|  |  |  |
| --- | --- | --- |
|  |  |  |
| Ramses Meza  April 2021  [https://github.com/lordtable/UTDSGC\_C2T](https://github.com/lordtable/UTDSGC_C2T2)3 |  | Data Analytics Course 2: Data Science with Python: Final Report |

Business Acumen

CreditOne revenue is based on correctly assessing the credit worthiness of clients’ customers. Failing to do so represents an existential threat to CreditOne’s profitability. The company has detected an increased rate of defaulting on credit lines that client companies have extended to their customers, which were previously vetted by CreditOne. It is the company expectation that applying Data Science techniques can help improving the credit worthiness assessment and provide grounds enhanced default risk mitigation, by correctly predicting the size of the line of credit to be approved to new customers, and assessing their risk of defaulting if approved.

Data Management, Cleaning & Pre-processing

The CreditOne dataset is located on a *MySQL* database, which has been retrieved to a local PC based on the SQL address provided by Credit One. This dataset has been locally stored as a .csv file. It is structured in a table format: each row represents an undisclosed customer already credit-rated by Credit One for the April to September 2005 period. Each column represents an attribute or feature collected post-rating, such as:

* **LIMIT\_BAL**: Amount of total credit given
* **SEX**: Gender of the customer
* **EDUCATION**: Highest level reached
* **MARRIAGE**: Marital status
* **AGE**
* **PAY\_0**🡪**PAY\_6**: History (6 months) of past payment.
* **BILL\_AMT1**🡪**BILL\_AMT6**: Amount of bill statement (6 months)
* **PAY\_AMT1**🡪**PAY\_AMT6**: Amount of previous payment (6 month)
* **Default payment next month**: Default status (Defaulted or not)

This data was managed as a Python Pandas’ DataFrame, with local backups in .csv format at major checkpoints.

Data cleaning/pre-processing fundamentally consisted on data type transformations to integers, removing extra or misplaced header lines, elimination of duplicated instances, header labels editing and feature categorizing. The cleaned data set consisted of 30,000 observations/instances. Of these, 70% were retained as instances used for training all the predictive models to be later described, while 30% of the instances were kept for models’ testing and evaluation.

Exploratory Data Analysis (EDA)

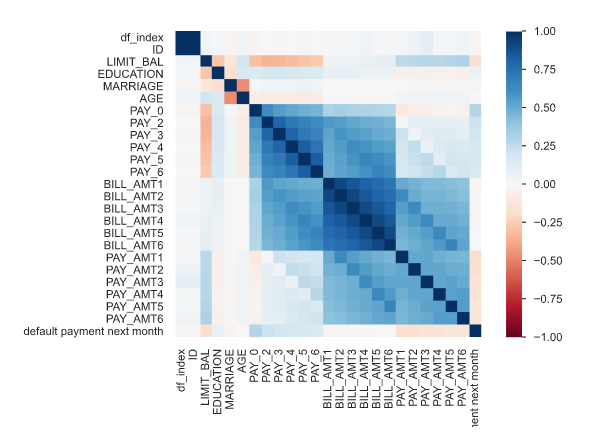
First, EDA allowed identifying several independent variables or predictors that are inter-dependent to each other. Some of the likely inter-dependent variables include the PAY variable group, the BILL\_AMT variable group, the PAY\_AMT variable group, and also a sizeable inter-dependency between variables on the BILL\_AMT variable group and the PAY\_AMT variable group. These observations can be inferred from Figure 1, which depicts a Feature Correlation Matrix. 

Figure 1: Feature Correlation matrix (Sperman’s r)

The same matrix allowed focusing our attention on the variables that are correlated to the target variables, facilitating the posterior feature selection for Machine Learning modelling. In the case of the target variable LIMIT\_BAL, we can see that the demographic variables (SEX, EDUCATION, MARRIAGE & AGE) are somewhat correlated (either positively or negatively), as well as the PAY\_ and PAY\_AMT feature groups. We are also interested on being able to predict customer clients’ default likelihood, so considering the proxy DEFAULT STATUS variable as the target or dependent variable, it is possible to make an early identification of those variables that are unlikely to be major drivers on any future predicting model. These variables are: Education, Sex, Marriage status, and Bill statements (April-Sept 2005) amount. This also played a role on the corresponding feature selection before building a predictive model for the default status.

Determining the Size of the Line of Credit: Regression approach

Three regression modelling methods were built using the same selected features to predict the LIMIT\_BAL variable: Supported Vector Regression (SVR), Random Forest Regression (RFR) and Linear Regression (LR). Baseline models were trained using their corresponding default parameters, and also systematically fined-tuned using 1-2 key parameters on each. Cross-validation, R2 and RMS error (RMSE) were the metrics used to compare the performance of the models (both using default and customized parameters) and select the method/model to use for the final prediction. I can state the following about the performance of each model/method:

* **Random Forest**: After tuning, the reduction on RMSE is negible, at the expense of a much lower R2.
* **SVR**: The tuned model shows a considerable improvement with respect of the default parametrization; however, it is much more computationally expensive.
* **Linear Regression**: For the current problem, LR is practically unsensitive to the main parameters tested, and ranked low in terms of metrics when compared to any Random Forest regression run tested.

Based on the above, I decided to perform the regression using RF with its default settings. The model evaluation yielded R2= 0.432 and RMSE=96,972. Plotting the predicted values (out of the test features) vs the actual LIMIT\_BAL values of the test set, as seen on Figure 2, allows to state that the regression model does not look reliable enough, despite of the efforts: There is plenty of dispersion in the predictions for a common actual value, and the model seems to have an overall bias towards under-predicting values as the actual values increase.

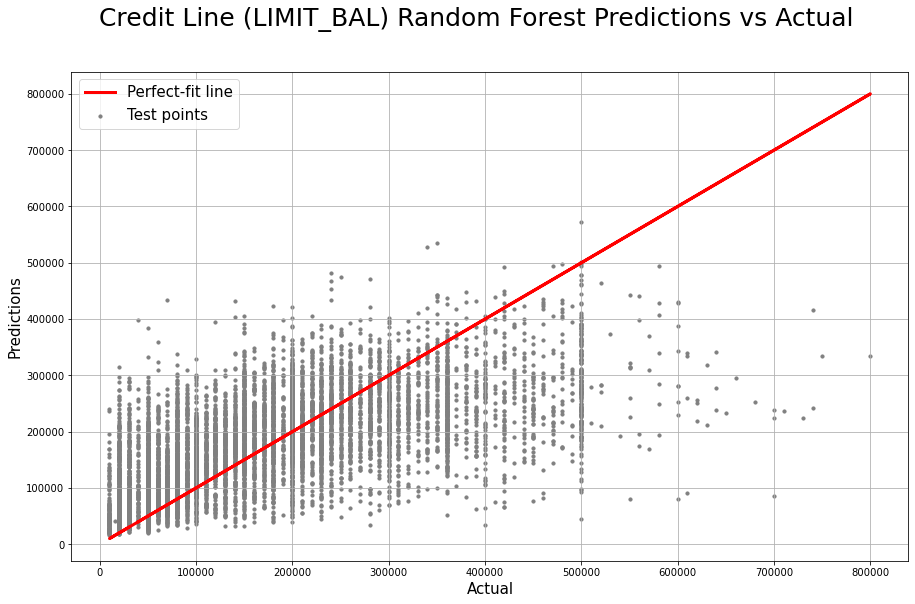


Figure 2: LIMIT\_BAL Predicted vs Actual values (testing dataset)

Determining the Size of the Line of Credit: Classification approach

Another observation from Figure 2 is that points on the actual value axis are not randomly scattered or dispersed along the x-axis, but rather having a “categorical” or “discrete” appearance, indicating that a classification, rather than a regression approach may be more suitable. With this in mind, I discretized the LIMIT\_BAL variable into four (4) bins or classes for the whole dataset, as follows:

* US$ 10,000🡪50,000
* US$ 50,000🡪140,000
* US$ 140,000🡪 240,000
* US$ 240,000🡪 1 MM.

These ranges allow for each bin to have roughly the same number of instances. Then the train/test data splitting was redone as before. Due to personal time-limit constraints, I only tested the Decision Tree Classifier (DTC), using the accuracy as a metric to QC the systematic fine-tuning of the algorithm parameters. The final DTC model was build using a maximum tree depth=5, allowing for an overall accuracy of 51%. The model tends to preferentially select variables of the PAY\_ and the PAY\_AMT\_ feature groups, while leaving the variables EDUCATION and AGE as minor drivers on the classification, towards the final branches of the tree. Given the depth of the tree and number of variables involved, it is not currently possible to state clear classification rules, however importance of features can work as a proxy, but that is pending.

The DTC performs a better job at triaging the LIMIT\_BAL than the regression approach, as inferred from its Confusion Matrix shown Figure 3. The model provides a reasonable accuracy, especially for the end-member classes (81% for US$ 10,000-50,000, and 77% for US$ 240,000-1 MM). The model allows a first-order discarding of credit amount ranges to be granted to the prospective client, based on the consistent large amount of True Negatives for each class.

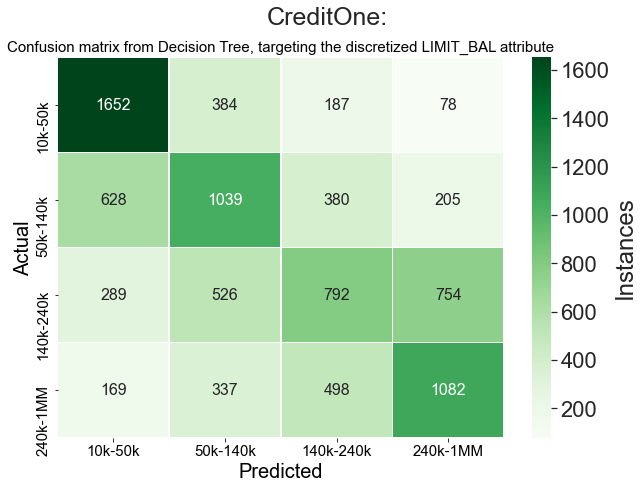


Figure 3: LIMIT\_BAL Classification Confusion Matrix

Estimating probable default status

The fact that the DEFAULT\_STATUS variable is binary makes it also suitable for a classification approach. Again, due to time constraints, only one classifier (a DTC) was built. In a similar way as the previous classification, a systematic testing of parameters was done for fine-tuning the model, which was finally run using a max\_depth=4, which significantly improved the precision with respect of the default parametrization, especially on predicting the DEFAULT class. The model tends to preferentially select variables of the PAY\_ feature group, while relegating and the PAY\_AMT\_ feature group towards the last stages of nodes of the tree. No demographic-type of feature was selected by the algorithm.

As seen on the corresponding confusion matrix depicted on Figure 4, the model performs very well at predicting clients unlikely to default on their credit line balances. The major concern for CreditOne would be the large number of False Positives (FP), because the model is predicting 1,223 defaulting clients as non-defaulting, which represents ~ 14% of the test samples and 63% of the actual defaulting instances. It is worth noticing that the DEFAULT\_STATUS variable is binary and severely skewed or imbalanced: from the total population sampled, about 78% is non-defaulting. This fact may generate a bias on the prediction a possibly under-estimating the predictive power of the model.

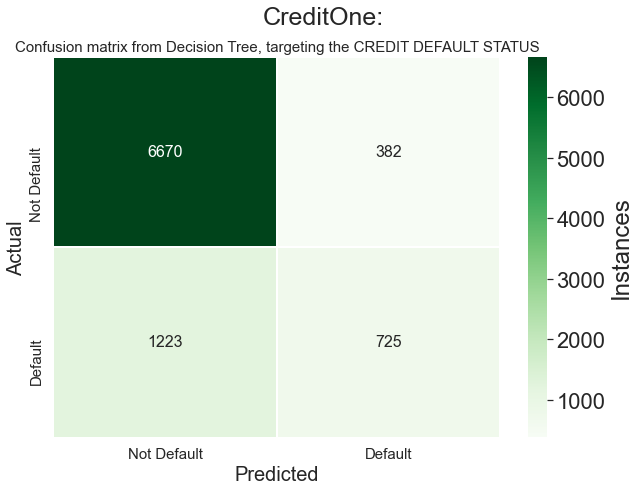


Figure 4: DEFAULT STATUS Classification Confusion Matrix

Recommendations and Future Work

* The lack of an INCOME variable may be a major factor on the current lack of across-the-board robustness of both regression and classification modelling. Current and new data collection should consider gathering such variable.
* DEFAULT STATUS classification yielding a large number of False Positives for the non-defaulting client class can still represent a threat to the revenues. This could be mitigated by deriving detailed information from the confusion matrix, convert that to probabilities or chances of failure, and feed those onto the CreditOne risk management system to determine potential losses and then determine whether those fall within CreditOne risk tolerance. This risk management needs to incorporate the LIMIT\_AMOUNT extended to each prospective client, since this will also drive the potential losses if the DTC for DEFAULT STATUS is wrong for a prospective client.
* Built and test at least two additional classifiers for each target variable analyzed.
* Determine a way to automatically derive the features used on each classifier and their importance (weight), in order to efficiently report classification rules.