Edvinas Vencevičius, Case 5 – Lending Club Loan Data – ML model GBM

For this task, I chose my pre-organized and concatenated dataset (LoanDataFixedA + LoanDataFixedB), the model choice was Gradient Boosting Machine.A screenshot of a computer

Description automatically generated

When creating the model, several unnecessary columns were discarded: Addr State, Table Names, Last Pymnt D, and key parameters were selected, such as ntrees – 50 (later changed to 100).

A screenshot of a computer

Description automatically generated

First test of the model with 50 ntrees (the number was increased to 100, because we wanted to see when exactly the logloss value would start to drop)

A graph with blue dots

Description automatically generated

Second (main) model test

A graph with a line graph

Description automatically generated

Training\_logloss indicator - a measure of the model's accuracy, fluctuates depending on the number of trees. As the number of ntrees starts to increase, it decreases in the range of 0 - 10 ntrees (on average from 0.0355 to 0.0211), which indicates improving results. The logloss value starts to rise again from an average of 15 ntrees and rises steadily to the highest limit - at 71 ntrees (0.0862), then decreases slightly, but remains higher than in the previous stages - at 100 ntrees (0.0812). This pattern may indicate a tendency to transplant above a certain number of trees.

So we see that a small number of ntrees undoubtedly indicates a higher model accuracy, but this does not make sense, so the optimal choice would be between 70 and 80 ntrees values, since from 85 ntress the accuracy starts to decrease again.

A graph with numbers and text

Description automatically generated

The trend shows that the indicator with the greatest impact on the forecasts is Recoveries, with a value of 1.0. Starting with Int\_Rate and other lower indicators, a very large gap is visible – 0.22 and below. The lowest values ​​are such as Pub Rec, Home\_Ownership, which is not surprising, since these are not numerical values ​​(perhaps it would have made sense to remove them before building the model).

A white sheet with black text and yellow squares

Description automatically generated

Charged Off: The model accurately predicts loans that are in default, with a recall of 0.99, which means that 99% of cases were correctly classified.

Fully Paid: The model also identifies fully paid loans well, with a precision of 1.0, which indicates that all actual fully paid cases were correctly identified.

On the other hand, for rarer categories such as Late (16–30 days) and Late (31–120 days), the precision is lower, at 0.87 and 0.83, respectively, indicating that these categories are difficult to accurately predict.

In summary, the precision scores are generally high across all classes, indicating a low number of false positives across all cases.

A screenshot of a document

Description automatically generated

Charged Off: The model’s performance in identifying charged loans shows high performance (0.98) and precision (0.99).

Fully paid: For fully paid loans, the model has high recall (0.99) and precision (0.99), indicating that the performance in this category is good.

However, again, the same trend is that the model has difficulty predicting the minority classes “Overdue (16-30 days)” and “Overdue (31-120 days)”), showing zero recall (0.0) and low precision. These are challenging cases.

In summary:

The Gradient Boosting Machine model showed quite good predictive ability to distinguish between the loan statuses “Paid” and “Fully Paid”, and therefore achieved the highest precision in these common categories. However, due to class imbalance, its performance weakened when processing less common states, such as “Late,” resulting in a decrease in cancellation rates and making it difficult for the model to make accurate predictions. Despite these limitations, the model’s performance in distinguishing between defaulted loans and successful repayments highlights its effectiveness in mainstream classifications.