

# **FINSEARCH**

# **CREDIT**

# **SCORE**

Prepared by  
**Aashish, Gaurav**  
**Likith, Prithvi**

Mentor  
**GANADEVA BANDYOPADHYAY**

# INTRODUCTION

---

One pillar of paramount importance of Capitalism is the fractional reserve system which allows banks and other certified institutions to lend to individuals who in turn become private entrepreneurs or consumers of goods and services.

As long as the borrower is returning the debt on time, the system is in harmony and the world is a good place. The problems begin when the borrower defaults on their loan and perhaps in the worst of cases declare bankruptcy.

If the lending institution faces many of such cases, it can run out of cash and even get forced to go under. If it is a giant bank, it may even cause a dent to the entire national or global economy just like Lehmann Brothers collapse of 2008 sent shockwaves across the planet.

There are also secondary players such as insurance companies selling policies to such banks, like AIG in 2008, who may not be aware about the quality of loans the bank lends out, leading them to charging lower premiums or adverse selection.



**Adverse selection** refers generally to a situation in which sellers have information that buyers do not have, or vice versa, about some aspect of product quality.



From all this we can see it is very important for a bank to only lend to borrowers who can most certainly return their debt + interest on time. That requires the bank to know as much as possible about the borrower, his credit history, his income and his net worth spread across different asset types. But there is a problem, the borrower always knows more about themselves than a bank can know about them. This phenomenon, in a nutshell, is known as INFORMATION ASYMMETRY.

“

**Asymmetric information**, also known as "information failure", occurs when one party to an economic transaction possesses greater material knowledge than the other party.

”

In order to bridge this gap we need a system that tells us who are the candidates worthy of getting a loan of a particular sum - *note that someone may be worthy of a particular sum but not of a higher amount* - in a quantifiable metric calculated based on the borrower's data, that can then be automated into a system that takes in these huge piles of data and gives us just one information over all other, should the bank lend to this particular borrower, this particular sum, or not. Enter CREDIT SCORE

“

A **credit score** is a number from 300 to 850 that depicts a consumer's creditworthiness. The higher the score, the better a borrower looks to potential lenders.

”

A credit score is based on credit history: number of open accounts, total levels of debt, repayment history, and other factors. Lenders use credit scores to evaluate the probability that an individual will repay loans in a timely manner.

Your credit score, a statistical analysis of your creditworthiness, directly affects how much or how little you might pay for any lines of credit that you take out.

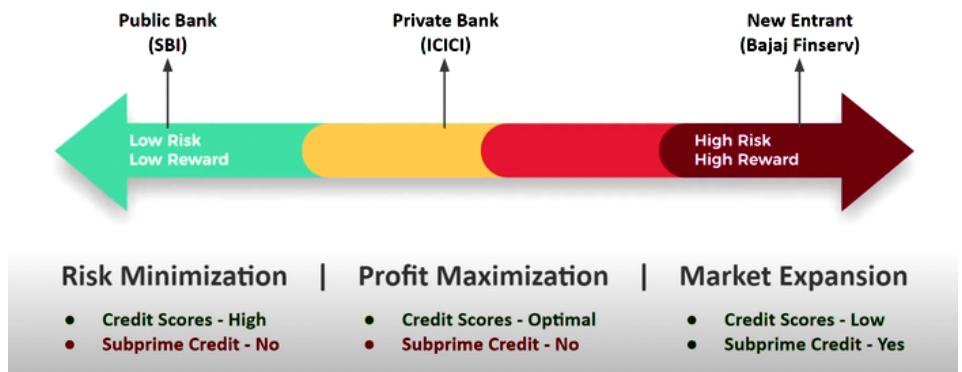
A person's credit score also may determine the size of an initial deposit required to obtain a smartphone, cable service, or utilities, or to rent an apartment. And lenders frequently review borrowers' scores, especially when deciding whether to change an interest rate or credit limit on a credit card

### **Business Case**

The second challenge depending upon your objectives as a bank, weather your objective is to maximise profit by ensuring minimum number of borrowers default on their borrowings or, to expand your business by lending out maximum number of loans which in turn implies lending to borrowers with relatively subprime credit ratings and hence suffer greater risks.

For example, if you are SBI Bank, your approach will be to lend only to individuals or businesses with a positive track record i.e. a good credit rating, but if you are bajaj fin-services your approach should be to take higher risk and lending to small scale entrepreneurs looking to create startups with tech

or otherwise for the opportunity to get maximum returns on a few bets even if it involves losing on maximum number of their lendings.



Now how to solve this issue ? Perhaps the best approach would be to have an algorithm that assigns the score to all the investments upon analysing the data input and then depending upon the risk appetite the firm can have a minimum threshold beyond which no matter what the firm will not lend to the borrower.

The second major issue is the lack of data about the borrower which makes it hard for the firm to make accurate predictions about the borrower's ability to repay. Now to solve this issue we can either use simple imputation i.e. filling the missing value with the mean median or mode, KNN imputation i.e. filling the missing value with related cases or just use removal of data which is often unfeasible as the maximum number of NA data values is 188 which can lead to huge data loss.

# DATA

Coming to the data sets, *data* plays an important role in the development, monitoring, and maintenance of credit scoring models.

The data used for credit scoring comes from diverse and multidimensional sources. For credit scoring, traditionally, credit data are used, including amount of loan, type of loan, maturity of loan, guarantees and collateral value, historical payment performance such as default information and payments in arrears, amounts owed, length of credit history, new credit, and types of credit. These data are factored into a credit score as indicators of willingness and ability to pay.

Most of data that credit companies deal with, including our data implemented in the model consists of the *Derogatories Count*.

“ —

**Derogatory marks** are negative, long-lasting indications on your credit reports that generally mean you didn't pay back a loan as agreed. For example, a late payment or bankruptcy appears on your reports as a derogatory mark.

— ”

These derogatory marks generally stay on your credit reports for up to 7 or 10 years (sometimes even longer) and damage your scores.

If you have a lower score coupled with a derogatory mark, you may have a hard time getting approved for credit or may get less-than-ideal credit terms. But the good news is that the impact to your credit of all derogatory marks decreases over time.

A **derogatory mark** can land on your credit reports in two ways:

- A creditor or lender may report negative information to the credit bureaus, which is then translated into a derogatory mark.
- Or the credit bureaus can add public records to your credit reports. These may include bankruptcies, civil judgments and tax liens.

However, thanks to stronger public-record data standards that the credit bureaus have recently agreed to, consumers nationwide will see fewer tax liens and civil judgments on their credit reports.

## **How do derogatory marks impact my scores?**

A **Derogatory mark** will damage your credit scores. But how much? That depends on a few factors:

- A *derogatory mark* typically affect a higher score more than it will a lower score. Also, a minor derogatory mark, which can be caused by a late payment, generally damages your scores less than a major derogatory mark, which can be caused by something like a foreclosure.

The amount of time a derogatory mark stays on your credit reports depends on what type of mark it is.

The chart below covers the different types of derogatory marks and how long they will likely remain on your credit scores.

<b>What can lead to a derogatory mark?</b>	<b>What is it and what happens?</b>	<b>How long might the derogatory mark appear on a credit report?</b>
<b>Late payments</b>	An account payment that is past due. This is generally the only form of a “minor” derogatory mark. After the payment is late, its severity may increase every 30 days it’s not paid.	Seven years from the date of a delinquent payment.
<b>An account in collections (or charge-off)</b>	When a creditor thinks you ultimately won’t pay what you owe, usually after several missed payments, it can write or “charge off” the account for tax purposes. After a creditor has charged off the account, it can sell it to a third-party collections agency. The collections company will try to get a payment from the borrower.	Seven years from the first date of a delinquent payment.
<b><u>Bankruptcy</u></b>	This is a special legal proceeding you can enter to request relief from debt obligations. You’ll either pay back some or none of your debt.	7 to 10 years from the filing date, depending on the type of bankruptcy.

<b>Civil judgment</b>	If you've lost a civil lawsuit that requires you to pay debt or damages, it can appear on your credit reports.	Paid civil judgment: Seven years from the date the judgment was filed.  Unpaid civil judgment: The seven-year time frame may be renewed depending on local laws.
<b>Debt settlement</b>	You and a creditor can reach an agreement where you pay back only part of the debt you owe.	Seven years from either the date the debt was settled or from the date of the first delinquent payment, depending on whether there were missed payments.
<b>Foreclosure</b>	A foreclosure can happen if you fall seriously behind or miss many of your mortgage payments. The bank will attempt to force a sale of the home, which is then used as collateral for the mortgage loan.	Seven years from the filing date.
<b>Tax lien</b>	If you fail to pay your taxes, the federal government will attempt to collect your debt by placing a lien, which is a claim, against your property.	Paid tax lien: Seven years from the filing date.  Unpaid tax lien: Can remain on a report

Just like derogatory marks have an entire history, we need a mechanism to track the history of every single open transaction and everything that happened in it, from the loan being issued, to a small amount being paid back to it being eventually defaulted or paid off, needs to be in chronological sync.

### Enter Trade Lines

“ —

**Trade line** is a number indicating the number of credit accounts a borrower has, be it: loan, credit card, or other debt obligations.

— ”

So, basically if you have a credit card and a home loan at a certain point in time, you have 2 trade lines open.

All credit reports contain a separate ‘tradeline’ for each credit account which depicts the nature of the user’s account, credit payment history, the date on which the credit was sanctioned, etc. All the information presented in the tradelines is used to determine the user’s credit score.

A tradeline records all the credit activities of the user including the nature of the credit account, number of accounts, total credit amount, etc.

Each credit account has a separate tradeline. For example- If Mr. X has 4 credit accounts, his credit report will show 4 separate tradelines consisting of the details of the credit provider, the total debt amount, payment history, etc.

The tradeline carries all the details like the current payment status, date of opening the credit account, the details of the lender, total credit amount, address, etc.

Tradelines are available on the credit report and are used to calculate the user's credit score

## **What are the Components of a Tradeline?**

**Tradelines** are created on the basis of users' total number of credit accounts and their credit activities. Credit bureaus like Equifax, Experian, TransUnion use tradelines to determine the user's credit score.

A tradeline consists of the following details-

- Name and address of the credit borrower
- Partial account number of the borrower
- Occupation of the borrower
- Date on which the account was opened
- Type of account- Loan or Credit Card
- Date of the user's most recent credit activity
- Loan or credit amount and current balance
- Credit payment history
- Minimum monthly payment, rate of interest, payment frequency, date of last payment, etc.
- Credit limit, sanctioned amount, current balance and amount overdue
- Account responsibility- whether the user owns the account

**Note** – Each Credit bureau like Equifax, Experian, TransUnion use the credit report differently; hence, there might be some variation in the credit report.

## What are the Types of Tradelines?

Number of credit accounts carries 'n' number of tradelines. No tradeline is identical to another tradeline and hence, are divided into different variants-

“ —

**Revolving accounts :** Available credit, payments, balance changes as and when the user makes payment or purchases

Example- Credit cards or lines of credit

**Installment loans :** These are the accounts in which the borrower borrows a fixed amount and is required to repay it on fixed terms

Example- Student loans, personal loans, mortgages, auto loans

— ”

# EXPLORATORY DATA ANALYSIS

Every single credit scoring company deals with tons of data that they analyse using models powered by K Neighbour Classifiers, Random Forest, Logistic Regression, etc. Now we will present to you a model comprising of data from 3000 customers, their individual IDs and 28 Variables for each customer based on their credit history and transactional behaviour categorising them as defaulter or punctual, providing hallmark for their credibility.

Following is the list of variables we have in our data set :

'ID', 'DerogCnt', 'CollectCnt', 'BanruptcyInd', 'InqCnt06', 'InqTimeLast',  
'InqFinanceCnt24', 'TLTimeFirst', 'TLTimeLast', 'TLCnt03', 'TLCnt12',  
'TLCnt24', 'TLCnt', 'TLSum', 'TLMaxSum', 'TLSatCnt', 'TLDel60Cnt',  
'TLBadCnt24', 'TL75UtilCnt', 'TL50UtilCnt', 'TLBalHCPct', 'TLSatPct',  
'TLDel3060Cnt24', 'TLDel90Cnt24', 'TLDel60CntAll', 'TLOpenPct',  
'TLBadDerogCnt', 'TLDel60Cnt24', 'TLOpen24Pct'

Variable Name	Label	Role
Target	Target = 1 (Defaulters), Target = 0 (Good Loans)	Target
BanruptcyInd	Bankruptcy Indicator	Input
TLBadDerogCnt	Bad Dept plus Public Derogatories	Input
CollectCnt	Collections	Input
InqFinanceCnt24	Finance Inquires 24 Months	Input
InqCnt06	Inquiries 6 Months	Input
DerogCnt	Number Public Derogatories	Input
TLDel3060Cnt24	Number Trade Lines 30 or 60 Days 24 Months	Input
TL50Utilent	Number Trade Lines 50 pct Utilized	Input

TDel60Cnt24	Number Trade Lines 60 Days or Worse 24 Months	Input
TDe160CntAll	Number Trade Lines 60 Days or Worse Ever	Input
T75UtilCnt	Number Trade Lines 75 pct Utilized	Input
Tel90Cnt24	Number Trade Lines 90+ 24 Months	Input
TLBadCnt24	Number Trade Lines Bad Debt 24 Months	Input
TLDel60Cnt	Number Trade Lines Currently 60 Days or Worse	Input
TLSatCnt	Number Trade Lines Currently Satisfactory	Input
TLCnt12	Number Trade Lines Opened 12 Months	Input
TLCnt24	Number Trade Lines Opened 24 Months	Input
TLCnt03	Number Trade Lines Opened 3 Months	Input
TSatPct	Percent Satisfactory to Total Trade Lines	Input
TLBalHCPct	Percent Trade Line Balance to High Credit	Input
TLOpenPct	Percent Trade Lines Open	Input
TLOpen24Pct	Percent Trade Lines Open 24 Months	Input
TLTimeFirst	Time Since First Trade Line	Input
InqTimeLast	Time Since Last Inquiry	Input
TLTimeLast	Time Since Last Trade Line	Input
TLSum	Total Balance All Trade Lines	Input
TLMaxSum	Total High Credit All Trade Lines	Input
TL_Cnt	Total Open Trade Lines	Input
ID	Customer ID	ID

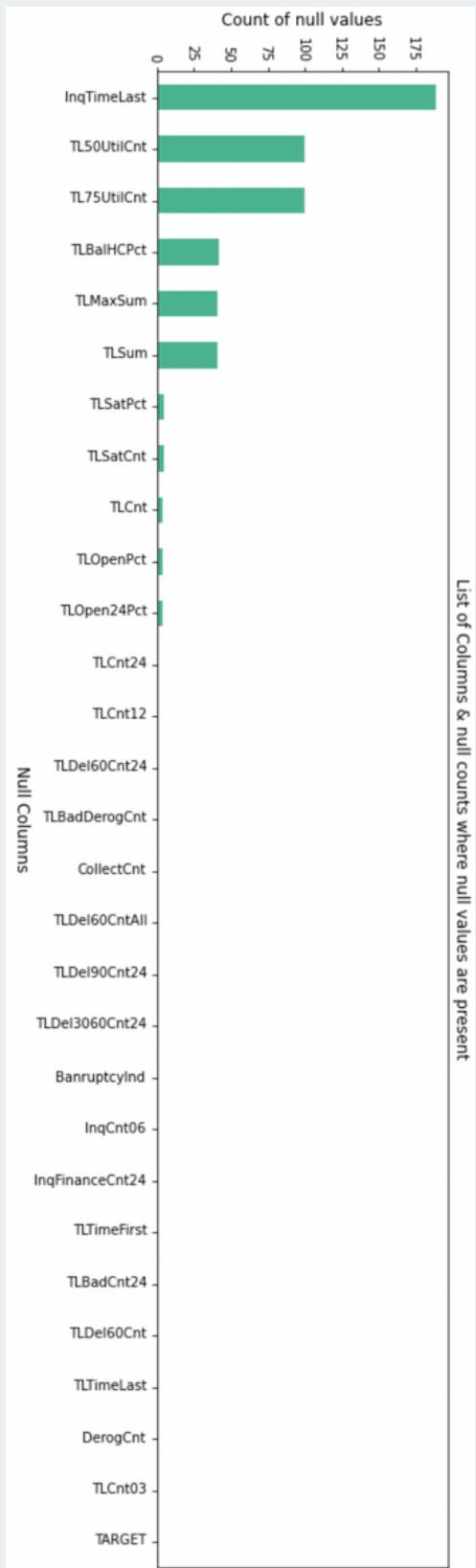
- As IDs are inert variables which do not impact our analysis, we will drop that column.
- We have taken the data of total 3000 customers, out of which 2500 are found to be normal, whereas 500 are fraudulent indicating a fraud rate of 20%..

## Identify and treatment of nulls in Data

#	Column	Non-Null Count
0	TARGET	3000 non-null
1	DerogCnt	3000 non-null
2	CollectCnt	3000 non-null
3	BanruptcyInd	3000 non-null
4	InqCnt06	3000 non-null
5	InqTimeLast	2812 non-null
6	InqFinanceCnt24	3000 non-null
7	TLTimeFirst	3000 non-null
8	TLTimeLast	3000 non-null
9	TLCnt03	3000 non-null
10	TLCnt12	3000 non-null
11	TLCnt24	3000 non-null
12	TLCnt	2997 non-null
13	TLSum	2960 non-null
14	TLMaxSum	2960 non-null
15	TLSatCnt	2996 non-null
16	TLDel60Cnt	3000 non-null
17	TLBadCnt24	3000 non-null
18	TL75UtilCnt	2901 non-null
19	TL50UtilCnt	2901 non-null
20	TLBalHCPct	2959 non-null
21	TLSatPct	2996 non-null
22	TLDel3060Cnt24	3000 non-null
23	TLDel90Cnt24	3000 non-null
24	TLDel60CntAll	3000 non-null
25	TLOpenPct	2997 non-null
26	TLBadDerogCnt	3000 non-null
27	TLDel60Cnt24	3000 non-null
28	TLOpen24Pct	2997 non-null

Upon observing the data set, we find that it contains a lot of null entries which maybe a result of domain specific problems, unavailability of significant credit history or corrupt measurements.

- InqTimeLast has 188 values missing
- TLCnt has 3 values missing
- TLSum has 40 values missing
- TLMaxSum has 40 values missing
- TLSatCnt has 4 values missing
- TL75UtilCnt has 99 values missing
- TL50UtilCnt has 99 values missing
- TLBalHCPct has 41 values missing
- TLSatPct has 4 values missing

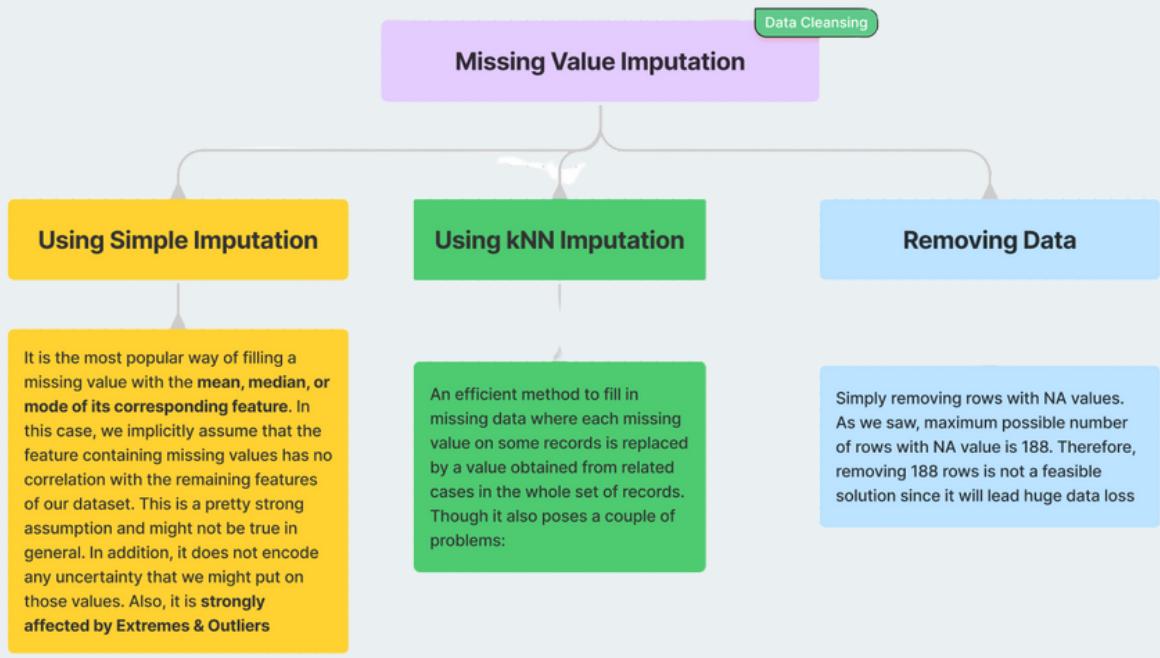


- TLOpenPct has 3 values missing
- TLOpen24Pct has 3 values missing

Now as we saw that the maximum number of NA values is 188 and their direct removal is going to cause significant data loss and hence this method is infeasible, henceforth we will use, imputation, the process of replacing missing data using substituted values.

The objective of these procedures is to create a collection of data objects that will practically support the next step of identifying useful features by making sure that all quantitative data meet quality requirements for automated processing.

We will create a model to predict the missing values, created for each feature that has missing values.



## Simple Imputation :

Is basically the process of completing our dataset by filling the missing information using the row average method for that feature variable assuming no correlation with other features in our dataset. The behaviour of particular customer's missing credit info is assumed similar to average of other customer. The model is also affected by outliers and lead to loss of a potential good customers due to highly non uniform behaviour of defaulters.

For example : As shown in the table the deviation between the mean and the maximum value of the data sets is humungous which are the outliers with non uniform behaviour skewing the mean. This can have adverse implications such as good customers losing out on loans as their score is demeaned using this skewed mean instead of the average of majority population.

	InqTimeLast	TLCnt	TLCnt03	TLCnt12	TLCnt24	TLCntAll	TLCnt06	TLCnt24Pct	TLCntAllPct	TLBadCnt	TLBadPct	TLOpenPct	TLOpen24Pct
count	1937500000	2084.000000	2052.000000	2052.000000	2086.000000	2014.000000	2014.000000	2051.000000	2086.000000	2084.000000	2084.000000	2084.000000	2084.000000
mean	3.255550	8.071497	20690.765595	33117.058480	13.744966	3.053128	4.048659	0.634520	0.590458	0.524225	0.585511	0.210127	0.455571
std	4.807047	5.580913	19547.089583	30071.449678	9.124619	2.562610	3.049617	0.264114	0.213296	0.213296	0.213296	0.213296	0.213296
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	1.000000	4.000000	6314.750000	11129.500000	6.000000	1.000000	2.000000	0.488600	0.461500	0.375000	0.375000	0.285700	0.285700
50%	1.000000	7.000000	16468.500000	26026.000000	12.000000	2.000000	3.000000	0.677500	0.608700	0.500000	0.500000	0.500000	0.500000
75%	4.000000	11.000000	29244.500000	47330.250000	19.000000	4.000000	6.000000	0.828700	0.733300	0.653800	0.653800	0.592300	0.592300
max	44.000000	40.000000	207979.000000	271036.000000	57.000000	17.000000	23.000000	3.361300	1.000000	1.000000	1.000000	6.000000	6.000000

## kNN Imputation:

is basically a model created for each feature that has missing values, taking as input values of perhaps all other input features. Using K-nearest neighbour model, a new sample is imputed by finding the samples in the training set “closest” to it and averaging the nearby points to fill in the value.

We will show that *kNN impute* appears to provide a more robust and sensitive method for missing value estimation and kNN impute surpass the commonly used *row average method* (as well as filling missing values with zeros). Configuration of KNN imputation often involves selecting the distance measure (e.g. Euclidean) and the number of contributing

TARGET	3000	non-null
DerogCnt	3000	non-null
CollectCnt	3000	non-null
BanruptcyInd	3000	non-null
InqCnt06	3000	non-null
InqTimeLast	3000	non-null
InqFinanceCnt24	3000	non-null
TLTimeFirst	3000	non-null
TLTimeLast	3000	non-null
TLCnt03	3000	non-null
TLCnt12	3000	non-null
TLCnt24	3000	non-null
TLCnt	3000	non-null
TLSum	3000	non-null
TLMaxSum	3000	non-null
TLSatCnt	3000	non-null
TLDel60Cnt	3000	non-null
TLBadCnt24	3000	non-null
TL75UtilCnt	3000	non-null
TL50UtilCnt	3000	non-null
TLBalHCPct	3000	non-null
TLSatPct	3000	non-null
TLDel3060Cnt24	3000	non-null
TLDel90Cnt24	3000	non-null
TLDel60CntAll	3000	non-null
TL0penPct	3000	non-null
TLBadDerogCnt	3000	non-null
TLDel60Cnt24	3000	non-null
TL0pen24Pct	3000	non-null

neighbours for each prediction, the k hyperparameter of the KNN algorithm.

The primary advantage of KNN imputation over simple imputation is its feature of not getting skewed due to certain outlier data sets. This was the primary reason good borrowers were losing out on debt due to a skewed mean, but since KNN only considers nearby neighbours, the problem is solved.

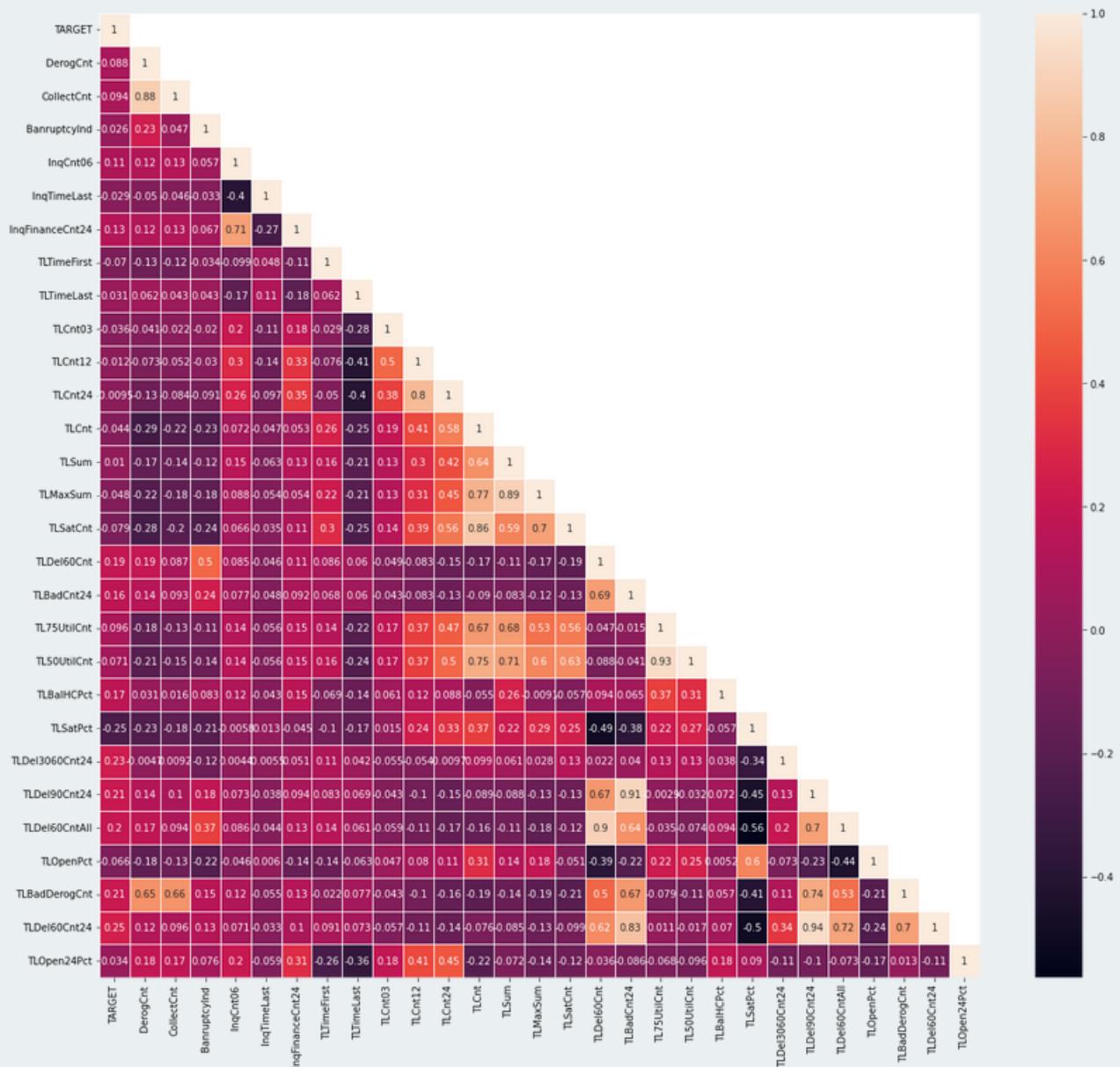
## **Outliers & Correlation In Data - Correlation**

Till now we have imputed for missing values and made our data useable by the computer. Now let us understand about the interdependence of variables (*correlation*) and how certain variables heavily impact our computation (*outliers*).

**Data Correlation:** Is a way to understand the relationship between multiple variables and attributes in our dataset using statistics

Using Correlation, you can get some insights such as:

- One or multiple attributes depend on another attribute or are a cause for another attribute. It helps us in imputation, assuming similar behaviours among customers with similar credit records.
- One or multiple attributes are associated with other attributes which may cause multicollinearity, hence, issues in prediction



Above is a correlation heat-map for our data.

The variables which are having maximum negative impact on the probability of a customer defaulting are (with their correlation coefficient):

- TLDel60Cnt24 (0.252026)
- TLDel3060Cnt24 (0.233709)
- TLDel90Cnt24 (0.211577)

Why is it so ? Lets' go back to what TLDelCnt denotes

**TelDel\_n\_Cnt24** denotes the number of trades lines currently held by the customer with a period of 24 months due past n days.

It represents the number of times the subject has been 30, 60, or 90 days delinquent, respectively.

With n as 60 days or worse, a 24 Month long trade-line is worsened and should highly reduce the credibility of the customer.

Similar behaviour is observed for trade lines with a period of 24 months due past 30 or 60 days

The variables that will have maximum negative correlation - positive impact on credibility of a customer are

- TLCnt03 (-0.036116)
- TLCnt12 (-0.012391)
- TLCnt24 (-0.009525)

“ —

A tradeline helps you improve your credit score until you close it within the given period, so it will reap all the benefits a good credit score enables you to achieve.”

— ”

Even seasoned trade-lines are considered to be the best trade-lines as they can significantly affect your credit score because of its credit history.

“ —

A report by Finance Monthly states that purchasing 2-3 seasoned tradelines can help increase your credit score by 720-850 in just one month.

— ”

The same is visible in our data also. **TLCnt\_n** denotes the number of trade lines opened in the last n months. Note that these trades may now be open or closed, paid as agreed, delinquent, or derogatory, and so on.

As more number of trade-lines are opened in a shorter duration, it has a higher impact on improvement of borrower's credibility.

Also, **TLSatPct** having a correlation coefficient of -0.253949, is having the maximum impact on improving one's credit score, since it denotes "Percent Satisfactory to Total Trade Lines" i.e. the percentage number of trade

TARGET	1.000000
TLDel60Cnt24	0.252026
TLDel3060Cnt24	0.233709
TLDel90Cnt24	0.211577
TLBadDerogCnt	0.208152
TLDel60CntAll	0.195863
TLDel60Cnt	0.185942
TLBalHCPct	0.167845
TLBadCnt24	0.163796
InqFinanceCnt24	0.132562
InqCnt06	0.106663
TL75UtilCnt	0.095996
CollectCnt	0.093748
DerogCnt	0.088100
TL50UtilCnt	0.070804
TLOpen24Pct	0.033585
TLTimeLast	0.030648
BanruptcyInd	0.025651
TLSum	0.009988
TLCnt24	-0.009525
TLCnt12	-0.012391
InqTimeLast	-0.028542
TLCnt03	-0.036116
TLCnt	-0.044204
TLMaxSum	-0.047951
TLOpenPct	-0.065911
TLTimeFirst	-0.069597
TLSatCnt	-0.079046
TLSatPct	-0.253949

lines closed on time by the borrower. The same is visible in our data also.

## **Outliers & Correlation In Data - Outliers**

Now we are clear about the correlation of data variables among themselves, it can be felt that if a certain dataset is significantly larger or smaller than the remaining pack, it can have skewing impact on our measure of central tendencies, in our model, the mean. An outlier maybe caused due to variability in data, or due to experimental or human errors, but it can also be due to a case of *fraudulent transaction*.

Thus detection of these frauds is in many ways similar to detection of the outliers. The frauds are data sets with high payments, high rate of purchase, data items that the customer never purchased, etc which reflect as outliers in our data more often than not.

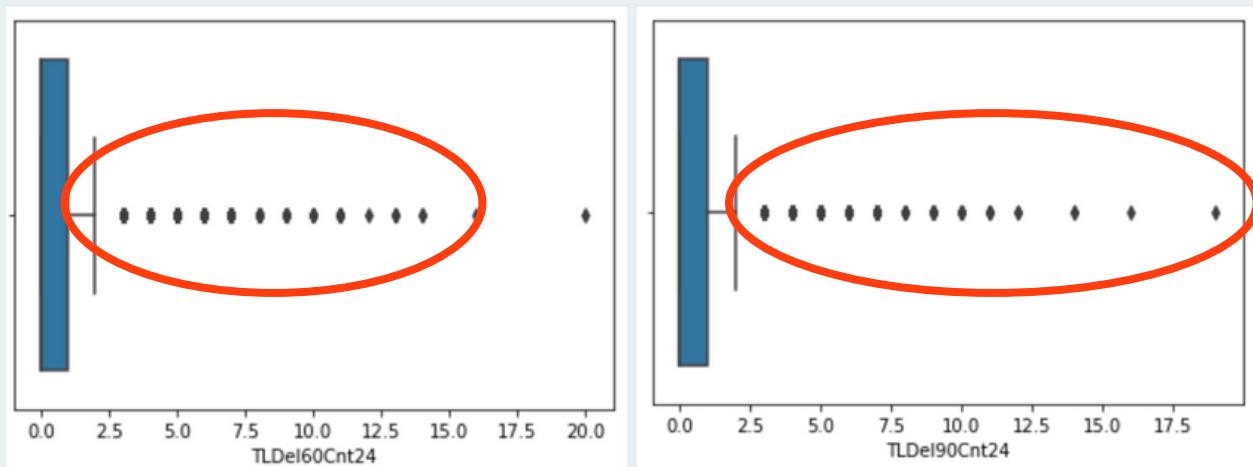
*As we saw above, outliers are already affecting our data and the credit record of our customers*

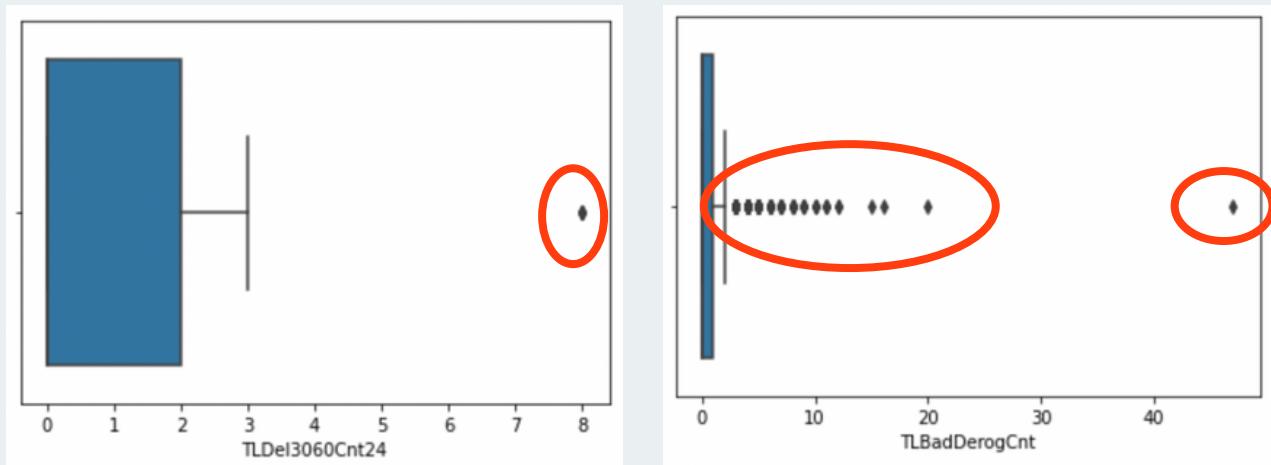
	InqTimeLast	TLCnt	TLSum	TLMaxSum	TLSatCnt	TL75UtilCnt	TL50UtilCnt	TLBalHCPct	TLSatPct	TLOpenPct	TLOpen24Pct
count	1937.000000	2084.000000	2052.000000	2052.000000	2086.000000	2014.000000	2014.000000	2051.000000	2086.000000	2084.000000	2084.000000
mean	3.255550	8.071497	20690.765595	33117.058480	13.744966	3.053128	4.048659	0.634520	0.590458	0.524225	0.585511
std	4.807047	5.580913	19547.089583	30071.449678	9.124619	2.562610	3.049617	0.264114	0.213296	0.210127	0.455571
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	1.000000	4.000000	6314.750000	11129.500000	6.000000	1.000000	2.000000	0.488600	0.461500	0.375000	0.285700
50%	1.000000	7.000000	16468.500000	26026.000000	12.000000	2.000000	3.000000	0.677500	0.608700	0.500000	0.500000
75%	4.000000	11.000000	29244.500000	47330.250000	19.000000	4.000000	6.000000	0.828700	0.733300	0.653800	0.833300
max	24.000000	40.000000	207979.000000	271036.000000	57.000000	17.000000	23.000000	3.361300	1.000000	1.000000	6.000000

For most of our feature variables, 75% of our data is far less than the maximum values present in our data set.

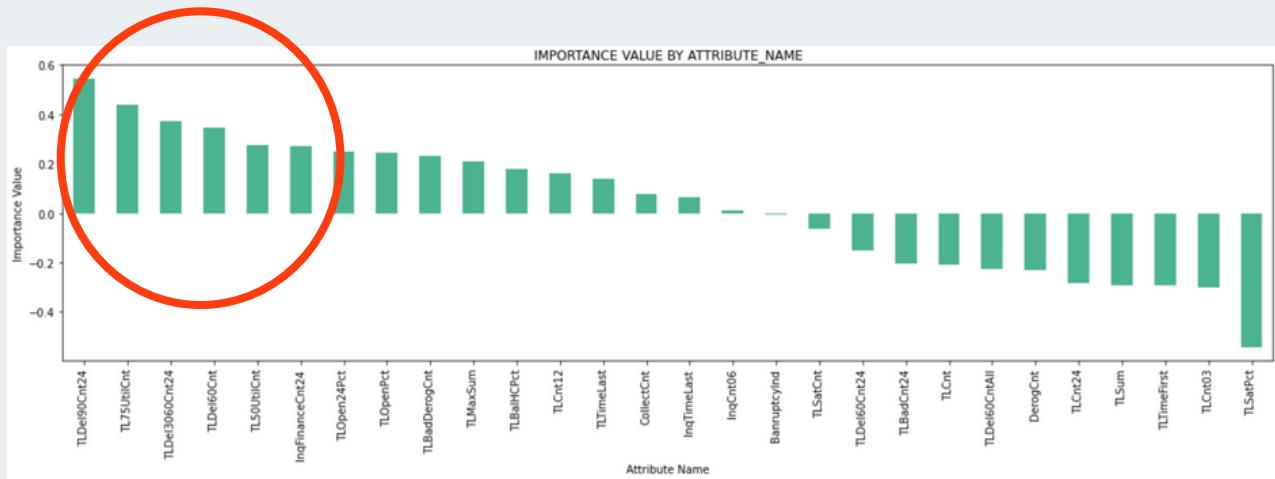
These outliers can play an important role in the model's forecasting behaviour. Although they represent events that occur with a small probability and a high impact, it is often the case that outliers are a result of system error. For example, a numerical variable that is assigned to the value 999, can represent a code for a missing value, instead of a true numerical variable. That aside, outliers can be easily detected by the use of boxplots. Box plots are used to show distributions of numeric data values, especially when you want to compare them between multiple groups. They are built to provide high-level information at a glance, offering general information about a group of data's symmetry, skew, variance, and outliers.

Some plots of highly correlated feature variables, i.e. those having greater weightage for precision are shown below:





Now coming to the treatment, not losing much on our data, we removed outliers for the feature variables which were highly positively or negatively correlated with our Target variable, hence having higher weights in our Model. This improved our various model's accuracy by a substantial magnitude. On these reduced attribute sets, our techniques will detect frauds with less memory and computation requirements. Let us analyse each of our model's performance to notice how this helped us avoid bad loans and ensure more of good ones.



## For Logistic Regression

<i>With Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	87	15
<i>Observed Good</i>	14	484

<i>Without Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	35	3
<i>Observed Good</i>	1	379

Percentage good lends made rose from 97.1 % to 99.7 %.

Percentage bad lends avoided rose from 85.3 % to 92.1 %

## For k Nearest Neighbour

<i>With Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	91	11
<i>Observed Good</i>	29	469

<i>Without Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	36	2
<i>Observed Good</i>	11	369

Percentage good lends made rose from 94.1 % to 97.1 %.

Percentage bad lends avoided rose from 89.3 % to 94.7 %

## For Random Forest Classifier

<i>With Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	87	15
<i>Observed Good</i>	10	488

<i>Without Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	36	2
<i>Observed Good</i>	1	379

Percentage good lends made rose from 98.0 % to 99.7 %.

Percentage bad lends avoided rose from 85.3 % to 94.7 %

## For Decision Trees

<i>With Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	70	32
<i>Observed Good</i>	36	462
<i>Without Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	34	4
<i>Observed Good</i>	49	331

Since Decision trees are not impacted by weights, the change in this case is ambiguous.

Percentage good lends made *fell* from 92.8 % to 87.1 %.

Percentage bad lends avoided rose from 68.6% to 89.5 %

This brings us to the end of our exploratory data analysis. We have completed a thorough understanding of correlation, outliers, their treatment and how it impacts our data sets. Now let us understand the various credit scoring methodologies and the machine learning models that can help us in producing a valid credit score used to predict the nature of the loans. Refer to the code attached with the report to see the detailed python implementation of the same

# CREDIT SCORING METHODOLOGIES

## Introduction

The methods used for credit scoring are evolving from traditional statistical techniques to innovative methods such as artificial intelligence, including machine learning such as random forests, gradient boosting, and deep neural networks.

With the central objective of making models that provide maximum utility to industry and avoiding unnecessary jargon, we researched upon various models and predictor variable combinations. We then realised the 5 models of *LogisticsRegression*, *Random Forest Classification*, *Decision Trees*, *K-Nearest Neighbours* and *Neural Networks* will provide a holistic viewpoint and necessary accuracy in our predication while at the same time ensuring our models can be explained to people with no prior professional technical experience in the arena of machine learning.

- We have analysed 2 datasets, one with outliers removed from the weighted feature variables based on correlations existing among them and one without losing any data with outliers intact.
- We also implemented a methodology for the selection of the key variables in order to establish a credit scoring model.



As one of the biggest challenge of building credit scoring models is also to decide which are the most relevant features to be selected for the task. In practice, the data used for the models may be collected from various sources, and sometimes the size of the data is small, in relation to the number of features considered. In addition, there might be some features in the data, which may not be significant to credit risk, or some of them may be correlated with each other. Thus, these data issues might result in a misleading interpretation of the credit scoring model and a very poor performance of it.

The *feature selection* process is a solution to these issues. Since, it is an arbitrary, and unsystematic task since there is no specific theory, and works differently on different data. therefore, for this report we have implemented feature selection, **forward stepwise selection**, on one of model which is by far giving most accurate results - The RandomForest Classifier.

## Splitting Data Train and Test

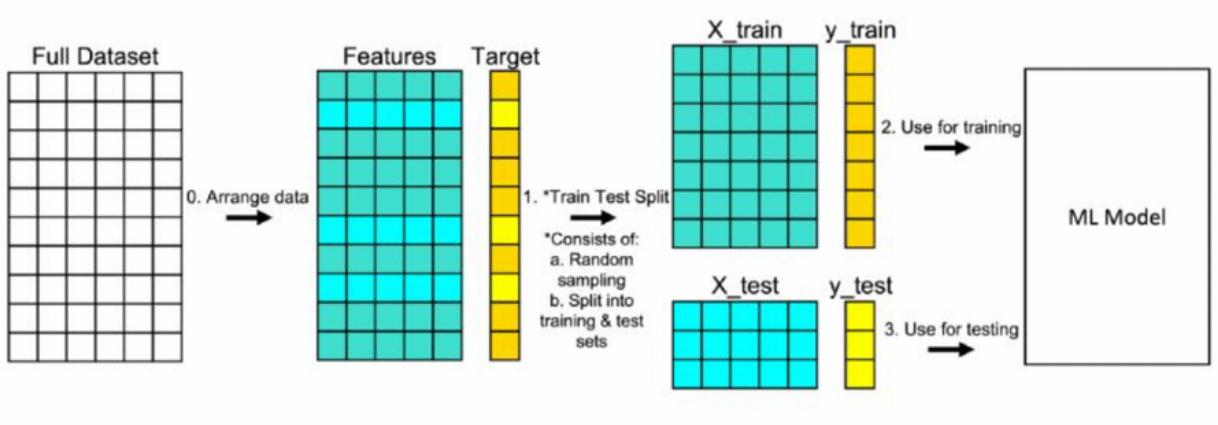
“ —

**Train test split** is a model validation procedure that reveals how your model performs on new data.

— ”



If you have new data, it's a good idea to see how your model performs on it. The problem is that you may not have new data, but you can simulate this experience with a procedure like train test split.



Train test split procedure. | Image: Michael Galarnyk

Now we will study about all these 5 models in brief and understand the various observations that we spot. Each of them have a different processing algorithm and perform differently on various metrics. Let us delve into the comparative study of these models to see which one emerges out on top.

# **Model Building:**

## **Logistic Regression**

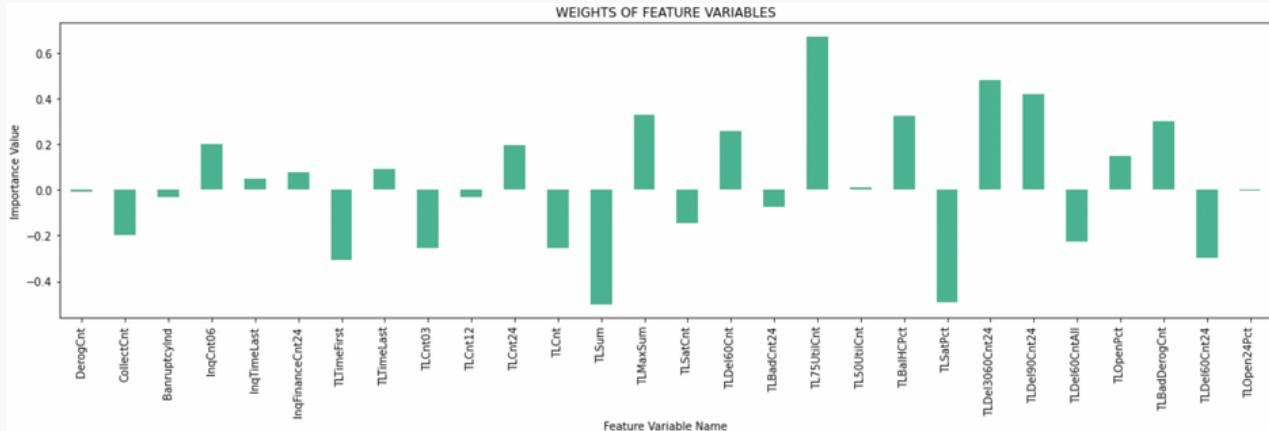
**Logistic regression** is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.

Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

$$Y("TARGET") = 1 \text{ if a borrower defaults or } 0 \text{ otherwise}$$

Since it provides the probability that the response variable is 1, a threshold must exist that classifies data into two or more categories; the default threshold is 0.5, and if a sample whose probability of defaulting is equal to or greater than 0.5, it is classified as 1 and 0 otherwise.

Firstly we trained the model over our dataset and weights are assigned to various features depending upon their impact on TARGET variables differentiating the borrowers as defaulter or punctual and predictions are compared with the actual behaviour of borrowers in our test dataset.



The variables with maximum weights, were cleaned against any kind of deviation due to outliers and on finally implementing the model it yielded an accuracy score of 91.38%.

## k Nearest Neighbour

**kNN** is one of the simplest supervised machine learning algorithm mostly used to make classifications or predictions about the grouping of an individual data point. It measures/stores all available cases and classify new cases based on a similarity measure. It serves as an example of a *non-parametric statistical approach* and is based on an assumption that similar inputs have similar outputs. Given a positive integer K and a test observation  $x_0$ , this classifier first identifies K points that are closest to  $x_0$  in the training data set represented by  $N_0$  and then kNN classifies the test observation  $x_0$  to the class that has the highest probability. This simple yet efficient approach made the KNN very popular and widely used in data mining and statistics.

We choose a metric on the space of applicants and take the k-nearest neighbour (k-NN) of the input pattern that is nearest in some metric sense. A new applicant will be classified in the class to which the majority of the neighbours belong.

Also, since it seems to depend on degree of correlation with feature variables, its performance increase as we remove the outliers from highly correlated feature variables.

<i>With Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	91	11
<i>Observed Good</i>	29	469
<i>Without Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	36	2
<i>Observed Good</i>	11	369

KNN gives an accuracy of 88.75% with outliers removed.

It is a fairly intuitive procedure and as such it could be easily explained to business managers who would need to approve its implementation. It can also be used dynamically by adding applicants when their class becomes known and deleting old applicants to overcome problems with changes in the population over time. Despite this, nearest neighbour models have not been widely adopted in the credit scoring industry. One reason for this is

the perceived computational demand not only must the design set be stored, but also the nearest few cases among maybe 100,000 design set elements must be found to classify each applicant.

## Decision Trees

**Decision Trees (DTs)** are a *non-parametric supervised learning method* used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.

It is used in credit scoring practice only as a supporting tool to accompany parametric estimation methods. It serves, for example, in the process to select characteristics with the highest explanatory power.

Decision Trees are not sensitive to noisy data or outliers since, extreme values or outliers, never cause much reduction in Residual Sum of Squares(RSS), because they are never involved in the split.

Hence, when we remove the non-uniform data, it does not improve the performance but due to less of data now available to train the model, the change in prediction values turn out to be ambiguous as shown:

<i>With Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	70	32
<i>Observed Good</i>	36	462

<i>Without Outliers</i>	Predicted Bad	Predicted Good
<i>Observed Bad</i>	34	4
<i>Observed Good</i>	49	331



The advantages of this method in credit scoring are that it is very intuitive, easy to explain to management and it is able to deal with missing observations.

The major disadvantage is the computational burden in case of large datasets since at each node every characteristic has to be examined. Very often the resulting tree is quite large so that the process of model-learning becomes too time consuming. Some empirical studies also note that often the trees are not stable since small changes in a training set may considerably alter the structure of the whole tree.

## **Random Forest Classifier**

We also experimented with the Random Forest Classifier. The **Random forest or Random Decision Forest** is a *supervised Machine learning algorithm* used for classification, regression, and other tasks using decision trees.

A random forest is a classification algorithm used commonly in both regression and classification problems given the easy interpretation and good reflection on human decision-making

- **Random Forest is capable of identifying important features**

As with many machine learning algorithms, the Random Forest model can be seen as a black box: it is not possible to define the process completely from input to output. However, there are ways



## Random Forest characteristics

<b>Pros</b>	<ul style="list-style-type: none"><li>• Limits overfitting</li><li>• High accuracy (pruning is not used)</li><li>• Easy choice of relevant variables</li><li>• Stable against the data (good handling of missing values)</li><li>• Can handle a high dimensional set of variable</li></ul>
<b>Cons</b>	<ul style="list-style-type: none"><li>• Low interpretability</li><li>• Parameter choice (number of trees, proportion of the observation in each sample, depth, etc.)</li><li>• Computation time</li></ul>

Table 1: Pros and Cons of the Random Forest algorithm applied to credit risk

The following insights are examples of Random Forest algorithm usage that prove its utility in the field of credit risk. Random Forest can be used to save time on data management steps and to identify the most important characteristics in a dataset.

to look inside the algorithm. An extremely powerful example is the ability to compute a feature's importance.

“

**Feature selection** is the most basic step in data pre-processing as it reduces the dimensionality of the data

”

There are two different categories of feature selection methods, i.e. *filter approach* and *wrapper approach*.

The **wrapper approach** uses a machine-learning algorithm to measure the goodness of the set of selected features. The measurement relies on the performance of the learning algorithm

such as its accuracy and precision values. The wrapper model uses a learning accuracy for evaluation. In the methods using the wrapper model, all samples should be divided into two sets, i.e. training set and testing set. The algorithm runs on the training set, and then applies the learning result on the testing set to measure the prediction accuracy.

As we implemented **forward stepwise selection**, sizes of selected subsets of varies from 20 to 28 features and target values are predicted.

Another feature is added to the subset, and values are redicted using RF Model. If the accuracy of the predicted values improve, the process is continued.

And we get an optimum set of feature variables which provide us with best results for the predicted Target variables for the borrowers in our test dataset.

After implementation of **forward stepwise selection** upon Random Forest Classifier, best subset comes out to be consisting of all28 feature variables. This shows data set contains only specific features .

Hence, Random Forest Model gives a final accuracy of 91.14% after outliers are removed and best subset of feature variables is chosen.

<i>With Outliers</i>	<b>Predicted Bad</b>	<b>Predicted Good</b>
<i>Observed Bad</i>	87	15
<i>Observed Good</i>	10	488
<i>Without Outliers</i>	<b>Predicted Bad</b>	<b>Predicted Good</b>
<i>Observed Bad</i>	36	2
<i>Observed Good</i>	1	379

## Neural Networks

A **Neural Network (NNW)** is a mathematical representation inspired by the human brain and its ability to adapt on the basis of the inflow of new information. Mathematically, NNW is a non-linear optimization tool.

The NNW design called multilayer perceptron (MLP) is especially suitable for classification. The network consists of one input layer, one or more hidden layers and one output layer, each consisting of several neurons. Each neuron processes its inputs and generates one output value that is transmitted to the neurons in the subsequent layer. Each neuron in the input layer (indexed  $i = 1; 2; \dots; n$ ) delivers the value of one predictor (or the characteristics) from vector  $x$ . When considering default/non-default discrimination, one output neuron is satisfactory.

There are two stages of optimization. First, weights have to be initialized, and second, a nonlinear optimization scheme is implemented. In the first stage, the weights are usually initialized with a small random number. The second stage is called the learning or training of the NNW. The most popular algorithm for training multilayer perceptrons is the back-propagation algorithm.

For each individual, weights are mentioned in such a way that the error computed from the output layer is minimized.



NNWs were described in the 1960s but their first use in the credit-scoring-related literature appears only at the beginning of the 1990s.

The major drawback of NNWs is their lack of explanation capability. The reasoning behind why and how the decision was reached is not available.

Neural networks are well suited to situations where there is a poor understanding of the data structure. In fact, neural networks can be regarded as systems which combine automatic feature extraction with the classification process, i.e. they decide how to combine and transform the raw characteristics in the data, as well as yielding estimates of the parameters of the decision surface. This means that such methods can be used immediately, without a deep grasp of the problem. In general, however, if one has a good understanding of the data and the problem, then methods which makes use of this understanding might be expected to perform better.

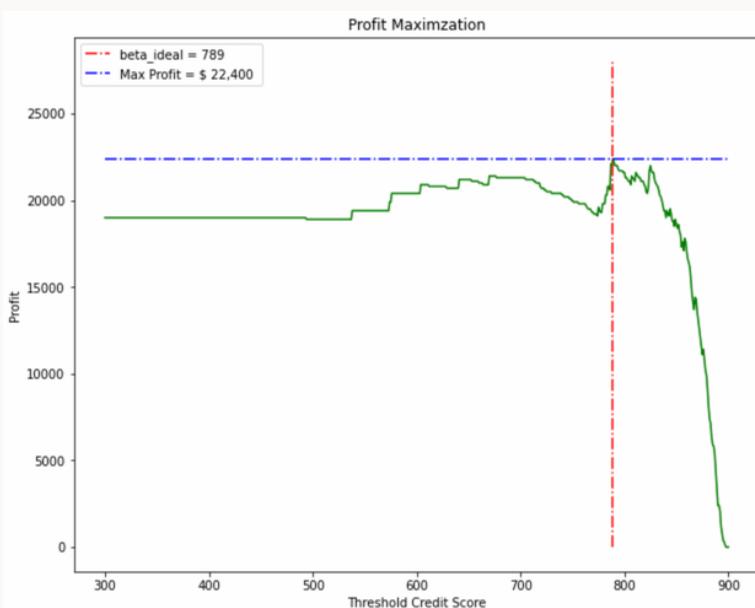
With a much better understanding of our data, we can outperform Model Prediction based on Neural Networks ( Accuracy Score of around 91% ) using other models used



# INTERPRETATION OF RESULTS

Accuracy Scores	With Outliers	Without Outliers
<b>Logistic Regression</b>	83.17 %	<b>91.38 %</b>
<b>K Nearest Neighbour</b>	80 %	88.75 %
<b>Decision Tress</b>	82.33 %	80.14 %
<b>Random Forest Classifier</b>	<b>83.83 %</b>	91.14 %
<b>Neural Networks</b>	Around 83 %	Around 91 %

In general there is no overall "best" method. What is the best will depend on the details of the problem, the data structure, the characteristics used, the extent to which it is possible to separate the classes by using those characteristics and the objective of the classification (overall misclassification rate, cost-weighted misclassification rate, bad risk rate among those accepted, some measure of profitability, etc.) The various methods are often very comparable in results. As explained and shown in the table the Random Forrest Classifier performs best with outliers @ **83.83 % accuracy** and the logistic regression model performs best when our data set is without outliers @ **91.38% accuracy**.



Now to get the credit score corresponding to being a punctual borrower we have scaled the probability of a good borrower obtained from the data of our logistic regression

model on the scale of 300 to 900 which is the commonly used scale by major credit score companies like CIBIL, TransUnion, Equifax, Experian, etc. From the 418 borrower's data sets out of which 380 yield a positive result and 38 of them end up as defaulters, this gives to us a scale which the bank can use to set a threshold above which it would be in the best interests of the bank to lend so as to minimize the defaulters, occurring due to information asymmetry and other factors. If we assume a scene where a bank can make \$100 profit from lending to a good customer and suffers a \$500 loss from lending to a bad customer, we will observe that a threshold credit score of 789 gives the bank the sweet spot of maximum profit (\$22,400).

## **Conclusion:**

The process of determining the credit score is a complicated endeavour, the hardships are further amplified due to lack of credit history of borrowers and cluttered data sets. The stakes during the practical application of our models are astronomical as even a tiny computational, syntax or logical error can lead to a *deserving firm miss out on loan* preventing their business to expand, a household missing out on necessary revenue in times of crisis even if they have the necessary assets or collateral or a bank can get into severe trouble if it ends up lending to sub prime borrowers or on the other hand, miss out on great relationships with night net worth powerful clients by denying them loans when they are more than capable of repayment.

—

In credit scoring, where people have been constructing scorecards on similar data for several decades, there is a solid understanding. This might go some way towards explaining why neural networks have not been adopted as regular production systems in this sector, despite the fact that banks have been experimenting with them for several years.

— //

On the voyage to solve this problem we developed a thorough understanding of information asymmetry, trade lines, derogatory marks and analysed a large data implemented in our model. We had the onus of exploiting the complete power of the strongest tool of humanity, artificial intelligence and computer science and put our best foot forward by implementing 5 complete models and providing the background about each in east to understand literature. We urge everyone reading this report to go through our model file and understand each step which will help paint the complete picture. We hope you enjoyed reading our work just as much as we loved presenting it to you. Thanks a lot for reading till the end.