# Machine Learning for Credit Scoring

Lucas BLANCHETON, Alexandre BONICEL, Gabriel OLLIER

Spring Semester 2022

# Contents

# 1 Introduction

## 1.1 Presentation of the subject

Bankers are the people who make loans, they are responsible for something which is very important and useful for the economy. This money creation process lets the economy works having something which makes exchanges possible. Also it lets people make investments, they can buy their houses for instance, and helps for the development of the economy trough the country or even worldwide.

Yet bankers can face a problem : the default risk which is a real threat for them. They can make loans but if they are not paid back by customers who make default, they can lose a huge amount of money. It can even lead to a bankruptcy in the worst case. Obviously, the aim of the bank is not to lose money but to make the maximum profit like most of the companies. Risk default is a crucial issue for each bank. If they impose drastic conditions to their borrowers it is for a reason, they do not want people to struggle too much with their loans and they do not want to lose money too.

In order to avoid to lose money, they do not lend to everybody. People need to have a good bank record otherwise they are not allowed to borrow money from the bank. In order to assess the risk of each borrower, banks use statistical methods to attribute a grade to each loan depending on the borrower. The grades show if people are more likely to respect their engagements and if they will use their credits properly. According to those grades, people cannot obtain the same credits. Banks make loans with different interest rates according to the default risk of each borrower. On his side regarding at the IFRS 9 regulation which is an international regulation relative to bank liabilities, provisions are made to counter the default risk. Basically, banks make money reserves so if a customer makes default, they can absorb the shock but in this case they lose money.

## 1.2 Goal of the project

Now bankers know that they need to grade borrowers. So our new issue is to know how to grade in the best way to avoid problems. Grades are not everything because unexpected events can happen but most of the time they represent the situation of each borrower properly.

To find a solution to that problem we will work on grade system of LendingClub. It is an American peer-to-peer lending company which uses grade from A to G. The best one is A and the worst one is G. Grades are not standard, we can find other systems.

Our goal is to make a machine learning model which could attribute a grade using some information about the borrower. To do that we will use a data set

coming from LendingClub and the idea would be doing a regression on the relevant characteristics of people.

We will explain you what is machine learning and why is it interesting for banks and companies to use it nowadays. Then we will built a model to grade borrower thanks to Machine Learning.

## 2 Team Presentation

We all three are attending courses at ISFA school in Lyon to become actuaries. We are doing our first year of master.

Lucas BLANCHETON

Alexandre BONICEL

Gabriel OLLIER

# 3 Challenges of the subject

## 3.1 The importance of credit scoring

As we said, the main risk for banks is the default risk and they want to limit it. To do that they have several possibilities. For instance they can apply a higher interest rate for people with more risks. Also they can refuse to borrow money to someone.

Even if it can seem a bit rude to select "good" borrowers and leave "bad" borrowers, it is important to do it. Otherwise it can be a catastrophe. The best example is the 2008 Sub-primes Crisis.

In the 2000s in the USA, bankers lent money to everyone without considering the default risk. They were paid by commission so it was very attractive for them to sell as many mortgages as possible. Americans lent a lot of money to buy their own houses and the huge demand made the prices exponentially increasing. It was not a problem as getting a loan was so easy. Unfortunately this was Utopian.
When the loans were not paid back, people had to sell their houses but as they were a lot in that case, the prices fell down. That means that even though they sold their houses they still had not enough money to reimburse the loan.
People in this horrible situation were so many that the default rate made the Lehman Brothers Bank going bankrupt. Then one of the worst worldwide financial crisis ever followed and we still are recovering from it in 2022.

We hope that it convinced you about the importance of managing default risk.

So banks select people to balance their default risk but we face another problem. Are we sure that banks select people with objective criteria? Indeed they can have bias like knowing the person, thinking the person is trusted according to its looking etc... It seems obviously wrong to act like that but in reality bankers are human and can have cognitive bias.

The best response is grading people. It sounds a bit like the favorite way of processing of a famous communist country but it reminds to be the safest way to avoid financial crisis. It offers an objective criteria to manage the default risk.

## 3.2 The evolution of machine learning

Machine learning is a process of artificial intelligence (IA). Giving some information to the computer, it can predict results on data. In our case we will give relevant information relative to borrowers and grades they have to the system. When the data set is trained, we use our model to predict grades for other customers by only giving information relative to the new ones. This is going to be

our use of machine learning but it can be used for several purposes, for instance to recognize images.

Machine learning is more and more used even if it is a kind of black box sometimes. We do not always know the functioning of algorithms which are behind the results. However it seems to give interesting results. Today it is not that complicated to use it because we can store a huge amount of data easily. It is what we call Big Data. Moreover with this amount of data, it is more simple to use machine learning model than making a manual process with Excel or Rstudio.

A lot of algorithms have been developed by many people who shared their works. As a consequence everybody can use it easily and this is the reason why every companies started to use it too.

Machine learning seems to give more accurate results than deterministic methods. Consequently more and more people are interested in this way of processing and solving some problems. (Why is it possible and efficient)

# 4 Theory of Machine Learning for Credit Scoring

## 4.1 Importance of loan

money creation cours 1st year

## 4.2 Functioning of Machine Learning

explication theorique, complexite par exemple

# 5 Methodology of the project

## 5.1 Data cleaning

## 5.2 Regression

# 6 Results

# 7 Project Management

# 8 Conclusion

# 9 Bibliography

*LendingClub Website* :
`https://www.lendingclub.com/`
*Loan grades information* :
`https://www.lendingclub.com/foliofn/rateDetail.action`
*Benefit of Credit Scoring to Banks* :
`https://www.herald.co.zw/benefits-of-credit-scoring-to-banks/`
*The Big Short* :
2015 movie on the Sub-primes mortgages crisis from Adam McKay.

# 10 Annex

## 10.1 The rough data

The data set that we chose comes from Kaggle. It can be downloaded **here**. It is data on many mortgages from LendingClub between 2007 and 2014. LendingClub is a peer-to-peer lending company headquartered in San Francisco, California. It was created in 2006.

In the data each row is a loan and each column is a characteristic of the loan. There are a lot of characteristics as there are 74 columns. Some of them are almost empty or useless for our work so we won't deal with them. Nevertheless we still have many and enough parameters.

Let's introduce the parameters :

**id** : A unique LC assigned ID for the loan listing.
**member_id** : A unique LC assigned Id for the borrower member.
**loan_amnt** : The listed amount of the loan applied for by the borrower.
**funded_amnt** : The total amount committed to that loan at that point in time.
**funded_amnt_inv** : The total amount committed by investors for that loan at that point in time.
**term** : The number of payments on the loan. Values are in months and can be either 36 or 60.
**int_rate** : Interest Rate on the loan
**installment** : The monthly payment owed by the borrower if the loan originates.
**grade** : LendingClub assigned loan grade.
**sub_grade** : LendingClub assigned loan subgrade.
**emp_title** : The job title supplied by the Borrower when applying for the loan.
**emp_length** : Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more.
**home_ownership** : The home ownership status provided by the borrower during registration.
**annual_inc** : The self-reported annual income provided by the borrower during registration.
**verification_status** : Verified, source verified or not verified.
**issue_d** : The month which the loan was funded.
**loan_status** : Current status of the loan.
**pymnt_plan** : Indicates if a payment plan has been put in place for the loan.
**url** : URL for the LendingClub page with listing data.
**desc** : Loan description provided by the borrower.
**purpose** : A category provided by the borrower for the loan request.
**title** : The loan title provided by the borrower.
**zip_code** : The first 3 numbers of the zip code provided by the borrower in the loan application.

**addr_state** : The state provided by the borrower in the loan application.
**dti** : A ratio calculated using the borrower's total monthly debt payments on the total debt obligations.
**delinq_2yrs** : The number of 30+ days past-due incidences of delinquency in the borrower's credit file for the past 2 years.
**earliest_cr_line** : The month the borrower's earliest reported credit line was opened.
**inq_last_6mths** : The number of inquiries in past 6 months (excluding auto and mortgage inquiries).
**mths_since_last_delinq** : The number of months since last delinquency.
**mths_since_last_record** : The number of months since the last public record.
**open_acc** : The number of open credit lines in the borrower's credit file.
**pub_rec** : Number of derogatory public records.
**revol_bal** : Total credit revolving balance.
**revol_util** : Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
**total_acc** : The total number of credit lines currently in the borrower's credit file.
**initial_list_status** : The initial listing status of the loan. Possible values are – W, F.
**out_prncp** : Remaining outstanding principal for total amount funded.
**out_prncp_inv** : Remaining outstanding principal for portion of total amount funded by investors.
**total_pymnt** : Payments received to date for total amount funded.
**total_pymnt_inv** : Payments received to date for portion of total amount funded by investors.
**total_rec_prncp** : Principal received to date.
**total_rec_int** : Interest received to date.
**total_rec_late_fee** : Late fees received to date.
**recoveries** : Post charge off gross recovery.
**collection_recovery_fee** : Post charge off collection fee.
**last_pymnt_d** : Last month payment was received.
**last_pymnt_amnt** : Last total payment amount received.
**next_pymnt_d** : Next scheduled payment date.
**last_credit_pull_d** : The most recent month LendingClub pulled credit for this loan.
**collections_12_mths_ex_med** : Number of collections in 12 months excluding medical collections.
**policy_code** : Publicly available policy_code=1 new products not publicly available policy_code=2.
**application_type** : Indicates whether the loan is an individual application or a joint application with two co-borrowers.
**acc_now_delinq** : The number of accounts on which the borrower is now delinquent.

There are a lot of parameters and we will have to select which ones we keep and

which ones we leave.
There are even more parameters in the original data set but we did not introduce whose with too many missing values.

## 10.2 Presentation of grades

LendingClub gives a grade for every borrowers. The grade reflects the risk from the borrower for LendingClub. Those grades are letters between A and G and have the following characteristics :

A : Risk of default is negligible, interest rate between 8.46% and 10.81%.
B : Risk of default is very low, interest grade between 13.33% and 16.08%.
C : Risk of default is moderate, interest rate between 17.30% and 20.74%.
D : Risk of default is average, interest rate between 22.62% and 30.99%.
E : Risk of default is possible, interest rate between 28.90% and 29.00%.
F : Risk of default is likely, interest rate between 29.35% and 30.75%.
G : Risk of default is very high, interest rate between 30.79% and 30.99%.

It is those grades that we want to predict using machine learning.

## 10.3 Suppression of null values

To begin with we want to remove the missing values. We tried to remove every rows with at least a missing value but it deleted the whole dataset. We decided to change the strategy by deleting columns with missing value. It is 'less professional' but it is an easy way to remove many parameters which seemed useless while keeping the same number of observations.
It is a good idea but we have a new problem, we delete some important parameters which have only few missing values such as *emp_length*. We need to change that.

Our new idea is to remove only parameters which have more than $k$ missing values. It let us to keep some important values. Then we can remove rows with missing values. We have to chose $k$ to select the columns we do not want to lose.

**Our final way to deal with missing values :**

We decide to remove columns with more than 10% of missing values. This value has been found after several trials. It appears to be the best one to keep features we want and drop those we do not want.

Then we remove rows with missing values. With this method we do not drop too many observations.

## 10.4 Qualitative study

We are going to delete parameters which are obviously useless. To do that we will discuss about them qualitatively.
**id** and **member_id** : They are clearly useless for our model. Indeed they are only random numbers. **We do not keep it**.
**int_rate** : We hesitated for that one. If the interest rate comes from the grades, it is not legit to use it to find the grade. Yet we are not sure that it comes from grades so **we keep it**.
**sub_grade** : This parameter comes from the grades so we cannot use it to find the grade. On the one hand, using it would be cheating and on the other hand we do not have it before having the grade. **We do not keep it**.
**emp_title** : There are too many null values, it should have already been deleted.
**loan_status** : It says if the loan is fully paid back or not. As we cannot know that at the beginning of the loan, it is irrelevant. **We do not use it**.
**pymnt_plan** : There are only 9 'True' and every other values are 'False'. It is not relevant to build a model. **We erase it as well**.
**url** : There is one unique categorical value for each row. It is useless. **We do not keep it**.
**desc** : It is a description of the use of the loan by the borrower. It is full sentences which are not usable. **We do not keep it**. Moreover there are some null values so it should have already been deleted.
**title** : As desc it is sentences written by the borrowers such as "*My wedding loan I promise to pay back*". It is funny but not usable. **We do not keep it**.
**zip_code** : There are too many different values. The base rate is 1%. **We delete it**.
**initial_list_status** : There is almost one unique value "f". It is useless for a regression or a classification. **We do not keep it**.
**application_type** : There is only one value "INDIVIDUAL". **We erase it**.

## 10.5 Dates rescaling

Now we are going to discuss about Dates Rescaling. In our dataset we can see that we have many date variables which are computed in a wrong format. That's why we are going to operate different modifications to make them fit for our needs.
First of all, we are going to modified the variable **term** to remove the **"month"** at the end of the variable.
After that, we create a function **emp_length_converter** to modify the variable **emp_length** by removing the different terminologies of the variable and only keep the number of years.
Finally, we are going to create the function **date_columns** to operate the difference between the date value in our dataset and today's date in order to have the same unit throughout the dataset i.e. the amount of months between two dates.We have applied that function to four variables : **earliest_cr_line**, **issue_d**, **last_pymnt_d** and **last_credit_pull_d**.

## 10.6 Balance of the dataset

We check for our base rate. As we have 7 outcomes, we should have a base rate around $1/7 \simeq 0.14$.
Unfortunately the current base rate is above 0.29 and it is for the categorical value $B$. We need to balance our dataset if do not want to have absurd results. Moreover the rate of categorical value $G$ is under 0.01. It is not balanced at all.

There are several ways to balance a dataset and we choose the easier one : to randomly delete observations in the over-represented groups.

The way we do it is simple. We know the number of observations for each categorical value and we want them to have the same number of the smaller group which is $G$.
We subtract the number of $G$ rows to each number of every other groups in order to know which number we need to delete in each other group.
Then we cut each groups in different dataframes. Now we have 7 dataframes; one for each grade.
For group of $A$, we define $N$ the number of rows that we need to remove. $N$ times, we randomly pick a number between 0 and the size of the dataframe of grade $A$ minus one and we delete the corresponding rows. At the end of the process we have a new dataframe for $A$ whose shape is the same as dataframe of $G$.
We do that for every other dataframe and we concatenate them. Finally we get our new balanced dataset.

While running the code we faced a little problem, the time of execution. Indeed it was approximately three hours to balance the dataset. So we had the idea to remove rows ten by ten to earn time and it worked. We could balance our dataframe in only twenty minutes.

To do that we change the following code :

```
for i in range(6):
    while  ListNbRowsToDelete[i] > 0:
        n = random.randrange(0, len(listGroup[i].index))
        listGroup[i] = listGroup[i].drop([listGroup[i].index[n]])
        ListNbRowsToDelete[i] = ListNbRowsToDelete[i] - 1
```

for that one :

```
for i in range(6):
    while  ListNbRowsToDelete[i] > 0:
        n = random.randrange(0, len(listGroup[i].index)-9)
        listGroup[i] = listGroup[i].drop([listGroup[i].index[n],
                                          listGroup[i].index[n+1],
                                          listGroup[i].index[n+2],
```

```
                                        listGroup[i].index[n+3],
                                        listGroup[i].index[n+4],
                                        listGroup[i].index[n+5],
                                        listGroup[i].index[n+6],
                                        listGroup[i].index[n+7],
                                        listGroup[i].index[n+8],
                                        listGroup[i].index[n+9]])
        ListNbRowsToDelete[i] = ListNbRowsToDelete[i] - 10
```

The second code is less random but still enough for us. The smaller group of grade that we transform has 12432 rows (group of $F$). When we delete 10 by 10, we still delete tiny parts ($10^{-3}$) each time.

## 10.7 Quantitative study

## 10.8 Label encoding

## 10.9 Rescaling