Predicting repayment ability of loan clients using imbalanced data with Random Forest and Adaboost

Ragnhild Skirdal Frøhaug[1,1]

Abstract

Purpose The purpose of this paper is to predict the repayment ability for loan applicants for loans in the Home Credit bank.

Data The data used to predict repayment ability is application data from the customer, as well as recent credit card history and installment payment history on previous loans.

Method The methods used for data preparation are manual feature engineering and principal component analysis to reduce the dimension of the application data. The methods used for prediction are Random Forest (RF) classification with stratified sampling and Adaboost classification with down-sampling. In addition 10-fold cross-validation (CV) was used to test the robustness of the Random Forest Classifier.

Results Both the RF with stratified sampling and the Adaboost classifier reached an AUC-score of 68% on the test set. The relative importance of the features were similar for the two models, however, the Adaboost classifier had a more even feature importance distribution.

Value This report adds value by proposing a method for feature engineering and classification methods to predict the repayment ability of future customers of Home Credit. The report further adds value by discussing the validity of the results.

1. Introduction

Many people struggle to get loans due to insufficient or non-existent credit histories. Home Group aims to provide loans to this group in a regulated, safe and trustworthy manner. The company was founded in 1997 and operates in 9 countries. Their business model is to offer loans to people with little or no credit history in a responsible manner. For many of the customers of Home Credit, the bank is the first regulated institution to extend credit to them. The company provides POS loans, cash loans and revolving loan products both online and through physical

Email address: s350110@oslomet.com (Ragnhild Skirdal Frøhaug)

distribution networks [1].

Home Credit uses various statistical and machine learning models to predict the repayment ability of their customers. In 2018, they released some of their data on Kaggle [2]. Their motivation was to see if the Kaggle community could provide new methods and new insights to the data. The data includes application data of new clients, demographic data and historical credit behaviour.

The purpose of this paper is to use some of the data provided by Home Credit to predict which clients of the company have repayment difficulties. The main challenge of the data is that it is highly imbalanced. Only a few of the customers in the client base have repayment difficulties. Hence, the challenge at hand is to create a predictive model that is able to deal with this imbalance and identify the clients that will not be able to repay their loan. Additionally, this report analyze and discuss the influence the different features have on the repayment ability of the clients. Lastly, this report discuss the ethics around predictive modeling of repayment-ability.

2. Data

The data used for this project was fetched from the open Kaggle database and is provided by the company HomeCredit. The data includes application data, credit card balance data and instalment payment data. The data sets consist of raw features, hence relevant features has been extracted from the data. The sections that follow describe the extracted features. Table 1 gives an overview of the number of extracted features from each data set.

There are some difficulties with this data that is easy to spot. First, the data is highly imbalanced, where only 0.08 % of the examples in the data represents clients with repayment difficulties. The methods section, present some techniques to deal with this imbalance.

In the next sub-sections, each data set is discussed. The features which are aggregated from the data is explained. For each data set, the correlation of the features with the target variable is shown as a color coded correlation matrix. Correlation with the target variable might indicate which features will be of importance for predicting the target variable correctly.

2.1. ***The target variable***

The goal of this project is to identify loan applicants with payment difficulties based on their application, as well as credit card data and installment data from previous loans. The target of the training data is 1 and indicates that the client has had repayment difficulties. Meaning

Table 1: Overview of extracted features and total raw features.

Data	**# features extracted**	**# raw features**
Application data	18	122
Installment data	5	6
Credit card data	7	16
Total	**30**	**144**

that the client had late payment more than X times on the first Y installments of the loan in the sample (The letters X and Y is used because Home Credit do not specify the exact number of of installments the client fails to pay to be considered a client with repayment difficulties). The target variable is 0 for all other cases.

2.2. ***Application data***

The raw features of the application data includes 122 features and 307 511 examples. Table 2 shows the number of numerical and categorical features. The application data is the core data set which also includes the target variable. All other data sets provided by Home Credit is related to the application data through a set of IDs. Since the problem at hand is to identify which clients will have repayment difficulties among the new loan applications, this data set represents the number if available data points at hand. The data set includes 307 511 unique client IDs.

The categorical features in this data set is excluded in this project. Each of the categories includes multiple categories (between 5 and 20), hence the one-hot-encoded data set was too large for the computational resources available for this project.

2.2.1. Numerical features in the Application Data

There are 106 numerical features in the application data. The type of data included in the application ranges from age, days since last phone change and the day the application was registered to flags marking if the client uploaded a certain document, or a rating of the region the client lives in. However, few of these features have any significant correlation with the target variable.

Table 2: Number of raw features in the application data.

Description	**Count**
Numerical features	106
Categorical features	16
Number of examples	307 511
Unique client IDs	307 511

One of the goals of this report is to identify the features that influences the prediction. However, features with low correlation are less probable of influencing the prediction. However, they might still contribute. The feature space can be reduced to make it more manageable and comprehensive. The technique Principal Component Analysis (PCA) can be used to achieve this, and still maintain the information of the less correlated features. However, using PCA comes at the cost of making the variables less interpretable. Therefore, the numerical features with a correlation above 0.04 (absolute value) are maintained as separate features.

Figure 1 shows the correlation matrix of the features with a correlation with the target variable above 0.04(absolute value). As seen in the figure, the most correlated features are the EXT_SOURCE_1, 2 and 3 variables. These variables are data sources external to Home Credit. Home Credit has concealed the meaning of these variables for privacy purposes and to protect some company knowledge from the public. In addition to the EXT_SOURCE_X variables, the *Days employed* variable is also somewhat correlated with the target variable. This feature indicates how many days before the application day the client started its current employment.

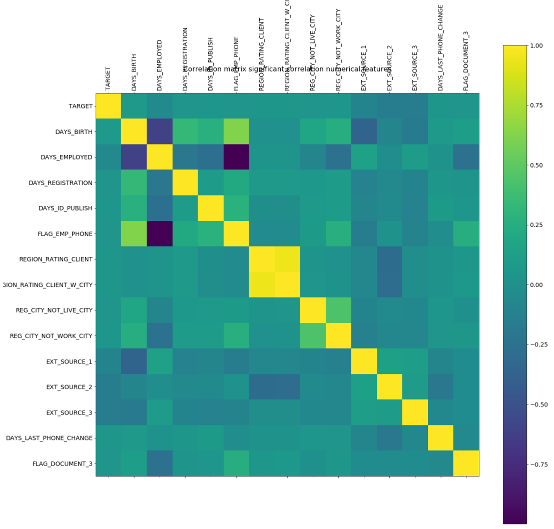
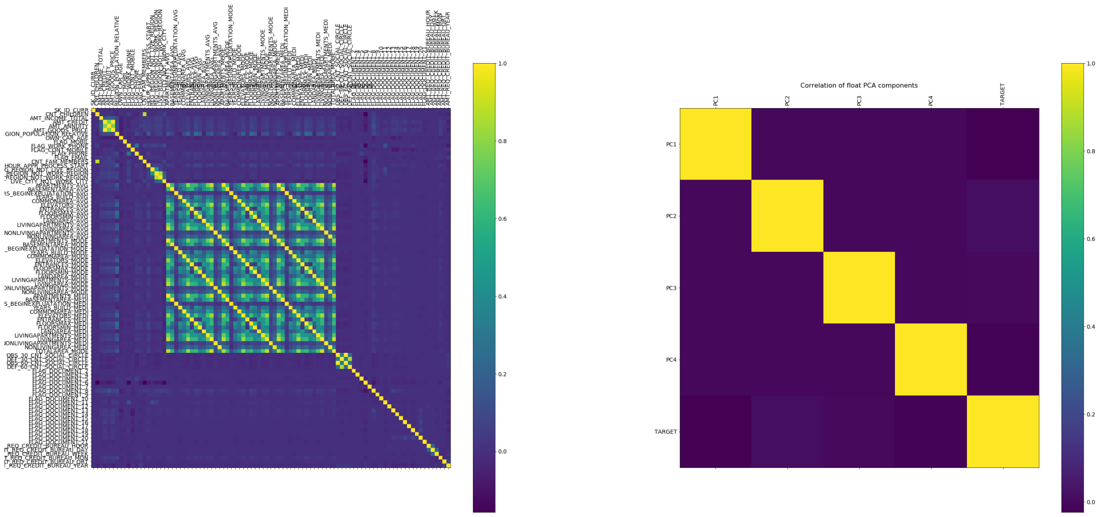


Figure 1: Correlation matrix of features with correlation above 0.04 (absolute value) with the target variable.

Figure 1 also shows some internal correlation among the features. This correlation could have been removed by using PCA on all the features in the application data, however this would also have concealed the meaning of all of the features. Hence, some correlation is accepted. The methods used to make the predictions are tree-based methods, hence some correlation among the features are accepted.

The correlation matrix for the features with correlation below 0.04 (in absolute value) with the target value is shown in Figure 2a . The green and yellow patterns means that groups of these features are highly correlated and contains some of the same information. Hence, the feature space can be reduce to remove this co-linearity. As mentioned, PCA can be used to reduce the feature space and at the same time maintain the information conatined in the features. In addition has PCA the advantage that the resulting principal components contains the maximum



(a) Correlation matrix of features with correlation below 0.04(absolute value) with the target variable.

(b) Correlation matrix of features with correlation below 0.04 (absolute value) with the target variable.

Figure 2: Correlation matrices for features below the correlation threshold of 0.04 (absolute value)

variation possible within the data. Further, the PCA features are also independent of each other. The features with an absolute correlation with the target variable below 0.04 are reduced to 5 features using PCA.

Figure 2b shows the correlation matrix of the PCA features and the target variable. The first PCA feature shows some negative correlation with the target variable, whilst the other components do not show significant correlation. However, the correlation matrix also shows that the features are independent of each other. Further, figure 3 shows the proportion of variance explained by each principal component, where the first component contributes considerably more to variation in the data than the other components.

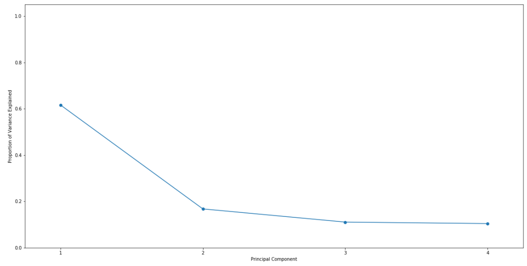


Figure 3: Propotion of variance explained by each PCA feature.

After principal component analysis and analysis of feature correlation with the target variable, the feature space of the application numerical features is reduced from 104 features to 14 features where 4 features are principal component features and 10 features are raw features.

An overview of the minimum, maximum, mean and standard deviation and count of each feature can be found in the appendix.

2.3. Installment data

The installment payment data contains repayment history for previously disbursed credits in Home Credit. The installment payment data is related to the application data with a Client ID. The data set only contains information on clients that have had previous loans in the Home Credit Bank. This means that some of the clients in the application data is not represented in the installment payment data.

One row represent repayment on one installment of a credit loan in Home Credit. There is one row for every payment that was made and one row for each missed payment. The data set contains 6 numerical features including the relative number of days since each installment payment from the application day. This is a negative number (e.g a -200 value means that the installment in this row was paid 200 days ago). Further, one column contains the relative number of days since the installment was due (e.g a -198 value means that the payment was due -198 days ago). The data also includes the size of the installment (the value to be paid), and the payment size (the value actually paid by the client).

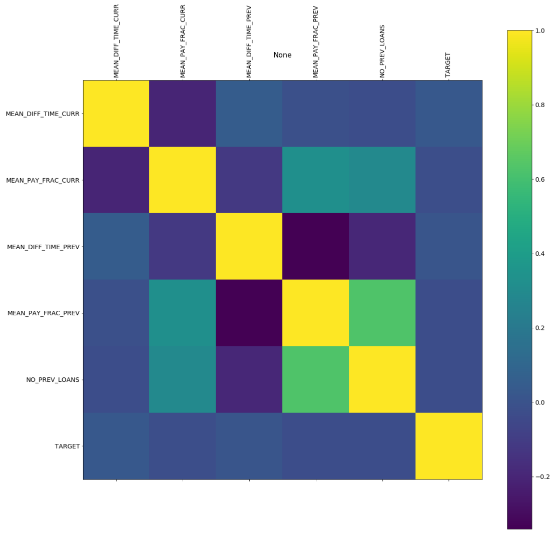


Figure 4: Correlation matrix of installment data with target value.

Table 3 gives a brief overview of the raw installment payment data. The data contains more unique customer IDs than the application data (hence, some of the examples are redundant), and it contains approximately 13.5 million rows.

In order to use the data for the classification problem, the data has to be aggregated to a client

Table 3: Number of raw features in the Instalment data.

Description	**Count**
Numerical features	6
Categorical features	0
Unique examples	13 605 401
Unique customer IDs	339 587

Table 4: Number of raw features in the Credit card balance data data.

Feature	**Description**
Mean diff payment time of current loan	Mean time between payment due date for the installment and the payment date. A positive number indicates that the installment was paid past the due date. A negative number indicates that the installment was paid before the due date.
Mean diff payment time of previous loans	Same as above, calculated as a mean of the previous loans of the customer. 0 if the customer only has one loan in home credit.
Mean payment fraction of current loans	Mean fraction of the installment payment paid for each installment on the current loan.
Mean payment fraction of previous loans	Same as above, calculated as a mean of the previous loans of the customer.
Number of previous loans	The number of loans the customer has had in Home Credit. 1 if the customer only has one loan.

ID-level. Since a customer might have multiple previous loans (the number of previous loans are in the range of 0 to 26). In total 5 features was generated from this data, and the raw features dropped from the data set.

The repayment ability of the client might vary over time, and the success of repayment might vary over time. Assuming that the most recent previous loan best represent the current situation of the customer, a set of features is created to represent the status on the most recent loan, and a set of features is created to represent the average of all other previous loans. Table 4 gives a description of the features aggregated from the installment data. An overview of the minimum, maximum, mean and standard deviation and count of each feature can be found in the appendix.

The features includes the mean difference between the payment day and the due date of the current loan as one feature, and for previous loans as one feature. It also includes the mean payment fraction on the current loan, and one feature for all previous loans. In addition, the number of previous loans are also counted. The value of 1 previous loans means that the client have had one previous loan. Figure 4 shows the correlation matrix of the engineered installment features with the target variable. There are some strong internal correlations among the features. Further the mean payment difference on previous loans and the mean payment fraction on the previous loans are the features most correlated with the target variable.

2.4. ***Credit card balance data***

The credit card balance data contains credit card history for current and previous credit card loans for the client applications in the application data. The original raw data contains 15 numerical features and one categorical feature (indicating whether the loan is active or com-

Table 5: Number of raw features in the Credit card balance data data.

Description	Count
Numerical features	15
Categorical features	1
Unique examples	3 840 312
Unique client IDs	103 558

pleted.).

One row in the raw data represent the credit card balance of one month both customer credit and cash loans. Hence, one loan of a client has multiple rows in the data. Each row representing a monthly credit card balance. Table 5 gives an overview of the raw credit card balance data. The data contains 103 558 unique client IDs, hence, not all clients in the application data is represented in the credit card balance data (which makes sense since not all applicants have previous loans in the bank).

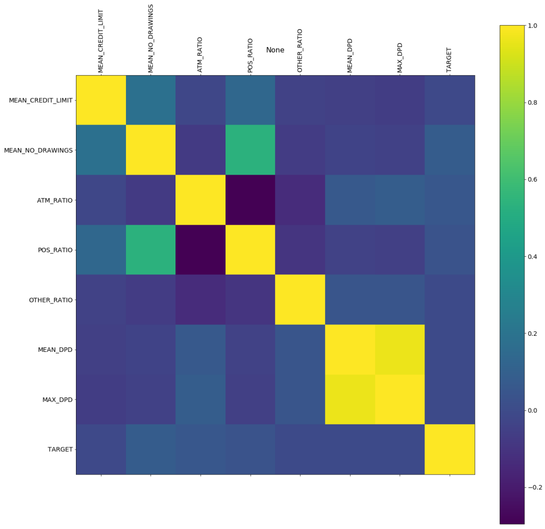


Figure 5: Correlation matrix of credit card balance data with target value.

Since the credit card balance data has monthly examples for each loan, new features was aggregated from the data. The features was aggregated at the per client ID-level. The correlation matrix of the features are displayed in figure 5 and a description of each feature is given in table 6. The features are based on the mean monthly behaviour of the client. The three ratios, ATM ratio, POS ratio and Other ratio sum in total to 1. These features each represent a fraction of total spending. An overview of the minimum, maximum, mean and standard deviation and count of each feature can be found in the appendix.

Regarding the correlation matrix, it is the same as for the other data sets, there are some internal correlation among the features and low correlation with the target variable. However, the *mean number of drawings*, *the ATM ratio* and *POS ratio* is slightly more correlated with the target variable.

Table 6: Number of raw features in the Credit card balance data data.

Feature	Description
Mean credit limit	Mean monthly credit card limit
Mean number of drawings	Mean number of monthly drawings
ATM ratio	Mean drawings from ATM
POS ratio	Mean drawings for buying goods
Other ratio	Mean other drawings
Mean days past due	Mean number of days past payment due date
Max days past due	Maximum number of days past payment due date

3. Methods

For this project, I chose tree based models with bagging and boosting to predict the repayment ability of new client loan applications to Home Credit. These methods were chosen as they are able to detect non-linear relationships in the data and do not assume any underlying distribution of the data.

3.1. ***Random forest classifier***

The random forest classifier fits a number of decision tree classifiers on various sub-samples of the data. The model is to prefer when you want to identify non-linear relationships between the target variable and the features.

The algorithm works as follows: When building a split in a tree, the algorithm is not allowed to consider all available features. Instead, it chooses to split on the feature which improves the prediction the most. It does so by choosing a feature from a subset of the features in the feature space. Hence, instead of the feature with the strongest prediction ability being chosen at the top split of each tree, different features are chosen for the top split. This allows for diversity amongst the trees in the forest. The trees are decorrelated [3].

There are several advantages of selecting the random forest classification technique to do predictions on the Home Credit data set. First, the model is less influenced by outliers than linear models such as logistic regression. Note that it is not totally robust towards outliers, but it is able to handle them to some extent. Second, the model does not make any assumptions about the underlying distribution of the data. Hence, it is able to handle any colinearity that might exist in the data. Lastly, and maybe the most important reason for choosing tree based models, is its ability to discover non-linear relationships among the features and the target variable. In a business perspective, you do not only want to be able to predict which clients have repayment difficulties, you also want to know the factors that lead to this repayment difficulty.

In addition, the Random Forest in the sklearn library in Python are implemented with a sampling technique which handles imbalanced data well. This method is called stratified sampling. In stratified sampling, the feature space is divided into two sub-groups based on the informativeness of the features with respect to the target variable. When the informativeness of each feature is computed, the features are ordered based on their relative informativeness and then divided into two subgroups. The features with informativeness above a threshold is in one group, and the rest of the features in the other.

When the algorithm then is to choose p features to consider at a split in the tree, it then selects $p_i(p_i < p)$ from the strong set, and $p_j = p - p_i$ from the weak feature set. A further advantahe of the stratified sampling method works well with high dimensional data with many features with less information [4].

The stratified sampling method is an option to the default setting in sklearn, random sampling. The random sampling chooses the features to be considered at each split at random.

3.2. ***AdaBoost Classification***

The AdaBoost algorithm uses weak learners called decision stumps (decision trees with a single split) to produce a strong learner. As the classification progress, AdaBoost uses information from the previous predictions to make new predictions. The new classifiers concentrate on instances not learned by the previous ones.

Similar to the Random Forest algorithm, AdaBoost is able to capture non-inearity, not captured by linear models. However, a challenge with AdaBoost is that it is not as good as random forest to handle imbalance in data. Because the minority class is prone to missclassification, one should think that the weights of the algorithm is able to handle imbalance. However, since the weighting strategy is optimized towards minimizing the total error rate this is not the case.

3.3. ***Implementation of a prediction procedure***

In order to test the performance of the two models, a prediction procedure was implemented. First a baseline prediction was done for both the Random Forest Classifier and the AdaBoost Classifier.

The procedure for AdaBoost is as follows: First do an imbalanced prediction on the full data set with a test set of 20% of the total data. Next, isolate the positive target examples and sample from the negative target examples in order to make predictions on a balanced data set

Table 7: Overview of the prediction procedure.

	Random Forest Classifier	**AdaBoost Classifier**
Baseline prediction	Random feature selection	Full prediction
Balanced prediction	Stratified sampling	Downsampling
Robustness testing	10-fold cross-validation	NA

(This technique is called down-sampling as the predictions are made for a considerably smaller data set).

For Random Forest, the baseline prediction was a prediction where the feature sampling for decision splits was done by random sampling from the feature space. Further, a balanced prediction was done on the full data set by using stratified sampling. To further test the robustness of the predictions, 10-fold cross validation on the stratified sampling prediction was done. Cross validation was not done for the AdaBoost classifier, as the down-sampling result in a form of cross validation. The algorithm samples from the negative target examples $round(totalnumberofexamples/numberofpositiveexamples)$ times. Hence, different negative examples are chosen each time. An overview of the prediction procedure is given in table 7

4. Results

This section present the prediction results of the different prediction procedures. Table 8 presents the confusion matrix of the different procedures, and table 10 presents the precision, recall, F1-score, AUC-score and accuracy of each prediction. Figure 11 gives an overview of the mean score for the different prediction procedures.

For both the Random Forest and Adaboost prediction, the baseline prediction (random sampling and baseline) receives higher accuracy than the RF stratified sampling and Adaboost downsampling methods. However, the confusion matrices in table 8 explains this result. Looking at the RF random sampling confusion matrix, we see that all examples are predicted to be negative. This result occur because the model is trying to achieve the highest accuracy as possible, and this accuracy is achieved by consequently predicting all examples as negative.

Table 8: Confusion matrix for the different predictions. For Adaboost, the last downsampled prediction was chosen

RF stratified sampling

	PN	**PP**
TN	39 210	17 403
TP	1 681	3 238

RF random sampling

	PN	**PP**
TN	56 613	0
TP	4 919	0

Adaboost downsampled

	PN	**PP**
TN	3 489	1 562
TP	1 557	3 325

Adaboost baseline

	PN	**PP**
TN	56 481	132
TP	4 807	112

The stratified sampling improves the Random Forest Classifiers ability to predict the positive examples in the data set. With stratified sampling, two thirds (66%) of the positive examples are classified correctly. However, the ability to predict a larger proportion of the positive examples comes at the cost poorer performance on the negative examples (0-examples). The result indicates that based on the current features, the model is still not fully able to distinguish the clients with repayment difficulties from the clients without repayment difficulties.

In order to test the stability of the Random Forest classification model with stratified sampling, 10-fold CV was used. For RF with stratified sampling, the average of 10-fold CV gets the same mean AUC score as the prediction on the full data set. There is also little variance in the performance, indicating that the model is able to produce robust predictions. However, the results also indicates slight over-fitting as the AUC score is slightly higher for the training set than for the test set.

There are also some interesting results comparing the RF stratified sampling method with the Adaboost down-sampling method. Both methods receive about the same mean AUC-score of approximately 68%. Though, there is an important difference between the predictions. As the confusion matrices show, the downsampling for the Adaboost prediction result in a significantly smaller data set. Usin the Adaboost model trained on the balance data set results in the same score as the baseline prediction. Hence the model trained on down-sampled data is not able to correctly predict on the actual distribution.

However, the Adaboost model is better at correctly predicting the positive examples (69% compared to 16%), however, this is under the assumption of a balanced data set.

In order to see how much each data set contributes to the prediction, each data set was successively added to each model. Thre results are shown in table 12. For Adaboost down-sampled, neither the installment data, nor the credit card data contributes significantly to the prediction. This might be because the negative examples are sampled at random for each generation. Hence, the procedure is prone to outliers or any differences in the negative sample. This is also probably why there is a negative change when adding the installment data.

However, for the RF stratified sampling, each data set contributes positively with a small amount, and the installment data contributes slightly more than the credit card data. The test of importance of each data set for Rf with stratified sampling is run on the whole data set with a test set of 20%.

Last, figure 6 show the relative feature importance of each of the features the green pillar

Table 9: Mean AUC-score from 10-fold cross validation using Random Forest Classifier

Random Forest	**Mean AUC-score**	**AdaBoost**	**Mean AUC-score**
RF random sampling	50.00 %	*AdaBoost baseline*	51.02%
RF stratified sampling	67.54 %		
Training data 10-fold	68.466 % ± 0.070 %	*AdaBoost downsampling train*	68.541 % ± 0.204 %
Test data 10-fold	67.670 % ± 0.490 %	*AdaBoost downsampling test*	67.870 % ± 0.538 %

Table 10: Evaluation metrices for the different predictions.

		Precision	**Recall**	**F1-score**	**AUC**	**Accuracy**
RF random sampling	**0**	0.92	1.00	0.96	0.500	0.920
	1	0.00	0.00	0.00		
RF stratified sampling	**0**	0.96	0.69	0.80	0.675	0.890
	1	0.16	0.66	0.25		
AdaBoost basline	**0**	0.92	1.00	0.96	0.5102	0.9197
	1	0.046	0.02	0.04		
Adaboost downsampling	**0**	0.69	0.69	0.69	0.6859	0.6859
	1	0.69	0.69	0.69		

Table 11: Mean AUC-score from 10-fold cross validation using Random Forest Classifier

Random Forest	**Mean AUC-score**	**AdaBoost**	**Mean AUC-score**
RF random sampling	50.00 %	*AdaBoost baseline*	51.02%
RF stratified sampling	67.54 %		
Training data 10-fold	68.466 % ± 0.070 %	*AdaBoost downsampling train*	68.541 % ± 0.204 %
Test data 10-fold	67.670 % ± 0.490 %	*AdaBoost downsampling test*	67.870 % ± 0.538 %

Table 12: The contribution of each dataset to the prediction.

	RF statified sampling	**Adaboost down-sampled**
Application data	66.98%	67.1502 % ± 0.3460 %
Installment data added	67.38%, 0.597 % change	67.1325 % ± 0.2710 %, -0.026% change
Credit card data added	67.54 %, 0.237 % change	67.870 % ± 0.538 %, 0.099 % change

represent the Adaboost down-sampled prediction, the blue pillar represent the RF stratified sampling with 10-fold CF prediction, and the red pillar represent the RF random sample prediction. Since there are 30 features, the expected feature importance if all features were equally important is 3.33%. However, some features stand out as significantly more important than the others. Further, the figure shows that for RF stratified and Adaboost downsampled, there is some variance in the importances between the runs, and Adaboost has slightly more variance than RF stratified.

For all prediction procedures, the most important features are the three EXT_SOURCE_X features. These features are raw features from the application data. However, their true meaning is concealed by Home Credit due to privacy constraints. Note that the RF predictions put more importance on these features than the Adaboost prediction. The feature importances of the Adaboost prediction is more evenly distributed.

Since the EXT_SOURCE features are concealed by Home Credit it is more interesting to look at some of the other features with higher importance. DAYS_BIRTH and DAYS_EMPLOYED do also have high feature importance, however they are also raw features from the application data. On the other hand, does some of the manually engineered features also gain some importance. This includes the mean number of drawings, and the mean time difference between the instalment payment due date and the actual instalment payment, as well as the mean fraction of the instalment payment value paid are also important for the prediction.

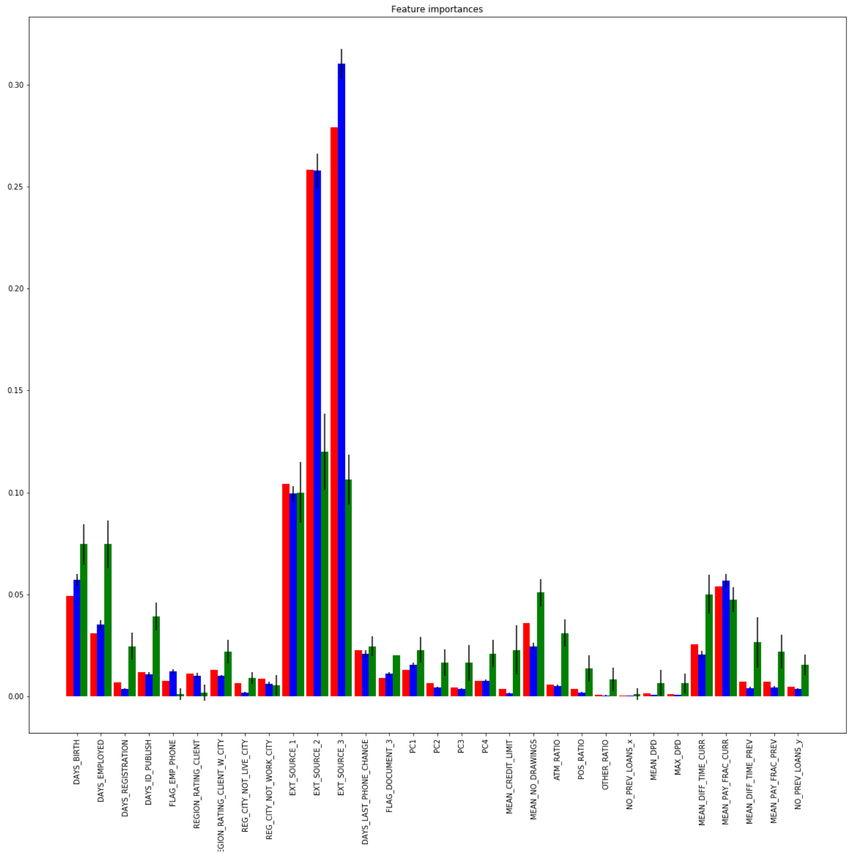


Figure 6: Feature importance for RF with random sampling (red), RF with stratified sampling (blue) and Adaboost downsampled (green).

5. Discussion

In this section, I will discuss some important discoveries made during the exploration of the Home Credit data set and the models to predict which clients have repayment difficulties.

The first point of discussion is the evaluation metrics. The RF stratified sampling model and the Adaboost down-sampling method gained an AUC socre of approximately 68%. This means that there is still a considerable amount of false positives and false negatives. For the RF stratified sampling method 17 403 clients out of 56 613 clients in the test-sample were falsely predicted as clients with repayment difficulties. Hence, the prediction is not precise enough to be used in a commercially.

Home Credit has two options for how they could treat clients which are predicted to have repayment difficulties. The first strategy could be to simply deny them a loan. However, this strategy would mean that they loose 17 403 possible clients which would have paid back their loan perfectly well. The next strategy would be to closer follow up the clients which are predicted to have repayment difficulties. However the models tested in this report is still not precise enough for this strategy either. With this strategy Home Credit would have to follow up 33% of their client base more closely. In addition 1 681 with repayment difficulties (0.03%of the clients) would not get any closer follow up.

Hence, an acceptable model for predicting repayment difficulties would accept some false positives (clients without repayment difficulties predicted as risky loan takers), and almost no false negatives (clients with repayment difficulties not labeled as risky loan takers). Given the closer follow up strategy, some clients would get a closer follow up than they need to be able to repay their loan. For the deny application strategy, there should be neither false positives nor false negatives to ensure that clients are not discriminated.

Further, testing the importance of each data set showed that (at least for) the RF stratified sampling method, feature engineering added more information to the model. However, the contribution from each data set was small. This shows that feature engineering helps, but is a tedious task. To succeed, one is dependent on both creating meaningful and impacting features. Hence, further feature engineering might have improved the model.

Next, the feature importance showed that creating meaningful features which also has business value adds value to this prediction problem. For some problems, the most important is the final prediction and its accuracy (measured in different ways). However, for prediction problems with business impact, it is also of interest which features are of importance to the predictions.

Hence, these features might be used as 'red flags'. However, basing business decisions based on single features should be done with caution because the features work together to make the final prediction.

5.1. Real world application challenges

Predicting future payment behaviour of clients based on the data provided by the Home Credit Group is related to some complicating factors. First, some of the variables provided by the applicant is not allowed to use for credit decisions in many countries. In example, deciding credit loans based on gender is discriminating. Hence, a lot of the features in the application data should possibly have been excluded in a real world scenario. As a data scientist, one should also be aware that techniques such as PCA do not remove features which might be discriminating, the technique simply do 'conceal' them and should therefore be used with caution.

Further, the Home Credit Group did not publish the time period in which the data was collected, hence we do not know the macro-economic factors which might have affected the payment behaviour of the clients.

In addition, basing the prediction on previous loans means that the amount of data in which each client is evaluated on differs. There will be much more data for a client with many previous loans in the bank compared to a client applying for his or her first loan. Hence, this is something which should also be considered in real world application of machine learning techniques for this type of problems.

6. Further Research

The main goal of further research should be to improve the models ability to correctly predict positive examples (and secondly to correctly predict the negative examples). Several paths should be considered to achieve this, the paths includes continued feature engineering, sophisticating pre-processing, parameter tuning and testing even stronger models.

*6.1. **Advanced feature engineering***

The Home Credit company also provide multiple other data sources for exploration, hence, a more precise prediction can possibly be made by adding features from additional data sets.

The repayment ability classification problem is partly a time series problem since the prediction is partly based on features generated from time series data. Further feature engineering of this problem might separate the most recent credit card behaviour from past credit card behaviour as the repayment ability might be more affected by recent behaviour than older behaviour.

6.2. ***Pre-processing and parameter tuning***

In this report down-sampling was used with the AdaBoost model. The challenge with this technique is that valuable information might be left out since the majority sample are reduced to the size of the minority sample.

Hence, other techniques which handle imbalance in data sets can be tested. Two such techniques are up-sampling and Syntetic Majority Over Sampling (SMOTE). Up-sampling is simply to make multiple copies of the minority target and thereby add significance to the minority variable. However, this technique might over-fit to the existing examples since it copies the minority examples to extend the data set.

The SMOTE technique is similar to up-sampling as it creates more examples of the minority class. Instead of copying examples in the minority class, it creates data points which lies around the existing data points in the class.

In addition to these pre-processing techniques. Parameter-tuning of the hyper-parameters might further improve the performance of the algorithms. For Random Forest this includes the number of threes in the forest, the depth of the trees, the number of leaf nodes etc. For Adaboost it also includes the number of trees as well as the learning rate.

6.3. ***Exploring XGBoost and LightBTM***

This report have used some of the standard machinlearning techniques provided by the sckit-learn library. However, there exist several other machine learning libraries with stronger predictive power. Two algorithms which have proven themselves many times are XGBoost and LightGBM. Both these algorithms are based on Gradient Boosting Decision Trees (GBDT) which combines the predictions from multiple trees by adding them together.

There are several advantages related to these algorithms. First, they are good at handling sparse data. In example, when one hot encoding categorical data, most instances will be zero. These algorithms suggest to ignore these zero-instances to reduce noise while predicting. In addition, these algorithms also has hyper-parameters for handling imbalanced data.

7. Conclusion and further research

The purpose of this project was to predict the repayment ability of loan applicants to the Home Credit bank. This however, proved to be a challenging task. This report shows that techniques to handle the imbalance of the data is crucial for meaningful predictions. In addition, feature engineering to find features that contributes to correctly identify the clients with repayment difficulties is important to further improve the prediction

References

[1] H. Credit, Our products, 2019. URL: `https://www.homecredit.net/about-us/our-products.aspx`.

[2] H. Credit, Welcome note from home credit, 2018. URL: `https://www.kaggle.com/c/home-credit-default-risk/discussion/57054`.

[3] G. James, D. Witten, T. Hastie, R. Tibshirani, An introduction to statistical learning, volume 112, Springer, 2013.

[4] Y. Ye, Q. Wu, J. Z. Huang, M. K. Ng, X. Li, Stratified sampling for feature subspace selection in random forests for high dimensional data, Pattern Recognition 46 (2013) 769–787.

8. Appendix

Table 13: Mean, variance, max and mind of the numerical features.

	mean	std	min	**max**
TARGET	0.0807	0.2724	0.0000	1.0000
DAYS_BIRTH	-16 037	4 364	-25 229	-7 489
DAYS_EMPLOYED	63 815	141 276	-17 912	365 243
DAYS_REGISTRATION	-4 986	3 522	-24 672	0
DAYS_ID_PUBLISH	-2994	1509	-7197	0
FLAG_EMP_PHONE	0.8199	0.3843	0.0000	1.0000
REGION_RATING_CLIENT	2.0524	0.5090	1.0000	3.0000
REGION_RATING_CLIENT_W_CITY	2.0315	0.5027	1.0000	3.0000
REG_CITY_NOT_LIVE_CITY	0.0782	0.2684	0.0000	1.0000
REG_CITY_NOT_WORK_CITY	0.2305	0.4211	0.0000	1.0000
EXT_SOURCE_1	0.5021	0.1394	0.0146	0.9627
EXT_SOURCE_2	0.5144	0.1909	0.0000	0.8550
EXT_SOURCE_3	0.5109	0.1745	0.000527	0.8960
DAYS_LAST_PHONE_CHANGE	-963	827	-4 292	0
FLAG_DOCUMENT_3	0.7100	0.4538	0.0000	1.0000
PC1	0.0000	4.1998	-12.6855	71.3886
PC2	0.0000	2.1911	-18.1865	29.8411
PC3	0.0000	1.7815	-7.7287	64.5897
PC4	0.0000	1.7293	-3.0326	36.0679
SK_ID_CURR	278 181	102 790	100 002	456 255

Table 14: Mean, variance, max and mind of the installment features.

	Count	Mean	Std	Min	Max
SK_ID_CURR	339 762	278 159	102879	100001	456255.000000
MEAN_DIFF_TIME_CURR	339 762	-11.87	15.50	-1 600	2 406
MEAN_PAY_FRAC_CURR	339 762	1.0070	0.1736	0.0000	5.4851
MEAN_DIFF_TIME_PREV	339 762	-8.12	16.99	-1 008.75	1 877.40
MEAN_PAY_FRAC_PREV	339 762	0.7081	0.4497	0.0000	3.5024
NO_PREV_LOANS	339 762	2.94	2.05	1	26

Table 15: Mean, variance, max and mind of the credit card balance features.

	Count	**Mean**	**Std**	**Min**	**Max**
MEAN_CREDIT_LIMIT	103558	207 320	190 229	0	1 350 000
MEAN_NO_DRAWINGS	103558	1.51	3.65	0	118.25
ATM_RATIO	103558	0.4550	0.4450	0	1
POS_RATIO	103558	0.1887	0.3293	0	1
OTHER_RATIO	103558	0.0348	0.1593	0	1
MEAN_DPD	103558	4.1	44	0	1 636
MAX_DPD	103558	16.4	141	0	3 260