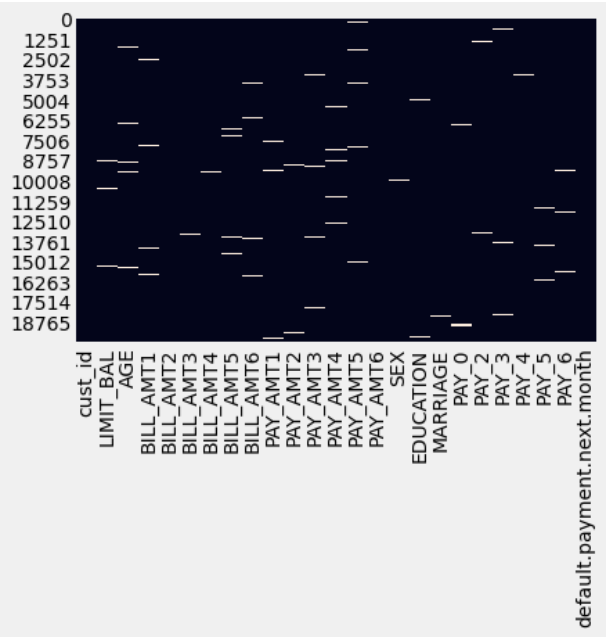




As constant variables do not contain much variance therefore information, we double check we don't have columns with less than 2 unique levels. In our case none of the columns have this issue. But this might represent a problem when working with demographic data, such as average waves in the same

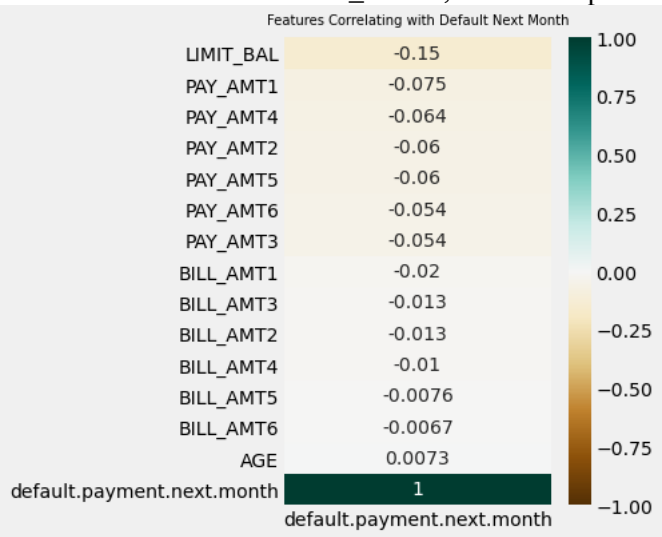
For the missing or null values, we graphically check that some columns have some, and for the numerical variables that represents 2700 observations and for the categorical variables that represents 1748. In this case we will check if the aggregation of missing values for more than 30% represents a change in our dataset, but in our case it does not. We proceed imputing with the mean the numerical variables and for the categorical we fill with "Missing".

When checking for outliers we find there range does not exceed 3%. Then we applied and ordinal encoder for the categorical variables and double check their data type.



Next, we proceed with correlations within variables and target, we can calculate Pearson correlation coefficient between the numerical variables and the target. This is not the greatest way of representing relevance, but it gives un idea of possible relationships within the data.

As we see here, there are no strong correlations but variables LIMIT_BAL, which is the amount of given credit in NT dollars and PAY_AMT1, amount of previous payment in September, 2005, are the two most correlated features to default.



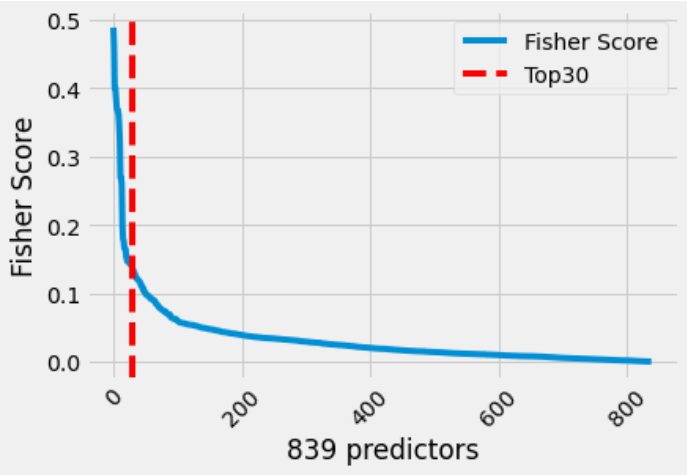
For checking the relevance of the rest of the variables we use Mutual Information (MI), which is the distance between two probability distributions, in comparison with Correlation which is a linear distance between two random variables, MI can also handle nonlinear relationships. If its equal to zero the random variables are independent and higher values mean higher dependency.

After this, we try to increase our AUC we test if adding polynomial terms for numerical variables would add any extra weight to our model but for this case it doesn't, with a threshold of 5% increase of AUC. Now we proceed with Value Transformation, grouping the categories into new categories using Decision Tree and only is the AUC is more than 0.5 and the number of the categories is more than 1. For Discretizing numerical variables, we proceed binning the values.

Then, we proceed with dummy encoding for categorical values, incidence replacement and Weight of Evidence conversion, given by your notebook and Dr. Coussement research papers.

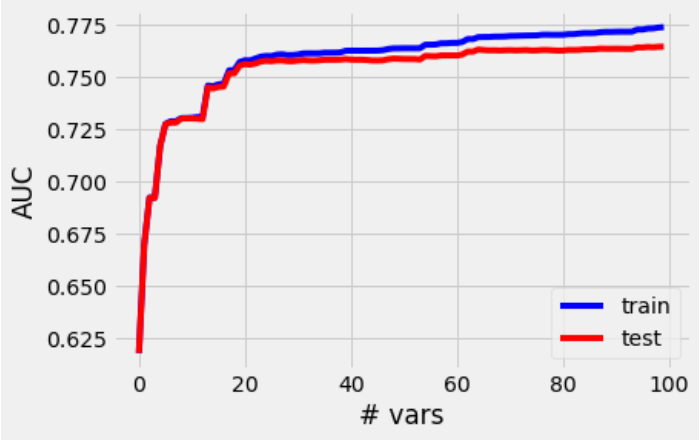
After all the transformations we export our Bastable than contains 2000 observations and 841 features.

As we have too many features, we need to apply Fisher Score, which is how much information about an unknown parameter we can get from a sample. In other words, it tells us how well we can measure a parameter, given a certain amount of data. The graphical representations let us know that from our predictors, about the top 30 would be enough for our fitting.



We can also observe this in the graph for the AUC fitted in a Logistic Regression, for our case taking 30 predictors would ensure we're not overfitting the model.

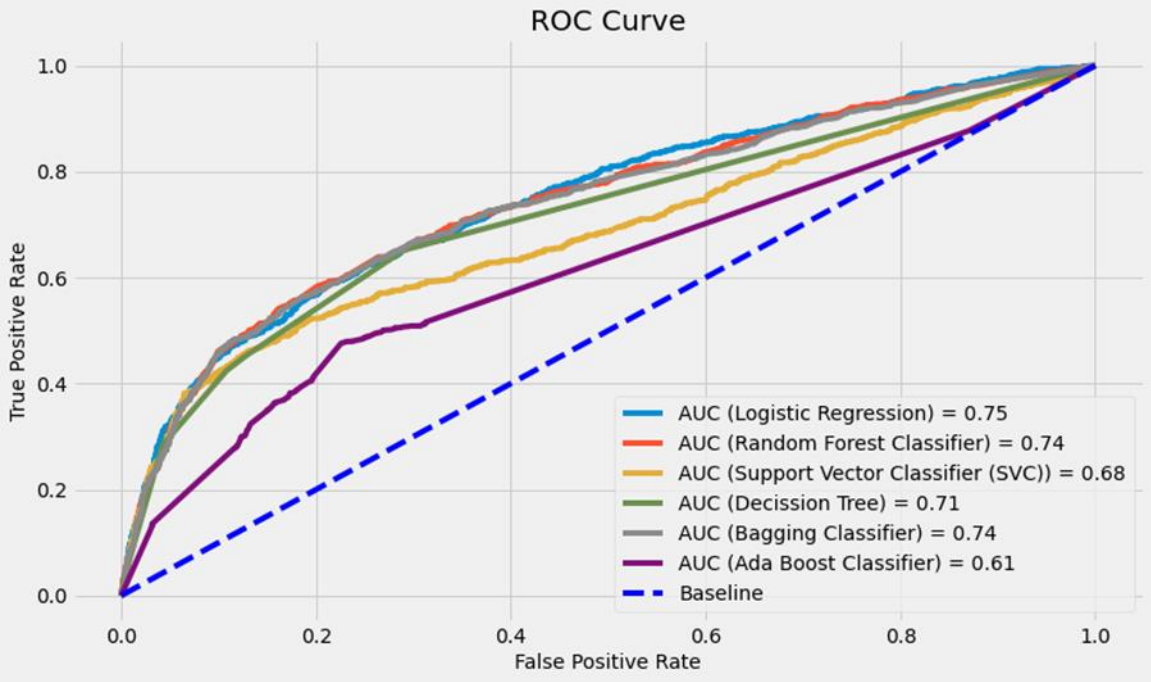
This is the reason we selected the top 30 features.



Then we proceed with Dimensional Reduction: PCA, this technique from linear algebra helps us with dimensionality reduction, and therefore if not, the machine learning algorithms would take more time to process.

This leaves us with a base table of 20000 observations and 30 columns.

We selected 6 algorithms for fitting and evaluating. Starting from Logistic Regression, which is the simplest and easer to explain one, and at the end we added two ensemble methods to see how they would perform comparing with the ones we saw on class. The cross-validation method we're using is holdout with train test split, the training set is what the model is trained on, and the test set is used to see how well that model performs on unseen data. We're using 80% of the data for training and 20% for testing. K fold cross-validation is when the dataset is randomly split up into 'k' groups. One of the groups is used as the test set and the rest are used as the training set. The model is trained on the training set and scored on the test set. Then the process is repeated until each unique group has been used as the test set. GridSearchCV applies k fold cross-validation to select from a set of parameter values; in this example, it does this by using k-folds with k=10, given by the cv parameter. For the SVC, Decision Tree, and the ensemble algorithms we only applied cross validation of 3 instead of the optimal 10, for time constrain reasons, algo the parameter grid for the GridSearchCV was a small one as this takes quite a long computing time. We perform these methods with accuracy as the optimizer and then later we evaluated Area Under the Curve (AUC) with the Receiver Operating Characteristic (ROC) curve, which would be a better indicator as it would imply the probability that the classifier will rank a randomly chosen positive example higher than a randomly chosen negative



As per our ROC curve comparison graph, we can observe that the best performing model for our dataset in the test set, is still the simplest one, Logistic Regression. There is 75% chance model would be able to segregate the points or rank correctly.

Note: I exported the Bastable in feather format as pickle and csv took a considerable larger space and could not be uploaded to Github.

References:

- https://medium.com/@hertan06/which-features-to-use-in-your-model-350630a1e31c
- https://medium.com/@eijaz/holdout-vs-cross-validation-in-machine-learning-7637112d3f8f
- https://vitalflux.com/grid-search-explained-python-sklearn-examples/#Grid_Search_and_Random_Forest_Classifier
- https://towardsdatascience.com/roc-and-auc-how-to-evaluate-machine-learning-models-in-no-time-fb2304c83a7f
- https://medium.com/analytics-vidhya/decisiontree-classifier-working-on-moons-dataset-using-gridsearchcv-to-find-best-hyperparameters-ede24a06b489
- https://www.datacamp.com/community/tutorials/adaboost-classifier-python
- Mutual information. Link: https://en.wikipedia.org/wiki/Mutual_information
- sklearn.feature_selection.mutual_info_classif. Link: https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.mutual_info_classif.html
- Coussement, K., Lessmann, S., & Verstraeten, G. (2017). A comparative analysis of data preparation algorithms for customer churn prediction: A case study in the telecommunication industry. Decision Support Systems, 95, 27-36.
- Verbeke, W., Dejaeger, K., Martens, D., Hur, J., & Baesens, B. (2012). New insights into churn prediction in the telecommunication sector: A profit driven data mining approach. European Journal of Operational Research, 218(1), 211-229.

- Lessmann, S., Baesens, B., Seow, H. V., & Thomas, L. C. (2015). Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. European Journal of Operational Research, 247(1), 124-136.File
- Ensemble methods. Link: https://scikit-learn.org/stable/modules/ensemble
- Dr. Phan Minh, MBD2021_InClass Kaggle_2_Modeling_Group1_Py_v5.3 - Data Processing
- https://towardsai.net/p/machine-learning/why-choose-random-forest-and-not-decision-trees
- https://blog.paperspace.com/adaboost-optimizer/#:~:text=AdaBoost%20is%20an%20ensemble%20learning,turn%20them%20into%20strong%20ones.

List of Variables:

There are 25 variables:

- ID - ID of each client
- LIMIT_BAL - Amount of given credit in NT dollars (includes individual and family/supplementary credit
- SEX - Gender (1=male, 2=female)
- EDUCATION - (1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)
- MARRIAGE - Marital status (1=married, 2=single, 3=others)
- AGE - Age in years
- PAY_0 - Repayment status in September, 2005 (-1=pay duly, 1=payment delay for one month, 2=payment delay for two months, ... 8=payment delay for eight months, 9=payment delay for nine months and above)
- PAY_2 - Repayment status in August, 2005 (scale same as above)
- PAY_3 - Repayment status in July, 2005 (scale same as above)
- PAY_4 - Repayment status in June, 2005 (scale same as above)
- PAY_5 - Repayment status in May, 2005 (scale same as above)
- PAY_6 - Repayment status in April, 2005 (scale same as above)
- BILL_AMT1 - Amount of bill statement in September, 2005 (NT dollar)
- BILL_AMT2 - Amount of bill statement in August, 2005 (NT dollar)
- BILL_AMT3 - Amount of bill statement in July, 2005 (NT dollar)
- BILL_AMT4 - Amount of bill statement in June, 2005 (NT dollar)
- BILL_AMT5 - Amount of bill statement in May, 2005 (NT dollar)
- BILL_AMT6 - Amount of bill statement in April, 2005 (NT dollar)
- PAY_AMT1 - Amount of previous payment in September, 2005 (NT dollar)
- PAY_AMT2 - Amount of previous payment in August, 2005 (NT dollar)
- PAY_AMT3 - Amount of previous payment in July, 2005 (NT dollar)
- PAY_AMT4 - Amount of previous payment in June, 2005 (NT dollar)
- PAY_AMT5 - Amount of previous payment in May, 2005 (NT dollar)
- PAY_AMT6 - Amount of previous payment in April, 2005 (NT dollar)
- default.payment.next.month - Default payment (1=yes, 0=no)
-

Amount of given credit in NT dollars

- PAY_AMT1 - Amount of previous payment in September, 2005 (NT dollar)