VRIJE UNIVERSITEIT AMSTERDAM

RESEARCH PAPER

Challenging LGD models with Machine Learning

Luc Severeijns

supervised by
Prof.Dr. Sandjai BHULAI

July 30, 2018

Preface

This internship report was written as the final part of the Master's program in Business Analytics at the Vrije Universiteit Amsterdam. The goal of the Master's program in Business Analytics is to improve business performance by applying a combination of methods that draw from mathematics, computer science, and business management. The internship was performed at ING. During this internship, I have focused on identifying risk drivers for LGD with the goal to improve the currently used models. The present thesis reports the results of my internship. I would like to thank ING for giving me the opportunity to complete my thesis. I would also like to thank Rikkert Hindriks for being my second reader. Finally, I want to give special thanks to my internal supervisors Maxence Lavalle, Yildiz Dündar, Svetlana Polenkova and VU supervisor Sandjai Bhulai. They have provided me with great guidance throughout the internship and useful feedback on the process and the report and I am very grateful for that.

Luc Severeijns
Amsterdam, August 2018.

Contents

Abstract

Currently, at ING a new LGD methodology is tried to better estimate their loss given default (LGD). The LGD is the loss of revenue due to customers which default on their mortgage. If the LGD is excellently modeled then ING can allocate their capital the optimal way, also the ECB will be very pleased. The difference in methodology is the shift of portfolio level to customer level. Now a vintage analysis is done to obtain the overall average of the mortgage portfolio. A vintage analysis is fast and gives easily interpretable results but does not distinguish customers based on their characteristics. According to the vintage analysis a customer which is 3 months in default has the same chance of curing as a customer which is 48 months in default. In this research, we will try to model the LGD based on customer characteristics instead of long-term averages. With the help of machine learning, we will try to estimate the LGD to see if this customer level approach can be used.

To estimate the LGD we will model two components: cure rate & recovery rate. The Cure rate gives the probability that a customer who has defaulted starts performing again. The Recovery rate gives the rate of the recovered amount of a non-cured-customer, so how much did ING recover when a customer is written off, in percentages. The LGD formula:

$$LGD = \frac{(1 - \text{cure rate}) * (1 - \text{recovery rate})}{EAD} + \frac{\text{costs}}{EAD}$$

EAD stands for 'Exposure at Default'. This is the amount a client currently owes ING This research contains 3 cases/scenario's: LGD through the cycle (through the cycle), LGD in default and LGD in downturn. The difference between LGD through the cycle and LGD in default is the customer characteristic/variable *Months in default*. With this variable we can predict the probability of cure/recovery of a customer per month. LGD through the cycle is to evaluate the overall performance of machine learning modeling and LGD in default is to predict future probabilities. The ECB obliges that ING has a downturn approach as well, therefore we have a third case/scenario LGD in downturn. This requires creating a model during downturn period. For all of these three cases, we are especially interested in the variables that predict the LGD the best. The best predictors are called Risk Drivers and are key points to watch for ING. First the variables are rated by a c-statistic, information value and Spearman rank coefficient. Thereafter, the variables are put in the machine learning algorithms to assess their predictive and discriminative power. At the end, we will conclude which are the Risk Drivers per component.

During this research, we will use five machine learning algorithms: Logistic/linear Regression, Decision Tree, Random Forest, Gradient Boosting and Neural Net. We will apply all five to each case and compare them afterward. We will compare them on performance but will also pay attention to the hyperparameters and important features. We see different results in algorithms for each case but usually Random Forest and Gradient Boosting outperform the other algorithms.

We end this research by evaluating if this new methodology: LGD modeling on customer level with machine learning is a good replacement for the current methodology. We conclude that this new methodology is a good replacement.

1 Introduction

With the sharpened rules regarding bank capital by the ECB, all banks are keen on optimizing their capital-models to get competitive advantage and avoid penalty. Loss given default (LGD) is an important factor in capital costs. The higher the LGD of a bank the more capital a bank, has to hold and thus less money to invest. An improved estimation of LGD leads to a healthier and risk-less allocation of capital. To estimate the LGD, ING uses the following formula:

$$LGD = \frac{(1 - \text{cure rate}) * (1 - \text{recovery rate})}{EAD} + \frac{\text{costs}}{EAD}$$

The goal is to better estimate these LGD components, cure rate and recovery rate, and get a more accurate estimate of the LGD.

To aim for better performance in LGD modeling, measuring current risk drivers and identifying new risk drivers of the LGD components is important and cannot be overemphasized. Risk drivers are metrics that ING include in their models which are of value in estimating the LGD. Evaluating current and new risk drivers will give more insight into the prediction power of these drivers. ING will know what drives their LGD performance.

There are already some basic risk drivers ING uses such as: NHG and EAD. NHG shows whether a client has collateral and EAD is the Exposure At Default. ING has done a lot of research in finding the right risk drivers. However, this research did not include machine learning techniques. All research so far has been done to estimate LGD components by deriving the long-term averages. In this research we will focus on the machine learning forecast of the risk drivers. This forecast will help ING retrieve insights and get the competitive edge over other banks in LGD modeling.

The scope of this thesis is challenging the current LGD models used by ING with machine learning. Next to providing a challenger model, this will include setting up an easy-to-use flow diagram in Enterprise Miner (SAS), downturn adjustments and visualization of variable importances. The simple long-term average model, which is the current model that is used for the prediction of LGD, is not accurate enough. The ECB will give penalty to banks that cannot estimate precisely so the predictions should be as accurate as possible.

In this research, the goal is to improve the prediction of the LGD using historical data. However, since the prediction will be reviewed by the ECB we will encounter restrictions that makes it challenging. One of these restrictions is the margin of conservatism, even if the created model would estimate all cases 100% correctly. The ECB will still demand that we estimate the LGD more conservatively because unexpected events can happen. Also the ECB demands a different model in case of economic downturn.

This data consists of mortgage data on client level of two databases. The largest database is the database of ING itself and the second database is of Westland Utrecht, a bank that ING has engulfed. This data is aggregated at customer level which means some data is lost, we will deal with this in the 'Data' section. This leads to the following research question: With how much can the forecast of LGD be improved compared to the current model using machine learning and sharpened rules by the ECB?

The structure of the thesis consists of eight chapters. The first chapter is the current introduction. Chapter 2 provides background information about ING, the risk drivers, LGD formula and the current model used in the bank now. Next, Chapter 3 gives a literature review about the common models used for time series forecasting and their application in forecasting risk drivers. Chapter 4 describes the data used followed by their pre-processing and analysis. Chapter 5 describes the methods and models that have been applied to the forecast. Chapter 6 describes the experimental setup and Chapter 7 presents its results. Lastly, chapter 8 reports the discussion about the results, recommendation for future research and the conclusion.

2 Background

2.1 Company Description

ING is a bank which focuses on innovation. ING empowers people to stay ahead in life and business. The department predictive analytics is focused on identifying risk cases in early stages and predicting numbers based on historical data. ING was founded in 1991 by a merger between NMB Postbank Group and Nationale-Nederlanden. During the past years, ING has become a multinational with diverse international activities

2.2 New methodology

ING wants to update their LGD methodology due to stronger regulations and more possibilities in data analytics.In the current model only models the cure rate and recovery rate through the cycle. In the new methodology we add in default modeling and in downturn modeling. The big change in the new methodology is the dataset we model on. The dataset of the current cure rate model contains all resolved cases (non dragging cases) + an estimation of the unresolved cases (dragging cases). Resolved cases are cases of which we already know the outcome at the end, so we know whether the client did cure in the end or did not. We could say that resolved cases is labeled data where unresolved cases are unlabeled data. The estimation of the unresolved cases is a migration matrix based on recent data (last 2 years). The recovery rate is divided into three sub recovery rates, primary, secondary and unsecured based on what is recovered of the customer. The new method takes another approach for both rates. The recovery rates are added together to form one recovery rate. For the estimation of unresolved cases a vintage analysis is done to calculate a maximum work out period. The vintage analysis calculates the long run average conditional probabilities of curing. All these probabilities are plotted to get a conditional cure rate. This cure rate is inverted to get a cumulative cure rate as illustrated below.

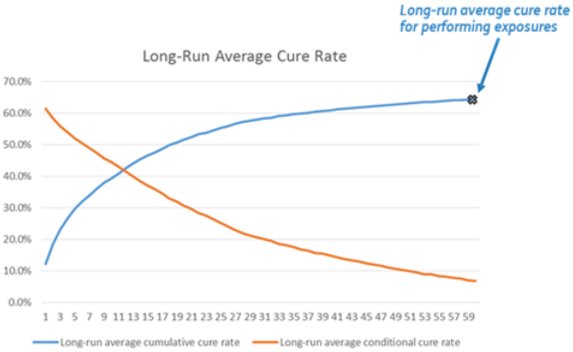


Figure 1: Long run average cure rate

The maximum work out period represents the number of months after which cumulative cure rate does not change (much) anymore. After a customer passes the maximum work out period it will automatically be written off since the chance of curing will be very small. This approach classifies customers based on months in default but not yet other characteristics. We will challenge the new methodology with machine learning based on risk drivers to see whether this is a good alternative to estimate the probability of cure and probability of recovery.

2.3 Scoping

The LGD team at ING made a very clear graph of the new methodology:

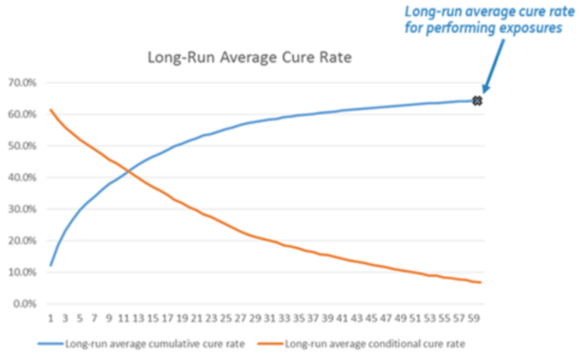


Figure 2: Long run average cure rate

The graph "Average observed LGD" at the left, this is the average LGD of the database of the resolved cases. We add the unresolved estimation to this to get the 'through the cycle' dataset also referred to as performing dataset. The whole second column can be seen as the performing part in modelling the LGD. In the third column we add post-default information and this is referred to as in default modelling. The scope of this research will be the green marked boxes.This will be the LGD TTC and LGD in downturn for the performing dataset and the LGD in default for the in-default dataset. We will model these datasets with machine learning models to observe if the LGD can be modeled well.

2.4 Definition of Default (DoD)

In the banking world, it is important to have clear rules about when a client is defaulted to prevent misalignment. A client is considered to be in default if payments on any mortgage under the same label are in arrears at least 3 terms. The default status is only lifted if all overdue payments have been repaid. As a result a client is in default if:

- the previous month the client was not in default and this month an arrear of at least 85 days exists, or
- the previous month the client was in default, and this month any arrears (also 1 or 2 months) exists, or
- the client has a loan modification (rating 19.9).

Indirect defaults occur when a client has more than one mortgage under the same label. If the client defaults on one mortgage, all mortgages of this client are in default. Indirect defaults are limited to mortgages under the same label.

Other terms Other key terms we will encounter during this research:
dragging: the mortgage of a client has not matured yet. This we will call dragging cases.
exit no loss: the client has been written off and ING recovered 100% of the outstanding amount.
exit with loss: the client has been written off and ING recovered less than 100% of the outstanding amount.
cured: A client was defaulted in the past but is now performing again (healthy again).
written off: A client which will not be able to cure again and ING starts the recovery procedure.
workout period Time it takes to deal with a client at the moment the recovery traject starts. This date is also called the order date. **resolved cases** records of client which are already closed (non dragging cases).

2.5 LGD formula

To compute LGD, ING uses the following formula:

$$LGD = \frac{(1 - \text{cure rate}) * (\text{EAD} + \text{additional drawings after default } \sum_i \text{discounted recoveries i})}{EAD} + \frac{\text{costs}}{EAD}$$

LGD is built up out of 4 components, EAD, Cure Rate, Recovery rate and additional costs. EAD is obtained in the database of ING itself and does not have to be calculated, $\frac{costs}{EAD}$ we will take as a constant. So we will need to estimate two components: cure rate and recovery rate.

Cure rate
The cure rate is defined as the percentage of clients which were in default but paid their debt back and are at the moment not in arrear anymore (healthy/performing clients). These customers we see as "cured" ergo we call the rate of these clients the cure rate. A client can default again, after being cured. In the data, we look at the last known record of this client whether the client is cured or not at the moment.

Recovery Rate
Recovery comes into play when a client is written off. The recovery rate is the recovered amount divided by the EAD (per client). We look at mortgage data so the collateral of the mortgage will be a house for example. The recovered amount will be the selling price of the house. It could be that the recovery rate is above 100% when the house gets sold for more than the client is in debt for. ING cannot keep this profit and will return this to the client. That is why we will cap the recovery rates at 100%.

2.6 Risk drivers

Risk drivers are the explanatory variables of the components of the LGD formula, Cure Rate and the Recovery Rates. We are going to estimate these factors with machine learning and the risk drivers are the variables in the model. ING already has a set of risk drivers, we will assess these risk drivers and possibly add more.

2.7 Margin of Conservatism (MoC)

During this research, we will be talking about conservatism. This means to install extra caution to be safe when a model does overestimate a probability of cure or recovery. Throughout the whole research we will be conservative when a choice presents itself. For example, when deciding to fill a missing value. We will choose a value which is more conservative than the average or if we have to fill a boolean we will always pick the most conservative one. In the end, we will see if we have to put an extra margin of conservatism on our model or if the model itself is conservative enough.

3 Literature review

The prediction of the risk drivers using their past observations relying on rules-based programming is known as statistical modeling. In this research, we will be using machine learning which does not rely on rules but creates an algorithm which invents own rules to group clients and estimate their rates. The aim of machine learning modeling is to collect and analyze the past observations of a dataset to develop an appropriate model that describes the structure of this dataset. This model is then used to forecast future values for new data. Thus, machine learning forecasting refers to the process of predicting future probabilities based on past observations using the characteristics of the data.

A popular machine learning model is the Logistic Regression. It is popular because of its flexibility and simplicity to represent different varieties of data. However, a major limitation is the assumption that the target variable has to be binary or categorical, LR requires distributional assumptions, independent observations and no multicollinearity. These assumptions do not hold in many complex data problems. Therefore, non-linear models are proposed such as neural networks. The major benefit of neural networks is their capability of flexible nonlinear modeling. It is not needed to specify a particular model form because the model is adaptively formed based on the features presented from the data.

These models have been applied in forecasting home equity. T. Slingerland & A. Pengel[6] have applied all kinds of different machine learning algorithms to predict home equity. The analysis has shown that ensemble methods and neural networks perform best for forecasting, especially in those series without obvious pattern. The models were trained on ING data in the Netherlands in 2017.

The machine learning algorithms we are going to use to challenge current models with quick description are given below:

Decision Tree Decision tree methods partition the feature space into regions and then fit a simple model in each one. Based on a splitting rule, the tree depicts the first split into pieces as branches arising from a root and subsequent splits as branches arising from nodes on older branches. The leaves of the tree are the final groups, the un-split nodes or terminal nodes.

Advantages

- No distributional assumptions
- Simple method
- Robust to outliers
- Can handle missings or irrelevant inputs
- Interpretability of the results and provides decision rules
- Nice visualization
- No multicollinearity issues.

Disadvantages

- Easily leads to overfitting of the data.
- At each step the best split for that step is made but not for the best ultimate tree.
- In general performs worse than the random forest method

[14]

Logistic Regression Logistic regression attempts to predict the probability that a binary target will acquire the event of interest as a function of one or more independent inputs. It is a linear regression model where with use of a transformation the dependent variable(target), which is a probability, is confined to values between zero and one: $y = f(\beta_1 x_1 + ... + \beta_n x_n) + e$, where f stands for the logistic function,β stands for the unknown parameter of interest that represents the unknown part of the relationship between the inputs and outputs, x stands for the input features and e is the corresponding error term. Parameter estimates are obtained by use of Maximum Likelihood.

Advantages

- Clear interpretation of relation features and resulting target.
- Has a good performance for a relatively simple method
- Confidence Bounds for forecasts.

Disadvantages

- Cannot easily deal with missing values.
- Susceptible to outliers.
- Susceptible to multicollinearity issues.
- Requires distributional assumptions and independence of observations.

[15][16]

Random Forest A Random Forest is an ensemble of decision trees that differ from each other in two ways. Firstly, each tree is trained on a different artificially created sample. Second, the input variables that are considered per tree for splitting a node are randomly selected from all available inputs. In other respects, trees in a forest are trained like standard trees. In the first step of the algorithm the bootstrap samples are drawn from the in-bag training data to create the different training sets for the individual trees in the forest. This procedure is called bagging which is short for “bootstrap aggregating”. The observations of the training data that will be used to create the bootstrap samples are called the bagged (or in-bag) observations, whereas observations that are excluded from the sampling are referred to as the out-of-bag (OOB) observations.

Advantages

- Deals well with nonlinear relations between input and output.
- Less susceptible to influence of outliers.
- Does not require independence of observations.
- Can easily deal with missing values.
- No multicollinearity issues.
- No distributional assumptions.

Disadvantages

- No clear interpretation for relation features and resulting target.
- Can be slow due to training a lot of trees.

[8]

Gradient Boosting Boosting is a procedure that combines the outputs of many weak classifiers (usually Decision trees, DCTs) to produce a powerful ensemble. Boosting bears a resemblance to bagging, however there are still major differences between the two techniques. The purpose of boosting is to sequentially apply the weak classification algorithm to repeatedly modified versions of the data, thereby producing a sequence of weak classifiers. The predictions from all of these classifiers will be combined and based on a weighted majority vote the final prediction is produced. Gradient Boosting is a special case of boosted tree models. A boosted tree model is a sum of decision trees induced in a forward stagewise manner. Forward stagewise boosting is in general a very greedy algorithm. That is where gradient boosting comes in. Gradient boosting involves three elements: A loss function to be optimized, a weak learner to make predictions and an additive model to add weak learners to minimize the loss function. The loss function used, depends on the type of problem being solved. For classification,one can use logarithmic loss function. As a weak learner a decision tree model is used. For each iteration the gradient of the loss function will be minimized and the next decision tree model will be fit on the residuals of the model one iteration ago. The predictions of all the models are added and a weighted majority vote will decide the classification.

Advantages

- Deals well with nonlinear relations between input and output.
- Less susceptible to influence of outliers.
- Does not require independence of observations.
- Can easily deal with missing values.
- No multicollinearity issues.
- No distributional assumptions.

Disadvantages

- GBDT training generally takes longer compared to random forest because of the fact that trees are built sequentially.
- More prone to overfitting than Random Forest
- No clear interpretation for relation features and resulting target.

[9][10]

Neural Network A Neural Network is a multiple stage nonlinear regression model that is typically represented by a network diagram. The central idea is to create derived features from the given input features and use these in a regression model, whereas in standard regression models the output is directly modeled as a function of the input variables.

Advantages

- Deals well with nonlinear relations between input and output.
- Builds forth upon regression techniques, so should have at least the same predicting power.

Disadvantages

- No clear interpretation of the relation between original features and target.
- Numerical errors may occur for complex networks.
- Missing values must be replaced prior to applying the model
- Susceptible to outliers.
- Needs a lot of data preparation

[13][12]

In this research we will use: Logistic Regression (LR), Decision Tree (DCT), Random Forest (RF), Gradient Boosting (GB) and Neural Net (NN). For the recovery rate we will use linear regression instead of logistic regression because our target variable is continuous.

4 Data

This section describes the data and its pre-processing. Moreover, data analysis will be performed.

4.1 Data Dictionary

First we started by inspecting the datasets, due to lack of documentation there was no description of the data yet. We provided a data dictionary by simply describing all the variables in the dataset. We provided all the names of the variables, the source of the variables and the description of them. If the variable was created next to the basic variables then also the method is provided. The results are in the appendix.

4.2 Cure rate and Recovery rate data

ING has different databases spread over multiple enterprises. The databases we will use are built up from the Vortex database, the Auction database and the Westland-Utrecht database. Vortex is the main database of ING for mortgages, the Auction database contains records of auction data for the collateral of the mortgages and the Westland-Utrecht database is a database of an engulfed company. These databases are combined into four main data sources: To estimate the two components we will use the respecting data source. Each database contains records on client level. Every data source contains a different target variable, for the cure rate this target variable is binary, cured or not cured, for the others this is a continuous variable, a rate. The recovery rate is split into three subcategories by the bank. the primary recovery rate which is the rate/amount the bank recovers from the collateral of the mortgage (usually a house). The secondary recovery rate is the rate of all other possessions and the unsecured recovery rate is the amount recovered which is insured by government or cash of the client. Most of the variables in the data sources are the same, some extra variables are added for the rates to have more information about the recovery. Notice that the records/clients in the data sources are not the same. The primary recovery rate can only have records where a client did not cure, because then the recovery traject will be started. The same holds for secondary recovery rate and the unsecured recovery rate. We will have different cases for every data source. The cure rate data source contains:

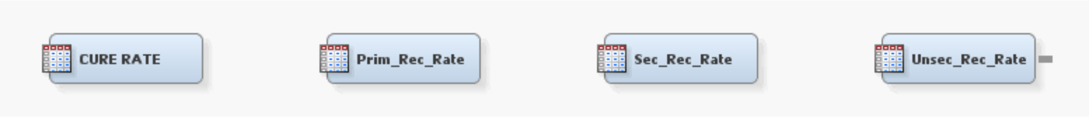


Figure 3: databases

Cure rate	
Variable name	Description
AGE_curr	Current age of client
DATE_FROM	Default date of client
DATE_WO	Write off date
DEFAULT_L12M	Flag if client was also in default 12 months ago
DEFAULT_L3M	Flag if client was also in default 3 months ago
DEFAULT_L6M	Flag if client was also in default 6 months ago
DELQ_L12M	Shows in how many months the client was delinquent last 12 months
DELQ_L3M	Shows in how many months the client was delinquent last 3 months
DELQ_L6M	Shows in how many months the client was delinquent last 6 months
EAD	Exposure at default
EAD12	Exposure at default 12 months ago
IR	Interest rate
ITI	Income to Interest ratio
LAST_DEFAULT	Last default moment of client in months
LAST_DELQ	Last delinquent moment of client in months
LTV_curr	Current loan to value
MONTH_limit	Number of months between the limit of the mortgage was updated
NHG	Flag if the mortgage has collateral
OS	Outstanding debt
OS_init	Initial outstanding debt
RATING_new	Rating ING gave to client
RATING_new12	Rating ING gave to client 12 months ago
RESIDENTIAL	Flag if the collateral is habitable
URBANISATION	Flag if the collateral is in a big city
arrears	Amount the client is in arrears
auction_costs	Costs of auction
auction_revenue	Revenue of auction
cure	Binary target variable, cured or not
current_instalment	Current installment payment due
customer_id	ID of customer
date_cancel	Date of auction cancellation, only when auction is cancelled
debt_on_auction_date	Debt of the client on auction date
euribor_3	Euribor rate
fix_interest_nom	Flag if the mortgage has a fixed interest
last_default_event	Date of last default event
limit_start_year	Year the limit of the mortgage started
loan_status	Status of the loan
loss_auc	Flag if there was a loss at the auction
marital_status_code	Relationship of client
months_workout	Workout period in months
order_date	Date of ordering auction (start workout period)
outdef_healthy	Flag if client is healthy/performing/cured
outdefault_date	Out of default date
outdefault_number	Continuous variable of outdefault_date
postal_code	Postal code of collateral
prd_bridge	Flag if mortgage is a bridge loan
prd_interest_only	Flag if mortgage is an interest only loan
prd_investment	Flag if mortgage is an investment loan
prd_life	Flag if mortgage is a life loan
prd_saving	Flag if mortgage is a savings loan
product_type_code	Code of product (bridge, interest,...)
remain_loan_period	Remaining period of mortgage
source	Database source (vortex, auction, Westland Utrecht)
system_id	Id of which enterprise the mortgage belongs to
writeoff_amt	Amount of write off

Additional variables for the recovery rates are:

Recovery rate	
Variable name	Description
auction_costs_PPR	Auction costs for primary recovery rate
auction_costs_SRR	Auction costs for secondary recovery rate
auction_costs_unRR	Auction costs unsecured recovery rate
disc_auction_revenue	Discounted auction revenue
disc_period_M	Discount period in months
disc_revenue_cons	Discounted revenue calculated conservative
disc_revenue_optim	Discounted revenue calculated optimistic
mortg_market_value	Market value of the collateral
mortg_market_value_WO	Market value of the collateral at write off date
rec_rate	Recovery rate

We will aggregate the different recovery datasets to one dataset: recovery rate. The target variable of all three datasets will also be updated to one new target variable: rec_rate.

4.3 External data

Besides the four data sources we have got an arrear database. This database keeps the number of arrears per month per client. So this database is on month-level. We have got the total number of arrears in the main data sources but due to aggregation to client level this information is lost, with the arrear database we can salvage this information. The variables of the arrear database are roughly the same but are per month.

4.4 Final dataset

For both components we will use a different final dataset to model it. The number of records we will use for the cure rate is 67,300 for the recovery rate it is 12,478. Both these numbers represent the performing datasets, the in-default datasets will be a lot bigger and both be around 1.2 million rows.

4.5 Data pre-processing & Data Analysis

To gain insight into the different variables in the database we will plot the data points of every variable versus the target variable, also the descriptive statistics are calculated per variable. Next we calculate the information value and c-statistic per variable and based on the univariate analysis we will transform or bin variables.

univariate analysis The univariate analysis is to visualize how the variable explain the target variable. For numerical variables the data points are drawn and we try to fit different lines through this data. The lines we try are: linear, square, log, exponential and cubic. All lines are compared to show the best fit. We need to examine if the best line is also the best fit for the respective variable. It can happen that a cubic line is the best fit but it does not make sense for that variable, this is usually caused by outliers. An example of the variable: *current Loan-To-Value*

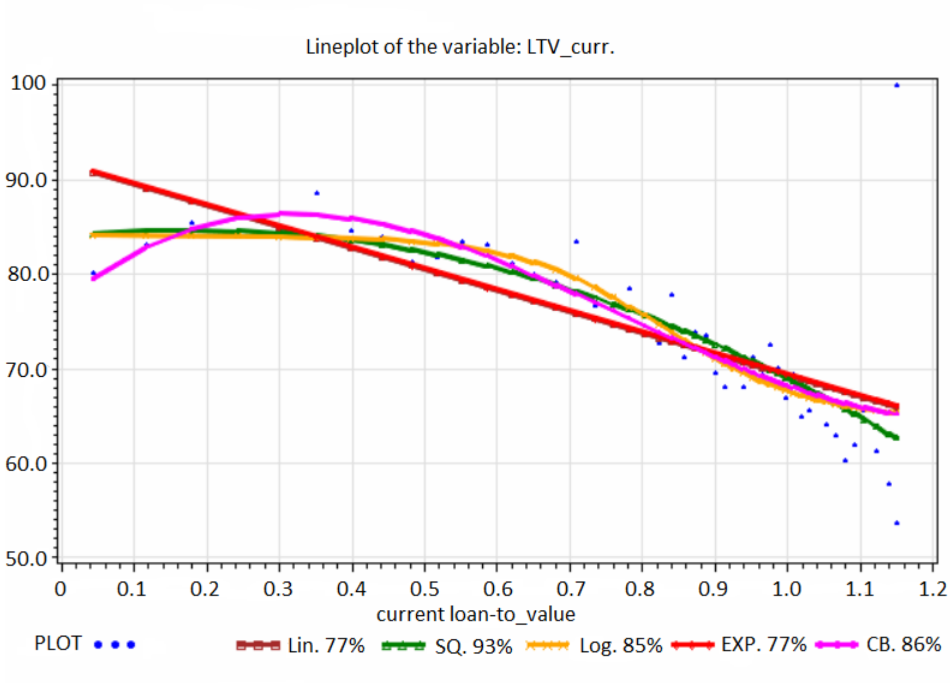


Figure 4: Lineplot of variable: current Loan to value

The blue dots are the datapoints of the ’LTV curr’ (current Loan-To-Value) values. The colorful lines are the fitted lines. In this example we see that a square line (green) is the best fit. When we inspect the plot we have no reason to suspect otherwise also it seems logic that if the current LTV is high then the chance to cure is lower. An example of the recovery rate:

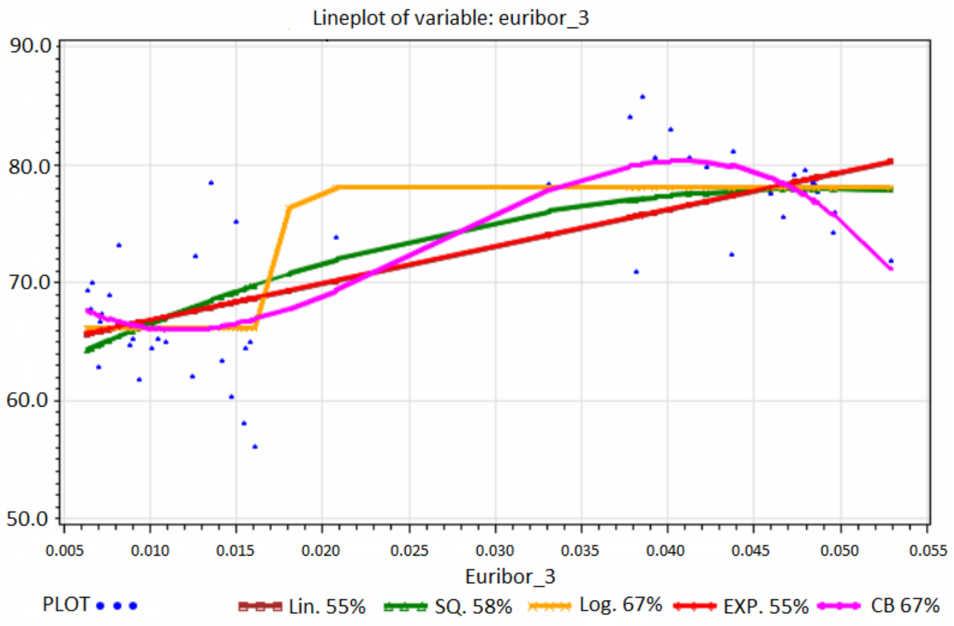


Figure 5: euribor 3 months

This plot is more tricky. The best fit is log (yellow) here but this does not seem as a natural line through the data. Euribor is an interest rate between banks. The duration of this euribor is three months so a lot of records have the same value for this variable. This makes the variable rather more discrete than continuous. We have no reason to assume a qubic relation between the recovery rate and euribor, it is more likely that euribor has a linear relation. Due to little difference in the datapoints it is hard to conclude which transformation to use (if needed). Maybe it is even better to bucket this and make a categorical variable, this will be experimented with during the modeling phase.

missings + outliers In this paragraph we will only discuss the missings and outliers of the variables that were included in the final model.The variables included in the model for the cure rate are:

- Delq_L12M, shows in how many months the client was delinquent last 12 months
- system_id, id of which enterprise the mortgage belongs to
- euribor_3, euribor rate
- limit_start_year, year the limit of the mortgage started
- remain_loan_period, remaining period of mortgage
- LTV_curr, current loan to value
- ITI, income to interest ratio
- _freq_, number of months client is in arrears database.
- same, #times client paid late - #times client paid back.
- fastrecover, #months of previous default period
- Speedcure2, #bucketed fast recover: fast, average, slow, no recover yet.
- diff_EAD, EAD12 - EAD.
- changeEAD, 1 if EAD12 - EAD > 0, 0 if EAD12-EAD = 0, -1 if EAD12-EAD < 0.

- LTV_risky1, Flag if LTV_curr > 1.
- arrearsflag, Flag if last record of client is an arrear in arrear database.

Missings + outliers			
Variable	type	missings	outliers
Delq_L12M	basic	0	0
system_id	basic	0	0
euribor_3	basic	0	0
limit_start_year	basic	0	0
remain_loan_period	basic	0	0
LTV_curr	basic	1757	3
ITI	basic	2811	60
freq	created	0	0
same	created	0	0
fastrecover	created	0	0
Speedcure2	created	0	0
changeEAD	created	5515	0
diff_EAD	created	5515	0
LTV_risky1	created	0	0
arrearsflag	created	0	0

The final dataset we are using contains 67,300 records. We do not want to delete records because of one of the risk drivers is missing so we will fill the missing values of ITI and LTV_curr with their mean. changeEAD and diff_EAD are created and contains missings because EAD and EAD 12 contains missings (the basic variables which they were created of). We will replace the missings of EAD and EAD12 with their means. By replacing these values we also remove the missing values in changeEAd and diff_EAD.

For the outliers we need to look more closer at the records. If we look at all records of ITI bigger than 2 we have 95 records. We can immediately see the outliers since there is a big difference between these values. The biggest reasonable record is 7.66. After 7.66 we only see values in 7000, 8000 or 9000. We decide that values above 7.66 are outliers and are deleted from the dataset. This results in 60 records deleted. For LTV_curr we only see three strange values in the dataset and all three are 9999.99. We decided to also delete these records from the dataset.

The records we are using for the cure rate and recovery rate are different. For the recovery rate we only use 12387 records, the datasets are already pre-made and can contain different clients. Therefore we will need to analyze the variables again even though a couple might already have been analyzed for the cure rate. For the recovery rate we use the following variables:

- Delq_L12M, shows in how many months the client was delinquent last 12 months
- system_id, id of which enterprise the mortgage belongs to
- euribor_3, euribor rate
- limit_start_year, year the limit of the mortgage started
- remain_loan_period, remaining period of mortgage
- LTV_curr, current loan to value
- ITI, income to interest ratio
- NHG, flag if the customer has collateral on his mortgage.
- same, #times client paid late - #times client paid back.
- fastrecover, #months of previous default period
- Speedcure2, #bucketed fast recover: fast, average, slow, no recover yet.
- arrearsflag, Flag if last record of client is an arrear in arrear database.

- month_limit, Flag if last record of client is an arrear in arrear database.
- prv_kort, Flag if last record of client is an arrear in arrear database.
- diff_EAD, EAD12 - EAD.
- current_instalment, Flag if last record of client is an arrear in arrear database.
- EAD, Exposure at default of client
- AGE_curr, current age of client.

Missings + outliers			
Variable	type	missings	outliers
Delq_L12M	basic	0	0
system_id	basic	0	0
euribor_3	basic	0	0
limit_start_year	basic	0	0
remain_loan_period	basic	0	0
LTV_curr	basic	235	0
ITI	basic	437	10
NHG	basic	172	0
same	created	0	0
fastrecover	created	0	0
Speedcure2	created	0	0
month_limit	created	0	0
prv_kort	created	0	0
diff_EAD	created	879	0
current_instalment	created	27	0
arrearsflag	created	0	0
EAD	created	27	0
AGE_curr	created	3	0

For LTV_curr, ITI, diff_EAD we apply the same methods as for the cure rate. Also we replace missing values of the variables, current_instalment, EAD, AGE_curr with their respective means. The only variable with a different approach is NHG, which is a flag (boolean) and represents a 0 or 1. Since we only have 12387 rows we do not want to delete any more records. We take a look at a bubble plot of NHG plotted against the recovery rate.

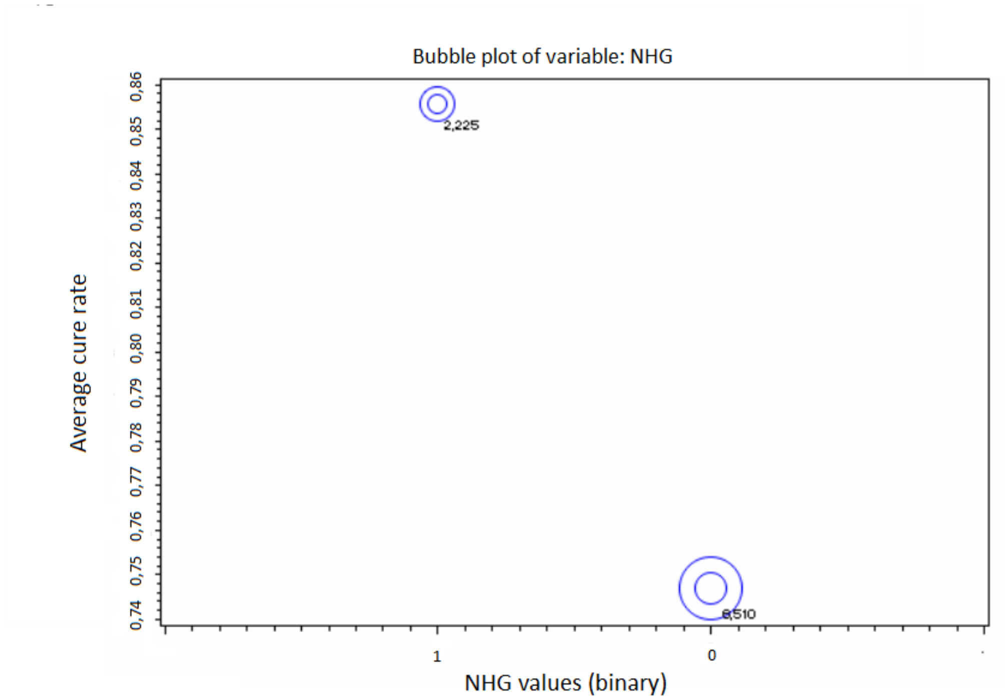


Figure 6: Bubble plot NHG

As we can see in the bubble plot mortgages without collateral (NHG = 0) recover on average less than mortgages with collateral. This is a logical result. To be conservative we say that if NHG is missing we will put a 0 instead. Our model will then forecast a little bit more on the conservative side which is less risky. If we look close at the plot we will see that the numbers do not add up to 12387, this is because we have run the univariate analysis on a training set and not on the whole set.

additional node At the end of our research we ran the machine learning models which can handle missing values (such as random forest) without replacing the missing values and without deletion. After comparison with the 'cleaned' dataset we saw that the models with the cleaned dataset performed better.

information value + c-stat information value (IV) and c-stat are both methods to assess the explanatory power of a predictive variable regarding the target variable. For categorical variables we look at Information value to classify predictive power. For continuous variables we look at the c-stat to determine the predictive power of a variable. IV and c-stat both help us select good predictors for our final model.

$$IV = \sum (\%\text{non-events} - \%\text{events}) * WOE$$

In this case an event would be a cure and a non-event would be a non-cure. WOE stands for weight of evidence which is derived by: $ln(\frac{\%\text{non-events}}{\%\text{events}})$. For the recovery rate our target variable is continuous and we cannot define properly an event and a non-event. The information value will only be calculated for the cure rate. For the recovery rate we will look at Spearman's rank correlation and feature importance of the algorithm plus the risk drivers of the cure rate models. The same holds for the c-statistic, it is a method to assess the predictive accuracy when the outcome is binary. We assess the variables by the following scoring rules:

IV	Predictive power
< 0.1	weak predictor
0.1 - 0.3	medium predictor
> 0.3	strong predictor

C-stat	Predictive power
< 55%	weak predictor
55% - 60%	medium predictor
> 60%	strong predictor

For every variable either an IV or c-stat was derived, depending on the type of variable. The results for the model variables of the cure rate are given below.

variable	IV	c-stat
FREQ	-	0.76
same	-	0.66
fastRecover	-	0.69
SpeedCure2	0.58	-
DELQ_L12M	-	0.59
system_id	0	-
euribor_3	-	0.58
limit_start_year	-	0.57
arrearsflag	0.22	-
Diff_EAD	-	0.54
LTV_curr	-	0.61
LTV_Risky1	0.14	-
ITI	-	0.52
changeEAD	0	-

As we can see changeEAD has no predicting power according to the IV. IV is very good in classifying binary variables but not that good in ordinal variables. changeEAD can be -1, 0 or 1, which are hard values for IV to do a good job, that is why the information value is low and we should not only look at the information value or c-stat for variable selection. The Spearman rank scores for the recovery rate variables:

variable	Spearman rank
arrearsflag	0.27246
DELQ_L12M	-0.10529
Diff_EAD	0.12476
fastRecover	-0.39146
limit_start_year	-0.09839
LTV_curr	-0.32306
NHG	0.14164
same	-0.32823
euribor_3	-0.08902
MONTH_limit	0.18308
remain_loan_period	-0.14226
ITI	-0.20902
current_instalment	-0.17216
EAD	-0.20025
AGE_curr	0.09325

The higher the Spearman rank score the more informative the feature is. The feature with the highest rank score is fastRecover (-0.39). The negative relation seems logical since it can be assumed that if the previous default was recovered fast that this default will have a higher recovery, due to reliable behavior. We observe that the features with the highest scores match the best features chosen by the algorithms.

transformations For logistic regression models we need variables on the same scale. Therefore a transformation (logistic, exponential, quadratic, linear or cubic) has been chosen for each variable as well, based on the c-stat or information value, the fit of the function and business rationale. In some machine learning models this is not necessary and we can use the original drivers. An example of the transformation-approach: The blue dots are the data points. We try to fit the best line through the points. the lines we use are, linear, square, logistic, exponential and cubic. For this plot, square and cubic are the best lines, which looks reasonable. We will summarize the results below, the plots can be found in the appendix. Only the continuous variables need to be transformed.

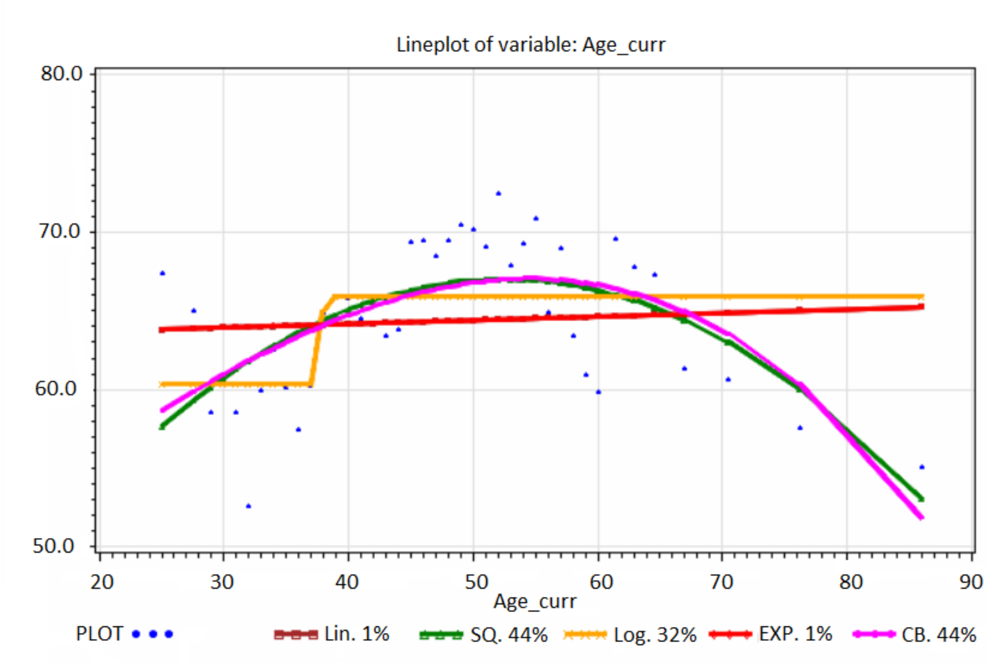


Figure 7: lineplot of variable: current age

cure rate variables	transformations
FREQ	log
same	log
fastRecover	log
euribor_3	cubic
limit_start_year	cubic
Diff_EAD	log
LTV_curr	cubic
ITI	cubic

recovery rate variables	transformations
Delq_L12M	log
euribor_3	log
limit_start_year	cubic
remain_loan_period	cubic
LTV_curr	log
ITI	log
same	log
fastrecover	log
month_limit	cubic
diff_EAD	log
current_instalment	cubic
EAD	cubic
AGE_curr	cubic

feature engineering After the calculation of IV and c-stat of the variables we tried to create own features where the values of the IV and c-stat were high, to see if we could explain the data even better. Next to creating extra features also the binning of several basic features was tried. A list of every created or binned feature:

1. leeftijdscategorie, binned age
2. Diff_months_OrderDate, difference between end of mortgage and order date of mortgage.
3. Diff_LDE, difference between end of mortgage and last default event.
4. Diff_NDM, length of performing period of mortgage.
5. Diff_Pod_Pid, lenght of previous default period.
6. sixmonthflag, Diff_months_OrderDate > 6 months
7. yearflag, Diff_months_OrderDate > 12 months
8. twoyearflag, Diff_months_OrderDate > 24 months
9. threeyearflag, Diff_months_OrderDate > 36 months
10. fouryearflag, Diff_months_OrderDate > 48 months
11. fiveyearflag, Diff_months_OrderDate > 60 months
12. Diff_EAD, EAD12 - EAD
13. Diff_OS, OS_init - OS
14. LTV_Risky1, LTV_curr > 1
15. CI_to_OS_ratio, $= \frac{\text{current installment}}{\text{OS}} * 100$
16. AuctionCancel, flag if auction_date is filled.
17. ComparedDebtandRecovery $= \frac{\text{debt\_on\_auction\_date}}{\text{NHG\_secrec}}$
18. flag_LS, flag if loan_status_key = "LS"
19. flag_MR, flag if loan_status_key = "MR"
20. flagTopThree, flag if product type code is ’p1182’,’p1175’,’p1168’
21. debtpermonthOD, arrears / diff_months_orderdate
22. debtpermonthLDE, arrears/ diff_LDE
23. flagDODlow, flag if DebtPerMonthOD < 41.86;
24. arrearclass, binned arrears
25. diff_limit_year, difference between start of limit and today
26. workoutflag, flag if months_workout is filled
27. nextintodefflag, flag if next_int_def is filled
28. diffDefaultflag, flag if last default event $\neq$ next_int_def
29. debtOnAUCflag, flag if there is debt on auction date.
30. changeEAD, 1 if diff_ead < 0, 0 if diff_ead = 0, -1 if diff_ead > 0
31. fastrecover, cure length of previous default period
32. sourceflags, binned source
33. flag65, flag if age_curr > 65

34. WWinst, mortg_market_value - EAD
35. regio, place of collateral
36. fleuribor, binned euribor
37. Delq_L12MBucket, binned delq_L12M
38. FEAD, binned EAD.
39. _freq_, months of client in arreardatabase
40. same, #times paid late - #times paid back
41. arrearsflag, flag if last record is an arrear.

Most of the features were for understanding the data but some explained the data really well and were implemented in the model. Some features were implemented but later removed because of correlation, bad predictive value or it gives information which may not be used. Some of these features can be improved and many more can be introduced this will be discussed in the follow-up research section.

5 Methods and Models

This section will discuss the models that are used and their purpose.

5.1 Models

historical average model The current model used at ING is a historical average model. This model is conducted by a vintage analysis. A vintage analysis gives a distribution of the cure rate per month based on months in default. The method is straightforward, we take a starting point in time and we keep track of the mortgages in default. Per month, we look at the cumulative cure percentage. Say we started with 100 mortgages, after the first month 10 mortgages cured. The cumulative cure percentage after one month is $\frac{10}{100} = 10\%$. In the second month, there are 8 mortgages extra cured and the bank sold 10 new mortgages. The cumulative cure rate will be: $\frac{18}{110} = 16,4\%$. With a sufficient number of mortgages this vintage analysis will generate a plausible longterm average per month. An example:

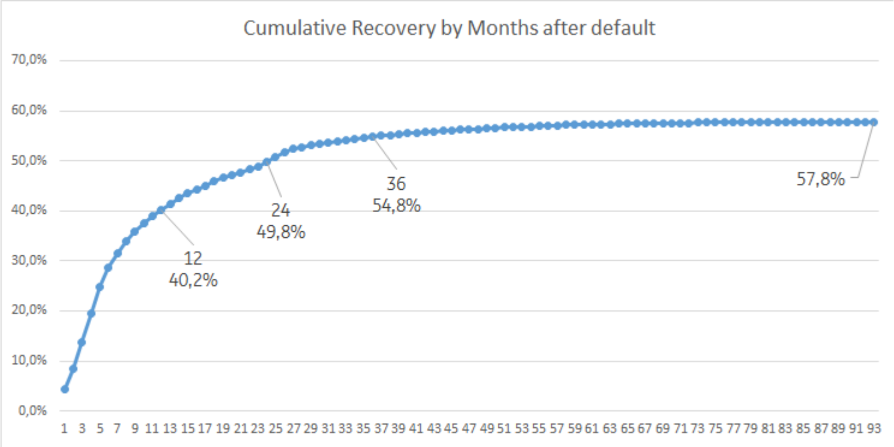


Figure 8: example of vintage analysis

This vintage analysis was done on the mortgage data, we see an exponential curve which flattens around 60 and has a long-term average of 57.8%. This method is easy to use and gives an indication of the overall cure rate but is limited in making a distinction between customers, since we only look at months in default.

machine learning models To estimate a probability of cure and recovery on customer level, we will use machine learning. The different machine learning techniques during this research have been discussed in the literature review. We will use: Logistic Regression, Decision Trees, Gradient Boosting, Random Forest and Neural Network. All models have different qualities and different learning algorithms. Logistic Regression and Decision trees are the simpler algorithms which are used to create a benchmark model and provide insights in feature importance. Gradient Boosting, Random Forest and Neural Network are algorithms which built upon the 'basic' algorithms such as DCTs and LR. We will train all models on the mortgage dataset and we will compare the results.

5.2 Evaluation

The evaluation of the models is different for the cure rate and recovery rate. This is caused by the different types of target variables. For the cure rate we estimate a binary target, for the recovery rate a continuous target. The cure rate target will be evaluated by accuracy/misclassification rate and the residual plots. The recovery rate target will be evaluated by the average square error and the residual plots.

6 Experimental Setup

To model the LGD we need to model the cure rate and recovery rate first. Both models will be trained and tested separately. We will have different sizes of datasets for both components as well. The cure rate will be evaluated on misclassification rate and the recovery rate on the coefficient of determination. We will estimate the LGD for three different cases. LGD through the cycle (through the cycle), LGD in default and LGD in downturn. For each LGD-model we will specify the test & training set, the features, the algorithms and the hyperparameters.

6.1 Training and Testing

The different sizes between the datasets are caused by different clients. Clients which do not cure will be in the recovery rate dataset. So our recovery rate dataset will always be smaller than our cure rate dataset for every case. For LGD through the cycle we use all data provided by ING on client level. For LGD in downturn we take the same database but only of 1 year. So the dataset for downturn will be much smaller. For LGD in default we will transform the database from client level to month level, this means that we will have a full overview per client. This results in an unfolded dataset which is much bigger than the orginal database. The different databases summarized:

Dataset	#rows
Cure rate through the cycle	67300
Cure rate in default	129744
Cure rate in downturn	7292
Recovery rate through the cycle	12478
Recovery rate in default	60300
Recovery rate in downturn	2000

We part the datasets into an 80% training set and 20% test set. The training set is large enough to train the algorithm sufficiently, a 20% test set is also large enough for testing. For LGD in downturn we have relatively few observations so we tested whether it is good to alter the training/test ratio.

Ratio	60/40	70/30	80/20	90/10
cure rate (misclassification rate)	0.047	0.047	0.045	0.047
recovery rate (ASE)	0.051	0.052	0.045	0.049

This test is based on a random forest algorithm, we test on misclassification rate for cure rate and average square error for recovery rate. We see that for the cure rate LGD in downturn the difference between the ratios is very small. We observe that 90/10 leads to a worse result than 80/20 due to overtraining. For LGD in downturn we will stay with the ratio 80/20. For the recovery rate we see bigger differences but also that the 80/20 ratio gives the best performance. A notable observation is that the model performs better under a 60/40 rate than a 70/30 rate. This can happen especially when we have little data.

6.2 Experiments

For each case we will run five machine learning algorithms. LR, DCT, RF, GB and NN, we compare every algorithm with each other based on important measurements. For the cure rate the measurements will be: *misclassification rate/accuracy, residual plot*. For the recovery rate it will be: ***average square error**, residual plot*.

6.3 Hyperparameters optimization

The more complex algorithms: Random Forest, Gradient Boosting and Neural net all have hyperparameters. These hyperparameters can be tuned for better results. For random forest and Gradient boosting we will use Grid Search to optimize the hyperparameters.

Random forest The hyperparameters which will be optimized for random forest are:

- Leaf size
- Maximum debt

- Number of variables to consider

The leaf size specifies how many observations there should be in the last leaf. This is to catch noise and not to overtrain the data with too many details. The maximum debt specifies how deep the trees can be, it reduces the complexity of trees and leads to lower chance of overfitting. The number of variables to consider is to handle the variance/bias trade-off. If we consider all the variables at each split random forest we continue to build the same tree over and over again which leads to low variance. If we consider only a small number of variables each split we could end up with a lot of bad trees and a biased model. The hyperparameter: 'number of variables to consider' is to find balance between these two possibilities.

Gradient Boosting For gradient boosting the hyperparameters are:

- iterations
- shrinkage
- depth
- samples

Iterations stands for how many trees are built by the algorithm before quitting. In theory the more iterations the better the algorithm can predict but it can take up to a very long time to train all these trees. The iterations hyperparameter optimizes the tradeoff between processing time and learning growth of the algorithm to find a fast and precise model. Shrinkage is the concept of multiplying each step of the algorithm (building a tree in this algorithm) by a factor between 0 and 1, called a learning rate. Shrinkage causes sample-predictions to slowly converge toward observed values. Samples that get closer to their target end up being grouped together into larger and larger leaves, resulting in a natural regularization effect. The hyperparameter depth is the same as Maximum debt for random forest so to reduce complexity. The hyperparameter samples is to generate more different tree splits, which results in more information for the model. Gradient boosting algorithms provide the ability to sample the data rows and columns before each boosting iteration, this results in more variance between the trees.

Neural Network For Neural Net we will run different auto neural nets which and see which performs best. These auto neural nets are all differently shaped networks, we will try a block structure, a cascade structure and a funnel structure. These networks provide a baseline model to see the type of network which performs best. The best performing network we will use for the competing neural network.

7 LGD in default

For LGD in default we will add "months in default" as a variable to the LGD through the cycle. The end goal is to give a prediction of the probability of curing/recovering per month. To avoid bias in the model we need a specific sampling approach.

7.1 Sampling approach

in default cure rate The modeling dataset is constructed by first extending the aforementioned starting dataset. Every default observation should be in the data as many times, as the months-workout. So for example, if a customer is 5 months in default, the number of rows for this particular customer should be 5 in the in-default dataset. This approach is to enable inclusion of post-default information by inclusion of reference moments. The risk driver information should be applicable at all the reference moments (e.g., LTV one month after default, two months after default, etc). Furthermore, the value of the months-in-default will increase through the cycle in the data. The variables that remain the same through the cycle should be copied throughout the history (like target variable that represents cure (cure flag) and the variable that represents the random number (used to select the development dataset etc).

The constructed dataset cannot serve as a modeling dataset yet. This is because of the following reasons:

- Since the average workout period for cures and losses differ, the dataset will be not representative.
- Since one customer exists multiple times in the dataset, autocorrelation will occur leading to biased estimations.

In order to deal with these issues, a stratified sampling approach should be applied (division into different subgroups or strata, then randomly select the final subjects proportionally from the different strata). The sampling is based on months-in-default, and the following conditions with regard to the original cure rate table and the sampled extended cure rate table apply:

1. distribution number of observations per months-in-default is equal
2. cure rate per months-in-default is equal
3. unique facilities is based on the same split of development vs. validation dataset ($\approx 80\%$)

The goal is to find the number of cures and non cures per workout months (months in default). To know the ratio of both we start by inspecting the starting dataset. We add up every observation of cures and non cures for different months in default. Naturally we see a decreasing line:

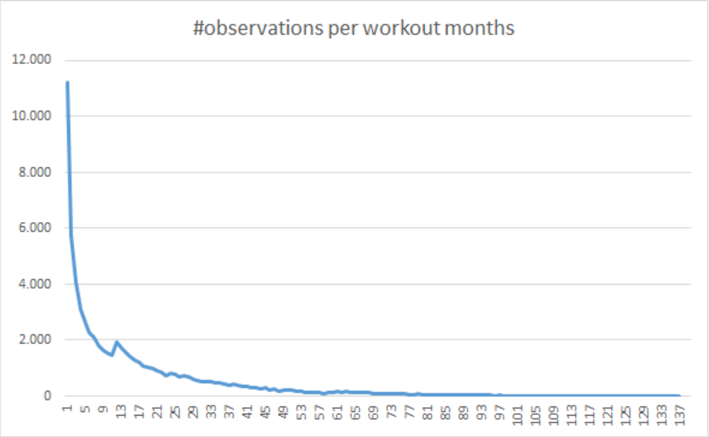


Figure 9: cure and non cure cases per workout months

We see a small peak at month 12. This can happen as we look at point in time observations, the reason for the peak is that there was somewhere in time a sudden boost in defaulted mortgages. As the data matures this peak will move to the right. We split these totals per month in cures and non cures to find the ratio. We use stratified sampling and calculate the sample size per different workout month and the ratios for cures and non cures for this method an optimal allocation formula is used. After the calculation we find the following ratios:

months_workout	prob_select_cure	prob_select_non_cure
1	40,6%	10,2%
2	24,9%	8,1%
3	19,3%	8,2%
4	15,4%	8,2%
5	13,9%	8,0%
6	12,6%	7,4%
7	12,3%	7,0%
8	10,3%	7,0%
9	10,0%	6,7%
10	9,7%	6,8%

The full table will be provided in the appendix. With this table we can generate our sample. We will include a variable "Random" in our extended dataset which represents a number between 0 and 1 per row. Next we look at the cure variable whether the observation is a 0 or 1. If this observation is a 1 we compare the 'Random' number with the "prob_select_cure" (table) percentage of the corresponding month. For example we have a record which did cure and is now in month 8, the prob_select_cure = 10,3%. We compare it to random, if the random number is smaller than prob_select_cure than we include it in our sample. The approach in pseudo code:

```
if cure $= 1$ and prob_select_cure > random then sample $= 1$;
else if cure $= 0$ and prob_select_non_cure > random then sample $= 1$;
else sample $= 0$;
```

This code is run per row which is already merged on months workout. At the end we keep all rows which have sample value 1.

in default recovery rate The recovery rate has the same sample methodology but we need to account for an extra factor. For recoveries, the outcome is not dichotomous (0 or 1) but may in theory range from 0-100%. The trick will be to define a number of buckets for recovery rate with different selection probabilities, to make sure the overall sample obeys all above requirements. The number of buckets will depend on the recovery rate profile for the specific portfolio. If the Dutch mortgage portfolio recovery rate profile shows a bimodal shape (a mixture of low recoveries and high recoveries), a two-bucket-model can model this situation well with a boundary lying somewhere between the two modes. Suppose that this boundary is equal to 50%. In that case, the calculation works the same as for cure rate, where the number of cures will be replaced with the number in the bucket above 50% and the number of non-cures will be replaced by the number in the bucket below 50%. The distribution of the recovery rate:

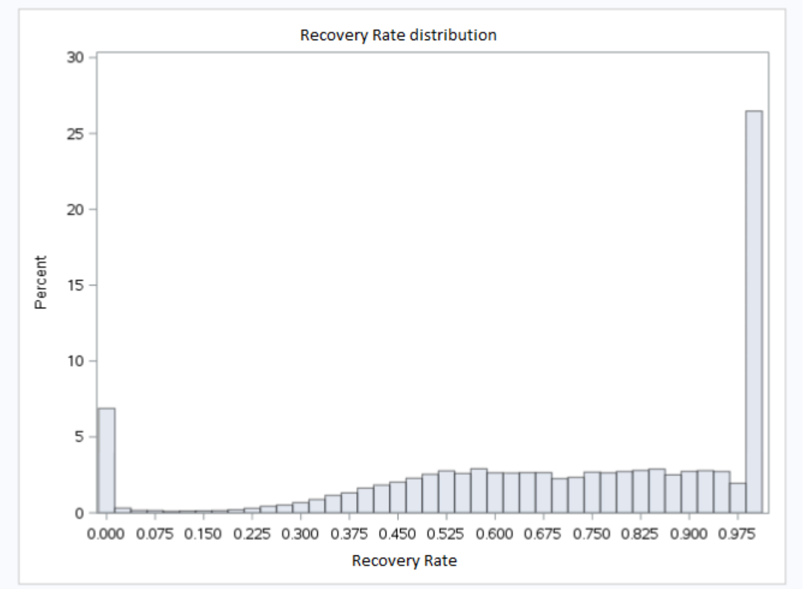


Figure 10: Distribution of recovery rate

As we can see above the distribution shape is bimodal but shifted to the right. A 50% cutoff will be inaccurate to use. That is why we are going to take the mean as cutoff point. The mean of the recovery rate is 70% so all recoveries above 0.7 will be set to 1 and those below to 0. This results in a dataset of 60300 rows with 4042 unique customers.

Subsample training data & validation data For the division into training data & validation data (80/20) we also need to sample again. This is needed because we need to keep the sample distribution for both sets. The same method is used but instead of deciding whether to keep a record or remove it, we divide the records into the training set and validation set. The distributions of the training and validation sets:

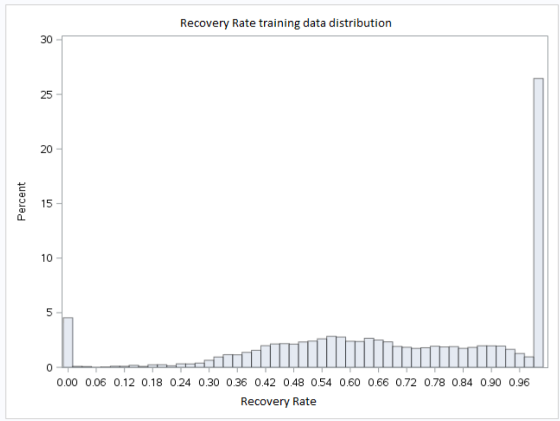


Figure 11: Cure rate average through the cycle

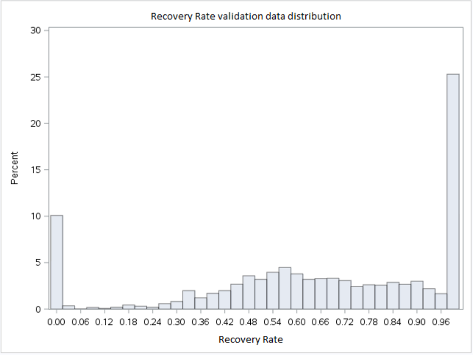


Figure 12: Recovery rate average through the cycle

The respecting means of the training and validation set are: 0.70 and 0.66. the two parted sets follow the distribution of the full set. The validation set contains more records with recovery rate 0. Since we only test on this dataset we are satisfied with the two created datasets.

It is important to notice that we use the real recovery rate to train the model on just like the distributions above. We only create a binary target variable to get a sample & subsample set, after that we model with the real recovery rate values.

8 Downturn Analysis

As we discussed before the current model uses multiple recovery rates. During this research there was a shift from using multiple recovery rates to using one recovery rate. However, the downturn analysis was done on the current approach with multiple rates. The reason We still use these results is threefold, first, using more rates for the recovery rate is a more extensive analysis than using one rate. Second, it is consistent with other projects within ING. Third, experts at ING do not think an analysis with one rate will change the downturn period.The recovery rates used here are: primary, secondary and unsecured.

For downturn analysis we first need to define what the downturn period of the data is. Article 4 of the draft RTS (CP/EBA/2017/02) specifies that the duration of economic factors should be one year and be defined as the worst period of each economic factor based on historical values observed in the preceding 20 years. We will start by looking at the average through the cycle of the cure- and recovery rate. The following two figures illustrate the cure and recovery rates through the cycle, together with the quarterly number of defaults. The analysis provided is performed by date of the default on a customer level.

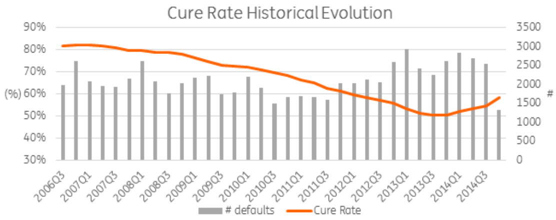


Figure 13: Cure rate average historical evolution

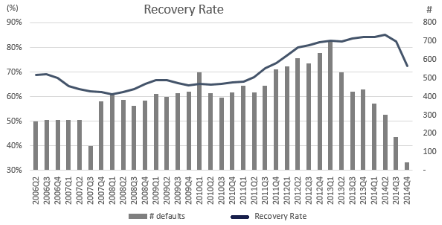


Figure 14: Primary recovery rate average historical evolution

The historical cure rates show a negative trend up to the end of 2013, where a minimum value of 50% is registered. During 2014 it slightly recovers showing a positive trend during this year. The resulting long-run number weighted average is equal to 68.4%. Generally, the primary recovery rate shows stable results between 2006 and 2010 with values around 65%, with an increase after 2011 and a decrease in the last two quarters of 2014. The resulting long-run number weighted average is equal to 71.0%.

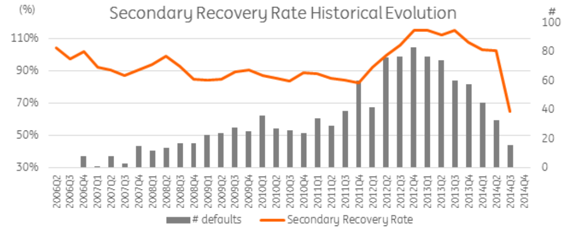


Figure 15: Secondary recovery rate average historical evolution

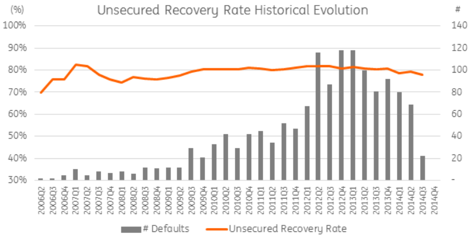


Figure 16: Unsecured recovery rate average historical evolution

Figure 14 shows a similar pattern although with a higher volatility due to a less number of observations as the historical primary recovery rates, with a long-run number weighted average is equal to 97.3%.

It should be noted that for the secondary recovery rate, the number of defaults available is limited in some periods, see also the graph presented above. For the next step in the analysis (linking each model component to economic factors), it is decided to only use the period 2009Q3 to 2014Q3 (5 years and 3 months). This period was chosen based on the quarters where the number of defaults is statistically relevant, defined by having at least 20 default observations in each quarter. This threshold ensures that a sufficient number of periods is taken into account whilst excluding the noise from periods with few default observations.

The unsecured recovery rate up until and including 2009Q2 is not statistically relevant due to the low number of observations. However, the unsecured recovery rate presents a stable value over time, with a long-run number-weighted average of 80.4%. As it can be seen, the stable behavior shows that no downturn effect exists for this component and therefore the described downturn approach is not applied for this component.

macro economic variables Next to the averages of all rates we will look at macro economic variables. To determine which economic factors are relevant for indicating the downturn period for each model component, a list of potential economic factors is considered, following Article 3(3) of the draft RTS (CP/EBA/2017/02). The initial selected economic factors for this analysis are described in the table below:

#	economic factor	Description	period	source
1	GDP	Absolute Growth Domestic Product of NL	Jan 95 - Jun 17	Eurostat
2	UR	Absolute Unemployment Rate in NL	Jan 95 - Aug 17	Eurostat
3	HPI	Absolute House Price Index, with the index at 100 in 2010	Jan 95 - Aug 17	CBS
4	Euribor 1M	Average interest rate at which Euro zone banks offer unsecured loans on the interbank market for a maturity of 1 months	Jan 95 - Apr 17	Bloomberg
5	Euribor 3M	Average interest rate at which Euro zone banks offer unsecured loans on the interbank market for a maturity of 3 months	Jan 95 - Apr 17	Bloomberg
6	Euribor 6M	Average interest rate at which Euro zone banks offer unsecured loans on the interbank market for a maturity of 6 months	Jan 95 - Apr 17	Bloomberg
7	Euribor 12M	Average interest rate at which Euro zone banks offer unsecured loans on the interbank market for a maturity of 12 months	Jan 95 - Apr 17	Bloomberg
8	inflation rate EU	Inflation rate in the Euro zone, with the index at 100 in 2015	Jan 97 - Aug 177	Eurostat
9	Inflation rate NL	Inflation rate in NL, with the index at 100 in 2015	Jan 97 - Aug 17	Eurostat
10	Dutch Mortgate Default rate	Internal average default rate	Apr 06 - Oct 16	Internal

For statistical analysis of dependencies between economic factors and model components it is among others required to take into account possible time lags between the realization of downturn in economic factors and the possible impact on the model components. Typically the impact of a downturn is only visible in model components several months or years after the downturn is identified in the considered economic factor. To find the correlation between the economic factors and the model components it is easier to have a smooth curve without any peaks in the data, also we are interested in the correlation of the trend between the component and the economic factor. For these reasons transformations and smoothing are used. Consequently, a number of transformations have been applied to each economic factor (see Table below), which means a total of 100 economic variables transformations are analyzed for each model component. We will use Moving Average over 1 or 2 years.

#	transformations	formula
1	1Y Growth Rate	(Var_T - Var_T-12M)/(Var_T-12M)
2	2Y Growth Rate	(Var_T - Var_T-24M)/(Var_T-24M)
3	Moving Average (MA) lag 1 year	Avg(Var_T-12M to Var_T)
4	Moving Average (MA) lag 2 years	Avg(Var_T-24M to Var_T)
5	Moving Average (MA) 1 year	Avg(Var_T to Var_T+12M)
6	Moving Average (MA) 2 year	Avg(Var_T to Var_T+24M)
7	Growth Rate 1Y - 6 Months lag	Var_T+6M/Var_T-6M - 1
8	Growth Rate 2Y - 6 Months lag	Var_T+6M/Var_T-18M - 1
9	MA 1year (1y forward)	Avg(Var_T+10M to Var_T+21M)
10	MA 2year (1y forward)	Avg(Var_T+10M to Var_T+33M)

To determine which economic factor(s) have the highest correlation with the model components, Spearman's rank correlation is calculated between each model component and the economic variables. Firstly, the transformed variable(s) with correlation higher than a threshold of 70% are selected. This threshold is commonly used in statistics to identify strong correlations. The resulting relevant economic variables for each model component are shown in the tables below.

cure rate		
economic factor	Spearman's rank	select transformation
UR	0.96	MA 2 year
euribor 3M	0.95	MA 2 year AVG (1Y fwd)
euribor 6M	0.95	MA 2 year
euribor 12M	0.95	MA 2 year AVG (1Y fwd)
euribor 1M	0.95	MA 2 year
HPI	0.94	MA 2 year

primary recovery rate		
economic factor	Spearman's rank	select transformation
HPI	0.91	MA 1 year
UR	0.90	MA 1 year
euribor 6M	0.82	MA lag 2 years
euribor 12M	0.81	MA 1 year
euribor 3M	0.81	MA lag 2 years
euribor 1M	0.81	MA lag 2 years

Secondary recovery rate		
economic factor	spearman's rank	select transformation
HPI	0.85	Growth Rate 2Y - 6 Months lag
Inflation rate NL	0.76	MA lag 2 years

The next step is to calculate the pair-wise correlation between the selected economic factors. The objective is to avoid selecting explaining variables which are highly correlated, to avoid multicollinearity. When this happens, some expert judgment will be used to select the one based on economic reasoning. Below only the relevant economic factors are provided (Spearman correlation > 0.7).[1]

Correlation matrix for economic factors considered for cure rate						
economic factor	UR	Euribor 3M	Euribor 6M	Euribor 12M	Euribor 1M	HPI
UR	**100**%	90%	90%	89%	89%	100%
Euribor 3M	90%	**100**%	95%	100%	95%	87%
Euribor 6M	90%	95%	**100**%	95%	100%	87%
Euribor 12M	89%	100%	95%	**100**%	94%	87%
Euribor 1M	89%	95%	100%	94%	**100**%	87%
HPI	100%	87%	87%	87%	87%	**100**%

It can be seen that all six economic factors are highly correlated with each other (all correlations are above 80%). When this happens, some expert judgment based on economic reasoning and business practice will be used to select the one which has the most influence on the specific model parameter. Taking this into consideration, unemployment rate (UR) is selected as the explainable variable for the cure rate component.

[1] The selected economic factors are discussed with the Economic Bureau

Besides being an important economic factor where a decrease in unemployment makes it more likely that clients will cure as it is more likely they will find a job and being able to pay their amount in arrears, this variable is also used earlier by ING in other models. This way, ING maintains consistency between different risk frameworks.

Correlation matrix for economic factors considered for primary recovery rate						
economic factor	UR	Euribor 3M	Euribor 6M	Euribor 12M	Euribor 1M	HPI
UR	**100**%	98%	91%	87%	91%	91%
Euribor 3M	98%	**100**%	88%	91%	88%	89%
Euribor 6M	91%	88%	**100**%	75%	100%	100%
Euribor 12M	87%	91%	75%	**100**%	75%	76%
Euribor 1M	91%	88%	100%	75%	**100**%	100%
HPI	91%	89%	100%	76%	100%	**100**%

The same six economic factors were selected as in the cure rate analysis and as such the same conclusion can be taken with all of them being highly correlated with each other. Again some expert judgment based on economic reasoning and business practice will be used to select the one which has the most influence on the specific model parameter. Therefore for the Primary Recovery Rate, the selected economic factor was the House Price Index as it directly impacts the recovery rate. As real estate prices move up or down, the value of the mortgage should also move in the same direction and therefore impacting the recovery amount.

Correlation matrix for economic factors considered for secondary recovery rate		
	HPI	inflation rate NL
HPI	**100**%	44%
Inflation rate NL 3M	44%	**100**

For the secondary recovery rate, the two selected economic factors do not present a strong correlation between themselves. As both have a strong link with the secondary recovery rate, they were selected as relevant economic factors. The final overview of the selected economic factors for the model component factors is present in the table below:

Component	Economic factors
Cure rate	UR
Primary recovery rate	HPI
Secondary recovery rate	HPI & inflation rate NL

identification of downturn period for each economic variable Next, the duration and severity of the economic downturn for each selected economic factor needs to be selected. As we stated before the downturn period has to be one year and be defined as the worst period of each economic factor based on historical values observed in the preceding 20 years. However, due to the global recession that was experienced after 2007 in the Dutch mortgages portfolio, the worst values of the macro-economic factors are to be expected in the period taken into account for model development (April 2006 to December 2014). To identify the downturn period for each economic factor we analyzed the period 1995 to 2016, except for inflation rates where data was only available after 1997 and for internal default rate for Dutch mortgages where we consider the period April 2006 to October 2016 due to data availability. In case the economic factor follows an upward/downward trend (depending on the economic factor) and therefore just taking the worst value in the observation period does not consider the economic cycle, the year with the highest/lowest annual growth rate is selected as the downturn period. Below, the historical graphs of these variables are shown and correspondingly the downturn periods are marked.

GDP: For GDP it can be seen a general consistent upward trend since 1996, registering only an annual decrease (one year change with negative value) in the years of 2009 and 2012. The steepest decrease is registered in 2009 and as such, this year was selected as the downturn period.

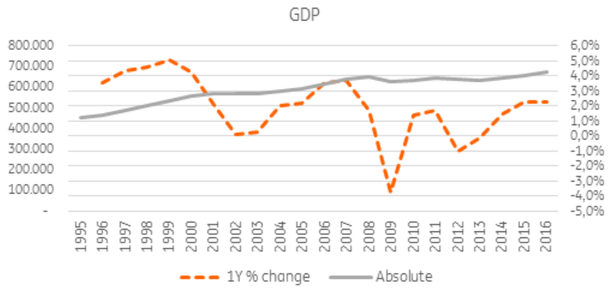


Figure 17: Identification downturn period GDP

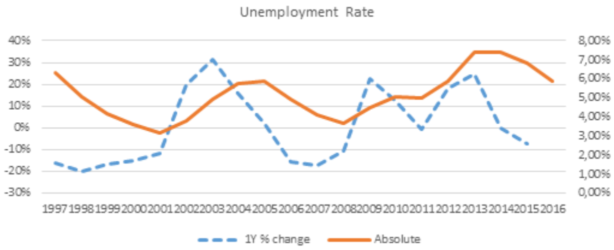


Figure 18: Identification downturn period UR

Unemployment Rate: The effects of the economic cycles in the last years for the unemployment rate are clearly visible, registering considerable increases in the years of 2004 and 2005, and 2013 and 2014, registering a maximum absolute value in the past 20 years of 7.4% in 2013. Therefore, this year is selected as the downturn period for unemployment rate.

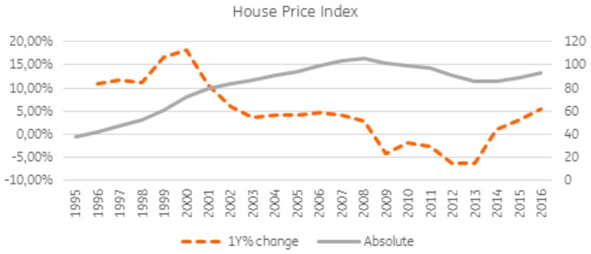


Figure 19: Identification downturn period HPI

House Price Index: a steady increase in the HPI is visible until 2008 when the global crisis hit the economy. HPI decreases in the following years and experiences a dip in 2013 where after it recovers. The year 2013 is selected as the downturn period as it is the year with the lowest absolute value after the crisis in 2007.

Euribor 1M 3M 6M and 12M: When the global crisis hit the Dutch economy, the Euribor experienced a steep decrease from 2008 to 2009. A steady negative trend in the Euribor in the following years is registered, with even small negative interest rates in 2016 (and 2015 for 1M and 3M rates). The relative change is the lowest in the year 2015, however this is due to very small negative rates (i.e., from 0.11% to -0.09% for 1M Euribor, resulting in a relative change of -188%). Therefore, it is decided to select 2009 as the downturn period, as the steepest decrease in absolute value is experienced in this year.

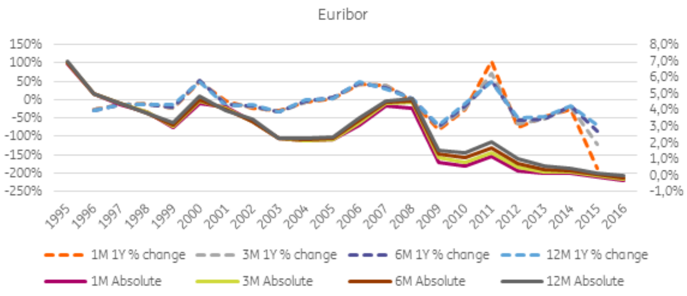


Figure 20: Identification downturn period euribor

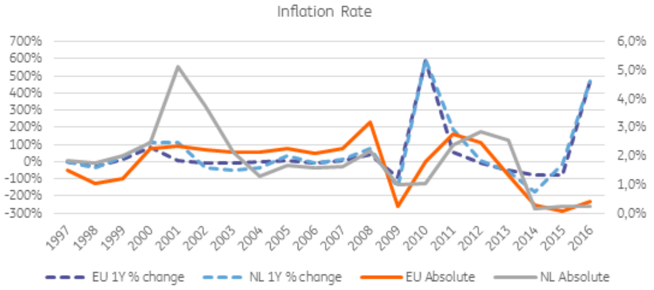


Figure 21: Identification downturn period inflation rate

Inflation Rates EU and NL: Both inflation rates for European Union (EU) and the Netherlands (NL) have registered more than one downturn period since 1997. The EU inflation rate recorded a peak in 2008 and 2011 where the NL inflation rate had a peak in 2001 and 2012. The year 2008 is selected as the downturn period for the EU inflation rate because the highest absolute value is observed in that year. For the NL inflation rate, the year 2001 is not selected as the downturn period because the passing on of the costs pertaining to the introduction of the euro may have been responsible for part of the high inflation rates. As this is not considered to be part of an economic downturn effect (and 2001 not observed as an economic downturn period), it is decided to select the year with the second highest NL inflation rate instead, which is the year 2012.

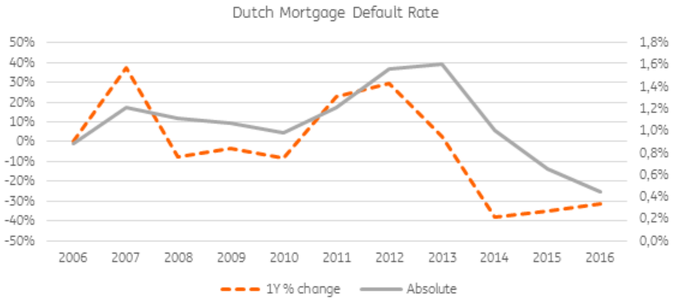


Figure 22: Identification downturn period default rate

Dutch Mortgage Default Rate: The Dutch mortgage default rate reflects the internal default rate for this portfolio. In there can be seen that after 2010 it started increasing achieving a peak in 2013. It can be said the default rate suffered its worst period in this year and therefore was selected as the downturn period for this factor. The overall picture for the economic downturn periods per factor is:

GDP	2009
UR	2013
HPI	2013
Euribor 1M	2009
Euribor 3M	2009
Euribor 6M	2009
Euribor 12M	2009
Inflation Rate EU	2008
Inflation Rate NL	2012
Dutch Mortgage Default Rate	2013

These results leave us two options, Scenario A: 2008-2009 and Scenario B: 2012-2013. The nature and economic background supporting the defined downturn scenarios is provided below:

Scenario A (2008 – 2009): This scenario represents the period where the global economic crisis happened, which affected the Dutch economy leading to a decline in GDP. After the crisis banks were more cautious in lending credit to clients which led to a supply issue in the market. Also during this period there was a combination of low affordability with high consumer confidence: the housing market was experiencing high prices, but people were still buying houses.

Scenario B (2012 – 2013): This scenario involves the change of fiscal deductibility applied by the government after 2012 which affected the consumer confidence. Also during the same period the Dutch government (NIBUD) applied austerity measures which lowered the affordability of mortgages. At ING there was an increase in the execution sales with the bank forcing the sales of the repossessed houses in order to clear their housing portfolio.

Computation of Downturn LGD for each Downturn scenario Once the downturn periods relative to different economic factors are grouped in different downturn scenarios, the LGD corresponding to each downturn scenario should be computed. In order to do this, we apply the following model formula to the development sample:

$$LGD = \frac{(\text{EAD}) - (\text{Secured recovery + Unsecured Recovery})(1 - \text{cure rate})}{EAD} + \text{indirect costs}$$

where,

$$\text{Secured Recovery} = \text{mortgage Collateral Value} * \text{Primary Recovery Rate} +$$

$$\text{Secondary Collateral Value} * \text{Secondary Recovery Rate}$$

$$\text{Unsecured Recovery} = \text{(EAD-Secured Recovery)} * \text{Unsecured Recovery Rate}$$

The inputs for the LGD formula are as expected the components identified in the first step of the downturn approach:

1. Cure rate
2. Primary Recovery Rate
3. Secondary Recovery Rate
4. Unsecured Recovery Rate
5. Indirect Costs

The estimation of the model parameters corresponding to each scenario depends on the relevant economic factor that is identified for each component. If the downturn period of the economic variable is present in the downturn scenario, the component will be estimated based on the downturn period of the economic factor. Otherwise, if the downturn period of the economic factor related to the model component is outside the downturn scenario, the long run average (LRA) of that model component will be used.

Year	Cure Rate	Primary Rec Rate	Secondary Rec Rate	Unsecured Rec Rate
LRA	68.4%	71.0%	97.3%	80.4%
2006	81.9%	69.9%	105.9%	60.8%
2007	82.1%	64.3%	92.5%	82.3%
2008	79.6%	60.9%	94.3%	74.5%
2009	76.5%	66.7%	84.7%	76.7%
2010	72.1%	65.2%	87.6%	80.4%
2011	66.5%	66.2%	88.6%	80.6%
2012	59.8%	76.7%	92.3%	81.7%
2013	53.3%	82.7%	115.3%	81.4%
2014	52.0%	84.2%	103.4%	78.5%

Consider the Cure Rate where the relevant economic factor for this component is the unemployment rate. Therefore the downturn period is the year 2013 which is present in the downturn scenario B. So for the scenario A, the LRA will be used to calculate the overall LGD where in scenario B the average cure rate value for 2013 will be used, i.e. 53.3%. When more than one economic factor is identified for a component factor, the worst value of the selected years is chosen.

Model component	Scenario A	Scenario B
Cure Rate	LRA	2013
Primary recovery Rate	LRA	2012
Secondary recovery Rate	LRA	2012
Unsecured recovery Rate	LRA	LRA
Indirect costs	LRA	LRA

This leads to the following downturn model component estimations and corresponding downturn LGDs:

Scenario A	Value	Period
Cure rate	68.4%	LRA
Primary recovery Rate	71.0%	LRA
Secondary recovery Rate	97.3%	LRA
Unsecured recovery Rate	80.4%	LRA
Indirect costs	0.32	LRA
LGD	**9.47**	

Scenario B	Value	Period
Cure rate	53.3%	2013
Primary recovery Rate	76.7%	2012
Secondary recovery Rate	92.3%	2012
Unsecured recovery Rate	80.4%	LRA
Indirect costs	0.32%	LRA
LGD	10.96%	

The downturn scenario associated with the highest LGD is chosen as the final downturn scenario. Below, it can be seen how the LGD through the cycle behaves for scenarios A and B compared to the observed loss cases (not including dragging cases):

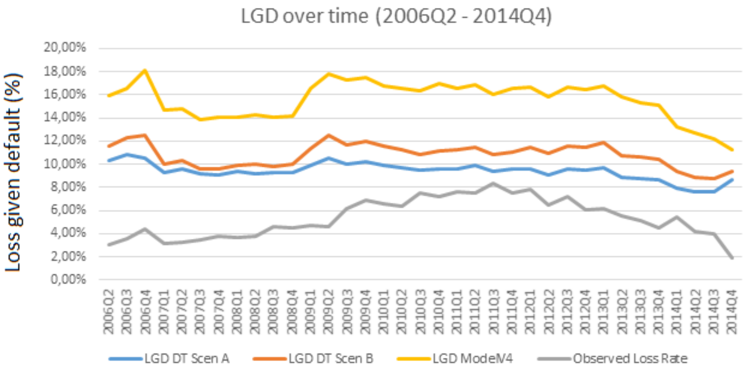


Figure 23: LGD through the cycle

Consequently, based on the results in the previous section and looking at the graph presented, Scenario B is chosen as the final downturn scenario. It can be seen that Scenario B always has a higher LGD value than Scenario A. Moreover, the LGD of scenario B is higher than the observed loss rate over the entire period while the current LGD model (LGD Model V4) has a much higher value than the LGD in both the Scenarios A and B. This demonstrates that the current model approach is sufficiently conservative

MoC for downturn Related to the Margin of Conservatism (MoC) and the downturn approach, a comprehensive analysis should be performed, calibrating the MoC around the assumptions made for the identification of the nature of the economic downturn. The MoC should be namely applied:

- When the analysis of the nature of the economic downturn shows no dependency between the model components and the economic indicators;
- When it is not possible to identify the severity of the economic downturn for a specific economic factor according to historical data;
- No data availability concerning the realized model components during the selected economic downturn period which are therefore estimated it for downturn adjustment purposes.

9 Results

In this section the results of each case will be discussed for the cure rate and recovery rate. Afterward, the LGD will be visualized and compared with the current model. The final cure rate models will be discussed in the section Experiment cure rate & Experiment recovery rate. First, we will show the findings we had with the first models of the cure rate component.

gap analysis The first benchmark models for the cure rate were flawed, it predicted well but there was a strange gap in the prediction, to illustrate:

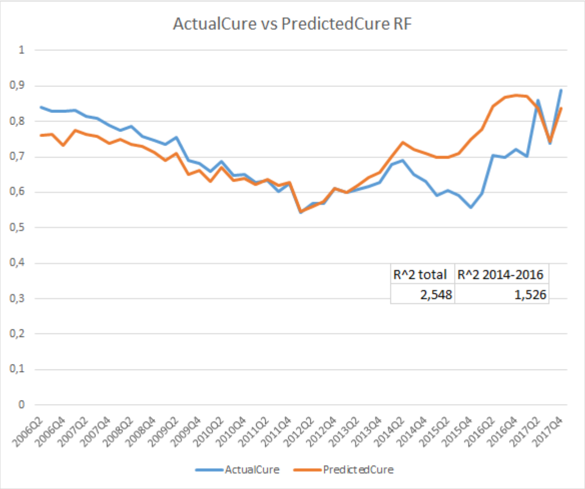


Figure 24: RF cure rate model 2014-2016

This illustration represents a random forest cure rate model. Per quarter from 2006-2017 the averages of the actual cure rate and the predicted cure rate are visualized The blue line is the actual average of the dataset, the orange line is the predicted average of the algorithm. The residual error is the overall difference between the actual cure and the predicted cure. As the graph above shows, the orange line follows the blue line very well except for the period 2014-2016. This model was calculated with a 70/30 ratio and the following variables:

- DELQ_L12M
- EAD
- LTV_Risky1
- NHG
- SpeedCure2
- _FREQ_
- arrearclass
- limit_start_year
- arrearsflag
- changeEAD
- fastRecover
- same

A residual error of 2.5 is not high when we consider the number of observations, it means that per observation on average we make an error of 7.13E-05. Per quarter it means an average mistake of 5%. These errors are mainly caused by the wide gap, so we analyzed this. Experts in the LGD field of ING did not recollect any big event in the period 2014-2016 which might lead to trend breaking. So we tried narrow the gap by using:

1. other algorithms
2. varying the training/test ratio
3. sampling

4. changing the input variables

For every variation on the random forest (or other) model we trained with the dataset and the same test set. Other algorithms gave the same trend as random forest. For the ratio and sampling changes we saw the following results:

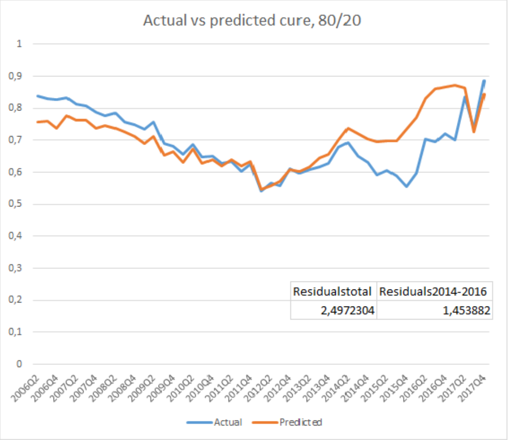


Figure 25: Ratio change 70/30 to 80/20

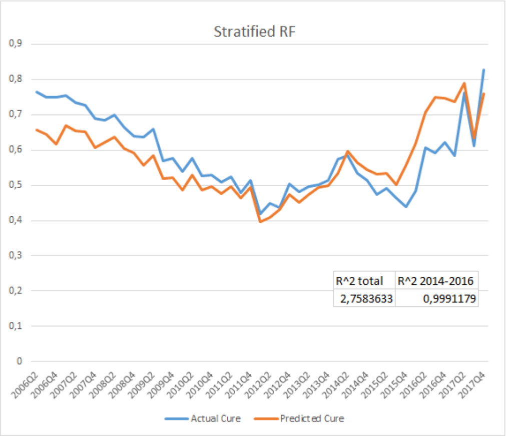


Figure 26: Stratified sampling applied on dataset

Both changes result in a small performance improvement in the 2014-2016 period. The residual error drops a little when we adjust the ratio, this is in line with our findings in the experimental setup section. If we apply sampling on the dataset we see that the error in the period 2014-2016 drops but overall the error increases, also the graph looks unstable. Both approaches do not narrow the gap sufficiently. Next we try changing the input variables of the algorithm. First we model the period 2014-2016 to see which variables are good predictors for that period. The results:

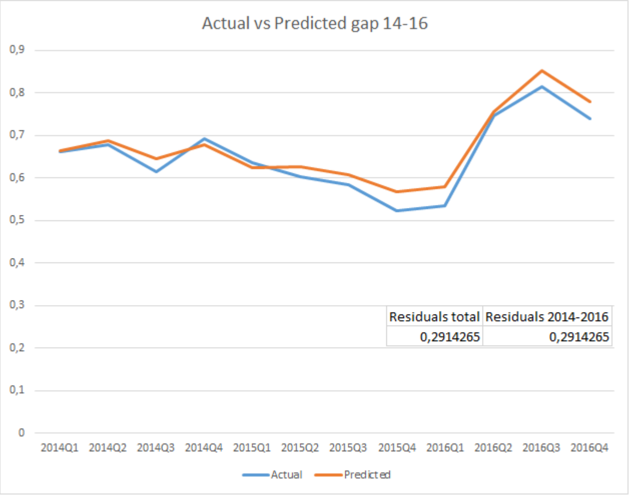


Figure 27: RF cure rate model 2014-2016

The fitted line estimates the actual line very well. When we take a look at the important features for this model we see some differences:

Differences in top variables between first model and 2014-2016 model	
2014-2016 model variables	First model variables
FREQ	DELQ_L12M
same	EAD
fastRecover	LTV_Risky1
SpeedCure2	NHG
DELQ_L12M	SpeedCure2
system_id	_FREQ_
arrearsflag	arrearclass
limit_start_year	arrearsflag
euribor_3	changeEAD
LTV_curr	fastRecover
remain_loan_period	limit_start_year
Diff_EAD	same

EAD, LTV_Risky1, NHG, arrearclass and changeEAD are out of the top predicting variables. system_id, euribor_3, LTV_curr, remain_loan_period and Diff_EAD replaced them in predicting ranking list. When we apply these variables to the full dataset we observe the current random forest model: The error has been cut from 2.5 to 1.25 and during the gap period the error decreased by almost 1. The gap analysis has shown us which variables predict better and which ratio to handle.

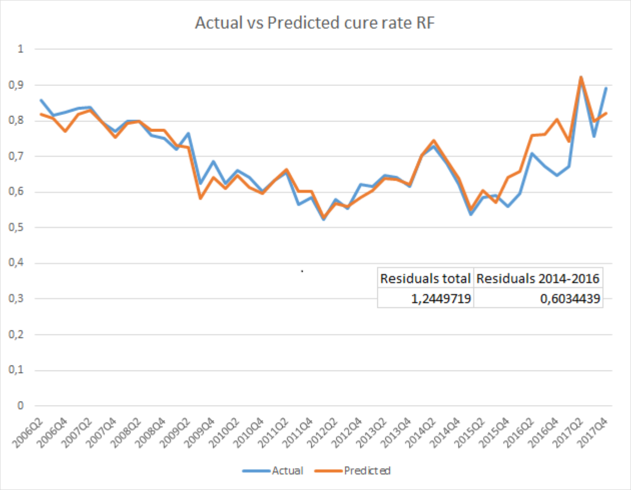


Figure 28: Cure rate model RF

9.1 Experiment cure rate

Here we will discuss the component cure rate, we will discuss the model parameters, the performance and the important features. In the end we will compare and choose the best model.

hyperparameters We will evaluate the hyperparameters we discussed in the experimental setup. We also discuss the difference between the hyperparameters in the three cases: LGD through the cycle, LGD in default and LGD in downturn. We will only show tables of the results. The graphs are in the appendix. First we inspect the cure rate.

hyperparameters cure rate LGD through the cycle	
Random forest	value
Leaf size	1
Maximum debt	35
Number of variables	10
Gradient boosting	
Iterations	500
Shrinkage	0.5
Debt	50
Sample	0.9
Neural network (NN)	
Structure	MLP, 10 layers

hyperparameters cure rate LGD in Default	
Random forest	value
Leaf size	1
Maximum debt	35
Number of variables	14
Gradient boosting	
Iterations	500
Shrinkage	1
Debt	50
Sample	1
Neural network (NN)	
Structure	MLP, 10 layers

hyperparameters cure rate LGD in Downturn	
Random forest	value
Leaf size	1
Maximum debt	5
Number of variables	10
Gradient boosting	
Iterations	200
Shrinkage	0.1
Debt	35
Sample	0.8
Neural network (NN)	
Structure	MLP, 10 layers

The neural net hyperparameters are constant. For RF and GB we observe more differences between the values of the hyperparameters per case. The shrinkage parameter is different for all three cases.

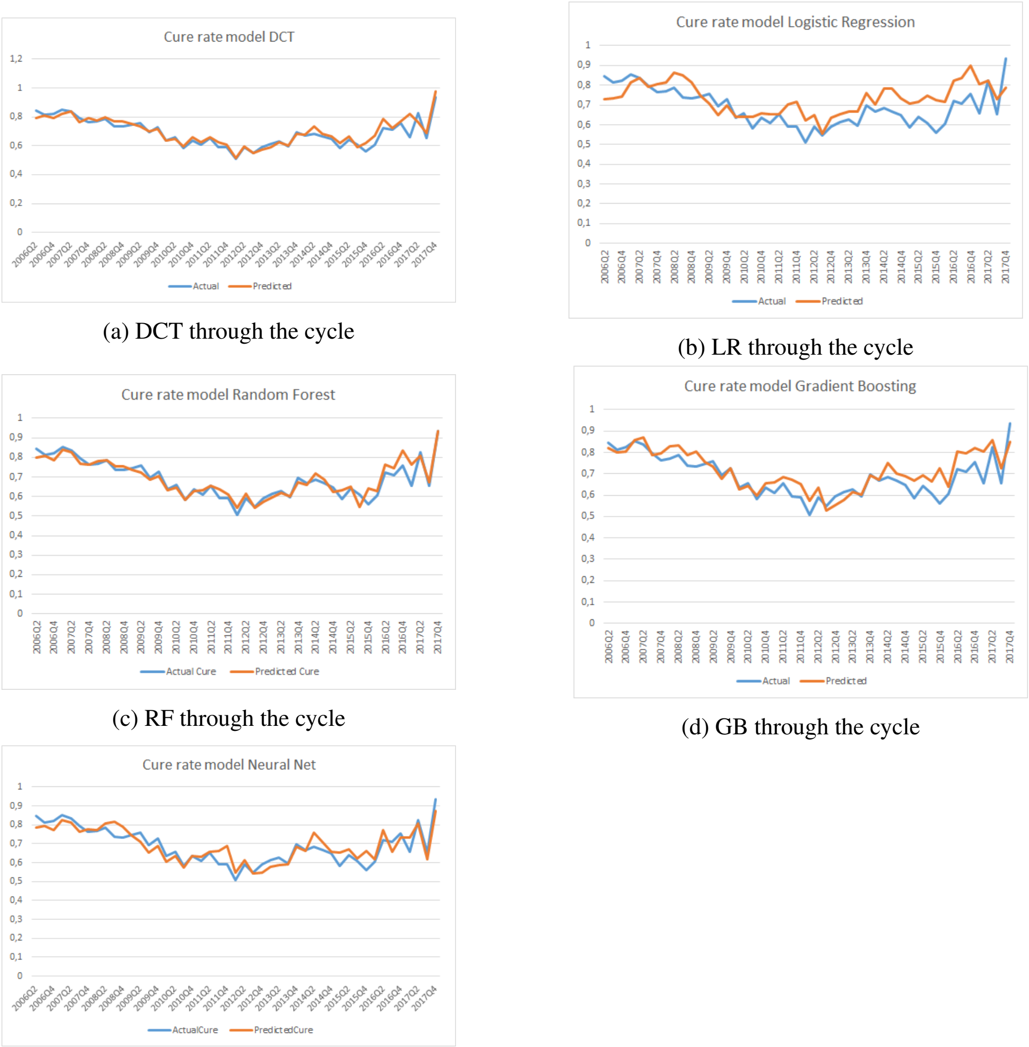
performance The performance of all algorithms is summarized below. First the evaluation metrics of the model are discussed, subsequently the features of the best algorithm are given. Of every case first the misclassification/accuracy is provided, second the model plots are provided. The model plots are residual plots of averages per quarter. At last the feature importance of the best algorithm is given. The importance of the feature is based on the discriminative and predicting power. For example in random forest every feature is evaluated on the number of splits and the ability to separate the data. We will start with the LGD through the cycle.

Model Scores cure rate LGD through the cycle		
Model	misclassification rate	Accuracy
logistic regression (LR)	0.15	85%
Decision Tree (DCT)	0.076	92%
Random Forest (RF)	**0.073**	**93%**
Gradient Boosting (GB)	0.011	89%
Neural network (NN)	0.089	91%

Model Scores cure rate LGD in Default		
Model	misclassification rate	Accuracy
logistic regression (LR)	0.23	77%
Decision Tree (DCT)	0.16	84%
Random Forest (RF)	**0.084**	**92%**
Gradient Boosting (GB)	0.20	80%
Neural network (NN)	0.19	81%

Model Scores cure rate LGD in Downturn		
Model	misclassification rate	Accuracy
logistic regression (LR)	0.13	87%
Decision Tree (DCT)	0.087	91%
Random Forest (RF)	0.082	92%
Gradient Boosting (GB)	0.080	92%
Neural network (NN)	**0.076**	**92%**

Above we see three tables of the model scores for each case. Each model is scored on their misclassification rate/accuracy. We see that Logistic Regression performs the worst each time. Random forest does the best job of classifying the customers. All scores are well above 80% and most even above 90%. This shows excellent models. We compare it with the residual plots of the model. In the next pages the model plots will be discussed.



(a) DCT through the cycle

(b) LR through the cycle

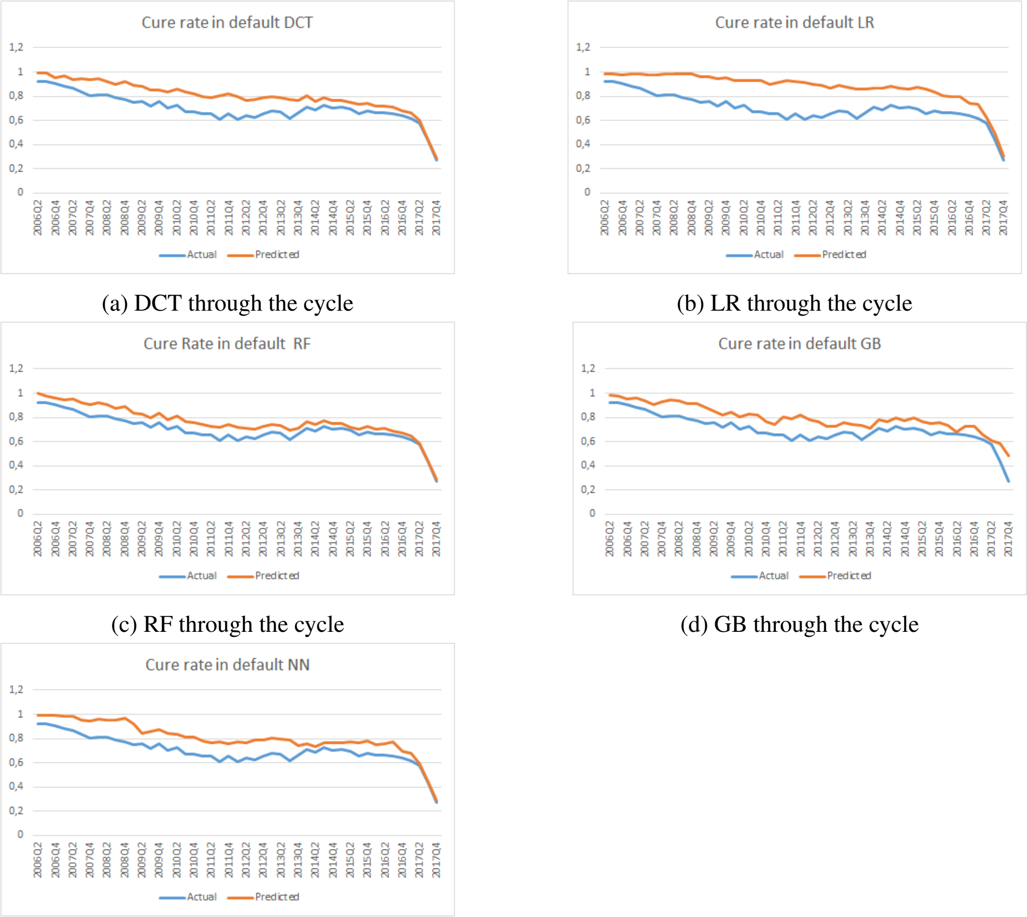
(c) RF through the cycle

(d) GB through the cycle

(e) NN through the cycle

Figure 29: residual plots of LGD through the cycle cure rate models

For the cure rate models through the cycle we see three nice fits: DCT, RF and NN. Random forest has the best fit overall and DCT has the best fit in the 2014-2016 period. This period has proved to be hard to estimate. In the model comparison paragraph more analysis is provided.



(a) DCT through the cycle

(b) LR through the cycle

(c) RF through the cycle

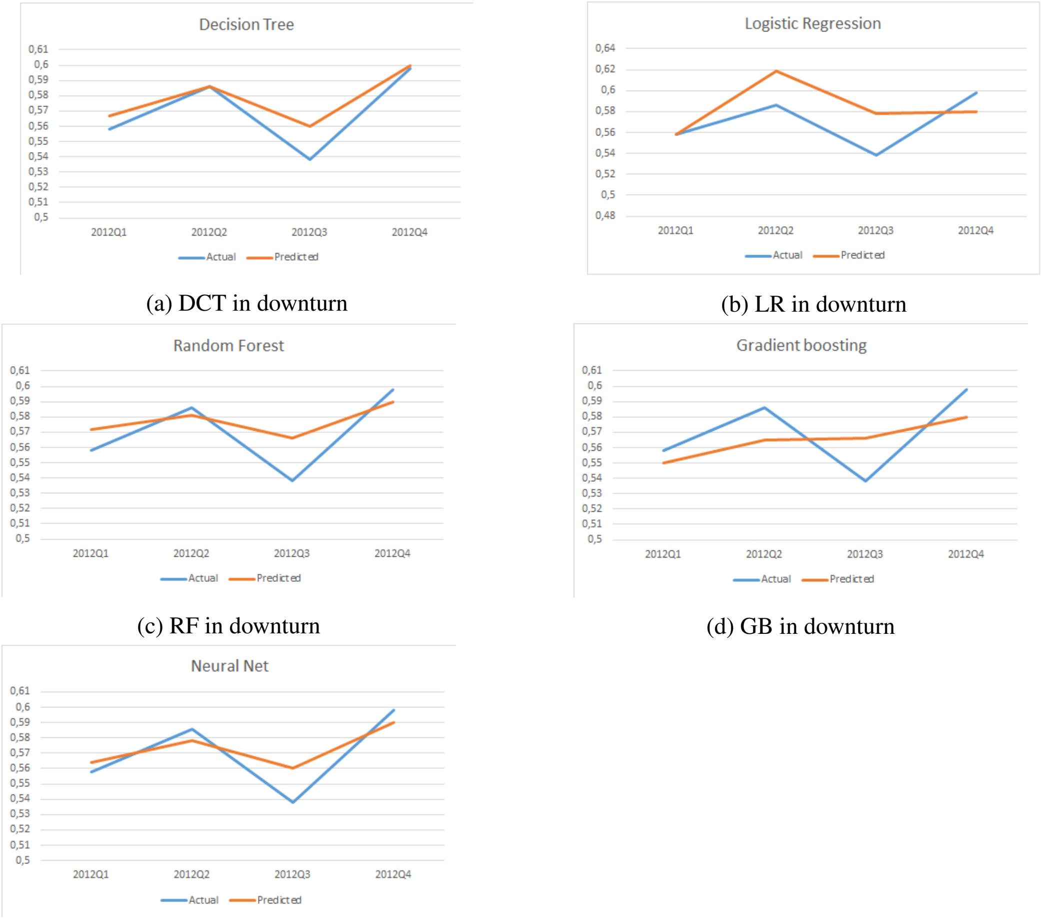
(d) GB through the cycle

(e) NN through the cycle

Figure 30: residual plots of LGD in default cure rate models

All models try to predict the steep drop in 2017 (at the end). Unfortunately for some of the models this results in a poor accuracy over the rest of the years. The only model that sufficiently models the cure rate in default is random forest. It is remarkable that random forest is the only great predictor when we look at other results. What also stands out is the plummet in accuracy of the in-default models. This is caused by a change of features in the model. Some features which were very predictive such as: "_freq_" and "same" are no longer available because they use the last known information of the customer. For in default modeling we are interested in the progress of the customer and using last known information would be cheating. This difference in features may also lead to the accuracy drop of most models.

h



(a) DCT in downturn

(b) LR in downturn

(c) RF in downturn

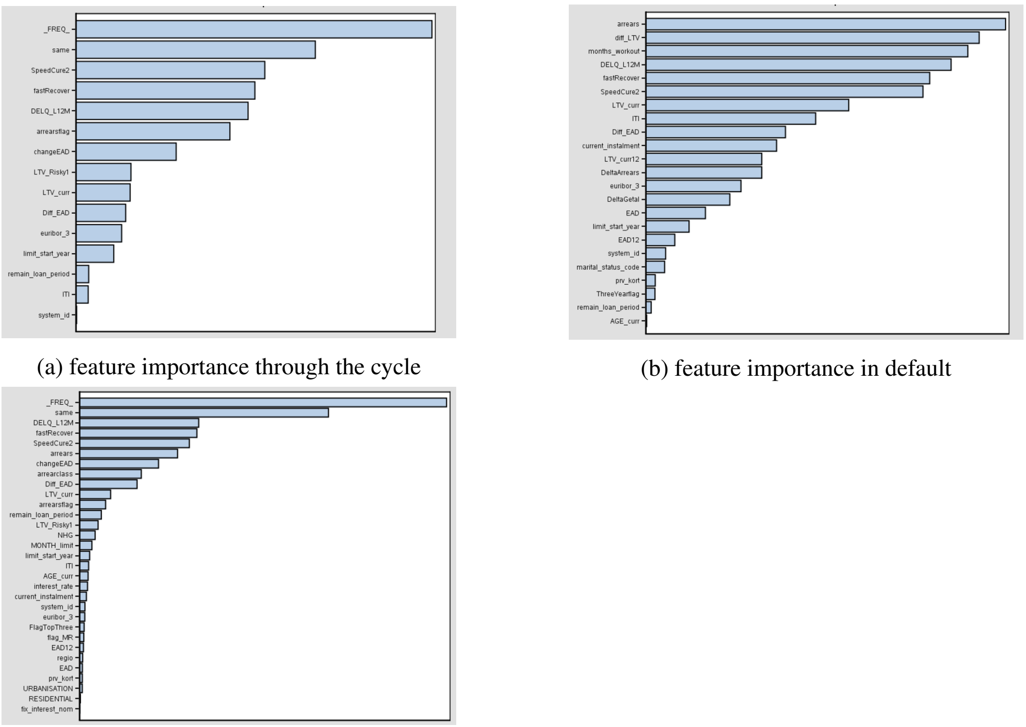
(d) GB in downturn

(e) NN in downturn

Figure 31: residual plots of LGD in downturn cure rate models

For LGD in downturn, we see that the Decision tree algorithm follows the actual trend the best. After that the Neural network, the random forest does a reasonable job and the Logistic regression and Gradient Boosting algorithm both do not align. Based on the trend Decision Tree and Neural Net perform the best.

feature importance Here we will discuss the different features of the models. All features are rated by the random forest algorithm even if this algorithm does not yield the best performance. This is done because random forest does a reliable job in rating the importance of the features and is easy to comprehend. Most features which are important for Random forest are also important for the rest of the algorithms. In the appendix there is a list of features of the final chosen algorithm. In all tables first the name of the feature is given and then the importance based on splitting rules, the ability to separate the data. Splitting rules indicate how often a variable is chosen to split the data on. The higher the number of splits the more important the feature is. For this analysis a random forest with 130 trees is used.



(a) feature importance through the cycle

(b) feature importance in default

(c) feature importance in downturn

Figure 32: feature importances of cure rate per case

We compare the three plots with each other and start by the comparison between cure rate through the cycle and cure rate in downturn. We see very much the same results for each variable. We only see minor shifts between the variables in importance this shows robustness in the model. Between the cure rate through the cycle and in default cure rate we see some differences. "_Freq_" and "same" are no longer available in default modeling since this is information we do not have at the moment of modeling in default. This loss we have to fill with the variable months workout and we see that the months workout account for these two variables due to its high feature importance. Two variables stand out in default modeling, diff LTv and arrears. Diff LTV is a new variable for in-default modeling since the database contains an LTV 12 variable (which we did not have in the dataset of cure rate through the cycle). Diff LTV = LTV 12 - current LTV. So the difference between the current loan to value and the loan to value previous year. It is not surprising that this variable is important since it is a direct relation between the value of the house and the loan itself. Typically a high loan to value relates to higher risk, so if the Diff LTV is positive this will result in less risk and more cures. It is a revelation that arrears is of such importance in default modeling since it was not important at all in the through the cycle model. This change probably arises from how the ING database is built. In the case cure rate through the cycle we take the last known record of the customer, the records of a customer are in the database until he is written off. If a customer is written off in August, the last record of this customer will be in September where the scenario variable will change to written off. Since it does not make sense to keep information of a customer which is already written off the data gathering after August will stop. So the record in September will not have any information at all but still be in the dataset. This results usually in a last record where the arrears are 0. For in default modeling we extract different records through the progress of a customer in the database where the arrears are non-zero values, making it a more informative variable.

model comparison We will compare the models based on their performance so the evaluating method and the residual plot.

Cure rate through the cycle We will discuss each case starting by LGD through the cycle. Based on the accuracy of the cure rate model random forest is the best performing algorithm, when we look at the residual plots we see no reason to reject the random forest algorithm. The predicted cure rate follows the actual cure rate very well and captures almost every peak. Random forest performs the best for LGD through the cycle.

Cure rate in default It is very clear that RF outperforms any other model for this case. Both the accuracy and the residual plot show the best results. RF should be used to model in default cure rate.

Cure rate in downturn At last for LGD in downturn we observe that NN does the best job in terms of accuracy. The residual plots show two algorithms which capture the actual cure rate sufficiently, DCT and NN. DCT seems to do a slightly better job but NN is more conservative. Since the ECB maintains strict rules about a conservative approach we will give the edge to NN for LGD in downturn.

9.2 Experiment recovery rate

For the recovery rate the same approach as the cure rate is taken. First the hyperparameters, then the performance and at the end the model comparison.

hyperparameters The algorithms for the recovery rate are the same, this means that the hyperparameters for the recovery rate are also the same. Logically the values can differ, we observe:

hyperparameters RR LGD through the cycle	
Random forest	value
Leaf size	1
Maximum debt	35
Number of variables	10
Gradient boosting	
Iterations	200
Shrinkage	0.1
Debt	50
Sample	0.9
Neural network (NN)	
Structure	MLP, 10 layers

hyperparameters recovery rate LGD in Def	
Random forest	value
Leaf size	1
Maximum debt	35
Number of variables	10
Gradient boosting	
Iterations	500
Shrinkage	0.9
Debt	35
Sample	0.9
Neural network (NN)	
Structure	MLP, 10 layers

hyperparameters recovery rate LGD in Downturn	
Random forest	value
Leaf size	1
Maximum debt	5
Number of variables	10
Gradient boosting	
Iterations	200
Shrinkage	0.1
Debt	35
Sample	0.8
Neural network (NN)	
Structure	MLP, 10 layers

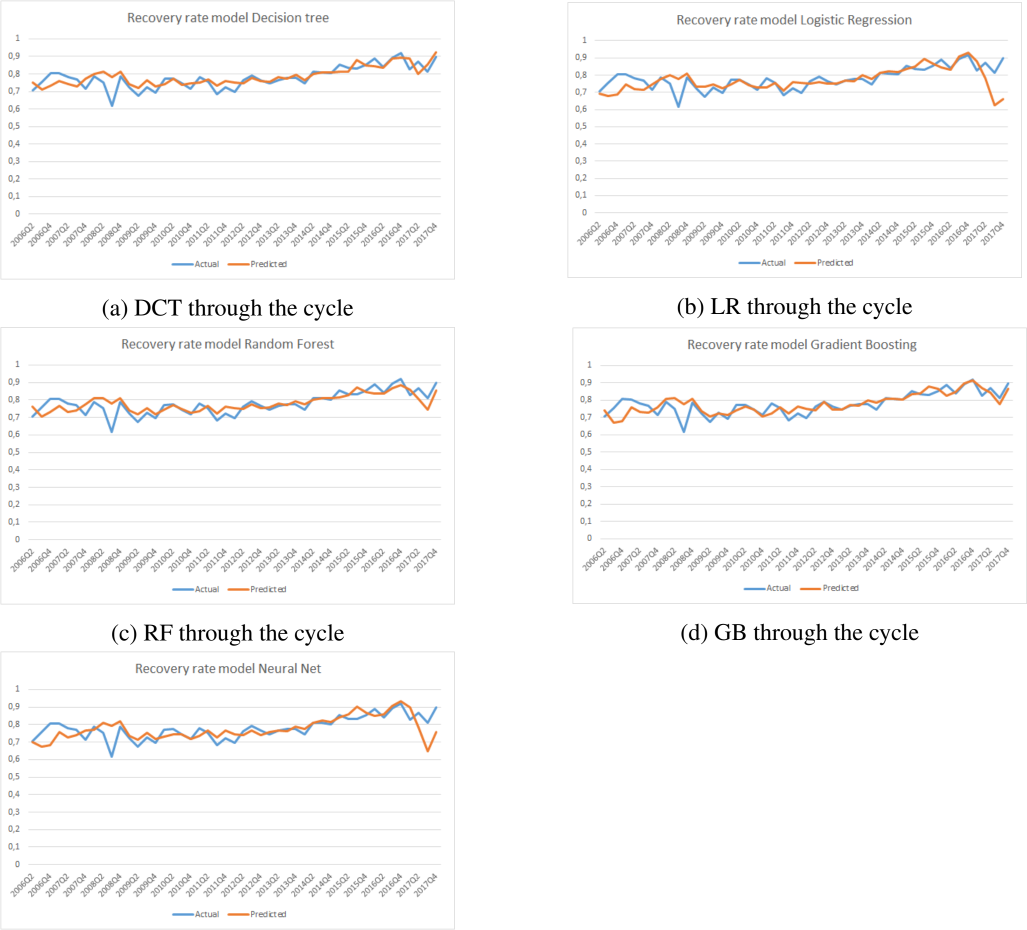
performance To find the right model for the recovery rate we will evaluate the performance of the models on Average square error (ASE), residual plots.

Model Scores recovery rate LGD through the cycle	
Model	Average Square Error (ASE)
linear regression (LR)	0.042
Decision Tree (DCT)	0.040
Random Forest (RF)	0.038
Gradient Boosting (GB)	**0.037**
Neural network (NN)	0.040

Model Scores recovery rate LGD in default	
Model	Average Square Error (ASE)
linear regression (LR)	0.057
Decision Tree (DCT)	0.032
Random Forest (RF)	**0.014**
Gradient Boosting (GB)	0.044
Neural network (NN)	0.047

Model Scores recovery rate LGD in downturn	
Model	Average Square Error (ASE)
linear regression (LR)	0.046
Decision Tree (DCT)	0.048
Random Forest (RF)	**0.045**
Gradient Boosting (GB)	0.046
Neural network (NN)	0.048

For each case the model scores have been given. We first compare the models based on their ASE. Gradient boosting does the best job in modeling the recovery rate through the cycle based on ASE. For the downturn period random forest performs better in terms of ASE. The differences are small so we need the residual plots to decide which is the best.



(a) DCT through the cycle

(b) LR through the cycle

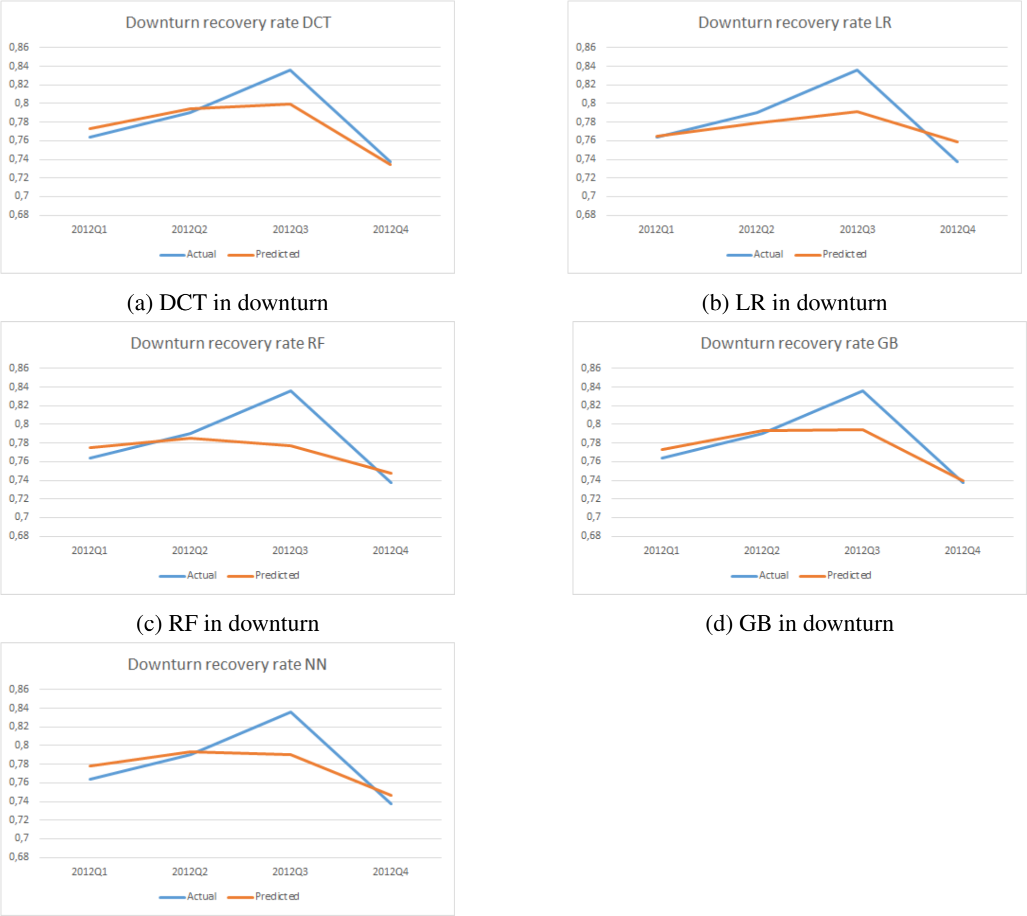
(c) RF through the cycle

(d) GB through the cycle

(e) NN through the cycle

Figure 33: residual plots of LGD through the cycle recovery rate models

Every model has a hard time forecasting the period 2006-2008. This period is hard because ING started gathering data around this time and the data gathered is generally of lower quality. We are more interested in the performance now than over 10 years ago so for now we will let this gap be. The trend of the recovery rate is slightly upwards through the cycle and we see that all models capture this. Three of the five models: DCT, RF and GB estimate the actual curve pretty well in the last period (2017). We observe that gradient boosting has a small edge over random forest and decision tree in terms of residual error. For the residual plots of recovery rate in downturn the same downturn period as for the cure rate is taken (2012-2013).



(a) DCT in downturn

(b) LR in downturn

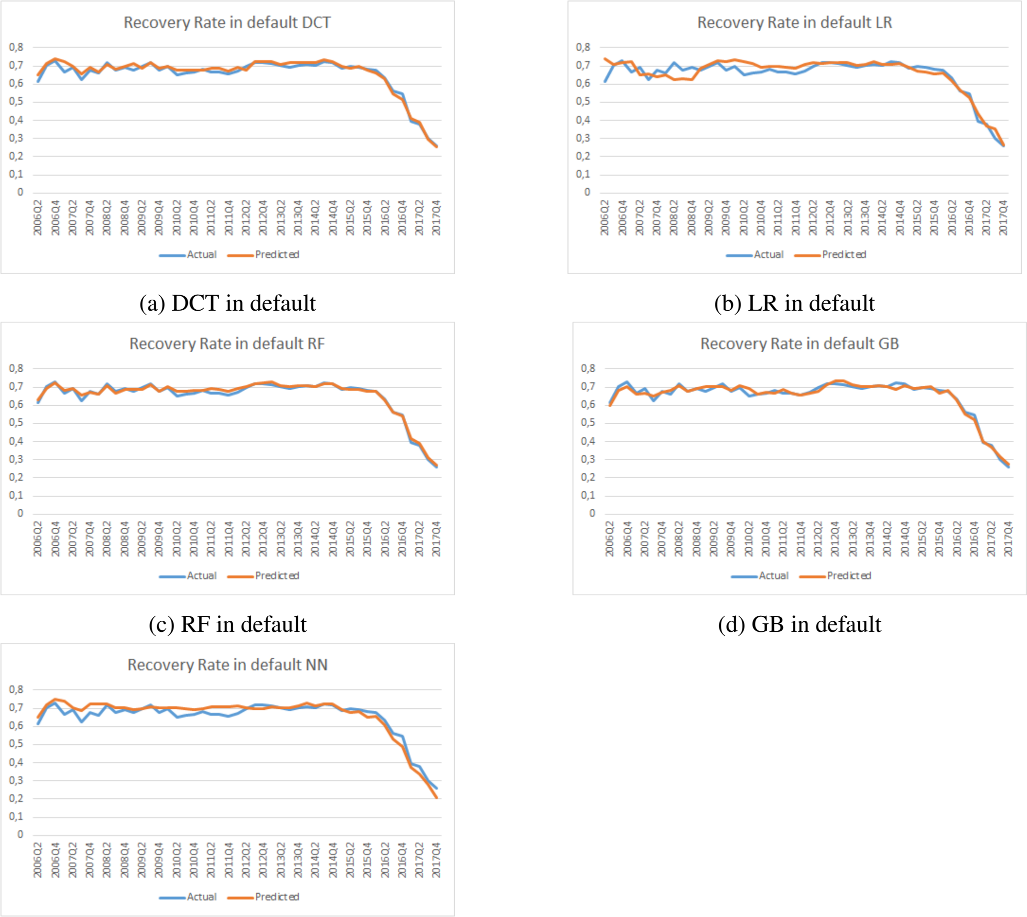
(c) RF in downturn

(d) GB in downturn

(e) NN in downturn

Figure 34: residual plots of LGD in downturn cure rate models

We see that none of the models captures the blue peak in the third quarter. When we look at the trend of the actual we see an increase until the third quarter and then it drops in the fourth. Only logistic regression and decision tree have an increase in prediction in the third quarter. Of those two decision tree captures the actual line the best. Decision tree does the best job in estimating the real recovery rate in the downturn period.



(a) DCT in default

(b) LR in default

(c) RF in default

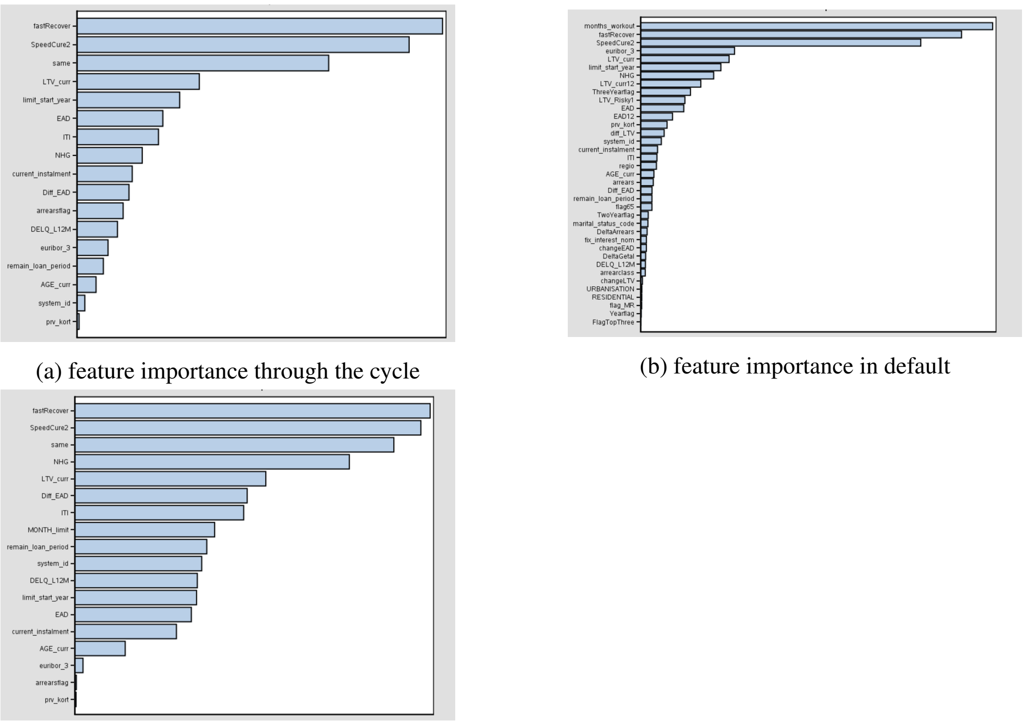
(d) GB in default

(e) NN in default

Figure 35: residual plots of LGD in default recovery rate models

The plots show that every model performs well. LR and NN have bigger deviations from the actual line than the others. It is hard to see which one is the best but if we calculate the total error we find that RF has the lowest error, followed by GB.

feature importance Let us have a look at the most important features for the recovery rate and the difference between the three cases.



(a) feature importance through the cycle

(b) feature importance in default

(c) feature importance in downturn

Figure 36: feature importances of cure rate per case

Again the difference in feature importance between recovery rate through the cycle and recovery rate in downturn is small. The importance of the downturn features is more spread out, probably due to less data. It is worth to mention that NHG has a more important role during downturn year than through the cycle. For recovery rate data we did not include the variable "_Freq_" since it correlated a lot with the variable same (unlike for the cure rate data). We observe that months workout replaces the variable "same" for in-default modeling. The rest of the features seem fairly constant for the recovery rate.

model comparison

LGD through the cycle For the recovery rate we use ASE instead of accuracy. The model with the smallest ASE is the best performing model. For LGD through the cycle this is the gradient boosting model. Gradient boosting performs slightly better than random forest so we will have to look at the residual plot for the final decision. Both struggle in the early years (2006-2008) but as we mentioned we are fine with this. We are much more interested in the last years and we see that gradient boosting does a better job in predicting these years than random forest. So gradient boosting is the best performing algorithm for the recovery rate through the cycle.

LGD in default Both the ASE and the residual plots show that RF models the LGD in default the best. The predicting power is extremely high which can be concluded from very low errors in the residuals. We see in both the cure rate model and the recovery rate model a drop in the last period (2017). This decrease in both models is caused by the sampling method. The stratified sampling approach we use, oversample the non cures and low recoveries. Therefore there are to many low values in the sample and the average drops. The stratified sampling approach does not imply oversampling, this oversampling is caused by little data in this last period (2017). We propose to train the algorithms on data until 2016. This way the model will be more informative and better describes the future cure rate and recovery rate.

LGD in downturn For LGD in downturn we observe very close results when looking at the ASE. However the residual plots show less matching results. We see large gaps in the third quarter for every model, this is due to the fact that we have very little data for that period. DCT does the best job in narrowing this gap and capturing the actual recovery rate the best. For this reason we would choose DCT as best model for the recovery rate. If we have more data for the downturn period we believe other algorithms would flourish more based on previous experiences and knowledge of the different algorithms.

9.3 In default adjustment

Our final experiment is to make an adjustment for the LGD in default models because they showed decreases in averages in the last year. This change in average was due to little data in the sample set which does not represent the original data well. To adjust for this we cap the cure rate model at January 2017 and the recovery rate model at July 2016. These thresholds have been chosen by the amount of records we oughted necessary to model. Both adjusted datasets will be trained by a random forest algorithm since it was the best performing algorithm before the adjustment.

Results adjusted models The cure rate model did not change much. The misclassification rate dropped slightly from 0.084 to 0.086. The feature importance list did not alter but the weights of the features did. These weights are more informative than the weights before due to the deletion of not accurate data. This adjusted model is more representative for the in default cure rate than the model on the full sample set. The number weighted analysis shows a slightly better plot for the adjusted model.

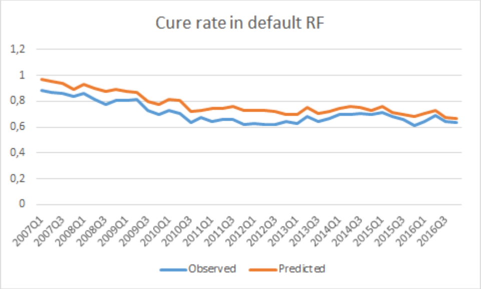


Figure 37: adjusted in default cure rate model RF

The recovery rate model improved by the adjustments.The ASE stayed the same 0.014 to 0.014 but the error of the fit decreased. Interesting to observe is also the change in feature importance.

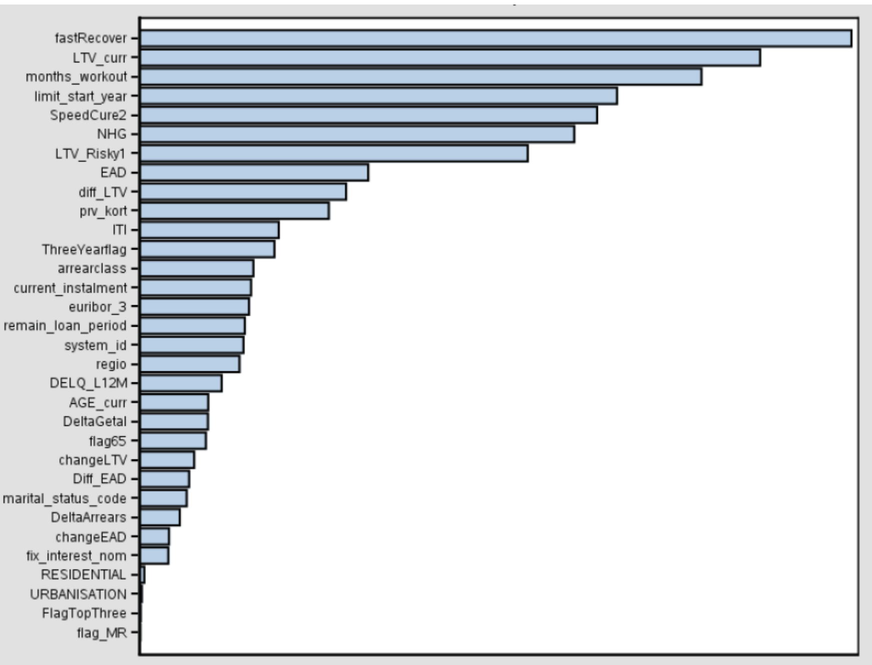


Figure 38: feature importance adjusted recovery rate model

Fastrecover rises to the most important feature, months workout and euribor3 become less important. Probably the importance of the fastrecover variable interfered with the noise in the data before. This feature importance list seems more reliable as it matches the recovery rate through the cycle featurelist more. As the error in the fit decreases the fit is also slightly better.

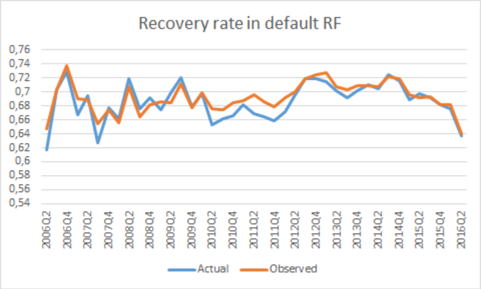


Figure 39: adjusted in default recovery rate model RF

9.4 Summary of results

To sum up the best algorithms for the six different cases:

Summary	
Case	Best suited algorithm
Cure rate through the cycle	Random forest
Cure rate in default	Random forest
Cure rate in downturn	Neural Network
Recovery rate through the cycle	Gradient Boosting
Recovery rate in default	Random Forest
Recovery rate in downturn	Decision Tree

Random forest dominates the table with being the best of 3 out of 6 cases. Neural network is surprising for cure rate in downturn as this algorithm usually needs lots of training data.

9.5 Scorecards

To get a better understanding of the models and how the different features interact with the predictions of the cure rate and recovery rate, we generated scorecards of the best algorithms per case we just discussed. These scorecards show how the model comes to the prediction based on points. The prediction of the target variable is a sum of these points which are between 0 and 1000. If an observation scores 1000 points than the algorithm will predicts a score of 100%, vice versa if an observation scores 0 points than the algorithm predict 0% recovery or cure. If a variable has a high feature importance this will also lead to a larger share of points to pass out. The scorecards are in the appendix.

10 Discussion and Conclusion

During this research, we modeled the Loss Given Default (LGD) under different conditions and with different models. We first explained the background of the problem and why we are taking this new approach, also we discussed several models of this approach. Secondly, we inspected the data, cleaned the data, made a univariate analysis and transformed some variables. This was done to get a dataset for each case we are investigating. These datasets were used to train models to predict the two key components of the LGD formula (cure rate and recovery rate). Next to manipulating the data we also tuned the hyperparameters with grid search. For in default modeling we had to use stratified sampling to get a representative dataset and for in downturn modeling we had to decide which year to pick for downturn modeling. All these steps led to the two experiments cure rate & recovery rate where we compared 5 models for each case to see which performs the best. The first goal we had, was whether the LGD could be modeled with machine learning and the second goal was to get the best performing model per case. We conclude from our research that the new LGD methodology, modeling based on characteristics of customers and estimating the cure rate and recovery rate with machine learning, is a promising and reliable way for banks to model their LGD. Having this said the models can be improved further and analysis could be further expanded. For instance, in the starting database of ING there were a lot of re-defaults. Most of them are filtered out but there all still some re-defaults. If these are filtered out we have a cleaner dataset. Another data-related improvement is the increase of recovery rate in downturn data. Right now we are restricted to using 1 year of data which is a constraint of the amount of data. If we would use multiple years this might improve predictions for the downturn model. A couple of minor improvements could be made such as providing the feature importance for all algorithms instead of just random forest to gain more insight. Different ensemble methods or machine learning techniques could be used to try to get better predictions. At last the hyperparameter optimization for recovery rate can be improved. At the moment it is not possible to do grid search for a continuous target variable, so the optimization was done by hand. This probably led to a local maximum of hyperparameters settings.

11 References

References

[1] E. Spuchlakova & J. Cug, *Credit Risk and LGD modelling*

[2] P Christian, *Estimating Loss Given Default – Experiences from Banking Practice*

[3] P. Crosbi, J.J.R. Bohn, *Modelling Default Risk, Modelling Methodology*

[4] D. Gorter, *Added value of machine learning in retail credit risk*

[5] F. Martinsson, *Exotic approaches for modelling Loss Given Default*

[6] Tim Slingerland, Arjen Pengen *Scorecard for analytics*

[7] Bill Huajian Yang and Mykola Tkachenko *Modeling of EAD and LGD: Empirical Approaches and Technical Implementation*

[8] Gilles Louppe *Understanding random forests*, section 4 (random forests)

[9] Ben Gorman *http://blog.kaggle.com/2017/01/23/a-kaggle-master-explains-gradient-boosting/*

[10] Alexey Natekin, Alois Knoll *Gradient boosting machines, a tutorial*

[11] Min Qi, Xinlei Zhao *Comparison of modeling methods for Loss Given Default*

[12] Guoqiang Zhang , Michael Y. Hu , B. Eddy Patuwo , Daniel C. Indro *Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis*

[13] Gilles Louppe *Understanding random forests*, section 2.4.3 (neural networks)

[14] Prashant Gupta *https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052*

[15] Sandro Sperandei *Understanding logistic regression analysis*

[16] Niklas Donges *https://towardsdatascience.com/the-logistic-regression-algorithm-75fe48e21cfa*