Prediction of credit card defaults.

Konstantin Lekomtsev
Data Science Career Track

Mentor: Nishant Gupta

Objectives

- Predict whether a client defaults as of next month (data for previous 6 months given).
- Identify main predictors.
- Understand key information about the dataset.

Dataset

- Contains historical data on defaults of the bank customers in Taiwan for the 6 months in 2005.

- Features:
  customer age,
  credit amount,
  educational level,
  marital status,
  gender,
  bill amount, payment amount, repayment status for each month Apr 2005 – Sept 2005.

- Imbalanced, 28% of customers defaulted in October 2005 and 72% who did not default.

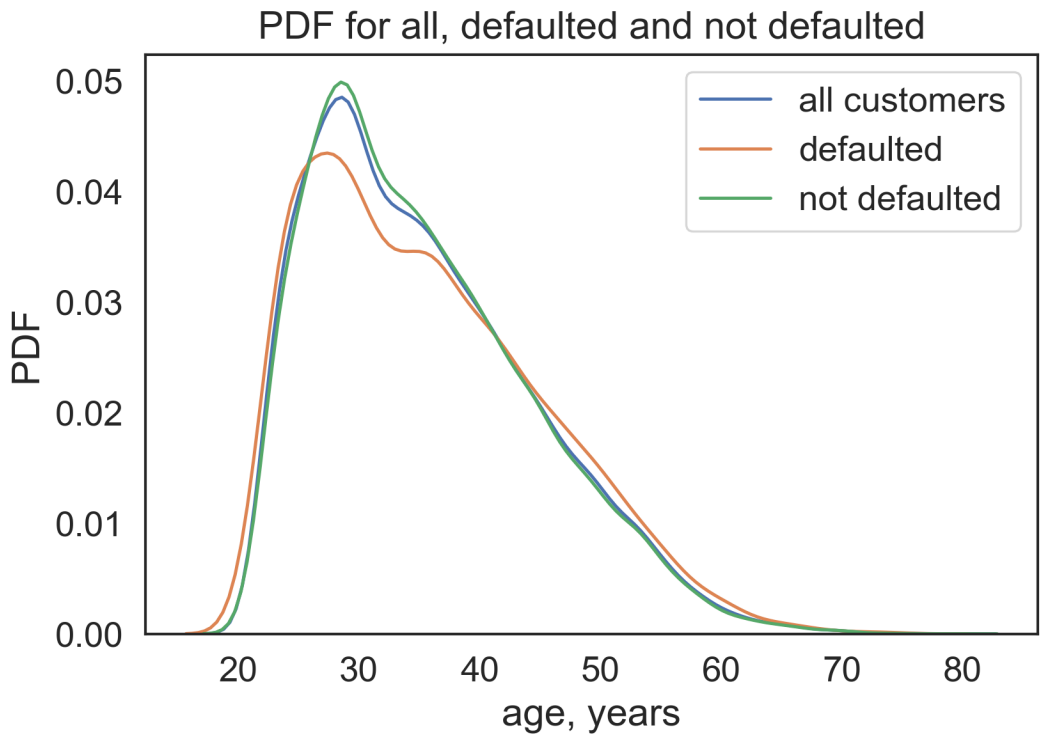
- 30,000 rows, 25 features.

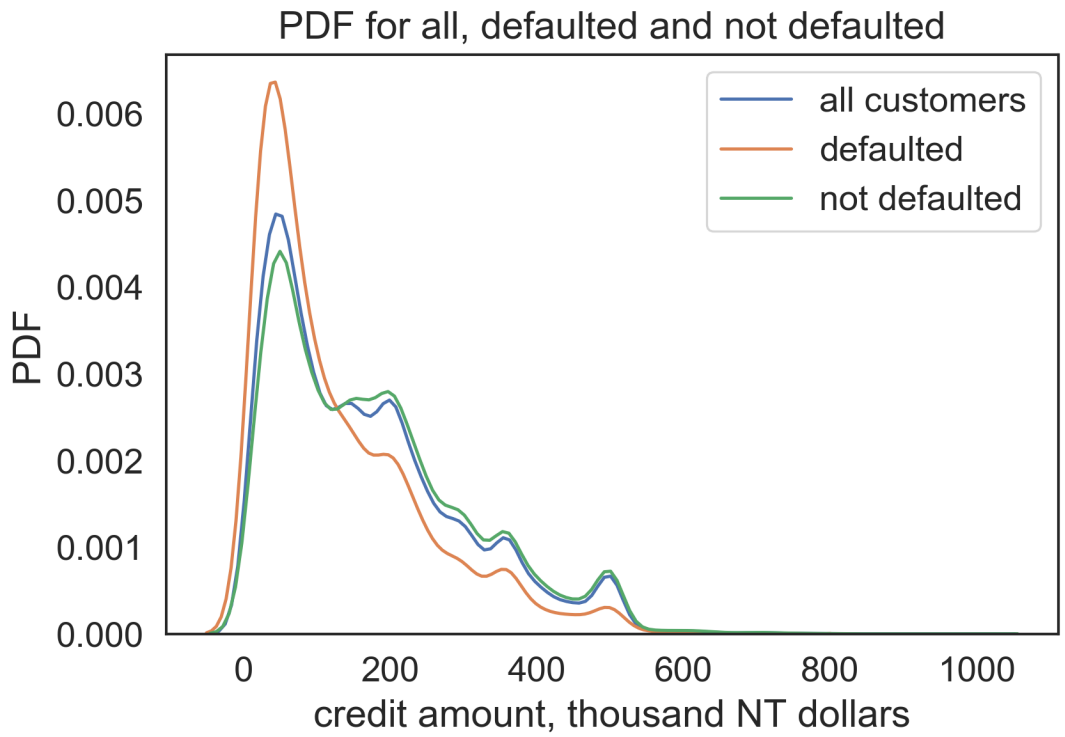
- Available both on Kaggle and at UCI machine learning repository.

Data story I

- The median credit amount for the defaulted customers was 90,000 NT dollars, compared to 150,000 NT dollars for not defaulted customers.

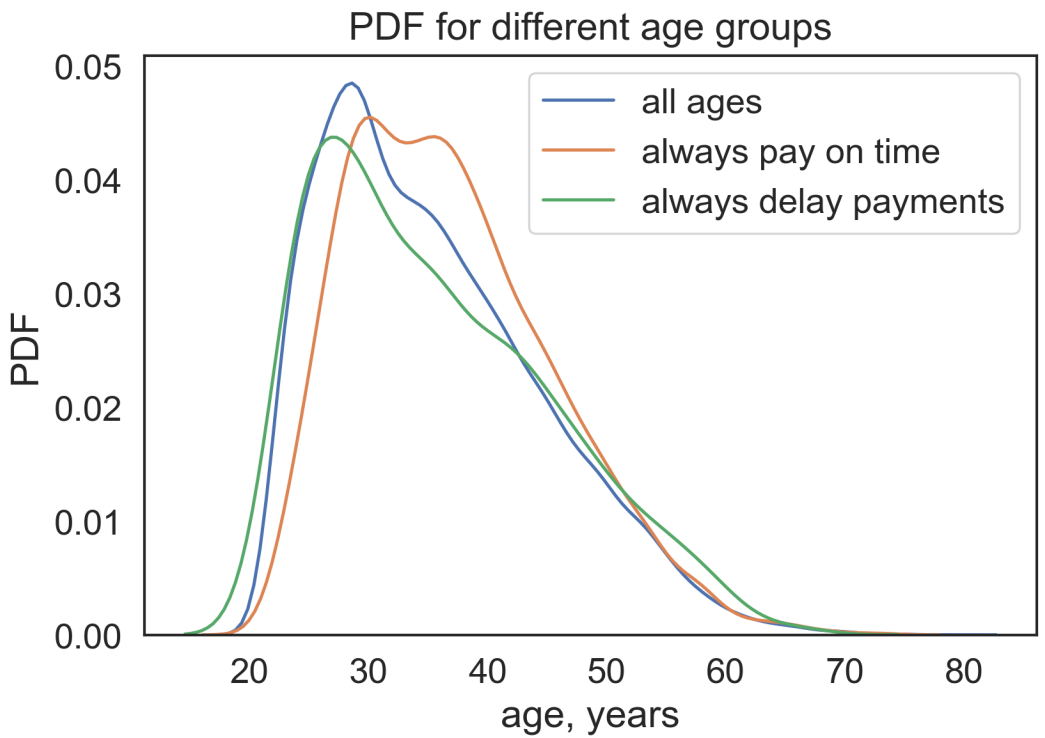
- The median age for both categories was 34 years old, however, there was a moderate shift of the distribution's mode for defaulted vs not defaulted customers from 27 to 29 years old.

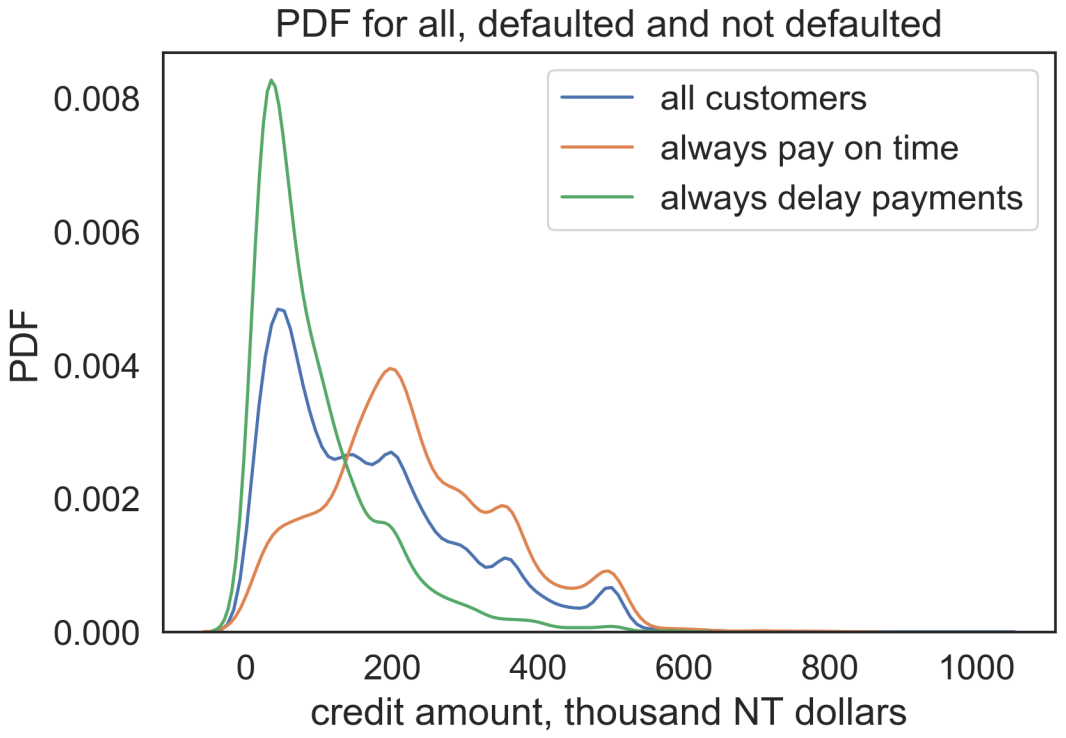




Data story II

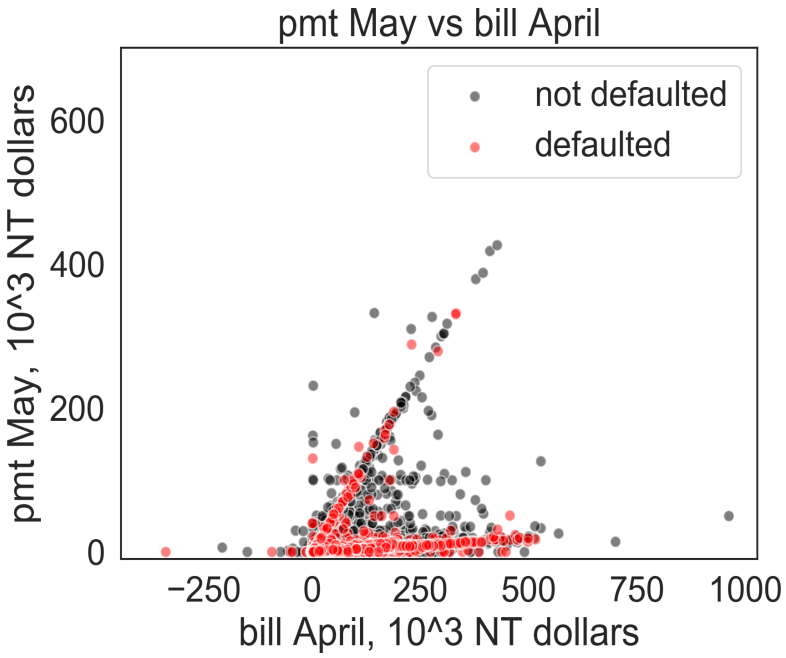
- The customers who always paid on time and always delayed payments were filtered out based on their repayment status.
- The customers who always paid on time had a median credit limit of 210,000 NT dollars and those who always delayed had only 70,000 NT dollars.
- The median value of age for customers who always paid on time was 36 years old, and for customers who always delayed payments it was 34 years old.

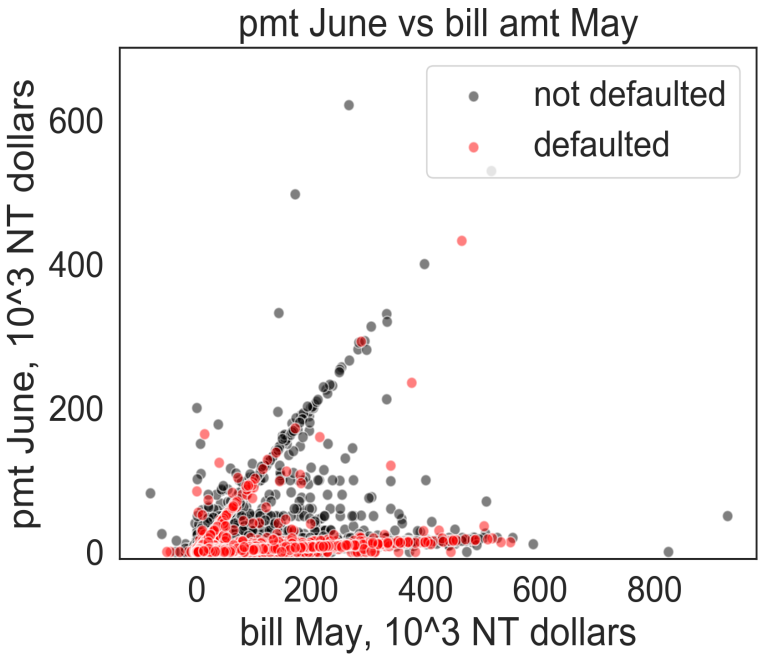


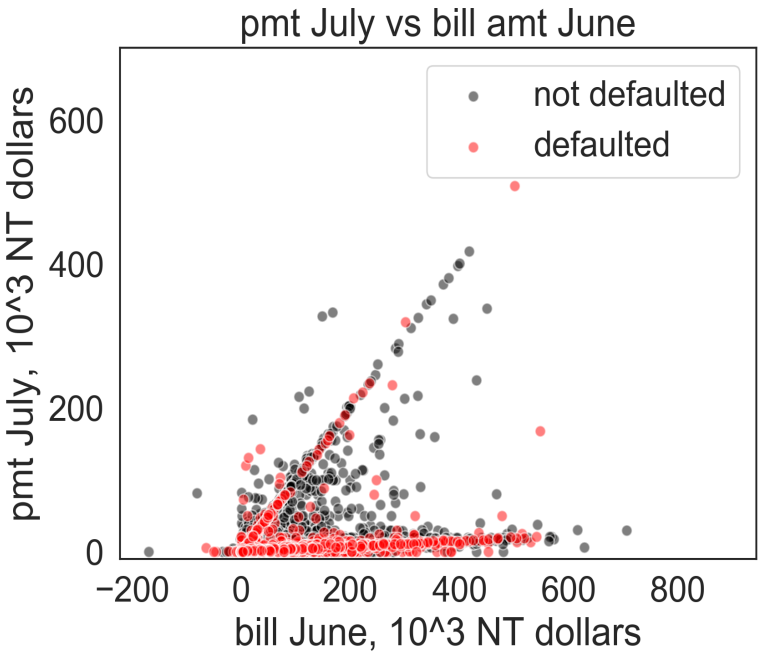


Data story III

- Investigated pairwise relationships between bill amount in each month vs payment amount in the following month using scatter plots.
- There are two clear trends: correlation between the minimum payments and the bill amounts, and the full bill repayments and the bill amounts.
- As expected, the customers who did not default form a larger proportion of those who paid more than a minimum payment each month.







Data story IV

- By this dataset, we were given a ”snapshot” of economically active adults in Taiwanese society in 2005.
- Young and educated society with a larger proportion of the female population, 60% female.
- Most customers are either married or single, with a very small proportion divorced. For example, 98.8% of customers with a bachelor’s degree are either married or single, the rest are either divorced or classified as other.
- 53.2% single customers in the data set vs 45.53% who are married.
- Out of those who are married 77.3% have at least a bachelor’s degree, out of those who are single 86.6% have at least a bachelor’s degree.

Data wrangling and preprocessing I

- There were no missing values, but as part of pre-processing workflow the functions imputing missing categorical and numerical features were included in the code. Imputing was done based on the values in the training data, assuming test data were not seen.

- There were outliers in the bill amount and payment amount columns, majority of the outliers showed the customers who paid much larger or smaller amounts compared to the previous month bill, or those who had a negative bill amount. All these situations are expected to occur in practice.

- The outliers in the numerical columns that were outside were replaced with median values.

$$lower\,bound = quartile_1 - 1.5 * IQR$$

$$upper\,bound = quartile_3 + 1.5 * IQR$$

Data wrangling and preprocessing II

- Performed 70/30 train - test split, validation set was introduced as part of stratified cross-validation.

- Training data were oversampled with replacement using imbalanced-learn API to generate balanced dataset.

- To prepare the dataset for machine learning models:
  -> the categorical features were converted to dummy variables,
  -> the numeric columns were standardised using sklearn standard scaler, first by fitting and transforming the training data and then by transforming the test data.

Feature engineering

- Credit utilisation for each month:

$$credit\ utilisation = \frac{bill}{credit\ amount}$$

- Cash flow for each month:

$$cash\ flow = pmt - bill$$

- The cashflow values are negative when a client pays less than the bill amount for a given month, zero when the bill is fully repaid, and positive when a client pays more than the bill amount.

ML models I

- Five different classification models were applied:
  -> logistic regression,
  -> Gaussian kernel support vector machine,
  -> random forest classifier,
  -> XGBoost classifier,
  -> voting classifier based on the above four.

- For all classifiers, the hyperparameters were assessed by:
  -> using validation curves for each parameter and a default classifier,
  -> then tuned using grid search cross-validation based on AUC scoring.

- For the optimized classifiers, a generalization performance on the training data was assessed:
  -> using stratified 5-fold cross-validation based on AUC score.

- Bias – variance trade-off was assessed:
  -> using learning curves for each parameter.

ML models II

- The following parameters were tuned:
  Logistic Regression -> regularization strength C,
  SVM -> regularization strength C and kernel parameter gamma,
  Random Forest -> max depth of trees and number of trees,
  XGBoost -> max depth of a boosted tree estimator, number of tree estimators, learning rate (controls step size in weight update), gamma parameter (minimum loss reduction required to make a further partition on a leaf node of the tree).

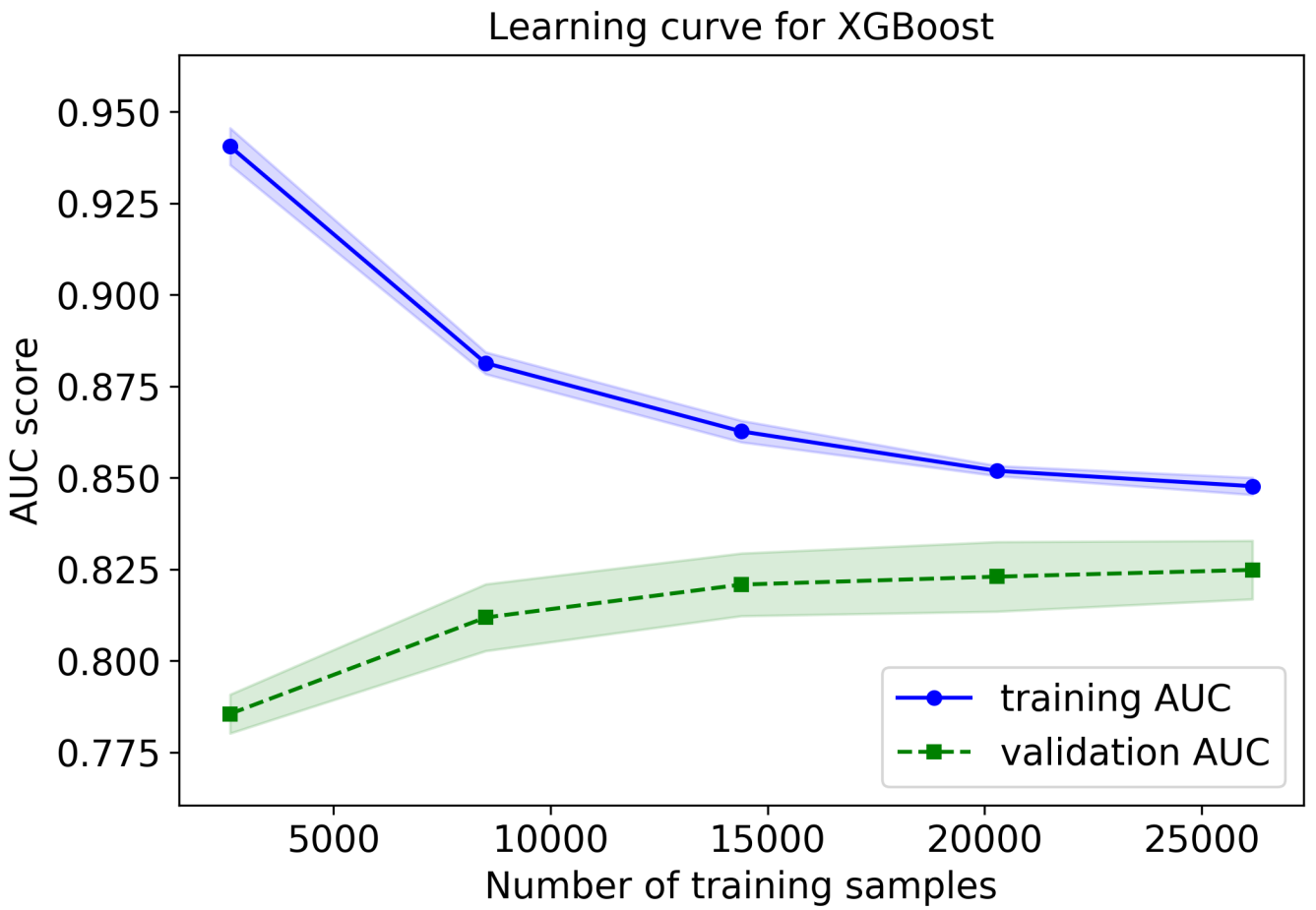
- Performance on the test data was assessed using AUC score, confusion matrix, ROC, and precision-recall curve.

	Logistic Regression	Random Forest	SVM	XGBoost	Ensemble
AUC on test	0.73	0.77	0.75	0.78	0.77

ML models III

The best performing classifier was XGBoost using oversampled training data.

XGBoost	AUC
train	0.82
test	0.78



Best classifier I

true label

Confusion matrix	defaulted	not defaulted
defaulted	1250 TP	741 FN
not defaulted	1437 FP	5572 TN

predicted label

XGBoost	precision	recall
defaulted	0.88	0.79
not defaulted	0.47	0.63

- There is a large number of false positives, the negative class (not defaulted) misclassified as the positive class (defaulted), which was a natural consequence of oversampling during training.
- However, for identification of customers with high probability of default, it is preferable to have more false positives rather than false negatives.

$$TPR = \frac{TP}{P} = \frac{FP}{FN + TP}$$

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN}$$

$$REC = TPR$$

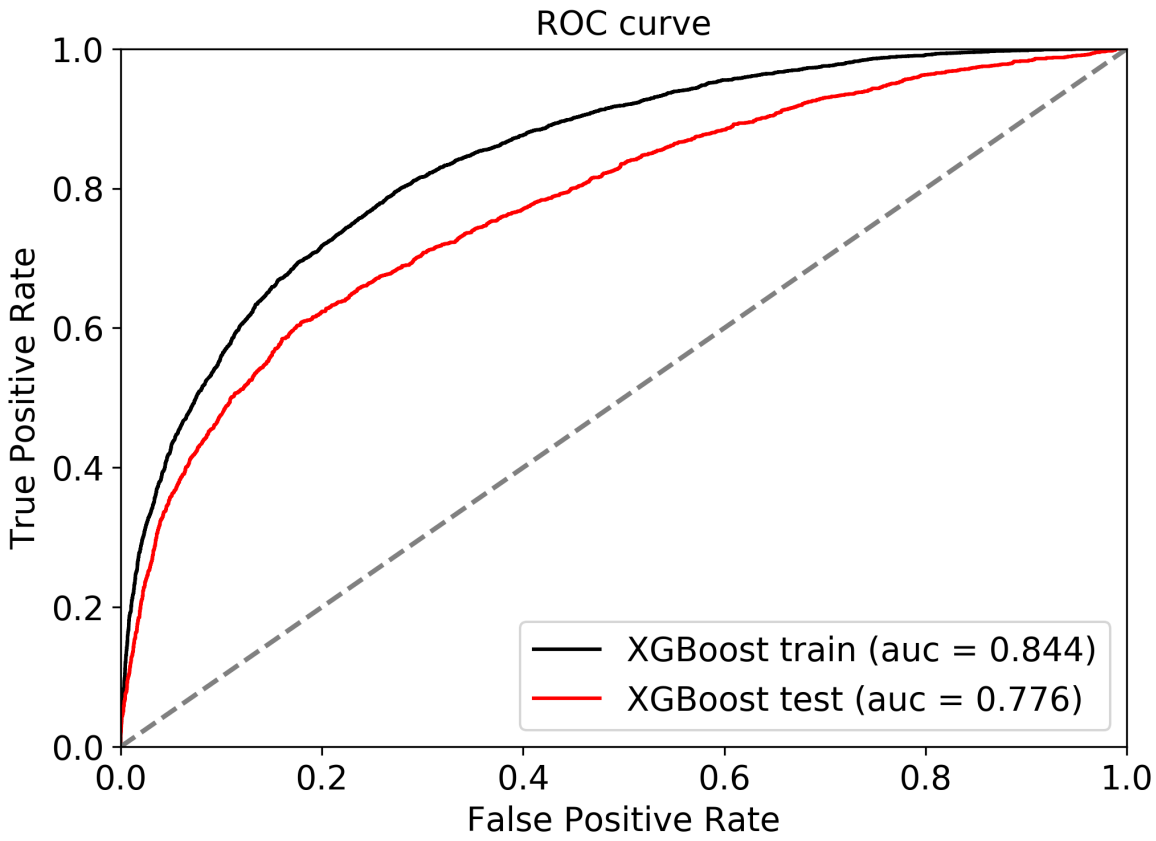
$$PRE = \frac{TP}{TP + FP}$$

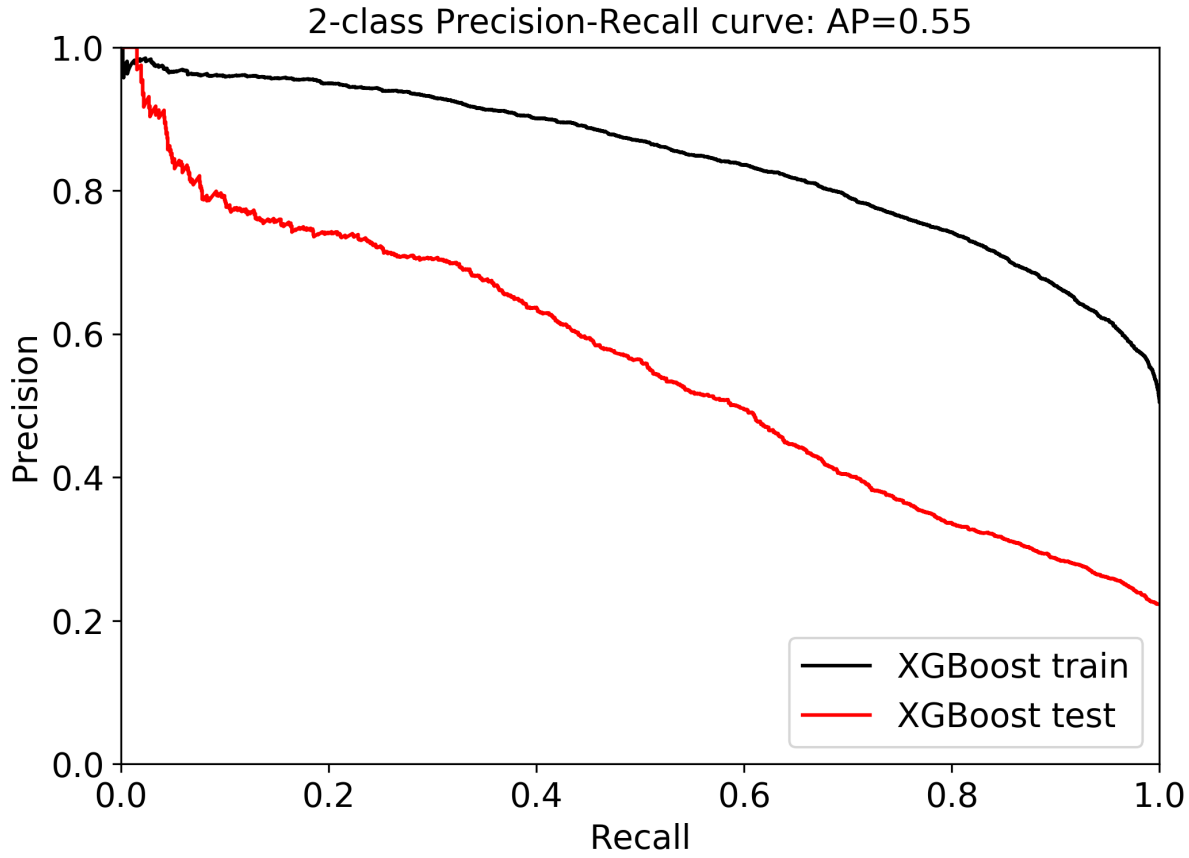
Recall: if I pick a random positive example, what is the probability of making the right prediction?

Precision: if I take a positive prediction example, what is the probability that it is indeed a positive example.

Best classifier II

- XGBoost ROC curve and precision-recall curve.





Features importance

Logistic Regression

Rank	Feature
1	payment status in September
2	bill amount in September
3	credit utilization in June

- ranked by the largest coefficient and p value = 0.000 after fitting training data using startsmodels

Random Forest

Rank	Feature
1	payment status in September
2	payment amount in September
3	credit amount

XGBoost

Rank	Feature
1	payment status in September
2	payment amount in August
3	credit amount

Conclusions

- The best performing classifier was XGBoost. However, its optimized performance was just marginally better than other considered classifiers.

- XGBoost performance was sub-optimal for predicting defaults with a large number of false positives, which was a result of using the oversampled dataset for training. Nevertheless, oversampling yielded better results, compared to putting more weight on the samples of less frequent class during training.

- The payment amount and payment status 1-2 months before October 2005 and credit amount were among the top 3 predictive features.

Future work:

- One of the ways to improve predictive performance could be adding even more engineered features.

- Getting data for additional 6months would help to take into account seasonality throughout the year.