CREDIT RISK MODELLING ACQUISITION MODEL

submitted in partial fulfillment of the requirements for the award of the degree of

Master of Computer Applications
(2018-2021)

by
Sreyashi Bhattacharjee
05104092018

Under the supervision of
Mr. Gaurav Indra
Assistant Professor

DEPARTMENT OF INFORMATION TECHNOLOGY
INDIRA GANDHI DELHI TECHNICAL UNIVERSITY FOR WOMEN, KASHMERE GATE, DELHI- 110006

© Sreyashi Bhattacharjee 2021

ACKNOWLEDGEMENT

I would like to acknowledge my faculty supervisor Mr.Gaurav Indra for his very helpful comments, support and encouragement.

I would like to thank my mentor Mr.Vansh Tuli, Module Lead for his constant supervision, for giving me a head-start and facilitating industry exposure. I would also like to thank Mr. Aakash Yadav, ML-AI Solutions Architect for overall guidance, vision and conceptualisation.

Finally, I am grateful to Sopra Steria (India) for providing a healthy, supportive and understanding environment. They allowed me the freedom to explore innovative models to simplify a complex business problem. This made my project work possible without any hindrance.

Sreyashi Bhattacharjee

Credit Risk Modelling: Acquisition Model

Sreyashi Bhattacharjee

May 2021

Abstract

Credit lending is one of the principle apprehension of banks as it incorporates the dangers of non-payment of instalments. This can be seen addressed in Basel I and II regulations. Basel I directed banks to develop their individual advanced risk management framework focusing mainly on credit risk. Basel II expanded this notion to credit risk of assets and differentiated between different types of credits. However, many have lost a significant amount of assets because the models they utilised neglected to precisely anticipate clients' defaults. Generally, banks use static models with a segment or static component to demonstrate credit risk patterns.

This paper attempts to fulfil the deficiencies in credit risk modelling, specifically regarding the acquisition of new clients. It provides a tool for induction into the lending segment of the banking sector. The methodology adopted exploits the existing machine learning algorithms to predict the risk of default by a potential customer. The risk evaluation adopts a probabilistic approach to evaluate incoming customer's credit default risk against a pre decided threshold. The classification is optimised with the help of hyperparameter tuning techniques and model evaluations.

Explainable Artificial Intelligence (XAI) is employed to get a clear picture of the attributes which contributed to this decision making. The algorithms that are explored include Logistic Regression, Support Vector Machine, Random Forest and Gradient Boosting Classifier. Random Forest was picked for further analysis and implementations. The project's results show that Random Forest gives an accuracy of 99.5% with a recall of 98%.The algorithm is then used to implement a predictor tool as part of the dashboard. The tool allows potential customers to input custom data to gauge their chances of rejection and the company to assess customer credibility.

Contents

List of Figures

List of Symbols

$\bar{A}$	Mean of A
χ	Greek letter chi used to symbolise chi-square statistic
γ	Greek letter gamma which can be used as variable
$\|A\|$	Norm of A which is used to calculate distance. Detailed explanation can be obtained from vector theory
σ	Small Greek letter sigma used to represent standard deviation and the square as variance
$\sim$	Approximation
$\sum_{lower}^{upper}$	Summation running from the lower limit to upper limit
θ^T	Theta transpose where theta is a Greek letter used to represent the parameter learnt in machine learning algorithm
$E()$	Expectation
N	Natural Numbers
X_i *or* Z_i	Sample

1 Introduction

This paper is a documentation of the project developed as a part of the final semester 6-month internship at Digital Transformation and Innovation division in Sopra Steria[1].

Sopra Steria[1] an information technology company deals with a wide variety of business domains and there has been a massive increase in the number of clients in banking and finance sector. The Digital Transformation and Innovation team is building a new service module to be added to their existing banking software to cater to the potential and current clients along with being made available as an independent service. The company plans to incorporate artificial intelligence/machine learning for Credit Risk Modelling.

Team Member	Designation	Work Description
Mr. Aakash Yadav	ML-AI Solutions Architect	project lead
Mr. Vansh Tuli	Module Lead	front-end development
Ms. Sreyashi Bhattacharjee	Intern	data science (machine learning) pipeline

The intern was entrusted with the execution of the entire data science pipeline. It included data collection, data pre-processing and transformation, feature engineering, model building, training, testing, evaluation, explainable artificial intelligence and deployment. The development and deployment of the dashboard was in collaboration with the software engineer.

1.1 Sopra Steria

Sopra Steria, is a $4.8 billion French multinational company, providing consulting, software development and digital services. They are the firm of choice in Europe and hold significant market dominance in other countries such as India. They assist clients in driving their digital transformation to achieve concrete and long-term benefits. They aim to provide end-to-end solutions to make large corporations and organisations more competitive. They bring together in-depth experience of a variety of industry sectors and cutting-edge technology with a truly collaborative approach.

Sopra Steria India[2] offers a fully integrated global delivery model to achieve delivery excellence, high value addition and cost adequacy for clients in different different business spectrum. Sopra Steria[1] spans across various Industries (Figure 1b) and Services (Figure 1a).

The **Digital Transformation and Innovation** division helps clients discover new ways to harness the power of digital technologies to fuel their transformation and growth. The team explores new technologies, incubates proof of values on them and strengthens their practices. The department focuses on emerging technologies (Figure 2) such as Block Chain, Smart Machines, Artificial Intelligence, Cognitive Computing, Social Media, Mobility, Analytics & Cloud computing, and Internet of Things.

[1] India

[2] https://www.soprasteria.in/

(a) Services

(b) Industries

Figure 1: Industries and Services covered by Sopra

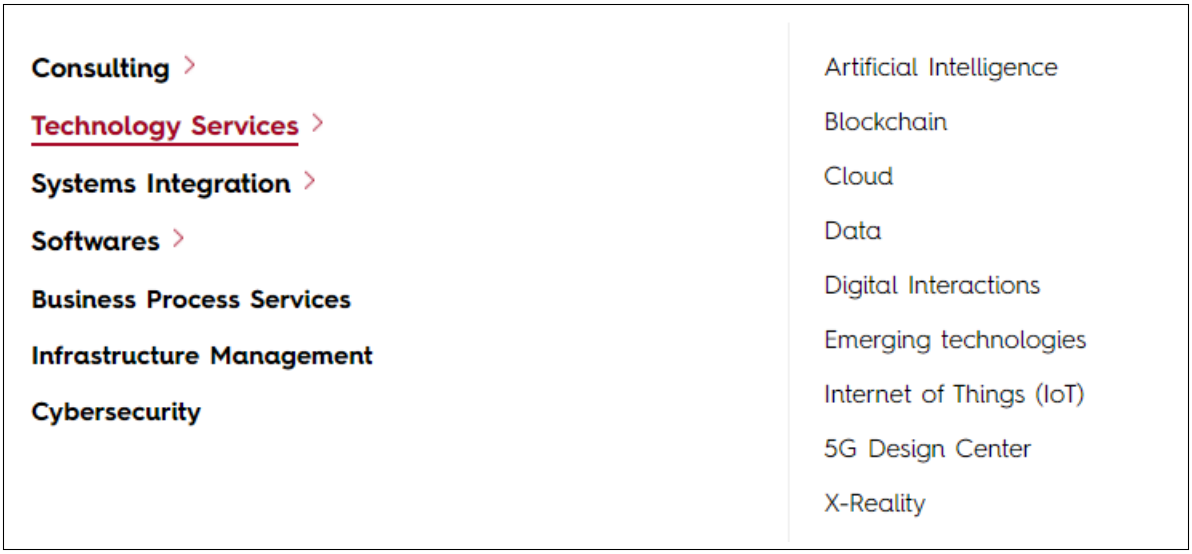


Figure 2: Technologies covered by Digital Transformation Division

1.2 Problem Statement and Objectives

> Top industries at risk of credit default in 2021
>
> S&P Global

A credit request by a potential customer comes with a lot of caveats and risks attached to it. It presents an opportunity for the lender that is both a potential risk and reward. Historically, the decision of whether to accept a request or defer it has been taken using statistical methods and expert knowledge base. But this method has a large margin of error because we are introducing the chances of human error. Combining the current on-going financial situation due to the pandemic with the inherent susceptibility to human error of the existing credit risk model, it seems to be a disaster waiting to happen.

With advancements in computational power and analytics which facilitates massive processing in minimum time combined with the amount of data available presents a very favourable scenario which needs to be exploited. Leveraging these resources to manage credit default risk can be a massive evolution and boon. This paper attempts to use advanced analytics to assist **credit decisioning**.

This paper addresses the following key aspects:

- New Clients : what is the probability of their default and accordingly the loan will be granted.
- Driving factors that lead to default.
- Help lenders (companies) to monitor portfolios and adjust risks accordingly.

1.3 Scope

Credit risk modelling encompasses three aspects.

- Acquisition Model
- Behavioral Model
- Recovery Model

This paper focuses on **Acquisition Model**. The use of machine learning techniques and analytics will help model the nonlinear relationships better. It will give a upper-hand to the lender on new customer acquisition, help avoid potential defaults and leverage potential gains. It will also allow the borrowers to get an inside look into the decision making process. It will allow the borrowers to understand the aspects which are necessary and improve upon them.

This is a proof of concept to showcase the capabilities of the model in handling and implementing credit risk modelling. In future, with company data, it can be up-scaled and brought to a larger forum.

1.4 Organization of Report

The entire paper is the result of approximately 6 months work as depicted in Figure 3.

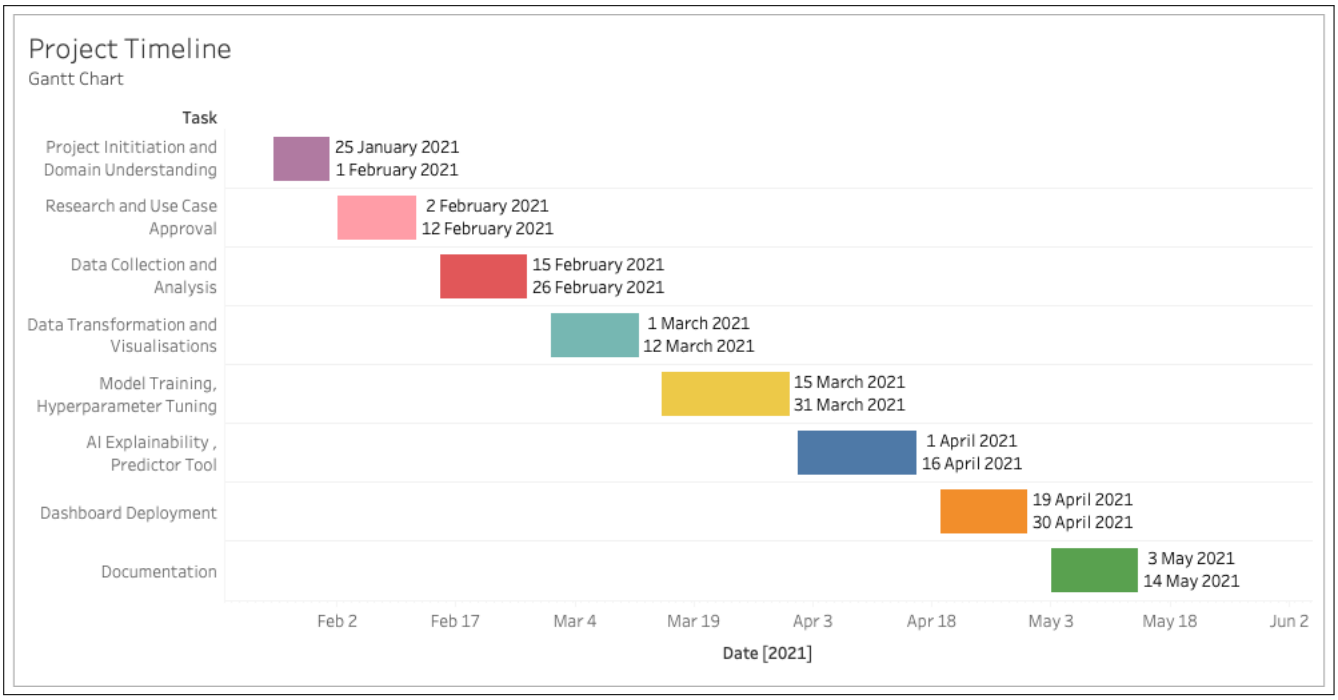


Figure 3: Project Timeline

The paper has 7 sections in total.

1. Introduction
2. Related Work
3. Software Requirement Specification (SRS)
4. Methodology
5. Experiment Setup and Results
6. Explainable Artificial Intelligence
7. Conclusion and Deployment

The **1**st section is further divided into 4 parts which provides a brief introduction to the nature of the project, about the company, the problem that is being addressed, the objectives, scope of this project and organisation of the paper.

The **2**nd section looks into the work done by other organisations in this segment and explains some of the key concepts that are essential.It introduces the proposed methodology along with the unique aspects of this paper.

The **3**rd section provides the software requirement specification document.

The **4**th section provides a complete breakdown of the methodology implemented and the corresponding output obtained. It also provides an in-depth knowledge of the models built and the algorithms used.

The **5**th section showcases the experiment setup and the results obtained. It also provides a detailed analysis of the results along with potential solutions.

The **6**th section deals with the salient features of this paper. It provides an insight into the driving factors that lead to default.

Finally, the **7**th section showcases the deployment of the project for the end users as a dashboard.

2 Related Work

Including a custom credit risk assessment system has become a norm in the banking sector especially after the mandates in Basel II. The amalgamation of the risk management tool kit and the power of machine learning is an area which is under exploration. The major participants in this area are **Deloitte**, **Moody's Analytics** and **S&P Global Market Intelligence**. In order to get an insight into the work done by these companies, their white papers were explored.

According to **Moody's Analytics** [1], credit risk models usually use leverage and benchmarks of profitability such as return on assets, to assess the risk profile of a lending decision. The predictive power of the model is greatly improved when other explanatory variables for instance liquidity ratio and behavioral factors namely loan/trade credit payment behavior are included. Including additional variables increases R^2 and reduces omitted variable bias. Most statistical inference techniques assume structural form of regression equation. This could produce biased results when this assumption is violated. Therefore, a linear regression model would lead to erroneous results when a nonlinear model achieves better fit. On the other hand, machine learning does not assume any structural form. It uses the data to deduce the structural form of the default risk and other explanatory variables which increases the fit of the model. Therefore, a machine learning based credit risk model frequently use Artificial Neural Network, Random Forest and Boosting algorithms to estimate default risk.

On the other hand, **S&P Global Market Intelligence** [13] states that private companies are highly heterogeneous and are often constrained with limited data availability. Therefore, there is a need for a robust credit default prediction model which is able to incorporate the inherit heterogeneity across private companies with limited data. The model they employ is able to counter these restriction through the use of limited financial data to capture the variability across S&P Capital IQ while ensuring comprehensive risk assessment coverage. The model tests multiple machine learning algorithms Altman Z-score, Logistic Regression, Support Vector Machine (SVM),

Naïve Bayes and Decision Tree. Each machine learning algorithm is evaluated using receiver operating characteristics (ROC) curve and corresponding area under the curve (AUC). In the in sample estimation, Decision Trees was the most efficient with high degree of classification accuracy. Logistic regression and SVM recorded similar AUC while Naïve Bayes and Altman Z-score were the least accurate. On the other hand, the performance of Decision Trees, which is able to marginally outperform other algorithms, deteriorates for out sample AUC, which more closely resembles real world situations. The volatility in accuracy of decision trees makes it unreliable.

Finally, **Deloitte** [10] gives a complete run down of the credit risk analytics system employed by the company. They provide a solution to a common problem in assessing the client's default likelihood. They apply a credit scoring model that exercises the decision making process of accepting and rejecting a loan. They use the most prominent method of credit scoring - logistic regression. It begins by analysing the data, remove the skewness ,the outliers, handle the missing values and decide on a benchmark for default decisioning. They further go on to discuss some of the difficulties that are encountered during this decision making and certain possible solutions that they employ like rejection inference. The paper then describes the working and methodology of the model. They also employ certain domain knowledge aspects like information gain, variable selection and error reduction. Further they explore the evaluation process of the built model. The evaluation is based on two objectives. One is to fit the data or the goodness of fit and the other is to give the correct probability of default or the predictive power. Finally, they wrap-up with a refinement of the model and it's interpretations.

2.1 Key Concepts

In order to facilitate the understanding of this paper, as few of the key aspects related to the domain knowledge is presented in this segment. Credit models are primarily of two types:

- Qualitative
- Quantitative

Quantitative credit models look into financial data. There are five types of quantitative credit models.

- Credit Scoring Model
- Structural Model
- Reduced Form Model
- Credit Migration Model
- Credit Portfolio Model

A few key terms:

1. **Credit Risk Modelling** - It is the process of utilizing information about an individual to estimate an individuals' credit default risk. The credit can be of any type, not necessarily credit card.

2. **Default Risk** - An individual's probability that they will fail to pay back the loan on time.

3. **Lender** - An entity that makes funds available to another entity with the expectation that the funds will be repaid. This entity can be an individual , a public or private group, or a financial institution.

4. **Borrower** - An individual or company that has received money from another party with the agreement that the money will be repaid.

5. **Probability of Default (PD)** - The probability that the borrower will be unsuccessful in fulfilling the credit obligations. It is expressed in the form of percentage.

6. **Basel Regulations** - A committee founded in 1974 by the central bank governors of G10[3] countries who meet in Basel, Switzerland each year to formulate regulations that make sure that the banks have enough capital to repay their depositors. Basel I, emphasised solely on credit risk and Basel II expanded this concept to operational and market risk. These regulations also mandated a custom credit risk model for each banking institution. The release of Basel III has been postponed to January 2023.

2.2 Proposed Methodology

This paper attempts to solve the credit risk default problem using machine learning models and predictive analytics.

The analysis starts with data exploration and exploratory data analysis. Following the initial exploration, the data are cleaned, features are engineered and manipulated. A few insights are drawn from interactive visualisations which help answer some of the preliminary questions. Then the models are built based on some machine learning algorithms. The algorithms include:

- Logistics Regression - base model
- Support Vector Machine
- Random Forest
- Gradient Boosting Classifier

The models are evaluated based on the accuracy and the recall value. The unique features added in this project are:

- Predictor Tool
- Explainable Artifictail Intelligence (AI)

Explainable AI will allow the users to get an in-depth understanding of the output along with the features that are affecting the rejection or acceptance of the applications. The predictor tool is a feature added to the dashboard where the client can input their custom data and get the acceptance or rejection decision in real-time along with the factors which contributed to this decision.

[3]Belgium, Germany,Canada, France,Switzerland,Italy, Japan, the Netherlands, Sweden,the United Kingdom and the United States

3 Software Requirement Specifications (SRS)

Product: Predictive Dashboard
Description: Credit Risk Modelling: Acquisition Model
Status: Proof of Concept
Version: 1.0

3.1 Introduction

3.1.1 Purpose

The aim of this document is to provide an overview of requirements and specifications of the project called Credit Risk Modelling: Acquisition Model. The goal of this project is to make a predictive dashboard showcasing a credit risk modelling mechanism which will help companies to predict the default chance before acceptance of a customer.

The tools used in this project are:

- Python programming language
- Scikit-Learn library for machine learning
- SHAP, LIME, ELI5 and Ethik AI libraries for explainable AI
- Dash by plotly for dashboard

3.1.2 Intended Audience

Anyone with some programming experience, with familiarity in Python and data analytics, can understand this document. The document is intended for data analyst, machine learning architects, data engineers, project managers, data scientists and documentation writers

This SRS also includes:

- Broad description of the product
- External interface requirements
- System features
- Nonfunctional requirements

3.1.3 Scope of product

The acquisition model powered by machine learning will help ease the issues discussed in this paper related to credit risk.

This predictive dashboard will help:

- Assess default vulnerability of potential clients
- Highlight factors that lead to default
- Monitor client portfolios

3.1.4 References

- This document is written in Latex
- IEEE Standard 830-1998 Recommended Practice for Software Requirements Specifications
- References used during this report mentioned in the appendix

3.2 Broad description of the product

3.2.1 Perspective of Product

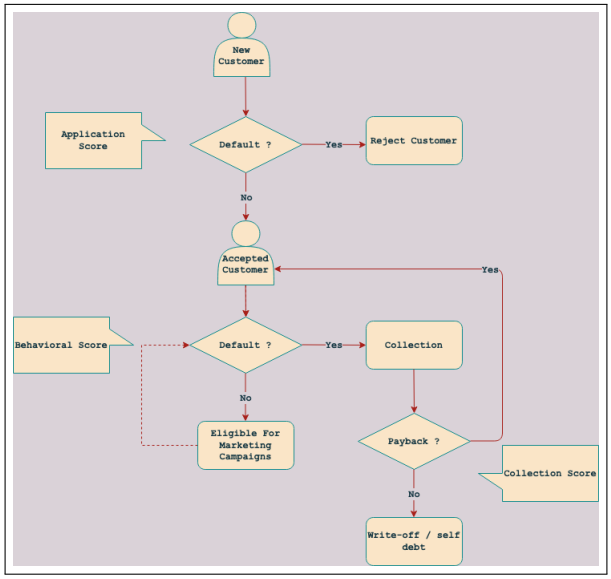


Figure 4: Flow Chart for Credit Risk Modelling

This system consists of following components:

- **Machine Learning Models**: have predictive capability regarding credit decisioning.
- **InnerData**: company's internal authentication based system for accessing predictive platform including data.
- **Training Data**: collected using open source API at this stage. This will be replaced by client's data.

3.2.2 Function

Dashboard functionality include:

- Analysis
- Model Analysis
- Predict for your customer

3.2.3 Users

- Lender
- Borrower - if the lender makes this functionality available.

3.2.4 Operating System Requirement

Following minimum specifications required:

- Operating System: Windows based or Linux based (MAC included)
- Processor: 2.5 GHz
- Network: 802.11n Wireless LAN
- Memory: 1GB or more
- Web browser

3.2.5 Constraints

- Processing power to run machine learning algorithms
- Stable and fast internet connection
- Huge data set set can help map non linear behavior better

3.3 External interface requirements

3.3.1 User Interface

The user will be able to access the predictor tool with the help of a web deployed dashboard. The user can upload custom data and get real-time feedback.

3.3.2 Software Interface

The dashboard is deployed on cloud using company specific platform. Data can be fed by navigating to the drive where the data is stored.

3.4 System features

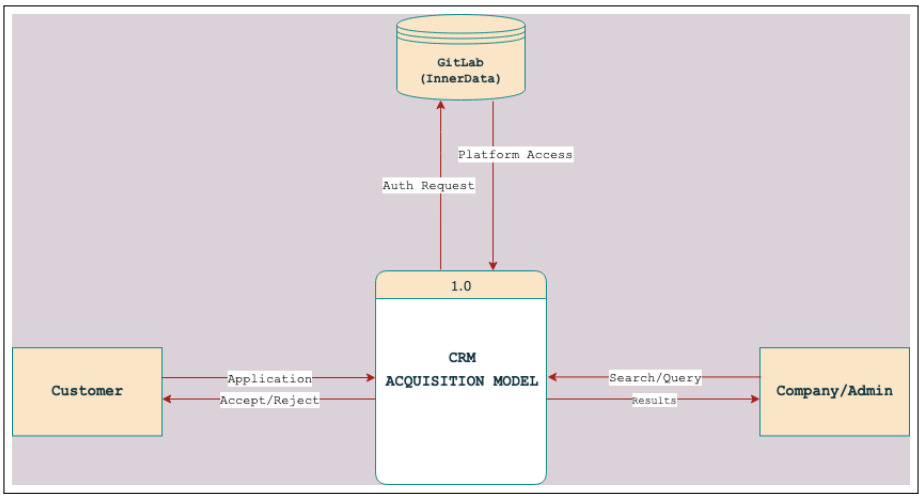


Figure 5: Context Diagram

- Feed in train data
- Select algorithm - if specific requirement, else will be trained on default.
- Train model on train set
- Display model analysis
- Feed test set or custom data
- Display accept or reject of loan application
- Display analysis of decision
- Suggest changes in application in case request rejected

3.5 Nonfunctional requirements

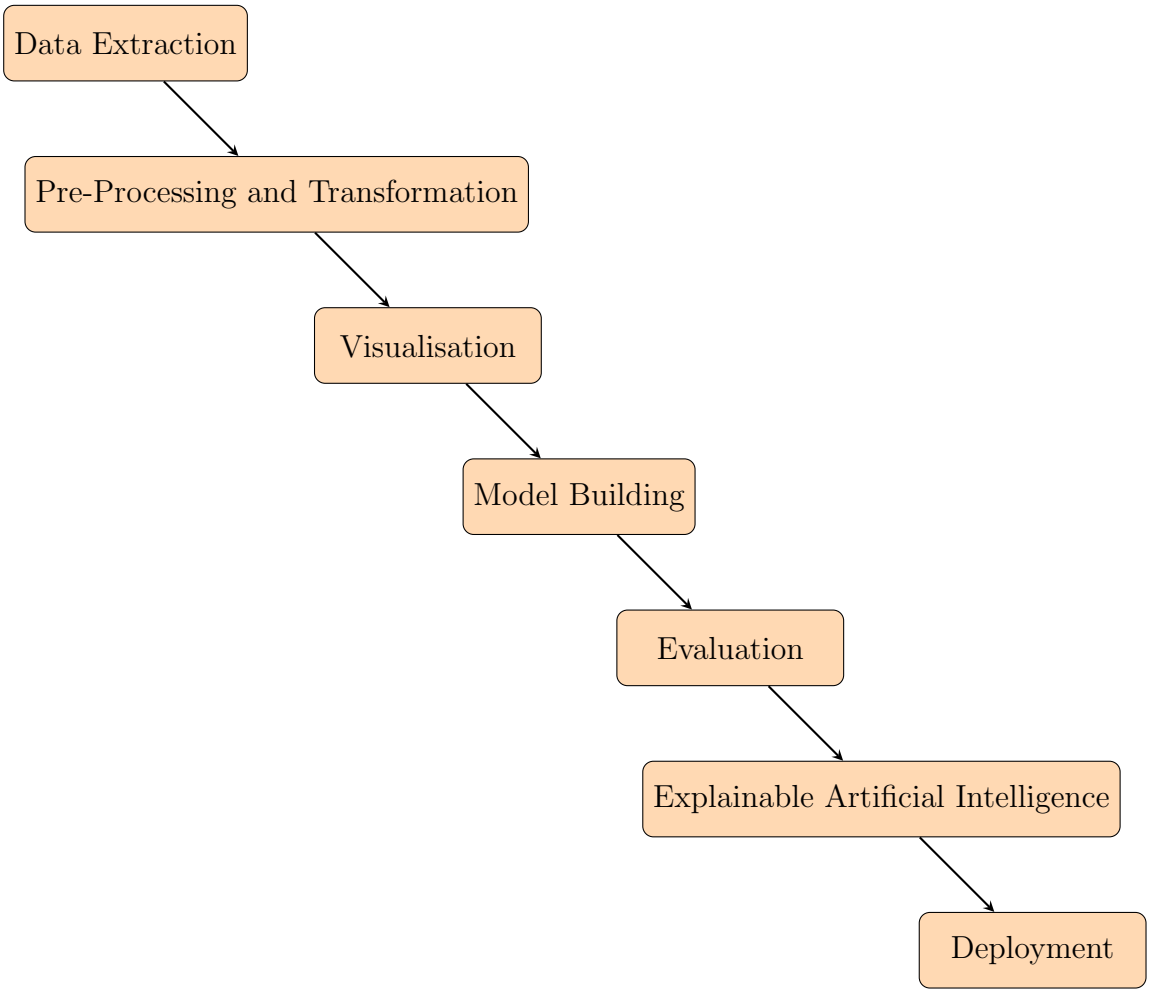
3.5.1 Security

- Data shared needs to be bound under non disclosure agreement
- The tool is deployed in a digital platform which requires prior authorisation

3.5.2 Rules of Business

- This predictive dashboard is NOT AN OPEN SOURCE. License is with Sopra Steria (India).
- Sopra Steria (India) needs to be contacted for demo.

4 Methodology



4.1 Dataset Description

After due deliberation and researching the data source[4] that is shortlisted belongs to the Lending-Club.

LendingClub is a peer-to-peer lending company, headquartered in San Francisco, California. They are the largest peer-to-peer lending group globally. Their data was consolidated and made available in kaggle. This data set has all the relevant attributes required for the execution of our methodology.

The entire project is done using Python Language version 3.7.10 [12]

[4]https://www.kaggle.com/ethon0426/lending-club-20072020q1

The data is extracted using the available API.

```
import json
token = {"username":"                    ","key":"                                    }
with open('/content/.kaggle/kaggle.json', 'w') as file:
    json.dump(token, file)
```

Figure 6: API for Data Extraction

The time frame of the data is from 2007Q1 (Quarter 1) to 2020Q3 (Quarter 3). This data set includes all the customers who's applications were initially accepted by the bank and later proved to be troublesome. This information is used to train the machine learning models. The new information is fed to make the acquisition decisions. Some of the key attributes of the data include:

Details	Count
Number of Customers	29,25,493
Number of Features	140

The data features are as follows:

Feature	Description
acc_now_delinq	The number of accounts now delinquent.
acc_open_past_24mths	Number of trades opened in past 24 months.
addr_state	The state provided by the borrower
all_util	Balance to credit limit on all trades
annual_inc	The self-reported annual income
annual_inc_joint	The combined self-reported annual income
application_type	the loan is an individual application or a joint application
avg_cur_bal	Average current balance of all accounts
bc_open_to_buy	Total open to buy on revolving bankcards.
bc_util	Ratio:total current balance to high credit all bankcard acct
chargeoff_within_12_mths	Number of charge-offs within 12 months
collection_recovery_fee	post charge off collection fee
collections_12_mths_ex_med	Number:collections in 12 months excluding medical
delinq_2yrs	Number:30+ days past-due incidences of delinquency past 2yrs
delinq_amnt	The past-due amount owed for the accounts:now delinquent.
desc	Loan description provided by the borrower
dti	Ratio:total monthly debt payments to monthly income.
dti_joint	Ratio: co-borrowers' monthly payments to combined income
earliest_cr_line	The month borrower's earliest reported credit line was opened
emp_length	Employment length in years
emp_title	The job title

Feature	Description
fico_range_high	The upper boundary range the borrower's FICO
fico_range_low	The lower boundary range the borrower's FICO
funded_amnt	The total amount committed to that loan
funded_amnt_inv	The total amount committed by investors for that loan
grade	LC assigned loan grade
home_ownership	The home ownership status
id	A unique LC assigned ID for the loan listing.
il_util	Ratio:total current balance to high credit on all install acct
initial_list_status	The initial listing status of the loan.
inq_fi	Number of personal finance inquiries
inq_last_12m	Number of credit inquiries in past 12 months
inq_last_6mths	The number of inquiries in past 6 months
installment	The monthly payment owed by the borrower
int_rate	Interest Rate on the loan
issue_d	The month which the loan was funded
last_credit_pull_d	The most recent month LC pulled credit for this loan
last_fico_range_high	The upper boundary range the borrower's last FICO pulled
last_fico_range_low	The lower boundary range the borrower's last FICO pulled
last_pymnt_amnt	Last total payment amount received
last_pymnt_d	Last month payment was received
loan_amnt	The listed amount of the loan applied for by the borrower.
loan_status	Current status of the loan
max_bal_bc	Maximum current balance owed on all revolving accounts
member_id	A unique LC assigned Id for the borrower member.
mo_sin_old_il_acct	Months since oldest bank installment account opened
mo_sin_old_rev_tl_op	Months since oldest revolving account opened
mo_sin_rcnt_rev_tl_op	Months since most recent revolving account opened
mo_sin_rcnt_tl	Months since most recent account opened
mort_acc	Number of mortgage accounts.
mths_since_last_delinq	The number of months since the borrower's last delinquency.
mths_since_last_major_derog	Months since most recent 90-day or worse rating
mths_since_last_record	The number of months since the last public record.
mths_since_rcnt_il	Months since most recent installment accounts opened
mths_since_recent_bc	Months since most recent bankcard account opened.
mths_since_recent_bc_dlq	Months since most recent bankcard delinquency
mths_since_recent_inq	Months since most recent inquiry.
mths_since_recent_revol_delinq	Months since most recent revolving delinquency.
next_pymnt_d	Next scheduled payment date
num_accts_ever_120_pd	Number of accounts ever 120 or more days past due
num_actv_bc_tl	Number:currently active bankcard accounts
num_actv_rev_tl	Number:currently active revolving trades
num_bc_sats	Number:satisfactory bankcard accounts
num_bc_tl	Number:bankcard accounts

Feature	Description
num_il_tl	Number:installment accounts
num_op_rev_tl	Number:open revolving accounts
num_rev_accts	Number:revolving accounts
num_rev_tl_bal_gt_0	Number:revolving trades with balance >0
num_sats	Number:satisfactory accounts
num_tl_120dpd_2m	Number:accounts currently 120 days past due
num_tl_30dpd	Number:accounts currently 30 days past due
num_tl_90g_dpd_24m	Number:accounts 90 or more days past due in last 24 months
num_tl_op_past_12m	Number:accounts opened in past 12 months
open_acc	Number:open credit lines in the borrower's credit file.
open_acc_6m	Number:open trades in last 6 months
open_act_il	Number: currently active installment trades
open_il_12m	Number:installment accounts opened in past 12 months
open_il_24m	Number:installment accounts opened in past 24 months
open_rv_12m	Number:revolving trades opened in past 12 months
open_rv_24m	Number:revolving trades opened in past 24 months
out_prncp	Remaining outstanding principal for total amount funded
out_prncp_inv	Remaining outstanding principal funded by investors
pct_tl_nvr_dlq	Percent of trades never delinquent
percent_bc_gt_75	Percentage of all bankcard accounts >75% of limit.
policy_code	publicly available policy code
pub_rec	Number:derogatory public records
pub_rec_bankruptcies	Number:public record bankruptcies
purpose	A category provided by the borrower for the loan request.
pymnt_plan	Indicates if a payment plan has been put in place for the loan
recoveries	post charge off gross recovery
revol_bal	Total credit revolving balance
revol_bal_joint	Sum of revolving credit balance of the co-borrowers
revol_util	Revolving line utilization rate
sec_app_chargeoff_within_12_mths	Number:charge-offs within last 12 months - sec appl
sec_app_collections_12_mths_ex_med	Number:collections within last 12 months - sec appl
sec_app_earliest_cr_line	Earliest credit line at time of application - sec appl
sec_app_fico_range_high	FICO range (low)- sec appl
sec_app_fico_range_low	FICO range (high)- sec appl
sec_app_inq_last_6mths	Credit inquiries in the last 6 months - sec appl
sec_app_mort_acc	Number of mortgage accounts - sec appl
sec_app_mths_since_last_major_derog	Months since most recent 90-day or worse rating - sec appl
sec_app_num_rev_accts	Number:revolving accounts - sec appl
sec_app_open_acc	Number:open trades at time - sec appl
sec_app_open_act_il	Number:currently active installment trades - sec appl
sec_app_revol_util	Ratio:total current balance to high credit for all revolving acct
sub_grade	LC assigned loan subgrade
tax_liens	Number of tax liens

term	int_rate	installment	grade	sub_grade	emp_title	emp_length	home_ownership	annual_inc
36 months	10.65%	162.87	B	B2	NaN	10+ years	RENT	24000.0
60 months	15.27%	59.83	C	C4	Ryder	< 1 year	RENT	30000.0
36 months	15.96%	84.33	C	C5	NaN	10+ years	RENT	12252.0
36 months	13.49%	339.31	C	C1	AIR RESOURCES BOARD	10+ years	RENT	49200.0
60 months	12.69%	67.79	B	B5	University Medical Group	1 year	RENT	80000.0

Figure 7: Data Frame Sample

The previous version of this project's data source[5] included the rejected applications as well. It contains information of the customers who's loan requests were rejected by the bank using the traditional static methods. It would be interesting to explore that data set as well.

The *app_yr* column which stands for application year is extracted from the *App_date* column. This is used to filter the data as well as further exploration via visualisation.

The rejected data set in the form of a data frame:

	Amount Requested	App_date	title	Risk_Score	dti	Zip Code	State	emp_length	policy_code	app_yr
755491	1000.0	2016-04-01	other	633.0	2.69	331xx	FL	0	0.0	2016
755492	4000.0	2016-04-01	debt_consolidation	633.0	28.26	834xx	ID	0	0.0	2016
755493	5000.0	2016-04-01	moving	633.0	-1	648xx	MO	0	0.0	2016
755494	1000.0	2016-04-01	moving	628.0	21.43	380xx	TN	0	0.0	2016
755495	3000.0	2016-04-01	debt_consolidation	633.0	8.49	895xx	NV	2	2.0	2016

Figure 8: Rejected Data

[5]https://www.kaggle.com/wordsforthewise/lending-club

4.2 Data Pre-Processing and Transformation

At first glance, it is visible that the data is huge. It also includes data which spans over more than 10 years. Therefore, it is decided that data from 2016 onward will be used for the model building. The primary reasons for this filtering are:

- Domain knowledge and research
- Infrastructure constraints

4.2.1 Data Transformation

First, the *issue_d* column is changed to a date-time data type and then the year is extracted from it saving it into a new column named *issue_yr*. This column is further converted into "int64" data type and used to filter the entire data set by discarding data before 2016.

After the filtering, the next step is to get an overview of the data.

```
Rows :  2038052

Columns :  141

Features :  ['loan_amnt', 'funded_amnt', 'funded_amnt_inv', 'term', 'int_rate', 'installment', 'grade', 'sub_grade'

Missing Values :  63634652

Unique values :
 loan_amnt                                  1561
funded_amnt                                 1561
funded_amnt_inv                             1585
term                                           2
int_rate                                     284
                                            ...
orig_projected_additional_accrued_interest  91742
hardship_payoff_balance_amount             158947
hardship_last_payment_amount                48871
debt_settlement_flag                            2
issue_yr                                        5
Length: 141, dtype: int64
```

Figure 9: Data Set Overview

It is evident that the data reduced to 20,38,052 and with the addition of *issue_yr*, the number of features increased to 141. There are 6,36,34,652 missing values which need to be addressed. The information about the composition of data types and basic statistics related to all the numerical features are also studied. The "datetime64[ns]" data type refers to the converted *issue_d* feature and the "int64" is for the *issue_yr*.

The basic statistics (Figure 11) are described in terms of the count of instances in each feature, the mean value, the standard deviation, the minimum value, the 1st quartile (25%), the median or 2nd quartile (50%), the 3rd quartile (75%) and the maximum value for each feature.

```
float64            106
object              33
int64                1
datetime64[ns]       1
dtype: int64
```

Figure 10: Data Type Composition

	loan_amnt	funded_amnt	funded_amnt_inv	installment	annual_inc	dti	delinq_2yrs	fico_rang
count	2.038052e+06	2.038052e+06	2.038052e+06	2.038052e+06	2.038052e+06	2.034946e+06	2.038052e+06	2.0380
mean	1.562161e+04	1.562159e+04	1.561771e+04	4.592281e+02	8.207569e+04	1.980314e+01	2.789296e-01	7.0293
std	9.886748e+03	9.886750e+03	9.886041e+03	2.842975e+02	1.268360e+05	1.797675e+01	8.352470e-01	3.4818
min	1.000000e+03	1.000000e+03	7.250000e+02	7.610000e+00	0.000000e+00	-1.000000e+00	0.000000e+00	6.6000
25%	8.000000e+03	8.000000e+03	8.000000e+03	2.504700e+02	4.800000e+04	1.216000e+01	0.000000e+00	6.7500
50%	1.300000e+04	1.300000e+04	1.300000e+04	3.816100e+02	6.800000e+04	1.831000e+01	0.000000e+00	6.9500
75%	2.100000e+04	2.100000e+04	2.100000e+04	6.190000e+02	9.800000e+04	2.533000e+01	0.000000e+00	7.2000
max	4.000000e+04	4.000000e+04	4.000000e+04	1.719830e+03	1.100000e+08	9.990000e+02	5.800000e+01	8.4500

Figure 11: Basic Statistics

A convention that is followed in a data science pipeline is to eliminate columns or features that have more than 70% missing values. These features have little to no affect on the target feature even after data imputation. Therefore, a custom function is created which calculates the percentage of missing values. This function helps to identify the features which need to be dropped and reinforce the decision to eliminate features with more than 70% missing values. After elimination, the number of features dropped to 109.

Before handling the remaining missing values, certain changes are performed as part of data preprocessing. The *home_ownership* column has three separate values - 'ANY', 'NONE', 'OTHER', which are clubbed into the existing category of 'RENT'. All the upper case categories are converted to lower case for convenience. The *verification_status* feature is also treated in a similar manner. A few of the features like *url*, *addr_state*, *pymnt_plan* and *policy_code* are dropped based on domain knowledge and data study.

The feature *emp_length* has two categories - 10+ years and <1 year. The "+" sign is dropped, the <1 year is converted to 0 years and the word "years" is dropped from all the instances effectively converting it into a numerical feature. Similarly, the word "months" is eliminated from *term* feature and the "%" sign is dropped from the *int_rate* and *revol_util* columns.

Once these changes are made, then for eliminating the missing values a custom function is created for imputation. It imputed the missing numerical values with the average of that feature and the categorical values are imputed with the most frequent value in column.

To ensure accuracy, all the values which are in float after imputation and represented year or month are converted to integers.

Your selected dataframe has 141 columns.
There are 68 columns that have missing values.

	Missing Values	% of Total Values
hardship_loan_status	1899566	93.2
hardship_reason	1899348	93.2
hardship_status	1899345	93.2
hardship_dpd	1899343	93.2
deferral_term	1899342	93.2
...	...	...
open_il_24m	61	0.0
last_credit_pull_d	28	0.0
pct_tl_nvr_dlq	2	0.0
inq_last_6mths	1	0.0
zip_code	1	0.0

Figure 12: Features in descending order according to percentage of missing values

Your selected dataframe has 106 columns.
There are 0 columns that have missing values.

Missing Values	% of Total Values

Figure 13: Percentage of missing values after Imputation

The final processing before feature selection is required to extract the target variable. This is done by observing the *loan_status* column. The loans classified as "Current" are not considered because they do not provide any information about their risk status. All the loans which are categorised as "Fully Paid", "In Grace Period", "Issued" are classified as *0* - No risk and the rest are classified as *1* - Risk and is appended as a new column *target*. The original *loan_status* column is dropped, giving us the final data frame before feature selection.

```
Current                1013507
Fully Paid              788161
Charged Off             205768
Late (31-120 days)       15761
In Grace Period           9738
Late (16-30 days)         2643
Issued                    2062
Default                    412
Name: loan_status, dtype: int64
```

Figure 14: Loan Status

funded_amnt	funded_amnt_inv	term	int_rate	installment	grade	sub_grade	emp_title	emp_length	home_ownership	a
12000.0	12000.0	36	7.97	375.88	A	A5	associate	10	own	
33000.0	33000.0	36	7.21	1022.12	A	A3	nurse	0	mortgage	
10000.0	10000.0	36	9.44	320.05	B	B1	therapist	3	mortgage	
8000.0	8000.0	36	16.02	281.34	C	C5	owner	0	mortgage	
12800.0	12800.0	36	13.59	434.93	C	C2	area director	5	rent	

Figure 15: Data Frame before Feature Selection (Snippet of few features)

4.2.2 Feature Selection

At this point, there are two types of features present in the data:

- Numerical - Discrete and Continuous
- Categorical - Nominal and Ordinal

For categorical, there are two methods that are explored for encoding before applying feature selection techniques:

- One-Hot Encoding
- Label Encoding

Before applying these methods, the column *emp_title* needs to be addressed. This column has too many unique values to be encoded using either of these methods. Therefore, the best way to test it's significance is by performing a **Chi-square test**.

Chi Square Distribution with n degrees of freedom χ^2_m

The distribution of the sum of n squared independent standard normal random variables is called the chi-squared distribution. It depends on m, which is called the degrees of freedom[6] of the chi-squared distribution.

$$Z \sim N[0,1] \Rightarrow Z \sim \chi^2_1$$

If Z_i, $i = 1, ..., m$ are independent $N[0,1]$, then

$$\sum_1^m Z_i \sim \chi^2_m$$

Moreover, if Z_i, $i = 1, ..., m$ are independent $N[0,\sigma^2]$, then

$$\sum_1^m (Z_i/\sigma^2) \sim \chi^2_m$$

Other properties include,

$$x_1 \sim \chi^2_{m_1}$$
$$x_2 \sim \chi^2_{m_2}$$

Then,

$$x_1 + x_2 \sim \chi^2_{m_1+m_2}$$

The amount of skewness declines as the number of degrees of freedom rises.

In practice, the χ^2_m is used to test the relationship between two categorical variables.

[6]Degrees of freedom refers to the number of logically independent variables.

H_0: The variables are not related
H_1 : The variables are related

The test statistic

$$Z_i = \sum \frac{Observed\,Value_i - Expected\,Value_i}{Expected\,Value_i} \sim \chi^2_m$$

At the 95% confidence interval, if p-value < 0.05, we reject H_0, else we are unable to reject the null. This hypothesis test is used to check the significance of the *emp_title* feature with *target* feature. In order to proceed with the statistical test, the *emp_title* feature is first converted into a crosstab and then a function is created to execute the test.

target	0	1
emp_title		
\tcfo	1	0
\tdrug test administrator	0	1
\tmultimedia supervisor	1	0
\tpasteurization	1	0
\tquality packer	1	0
...	...	...
{job title}	0	1
{owner}truck driver	1	0
| principal business solution architect|	1	0
associate tech support analyst	1	0
senior it field support	1	0

Figure 16: Emp_title Crosstab

The result of the test tells that the p-value $\not< \mathbf{0.05}$, therefore the null is not rejected and the feature is not significant. Hence, this feature is dropped.

```
significance=0.050, p=1.000
Variables are not associated(fail to reject H0)
```

Figure 17: Null Hypothesis Not Rejected - Feature Not Significant

Now, the remaining categorical data are encoded. The choice of type of encoding method depends on the nature of the variable. For the variables which are nominal[7] in nature, One-Hot Encoding is the method that is used. In One-hot encoding a column which has categorical data is taken, it is then label encoded and then split into multiple columns. In this data, each nominal feature has a lot of categories which will lead to a massive increase in number of columns and will lead to increase in space taken by the data. Therefore, a function is created which takes the top 10 categories from each feature and one-hot encodes them.

Joint App	N	Y	36	60	Verified	Not Verified	w	f
0	1	0	1	0	1	0	1	0
1	0	1	1	0	1	0	1	0

Figure 18: One - Hot Encoded Data (Sample)

For the variables which are ordinal[8] in nature, Label Encoding is the method that is used. In Label encoding a column which has categorical data is taken and then label encoded based on an hierarchical order. In this data, we had multiple columns which are ordinal in nature. Hence, a function is created to handle multiple features and label encode them simultaneously.

term	int_rate	installment	grade ↓	sub_grade ↓	emp_length
36	7.97	375.88	0	4	10
36	7.21	1022.12	0	2	0

Figure 19: Label Encoded Data (Sample)

[7]The data is labeled with no specific order.
[8]The data is labeled with a specific order.

Once the entire data is encoded, the chi-square test is applied to all the categorical features and features that fail this test are eliminated.

```
(array([6.01845279e+04, 2.60234408e+05, 2.96406803e+03, 3.43256619e+03,
        1.55183415e+01, 4.83329945e+02, 5.49059318e+02, 3.92257363e+02,
        7.92660037e+00, 9.89818016e+00, 1.86859817e-01, 2.66357253e+02,
        5.81548023e+02, 4.02593627e+01, 1.82221895e+01, 4.84009176e+02,
        5.73447020e+02, 3.95971375e+02, 7.86275622e+00, 9.60646450e+00,
        2.43728223e-01, 2.64319078e+02, 5.80474469e+02, 4.48306455e+01,
        1.75455751e+01, 7.54962151e+01, 9.75145173e+02, 9.44020215e-01,
        3.63346447e+02, 6.94265128e+03, 2.38267667e+04, 1.54639510e+03,
        3.00034172e+03, 1.69923130e+00, 6.24188529e+00]),
 array([0.00000000e+000, 0.00000000e+000, 0.00000000e+000, 0.00000000e+000,
        8.17085842e-005, 4.02865593e-107, 2.01663385e-121, 2.66962494e-087,
        4.87132384e-003, 1.65442297e-003, 6.65543117e-001, 7.06073041e-060,
        1.72724920e-128, 2.22387007e-010, 1.96575257e-005, 2.86657574e-107,
        9.98852119e-127, 4.14884047e-088, 5.04634090e-003, 1.93893582e-003,
        6.21526268e-001, 1.96376503e-059, 2.95715509e-128, 2.14833909e-011,
        2.80502782e-005, 3.66108013e-018, 4.53813367e-214, 3.31246085e-001,
        5.25912970e-081, 0.00000000e+000, 0.00000000e+000, 0.00000000e+000,
        0.00000000e+000, 1.92388537e-001, 1.24763596e-002]))
```

Figure 20: Chi Score

But, what about the numerical data? For feature selection with numerical data a correlation measure is used called **Point Bi-Serial Correlation** because the target variable is dichotomous[9] in nature.

Point Bi-Serial Correlation

Point bi-serial correlation is a measure of the relationship between an continuous variable Y and a categorical or dichotomous variable X_i, where $i = 1, 2$. It is also used to estimate the direction of correlation, positive or negative. This measure assumes that Y is normally distributed (approximately) and has equal variance for both categories of the dichotomous variable. The Shapiro-Wilk test and Levene's test are usually used to check the validity of these assumptions. If all the assumptions are satisfied the point bi- serial correlation is given by the following.

$$X_i = \begin{cases} 0, & \text{Sample Size} = n_1 \\ 1, & \text{Sample Size} = n_2 \end{cases}$$

Therefore, the total sample size is $n_1 + n_2$

$$r_{pb} = \frac{E(Y|X_i = 1) - E(Y|X_i = 0)}{s_n} \sqrt{\frac{n_1 n_2}{n_1 + n_2}}$$

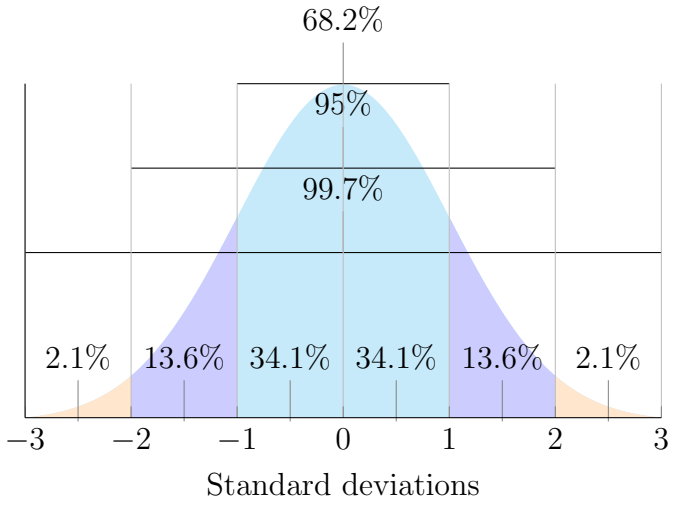
[9]Nominal variables with only two categories or values

where s_n is the standard deviation of the continuous variable Y

$$s_n = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})}$$

For feature selection, the r_{pb} is calculated for all continuous variables and the dichotomous variable - *target*. Using a threshold coefficient of **0.5**, features with high degree of correlation were selected. The final steps, before model building are removal of the outliers[10] and converting the date-time features into numerical values for ease in model building. The date-time features were converted to ordinals with the proleptic Gregorian ordinal 1/1/1 as the starting point. For detection and elimination of outliers **Z-Score** is used.

Outlier detection using Z-Score



The z-statistic is used to detect outliers. The z score is an estimate of the distance of the data point to the sample mean. The z score is calculated using the following formula.

$$Z_i = \frac{x_i - \bar{x}}{\sigma}$$

where σ represents the sample standard deviation. Using a threshold of 3, the z score is used to eliminate data points which are far from the sample mean. If $Z_i > 3$ and $Z_i <$ -3, the data point is an outlier.

Now, the final data frame will be used for analysis using visualisations as well as model building for predictions.

[10]Data points that vary significantly from other observations

sub_grade	emp_length	annual_inc	issue_d	dti	delinq_2yrs	earliest_cr_line	fico_range_low	fico_range_high
4	10	42000.0	736573	27.74	0.0	728811	715.0	719.0
11	5	90000.0	736573	22.63	0.0	726072	660.0	664.0
22	6	40000.0	736573	21.42	0.0	732281	675.0	679.0
19	10	46800.0	736573	32.23	0.0	724092	680.0	684.0
17	10	62500.0	736573	35.02	0.0	731672	725.0	729.0

Figure 21: Final Data Frame (Snippet of few features)

4.3 Visualisation

Visualisations help gain insights and answer a lot of questions, which might help understand the problem statement better and provide solutions. Let us try answering some of the questions related to this problem statement.

Question.1: *What is the distribution of loans issued over the years?*

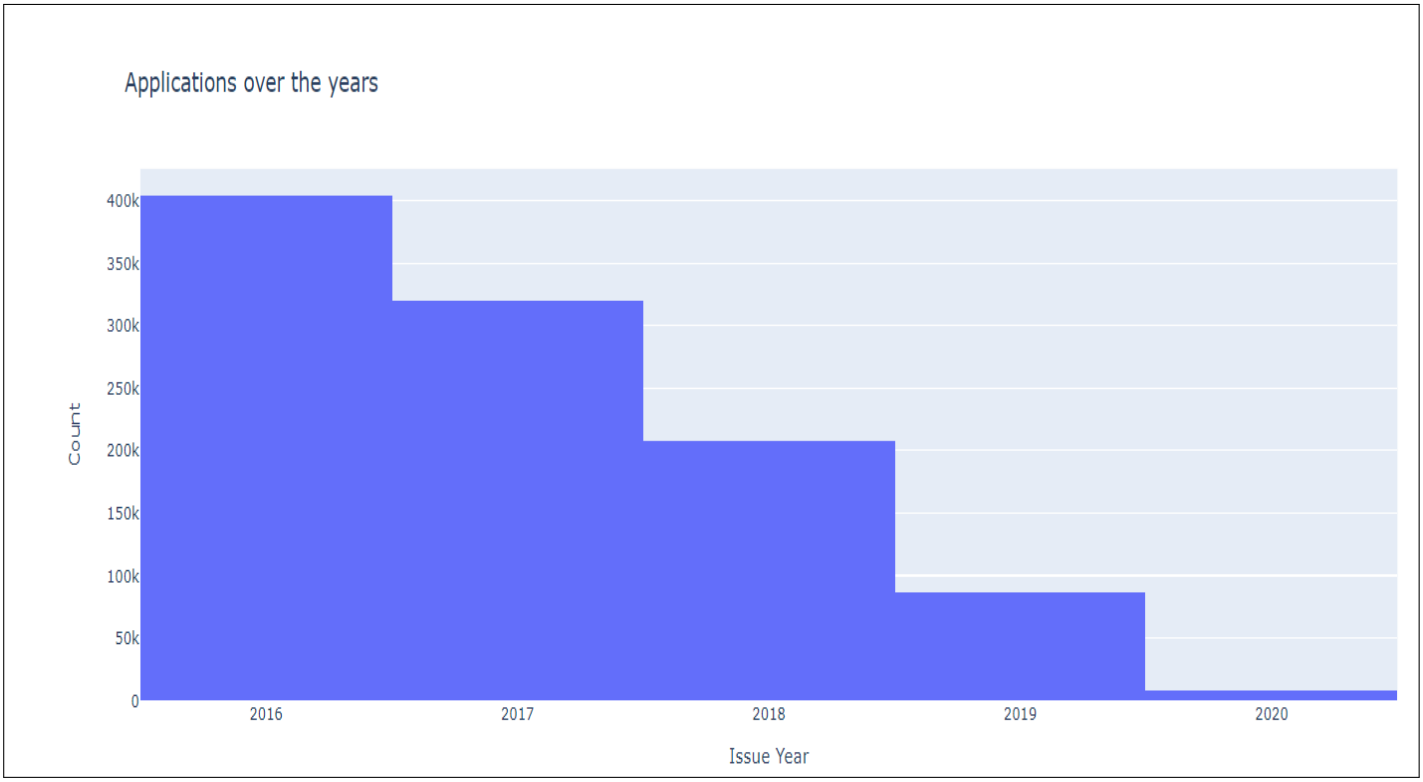


Figure 22: Loans Issued over the years

Answer. According to Figure 22, there has been a gradual decrease in the number of loans issued over the years. Year 2016 has the maximum loans issued and year 2020 has minimum loans. This could also be due to the fact that data of 2020 is till Q3 (Quarter 3) and the company based their judgement of loan issuance on previous year's defaults coupled with the onset of pandemic.

Question.2: *What is the composition of various home owners?*

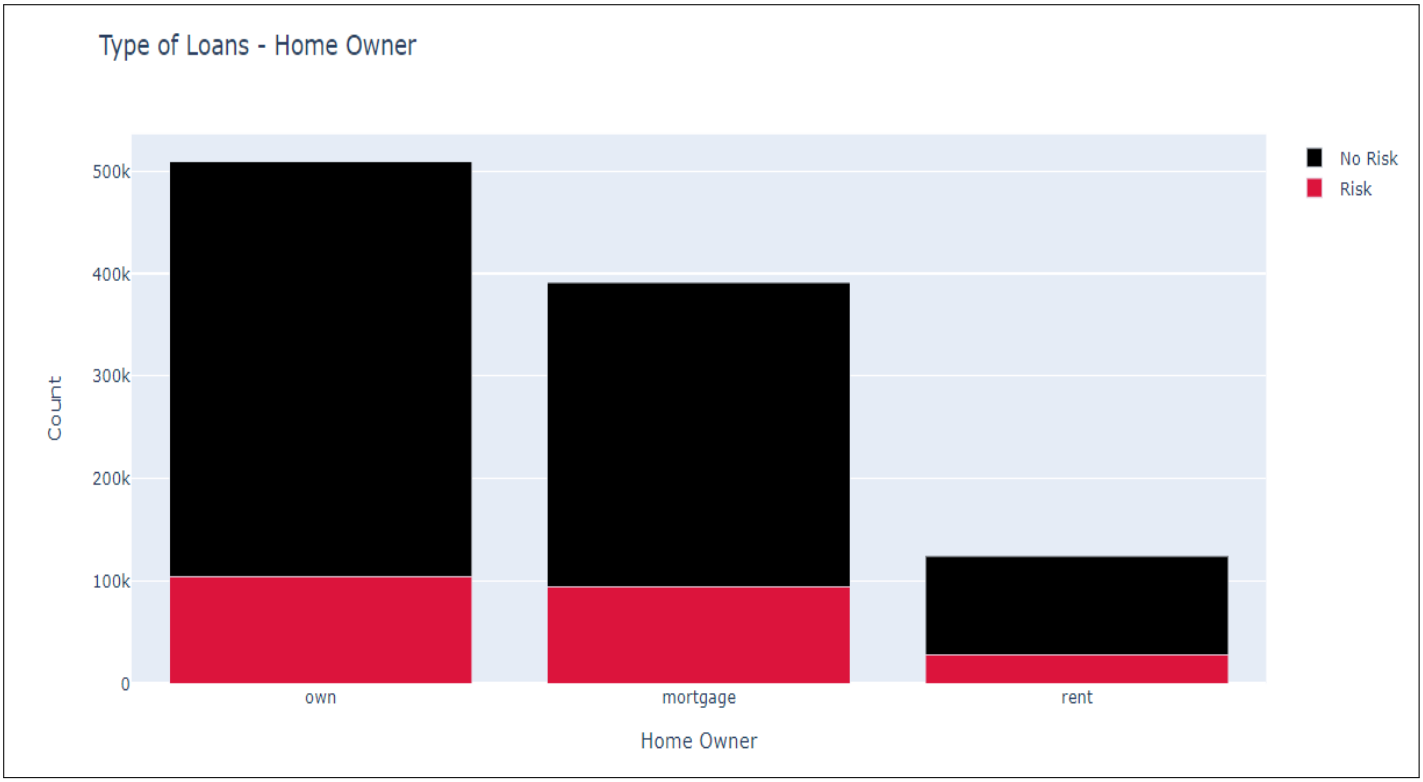


Figure 23: Composition of Home Owners

Answer. According to Figure 23, the maximum number of customers own the house and they do not default in loan repayment that much. Whereas, the number of customers that live in rented accommodation is relatively low but they tend to default more on loan repayment.

Question.3: *How is the target distributed?*

Answer. According to Figure 24, the data has more clients who have not defaulted in loan repayment than clients who have defaulted in the past. This is also a demonstration of a skewed data. The class has an imbalance. While model building and training this will have a significant impact on the predictions and the results. Therefore, class re-balancing needs to be taken into consideration while model building.

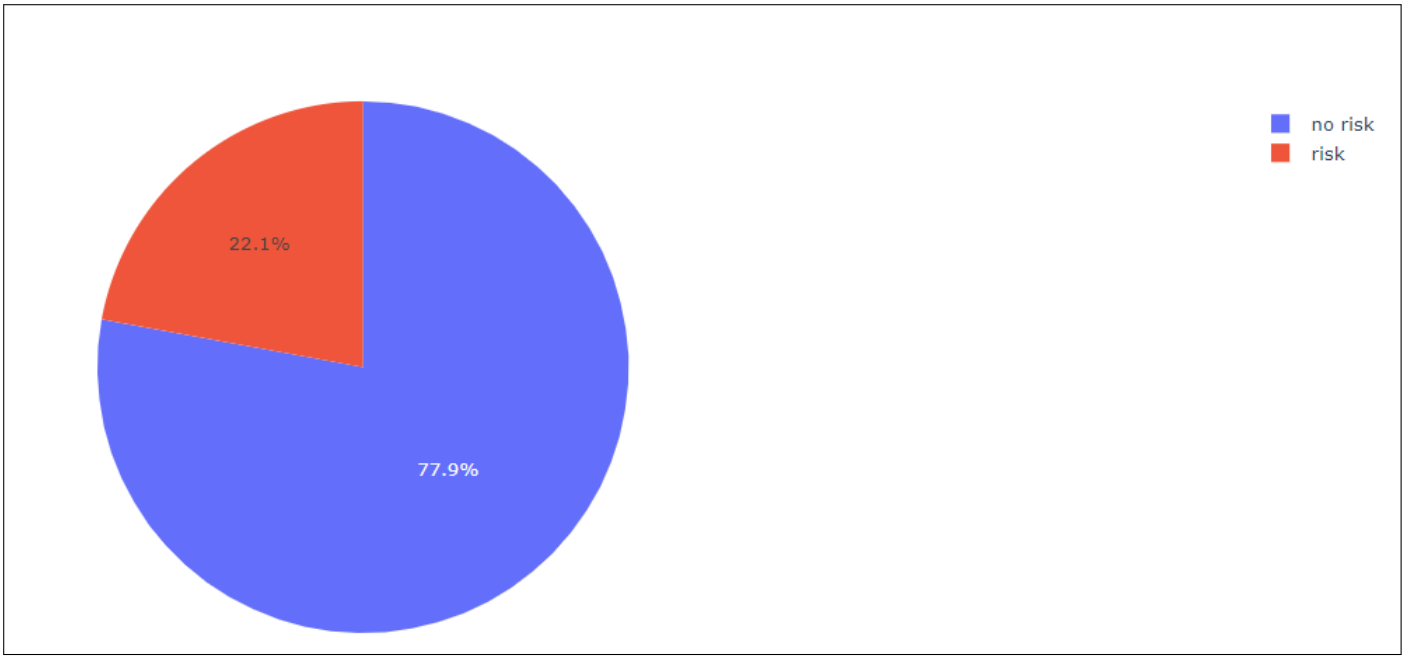


Figure 24: Target Distribution

Question.4: *What is the interest rate over time and grade?*

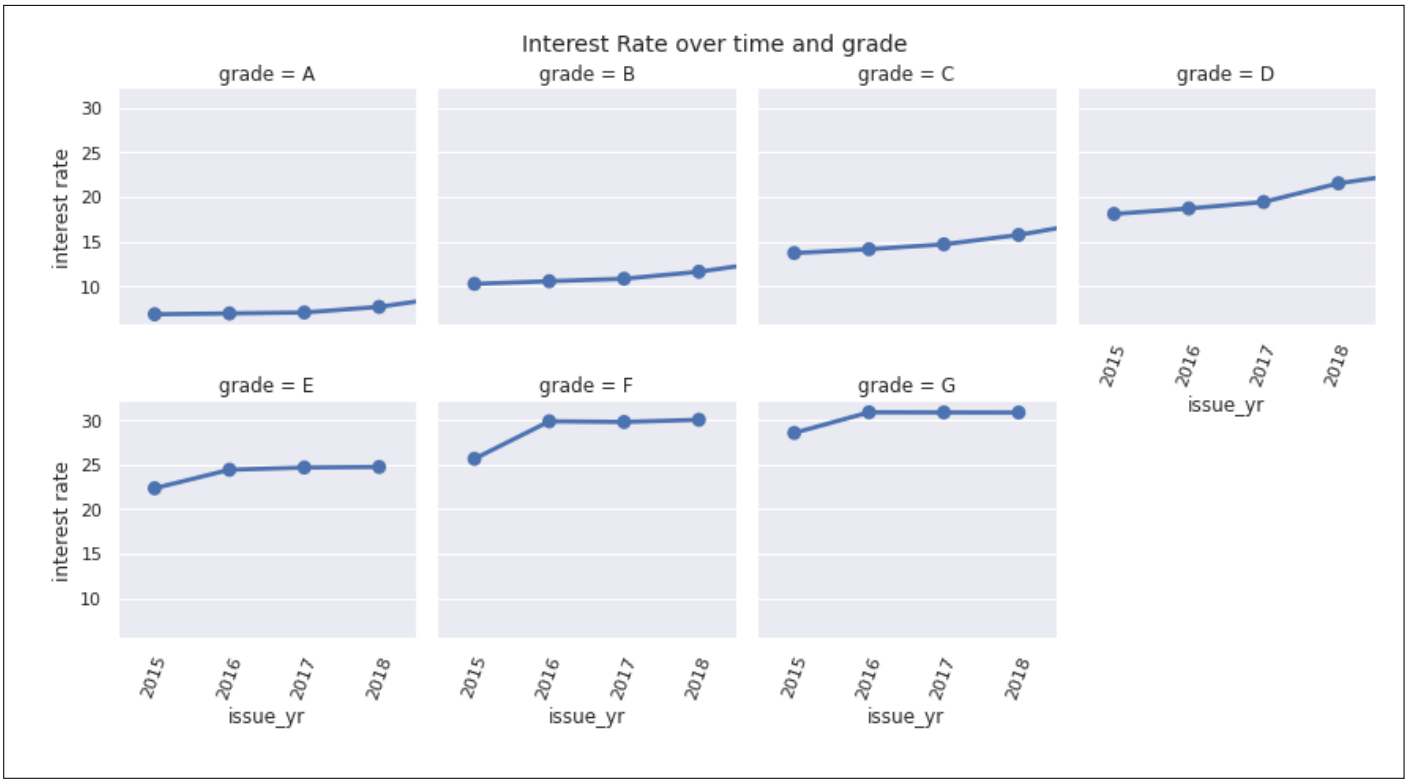


Figure 25: Interest rate over Time and Grade

Answer. According to Figure 25, higher the grade of the loan (A is the highest and G is the lowest grade), lower is the interest rate charged by the bank. This is due to the risk factor. Lower the grade of the loan, riskier is the proposition.

Question.5: *How are the features correlated?*

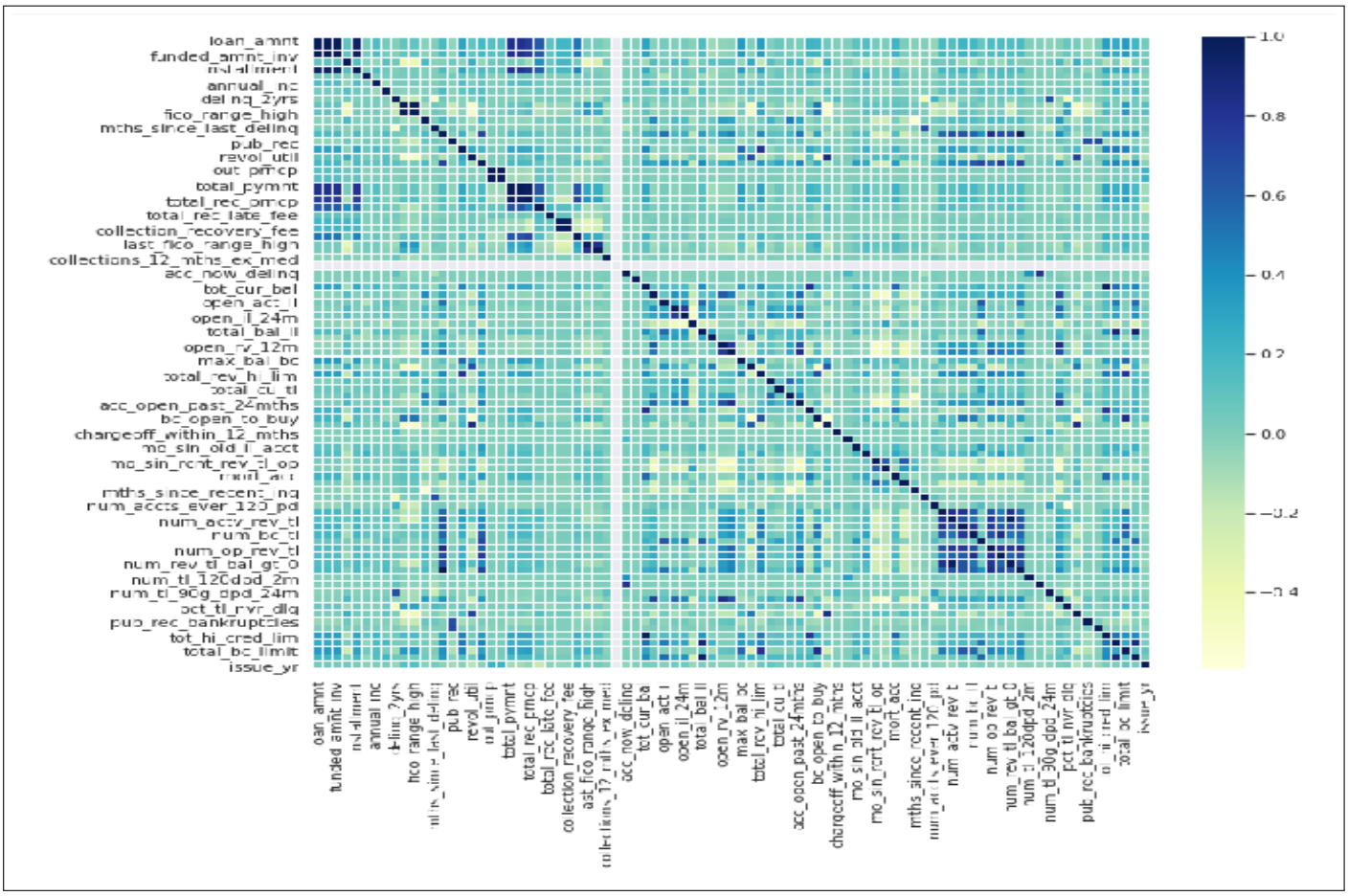


Figure 26: Feature Correlation

Answer. Figure 26 shows the correlation among all the numerical data using a correlation heatmap. The measure of correlation used is the **Pearson Correlation**. In this the strength of linear correlation between two variables is measured (r).The range of r is from -1 to +1. 0 (zero) denotes no correlation, greater than 0 denotes positive correlation and less than 0 denotes negative correlation.

$$r = \frac{\sum (m_i - \bar{m})(n_i - \bar{n})}{\sqrt{\sum (m_i - \bar{m})^2 \sum (n_i - \bar{n})^2}}$$

where, r = correlation coefficient
m_i = value of m-variable
$\bar{m}$ = mean of values in m-variable
n_i = value of n-variable
$\bar{n}$ = mean of values in n-variable

Question.6: *How is the debt-to-income ratio distribution?*

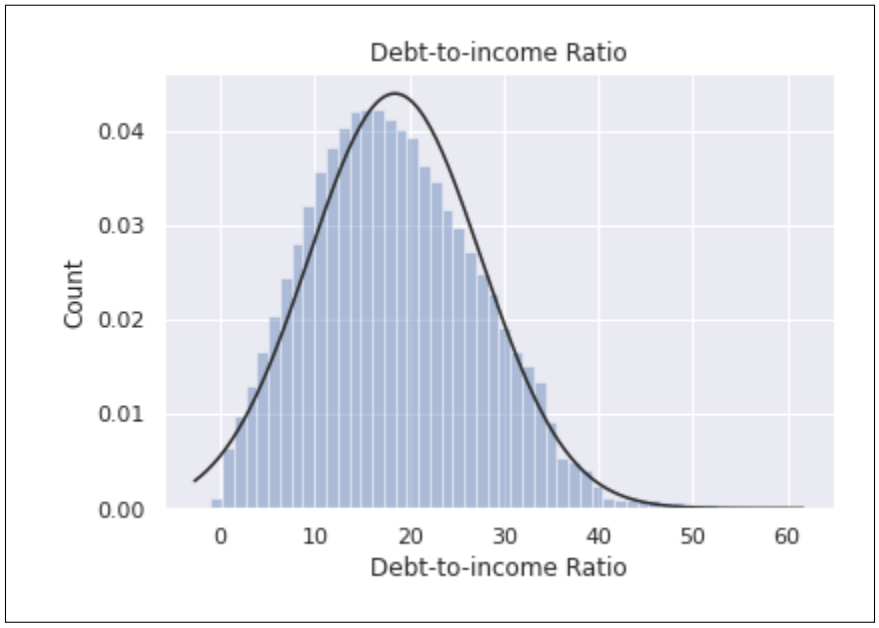


Figure 27: Debt-To-Income Ratio

Answer. Figure 27 shows the debt-to-income ratio is normally distributed with a peak at around 20%. Dti is one of the ways in which the bank (lender) can measure the repayment capability of the borrower.

$$Debt - to - Income\ Ratio = \frac{Monthly\ Debt\ Payments}{Gross\ Income} \times 100$$

Question.7: *What is the grade and sub-grade of loan distribution?*
Answer. The loans that were given to the customers were classified according to the grade. These grades in turn were devised based on the risk factor. The grades were further divided into sub-grades. Grade A is the highest with least risk associated with it and Grade G is the lowest with highest risk. In the sub-grades, the riskier the loan within the grade higher will be the number. For example, sub-grade *A1* has a lower risk and higher sub-grade than sub-grade *A5*.

Studying the distribution of these grades help determine the composition and the nature of the data. Figures 28 and 29 depict these distributions. It can be seen that the data contains maximum amount of Grade A loans and minimum amount of Grade G loans. The security of these grades are visible from the composition of risk to no risk as well. Higher grades have lesser default history than loans with lower grades. Similar, pattern can be observed in the sub-grades as well. Loans with higher sub-grades have lesser defaults than loans with lower sub-grades. The composition of the sub-grades illustrates maximum clients with sub-grade *A1* and minimum clients with sub-grade *G1*.

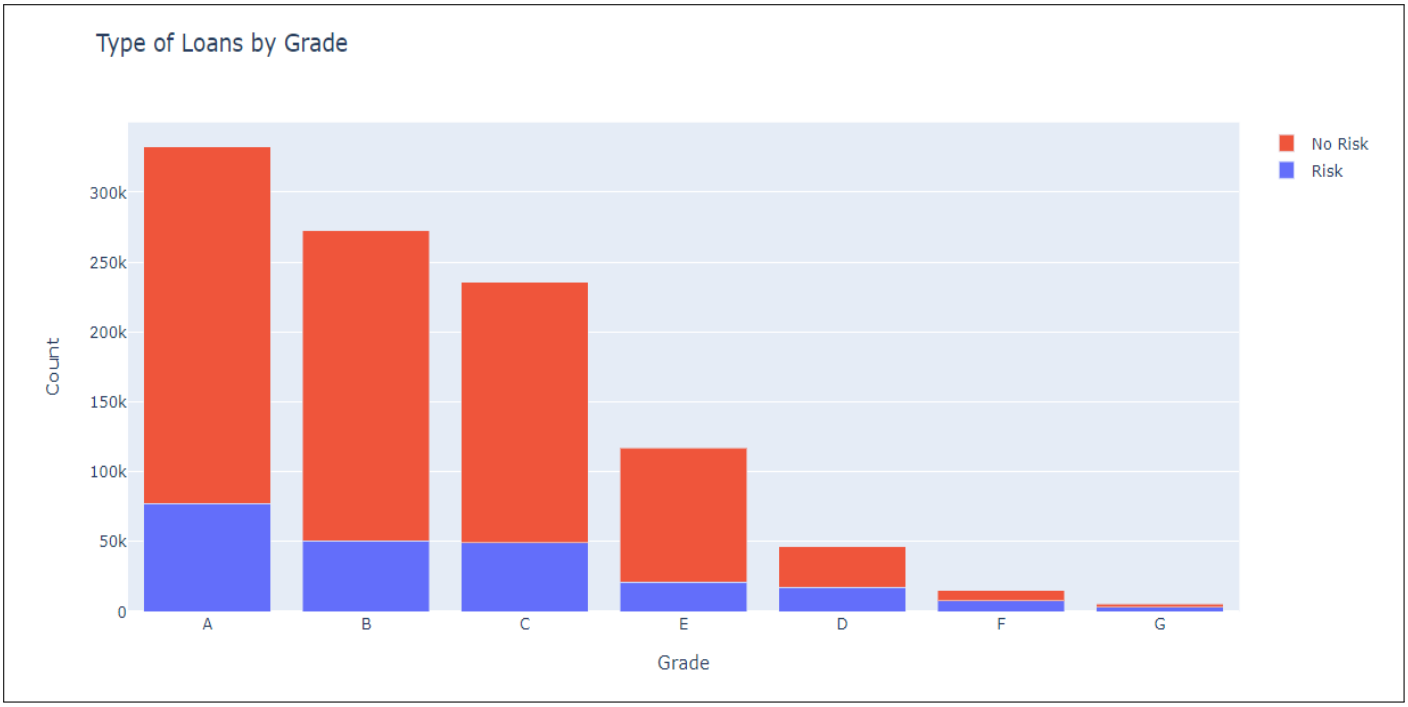


Figure 28: Grade of loan

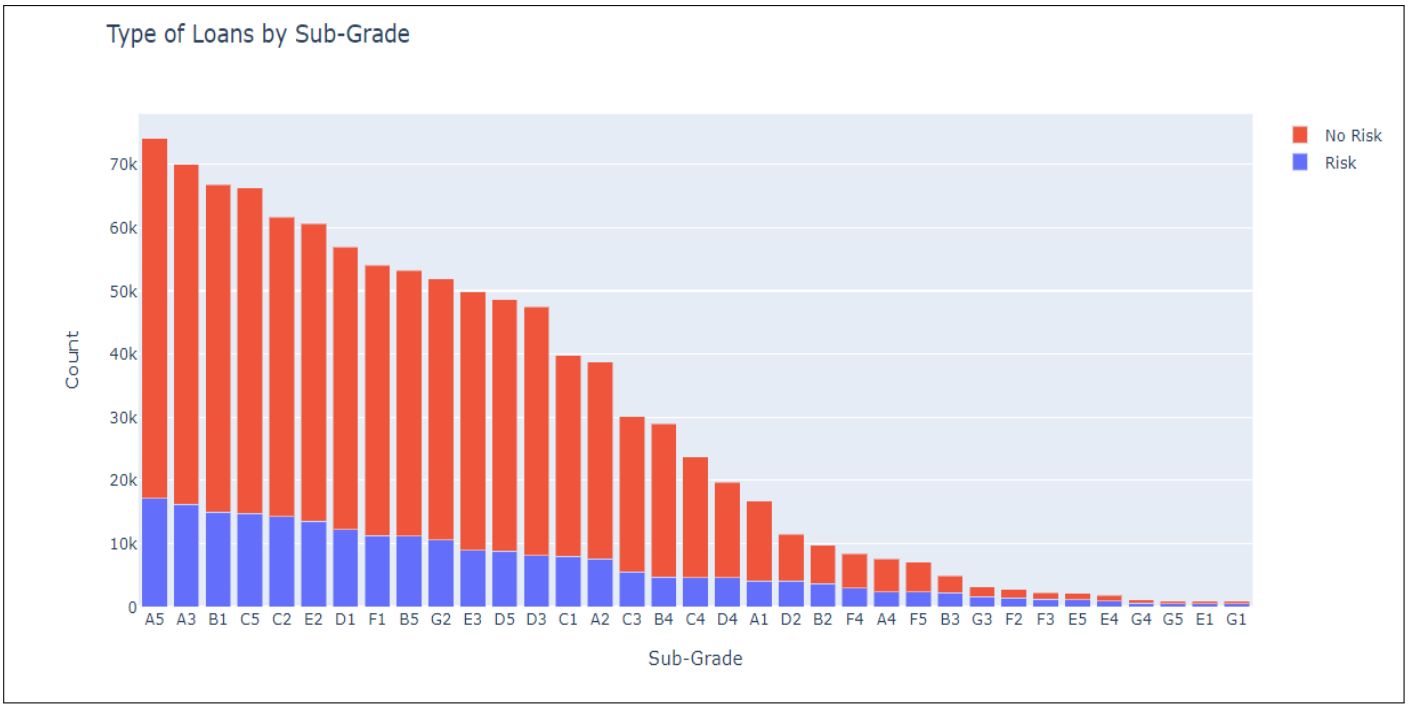


Figure 29: Sub - Grade of loan

Question.8: *What could be the projected loan amount over next 3 years?*

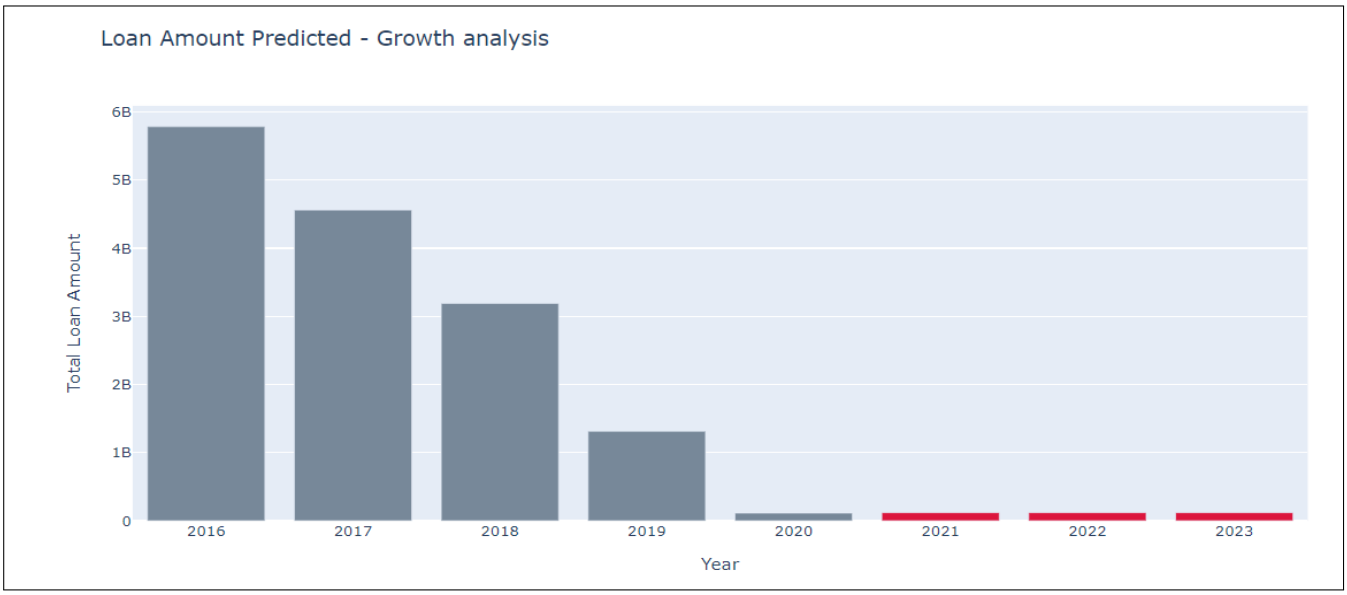


Figure 30: Projected Loan Amount over next 3 years

Answer. Figure 30 shows that the predicted amount of loan given each year would not be that high. This could be due to various factors.As the amount depends on previous years' numbers, it is consistent with the yearly decrease, but the amount is more than in year 2020.

Question.9: *What could be the projected number of loans issued over next 3 years?*

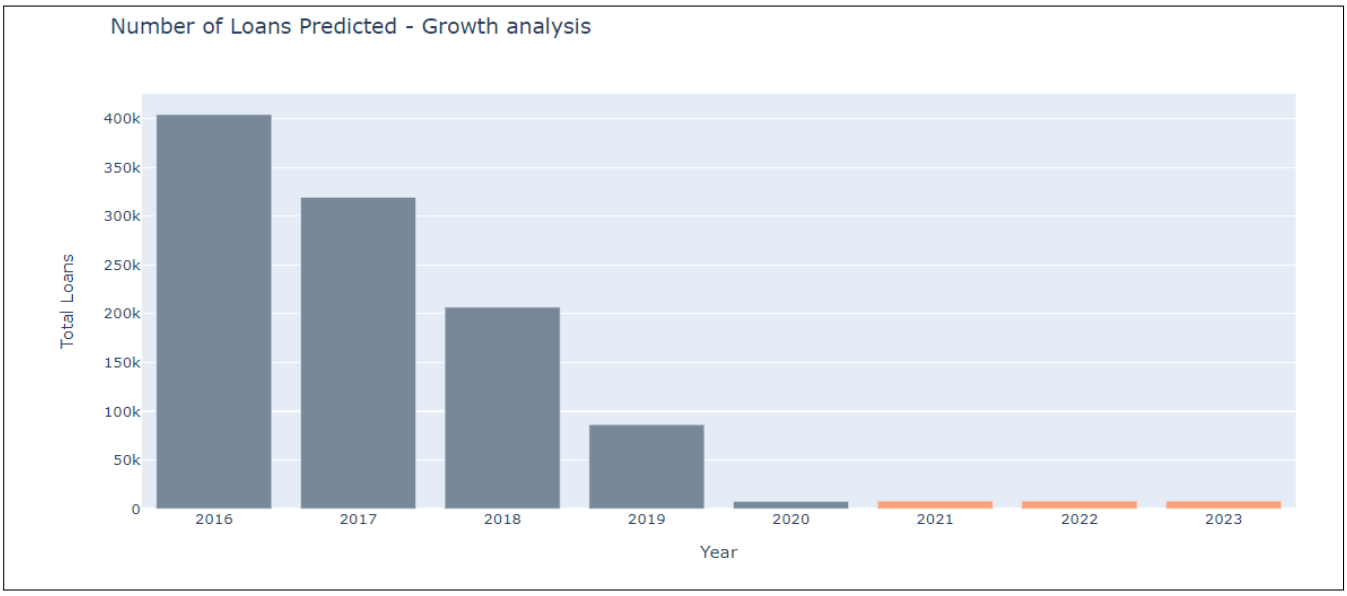


Figure 31: Projected Loans issued over next 3 years

Answer. Figure 31 is in line with the projection in Figure 30. As the number of loans to be issued in the next 3 years have been projected low therefore, the amount projected is also less. It is interesting to note that a function is created which would allow the customer to change the filters and view the visualisations accordingly. This feature will be added to the dashboard.

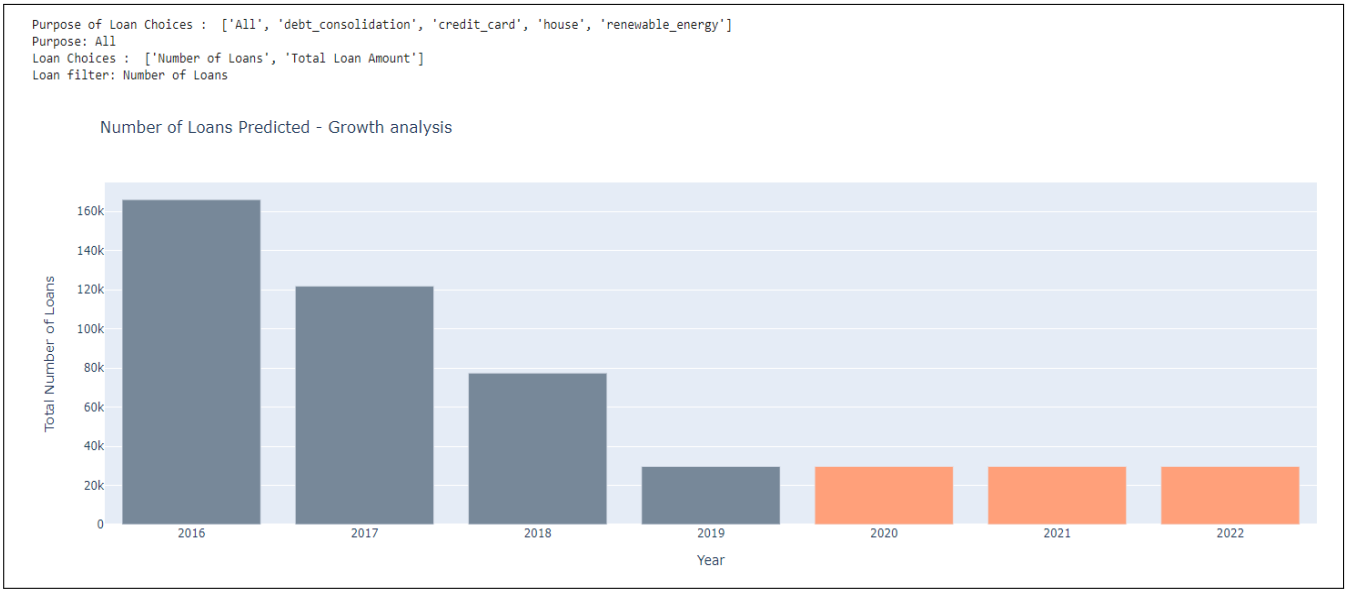


Figure 32: Visualisation Function

Question.10: *For what purpose did the defaulters take the loan?*

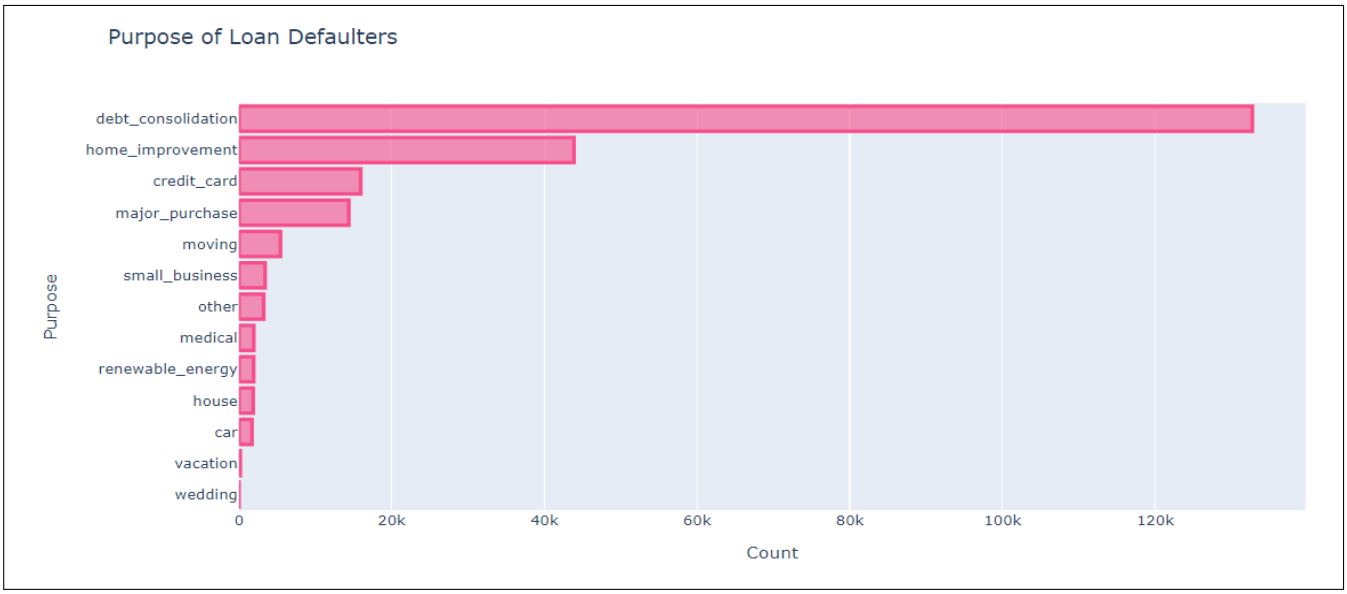


Figure 33: Purpose of loan - Defaulters

Answer. It is evident from Figure 33 that most of the defaulters took loan for **debt-consolidation**[11] .The next highest loan defaulters took the loan for **home improvement** followed by **credit-card** bill payment.

Question.11: *How much amount was spent on funding loans in each state?*

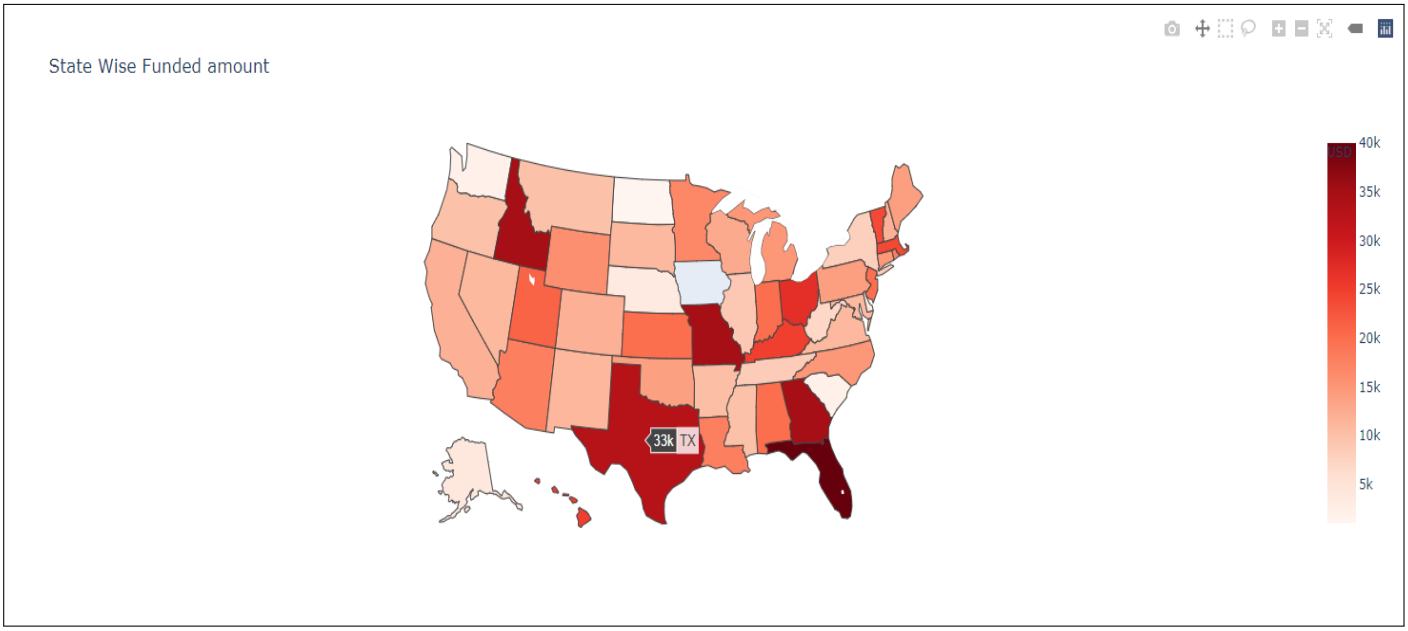


Figure 34: Amount spent on funding loans in each state

Answer. Figure 34 depicts that the maximum amount was spent on Florida. This could also be due to the population of the state and the frequency of the loan applications.

As mentioned earlier, there is another data set which contains the applications that were rejected by the company based on traditional static methods. It would be a good idea to analyse those data as well.

Many questions can be better answered with the help of the rejected applications. For example, the amount spent on funding loans in Florida could be due to the population, number of applications or any other factor. The analysis of the rejected applications could provide a wider picture and help in better understanding.

[11] Loan taken for repayment of other high interest loans

For the purpose of a recap, the details of the rejected applications is given in Figure 35.

```
Rows :  21339229

Columns :  10

Features :  ['Amount Requested', 'App_date', 'title', 'Risk_Score', 'dti', 'Zip Code', 'State', 'emp_length', 'policy_code', 'app_yr']

Missing Values :  16675458

Unique values :
 Amount Requested      3099
App_date              1096
title                   27
Risk_Score             691
dti                 113052
Zip Code              1001
State                   51
emp_length              11
policy_code              2
app_yr                   3
dtype: int64
```

Figure 35: Rejected Data Overview

On revisiting **Question.1** for the rejected data[12], it depicts a pattern consistent with the accepted application data set. More applications have been rejected over the years.

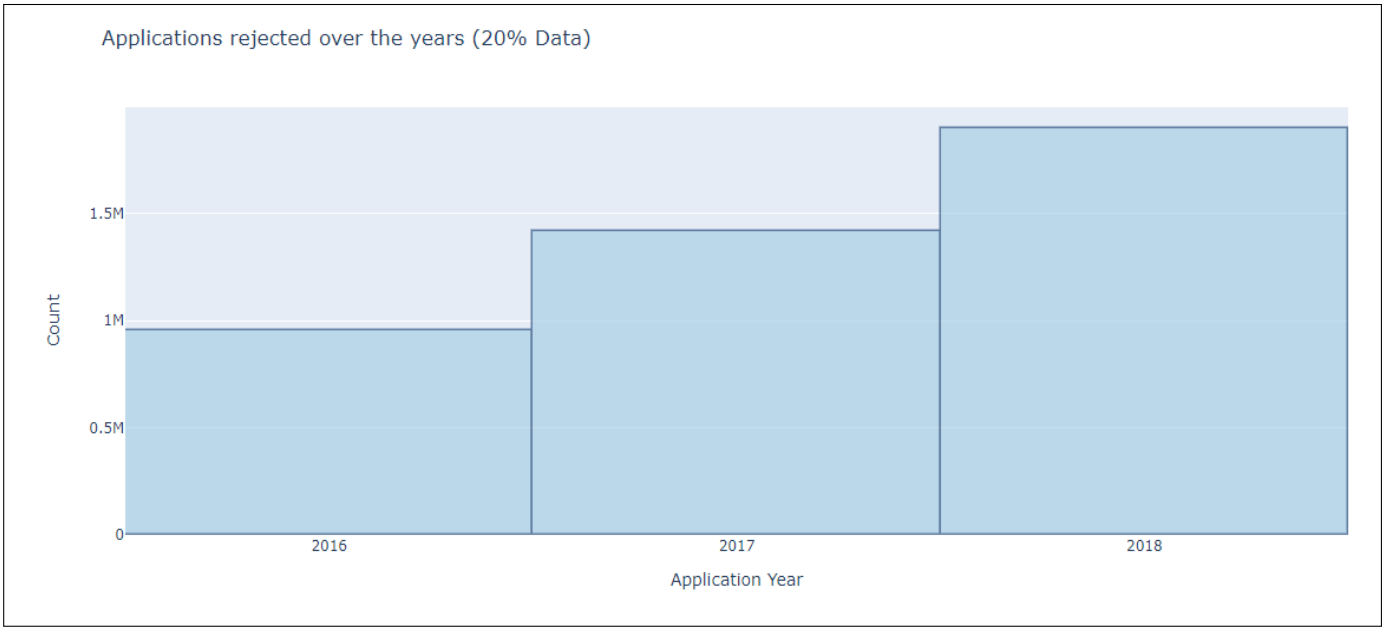


Figure 36: Rejected Applications over the years

[12]20% of the data is used due to infrastructure constraint. Data is shuffled and the composition is kept intact as in original data set.

The purpose of loan for rejected applications is shown in Figure 37. The maximum loans which have been rejected stated their purpose to be **other** followed by **debt-consolidation**. The ambiguity in stating the purpose as other is the main reason for the rejection. For debt-consolidation, the previous years' defaults with similar cases could be a reason.

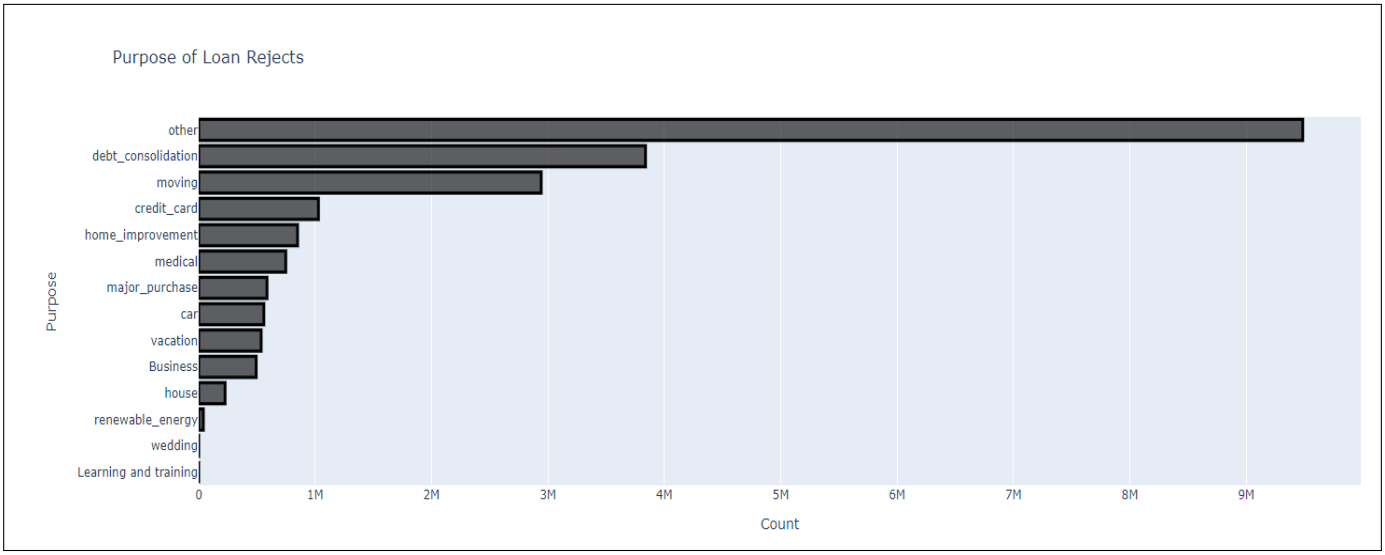


Figure 37: Purpose of loan for rejected applications

For **Question.11**, Figure 38 provides more clarity. It is evident that the number of rejected applications are also high in Florida. Therefore, the number can be attributed to the population and amount of applications.

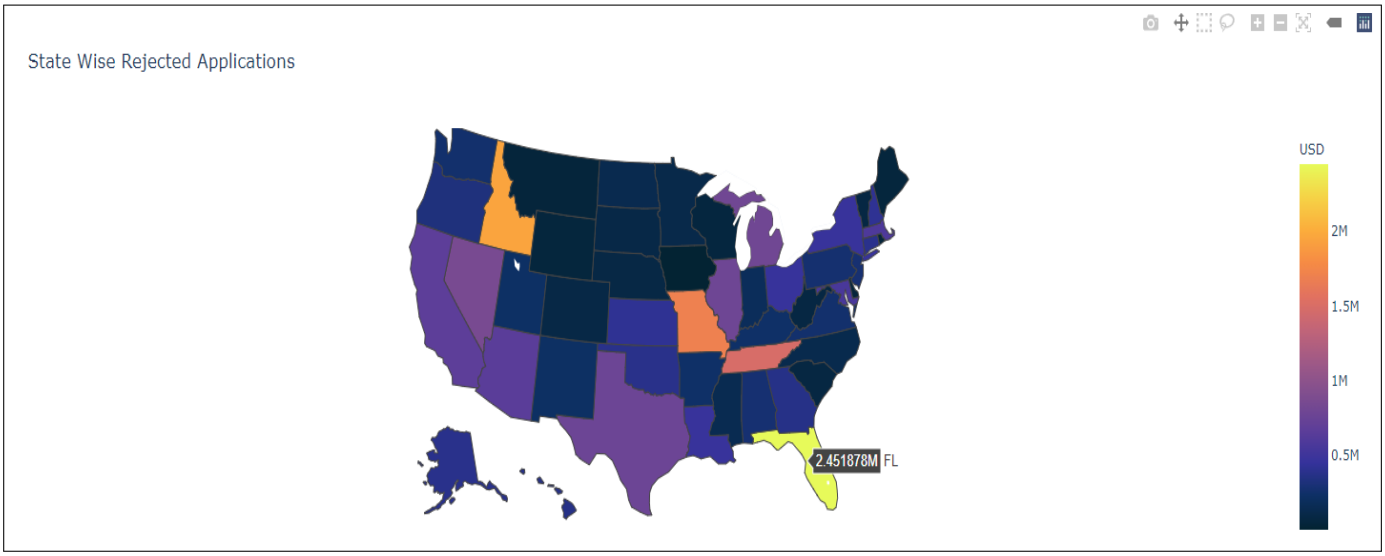


Figure 38: State-wise rejected applications

4.4 Algorithms

For model building there are many algorithms. To select the best out of the lot, let us implement a few. The selection of the algorithms are based on subject matter knowledge.
The machine learning algorithms which will be used are:

- Logistic Regression
- Support Vector Machine
- Random Forest
- Gradient Boosting Classifier

4.4.1 Logistic Regression

Logistic Regression [5] is a popular classification model used for binary classification. It can, however, be used for multi-class classification as well. In the CRM [13], the two classes are *0* and *1*. If a person is likely to default in payment of loan the algorithm will return *1* and if a person is not likely to default loan payment then the algorithm should return *0*. The logistic regression model generalises from the linear regression hypothesis.

$$h_\theta(x) = \theta^{\mathrm{T}} x$$

$$h_\theta(x) = g(\theta^{\mathrm{T}} x)$$

$$g(z) = \frac{1}{1 + e^{-\mathrm{z}}}$$

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

The $g(z)$ function is the **Logistic Function** or the **Sigmoid Function**. The Sigmoid Function[14] is asymptotic at *0* and *1* and cuts the y-axis at *0.5* .Therefore, it essentially returns values between *0* and *1*. If it returns value *less than 0.5* then the value *0* is assigned to it and if it returns *0.5 or more* then the value *1* is assigned to it.
Since CRM has 121 features, the logistic regression hypothesis will be:

$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + ... + \theta_{121} x_{121})$$

The logistic regression classifier will predict *Risk* if:

$$\theta_0 + \theta_1 x_1 + ... + \theta_{121} x_{121} \geq 0$$

because the *threshold* is set at $g(z) = 0.5$ A non-linear decision boundary can be visualised to understand the demarcations clearly.

[13] Credit Risk Modelling: Acquisition Model
[14] Figure 39

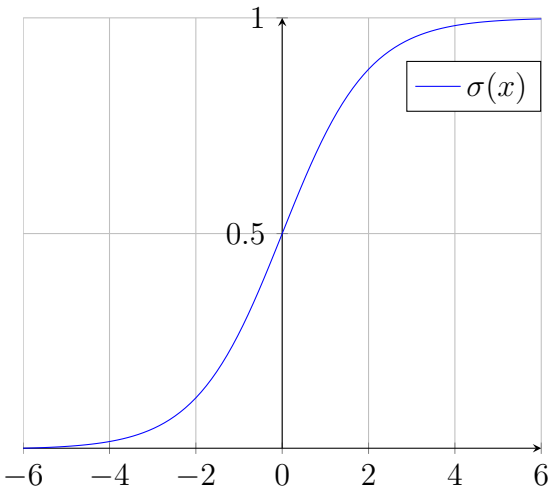


Figure 39: Sigmoid Function

The logistic regression cost function for the two points of interest:*1 (Risk)* and *0 (NoRisk)* are given by:

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n} Cost(h_\theta(x^{(i)}), y^{(i)})$$

$$Cost(h_\theta(x), y) = \begin{cases} -log(h_\theta(x)), & if\ \text{y} = 1\ (Risk) \\ -log(1 - h_\theta(x)), & if\ \text{y} = 0\ (NoRisk) \end{cases}$$

Simplified:

$$J(\theta) = -\frac{1}{n}[\sum_{i=1}^{n} y^{(i)} log(h_\theta(x^{(i)})) + (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}))]$$

To fit parameter θ:

$$\min_{\theta} J(\theta)$$

This function is a convex function and the above optimisation problem needs to be solved in order to find θ. One of the methods that can be used is **Gradient Descent**.

Algorithm 1 Gradient Descent

Let $\mathcal{J} := \{1, ..., m\}$.

while Some convergence criteria is not fulfilled ... **do**

for each $j \in \mathcal{J}$ **do**

$\theta_j \leftarrow \theta_j + \alpha \sum_{i=1}^{n} (y^{(i)} - h_\theta(x^{(i)}))\ x_j^{(i)}$

(Simultaneously update all θ_j)

end for

... update some convergence measure

end while

The applications which are labelled as *1* are rejected and the applications labelled as *0* are accepted.

4.4.2 Support Vector Machine

Support Vector Machines [11] are supervised machine learning models. They can be used for classification - **Support Vector Classification (SVC)** or regression - **Support Vector Regression (SVR)**. SVM[15] can be linear and nonlinear models. The model is said to be linear if the data domain can be demarcated with a straight line or hyperplane to separate the classes in the original domain. As, CRM[13] is a classification problem with two classes demarcated clearly, it falls under the category of SVC linear model.SVC essentially uses $y = wx' + \gamma$ and manoeuvres it to allow linear domain demarcation. The classification technique includes parameterisation and optimisation. In the CRM[13] data domain, the straight line or **hyperplane** divides the data into two sub domains - *0 (No Risk)* and *1 (Risk)*.

For ease of depicting let the domains be D_1 and D_2.

$$D_1 = x : wx' + \gamma \leq 0$$

$$D_2 = x : wx' + \gamma > 0$$

The SVC aims to find the best hyperplane[16] or decision boundary that best demarcates the two classes.As CRM[13] has more than 2 features therefore a hyperplane will be fit to demarcate the applications into the two classes. The above domains (D_1 and D_2) are hyperplanes separating the data points and the support vectors are the data points which are located nearest to these hyperplanes. These support vectors can alter the positions of these hyperplanes. The optimisation goal is to calculate the distance between the hyperplanes and search for parameter values that maximises this distance.

The distance is calculated using:

$$d = \frac{\pm 2}{\|w\|^2}$$

The error function is defined by:

$$e = 1 - y(wx' + \gamma)$$

and w and γ must be selected such that the error $e \leq 0$ (Refer Figure 40).

Combining the minimisation goals for two-class support vector machine and extending it to multidimensional data domain:

$$\underset{w,\gamma}{Minimise} \frac{\pm 2}{\|w\|^2}$$

subject to:

$$s(wx' + \gamma\, I) \geq I$$

The applications which are labelled as *1* are rejected and the applications labelled as *0* are accepted.

[15]Support Vector Machine

[16]It is a (N-1)-dimensional subspace for an N-dimensional space.

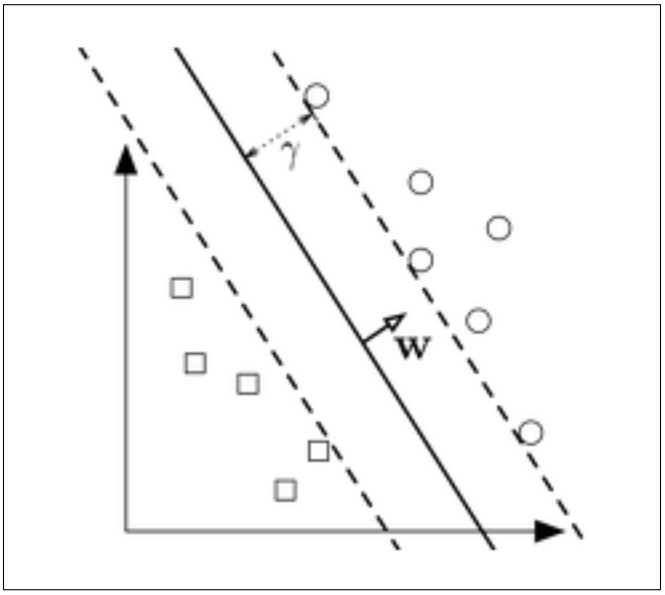


Figure 40: Support Vector Machine - 2 dimension sample illustration

4.4.3 Random Forest

The Random Forest [**?**] is a supervised learning algorithm that uses the concepts of ensemble learning, bagging and bootstrap. Ensemble learning deals with combining more than one algorithm of same or different kinds for classification task. Random Forests are structures which are like forest formed with decision trees that are produced using random sampling with replacement. The fact that random forest provides multiple trained decision tree classifiers makes it a preferred choice over regular decision tree classifier. Figure 41 illustrates the steps and the parallelization feature of random forest.

An important role in classification and optimisation is played by statistical techniques of bootstrapping and bagging. **Bootstrapping** is a randomisation technique which generates many subgroups from a under observation data by randomly selecting the same number of observations as the original data set, but with the replacement. This technique is applied in the training phase. The selection of parameters for the bootstrapping stage is crucial. A typical number of bootstrapping samples used is 10. However, it is best to apply cross-validations and get suitable ranges of values.Several papers have used a random selection of $\sqrt{p}$ number of features for the nodes of the trees where p are the features in the data domain.

Bagging is applied in the testing phase. The term bagging comes from bootstrapping aggregation. It averages the predictions (or classification) the bootstrap samples give to get the final prediction (or classification) result. The selection can be derived as followed: Let there be q features in the data domain, and m of them are good ones, but they are unknown. Fraction of good ones:

$$\frac{m}{q}$$

Assume, 1 best among good ones. Fraction among good ones:

$$\frac{1}{m}$$

Hence:

$$\frac{1}{m} \geq \frac{m}{q}$$
$$m^2 \leq q$$
$$m \leq \sqrt{q}$$

In random forest, parameterization and optimisation steps are performed at the same time. The main optimisation measures are **entropy** and **information gain**.

$$Entropy(x) = -\sum(P(x = k) * log_2(P(x = k))$$

where, $P(x = k)$ is the probability that a feature x takes value k

$$Information\,Gain(feature) = Entropy(dataset) - Entropy(feature)$$

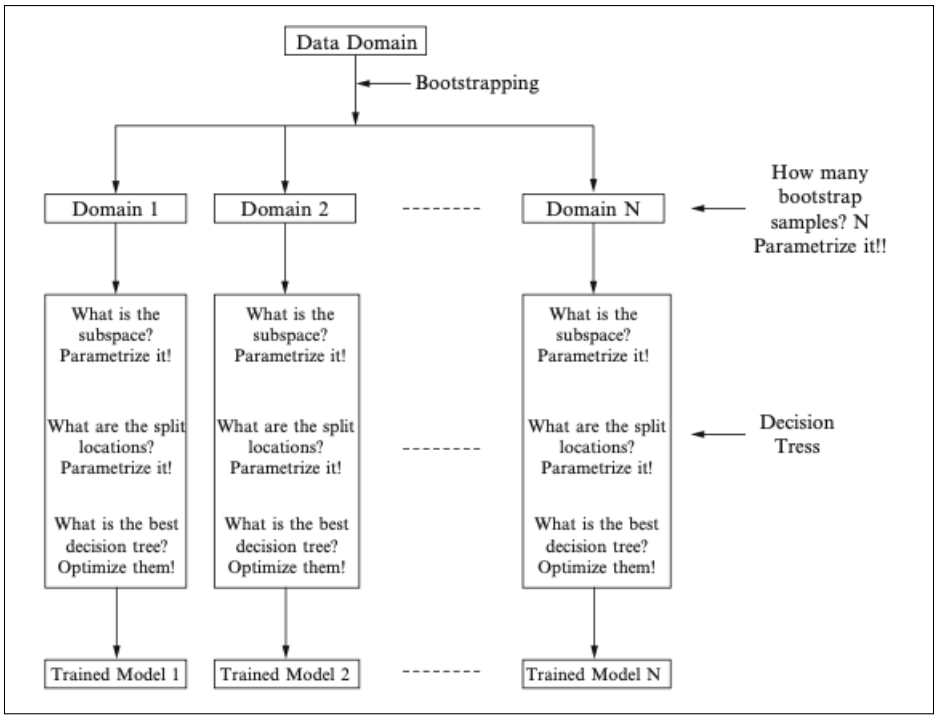


Figure 41: Demonstration of Random Forest Technique

The applications which are labelled as *1* are rejected and the applications labelled as *0* are accepted.

4.4.4 Gradient Boosting Classifier

Gradient Boosting [3], a machine learning technique for regression and classification problems uses ensemble learning. Similar to random forest, it produces more than one decision tree and then combines their output. GBM [17] uses the concepts of **boosting** and **gradient descent** to ultimately minimise the loss. The model starts with a sample tree and the subsequent trees are added one at a time with the modifications based on the loss of the previous tree. The trees learn from the "mistakes" of the previous trees. Hence, the observations have an unequal probability of appearing in ensuing trees and ones with the highest error appear most. This concept is called boosting.

Unlike bagging, boosting is a technique where the predictors are make sequentially not independently. GBM trees are shallower than random forest trees and they take less iterations to reach the result.

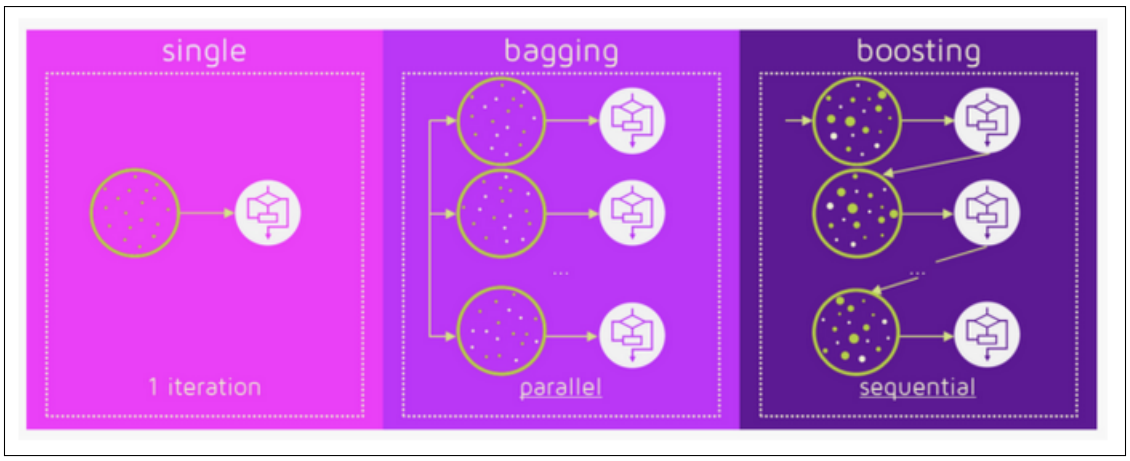


Figure 42: Bagging Vs Boosting

The weakness of the previous models are defined by gradient descent(1). To improve a model F, the loss function $Loss(Y, F(x))$ is minimised. Loss Function used in GBM classifier is the **Logistic Loss/Binomial Deviance**.

$$Loss\ Function\ L = Binomial\ Deviance = log(1 + e^{-y_i f(x_i)})$$

where

$$y \in \{-1, 1\}$$

Minimisation is performed by fitting estimator H on $(X_i, \frac{\delta L}{\delta X_i})\forall i$. $H(X) + F(X)$ is an approximation of gradient descent $\hat{F}(X_i) = F(X_i) - \frac{\delta L}{\delta F(X_i)}$

The applications which are labelled as *1* are rejected and the applications labelled as *0* are accepted.

[17] Gradient Boosting Machine

5 Experiment Setup and Results

All the algorithms are implemented using the Scikit-Learn toolkit [8] along with hyperparameter tuning and custom functions which cannot be disclosed due to confidentiality agreements. All the algorithms will be compared on the basis of accuracy (before any tuning) and recall. The base model is **Logistic Regression**[18]. The results are as follows:

```
Accuracy of logistic regression classifier on test set:  0.9944391001299512
```

Figure 43: Accuracy - LogReg

```
0.9951901169561035
```

Figure 44: Accuracy after SMOTE - LogReg

```
[[97994   465]
 [  194 19853]]
```

Figure 45: Confusion Matrix - LogReg

	precision	recall	f1-score	support
0	1.00	1.00	1.00	98459
1	0.98	0.99	0.98	20047
accuracy			0.99	118506
macro avg	0.99	0.99	0.99	118506
weighted avg	0.99	0.99	0.99	118506

Figure 46: Classification Report - LogReg

[18]https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression

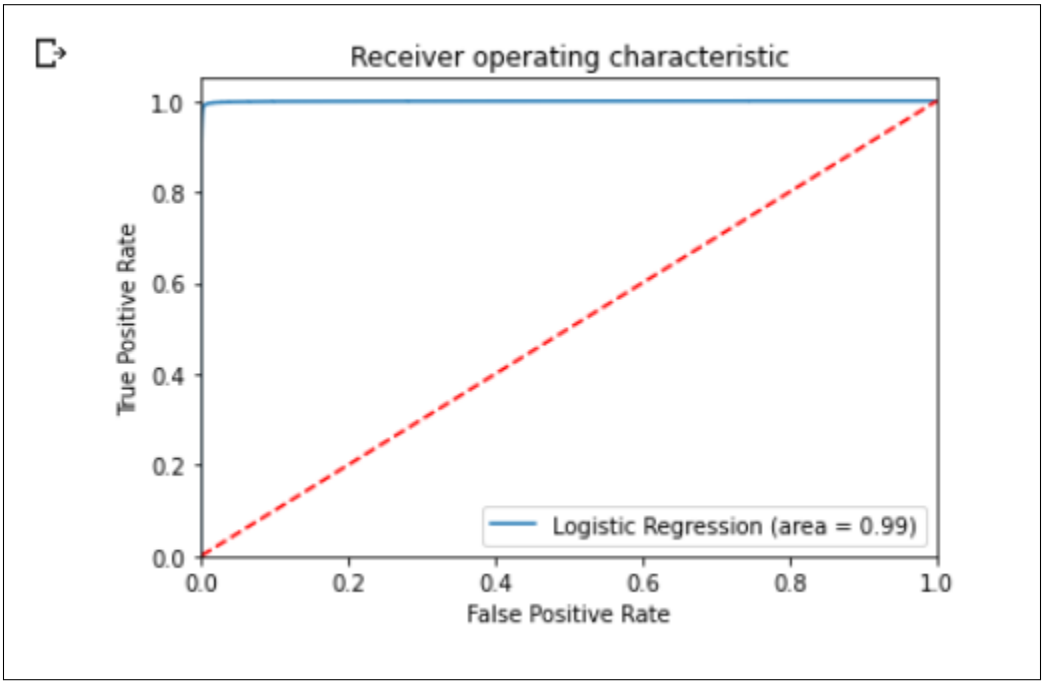


Figure 47: ROC-AUC - LogReg

```
Best Score: 0.9883827787741405
Best Hyperparameters: {'C': 10, 'penalty': 'l2', 'solver': 'lbfgs'}
```

Figure 48: After Hyperparameter Tuning - LogReg

The logistic regression has an accuracy of **99.4%** with a 98.8% after hyperparameter tuning. It's recall is at **99%** with the best hyperparameters depicted in Figure 48.

Next, is the **Support Vector Machine**[19]. The results are as follows:

```
Accuracy of SVM classifier on test set: 0.9949454036082561
```

Figure 49: Accuracy - SVM

[19]https://scikit-learn.org/stable/modules/svm.html

```
[[98224   235]
 [  364 19683]]
```

Figure 50: Confusion Matrix - SVM

	precision	recall	f1-score	support
0	1.00	1.00	1.00	98459
1	0.99	0.98	0.99	20047
accuracy			0.99	118506
macro avg	0.99	0.99	0.99	118506
weighted avg	0.99	0.99	0.99	118506

Figure 51: Classification Report - SVM

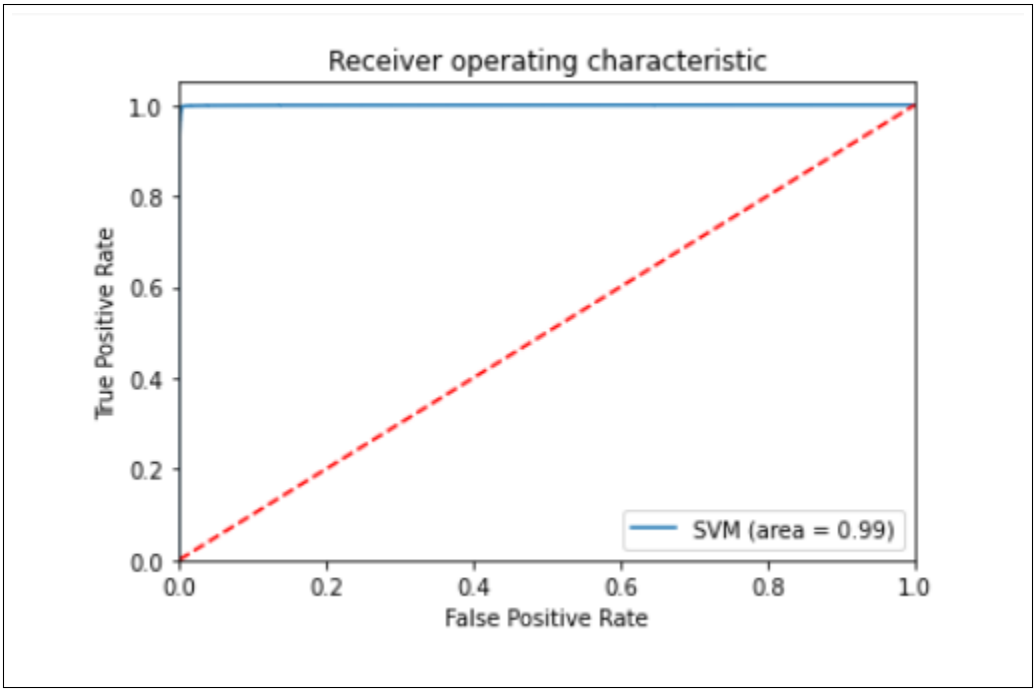


Figure 52: ROC-AUC - SVM

The Support Vector Machine has an accuracy of **99.4%** which is same as logistic regression but a recall of **98%** which is less than logistic regression.

The third algorithm is **Random Forest**[20]. Here are the results:

```
The accuracy of the Random Forests model is :    0.9950129107386968
```

Figure 53: Accuracy - RF

```
[[97994   465]
 [  194 19853]]
```

Figure 54: Confusion Matrix - RF

	precision	recall	f1-score	support
0	1.00	1.00	1.00	98459
1	0.99	0.98	0.99	20047
accuracy			1.00	118506
macro avg	0.99	0.99	0.99	118506
weighted avg	1.00	1.00	1.00	118506

Figure 55: Classification Report - RF

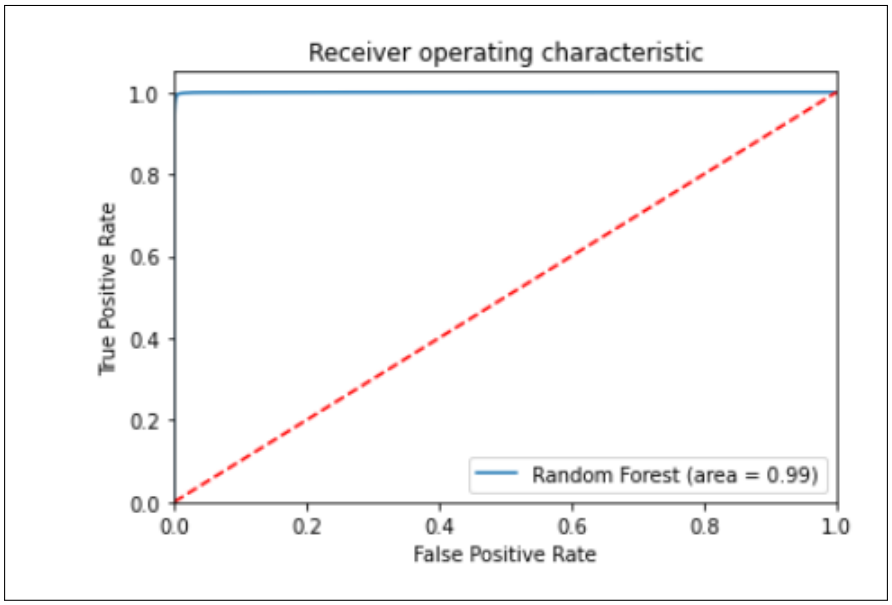


Figure 56: ROC-AUC - RF

[20]https://scikit-learn.org/stable/modules/ensemble.html#forests-of-randomized-trees

Feature Importance

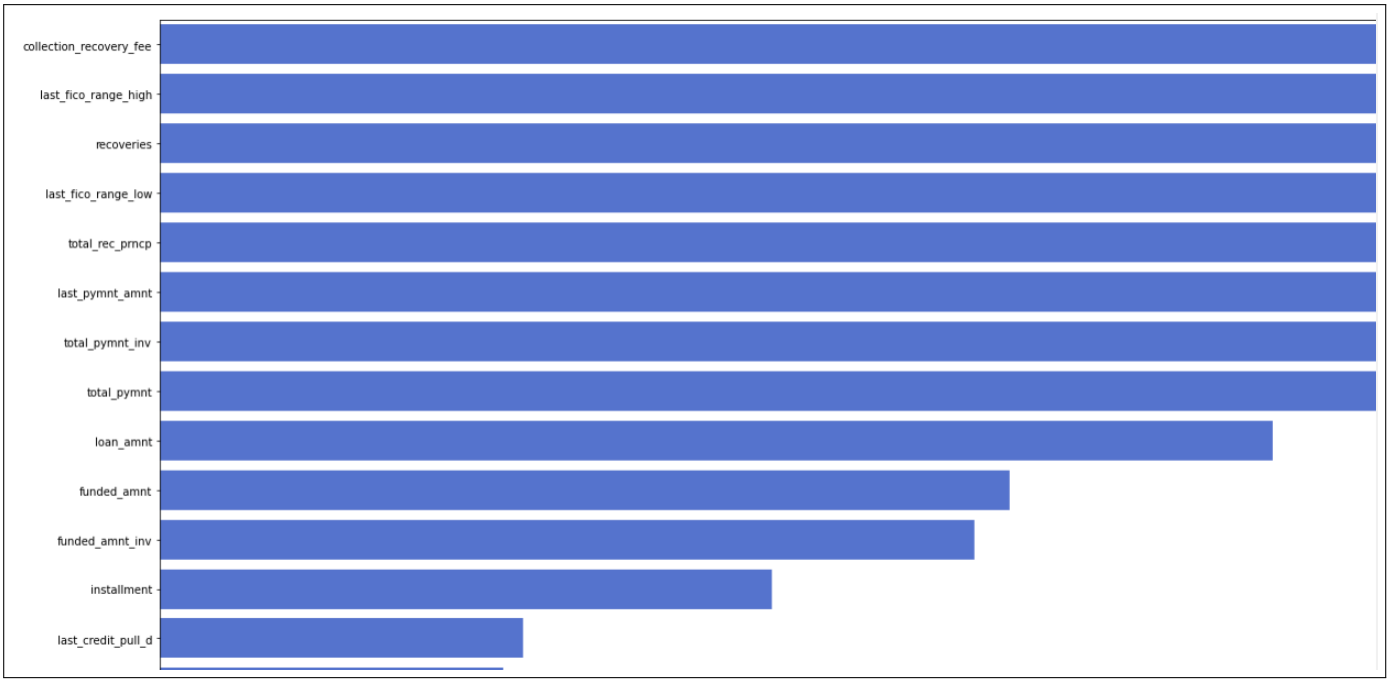


Figure 57: Feature Importance - RF

Hyperparamter Tuning

```
{'bootstrap': [True, False],
 'max_depth': [10, 55, 100, None],
 'max_features': ['auto', 'sqrt'],
 'min_samples_leaf': [1, 2, 4],
 'min_samples_split': [2, 5, 10],
 'n_estimators': [200, 600, 1000]}
```

Figure 58: Custom Random Grid

```
Fitting 3 folds for each of 3 candidates, totalling 9 fits
[Parallel(n_jobs=3)]: Using backend LokyBackend with 3 concurrent workers.
[Parallel(n_jobs=3)]: Done   9 out of   9 | elapsed: 254.1min remaining:    0.0s
[Parallel(n_jobs=3)]: Done   9 out of   9 | elapsed: 254.1min finished
```

Figure 59: Cross Validation

```
{'bootstrap': False,
 'max_depth': 100,
 'max_features': 'auto',
 'min_samples_leaf': 4,
 'min_samples_split': 10,
 'n_estimators': 200}
```

Figure 60: Best Parameters - RF

```
Best Score: -0.008966444917053675
```

Figure 61: Best Score: Negative MAE - RF

The random forest has an accuracy of **99.5%** which is better than both logistic regression and support vector machine and a recall of **98%**.

The final model is **Gradient Boosting Classifier**[21]. The results are as follows:

```
Accuracy of Gadient Booster classifier on test set:  0.9936374529559685
```

Figure 62: Accuracy - GBC

	precision	recall	f1-score	support
0	0.99	1.00	1.00	98459
1	0.99	0.97	0.98	20047
accuracy			0.99	118506
macro avg	0.99	0.98	0.99	118506
weighted avg	0.99	0.99	0.99	118506

Figure 63: Classification Report - GBC

[21] https://scikit-learn.org/stable/modules/ensemble.html#gradient-tree-boosting

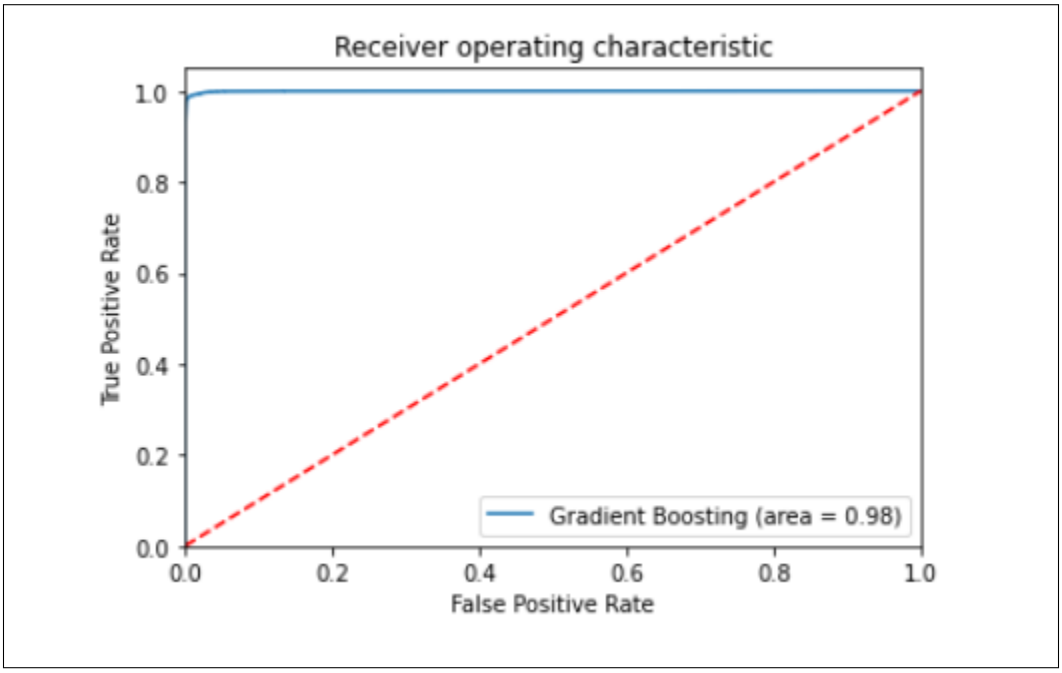


Figure 64: ROC-AUC - GBC

The Gradient Boosting classifier has an accuracy of **99.3%** and a recall of **97%** which is less than all the other models.

In conclusion, the model that is chosen for further analysis is **Random Forest**. This model will be used to get a better insight into the results using **Explainable Artificial Intelligence**.

A few things to note here:

- All the models were trained on the same composition of data and class imbalance was taken care.
- The data used for training is from the original accepted applications data set.
- The training set was taken as 70% of the data and 30% of the data was treated as the testing set.
- The techniques of Grid Search and Random Search Cross Validations were used for hyper-parameter tuning.
- The feature importance displayed for Random Forest was based on build-in applications in scikit-learn.
- All the manipulations in application are based on domain knowledge and infrastructure constraints.

6 Explainable Artificial Intelligence (XAI)

Machine Learning models which are used in this project are not very opaque nor can they be understood by the user at first glance. By looking at the results, the user can have many questions.

- Why was this particular model selected ?
- Why wasn't a different approach taken?
- How did the model reject a particular application?
- What features contributed to this rejection?

These models essentially become a black box for the end user. This in turn puts a question on the model and the developer. This is where the concept of **Explainable Artificial Intelligence (XAI)** [7] comes into existence.

XAI is essential for the end user to understand, appropriately trust, and effectively manage machine partners. It creates a suite of machine learning techniques that:

- Create more explainable models with a high level of learning performance
- Enable end users to understand the final results and trust the model and output.

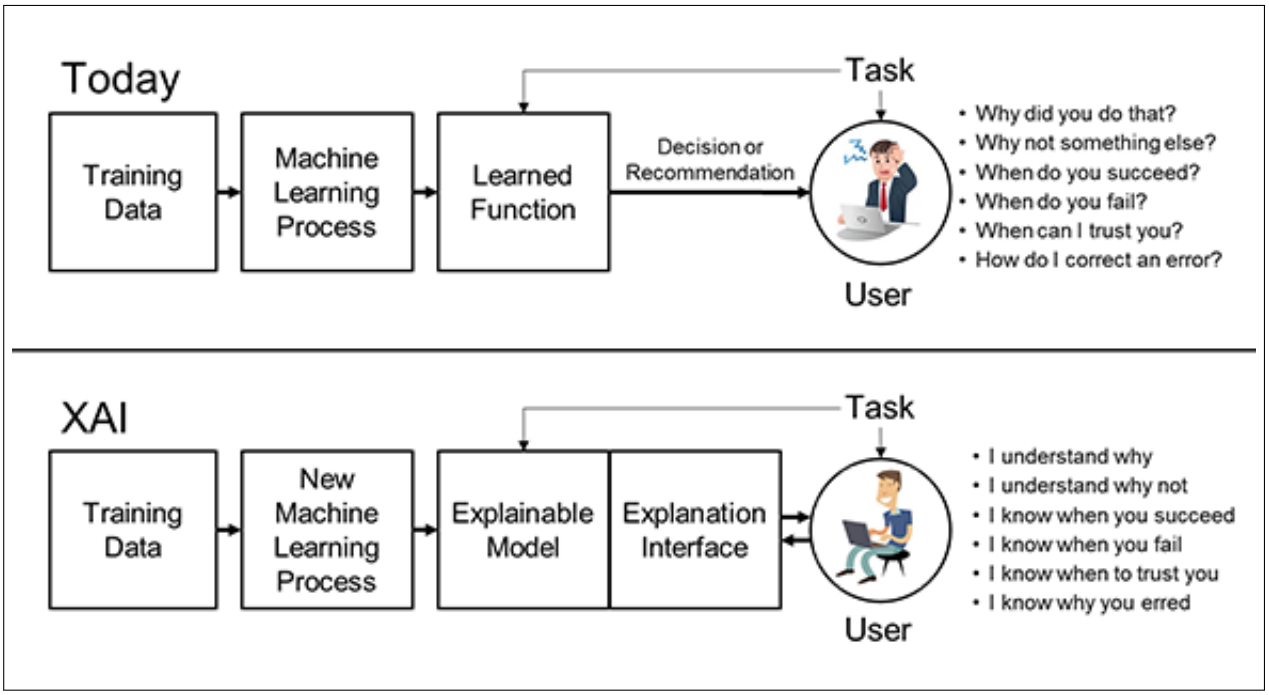


Figure 65: Explainable Artificial Intelligence Concept

In CRM[13], these concepts are used to explain the features that contributed to the **acceptance** or **rejection** of the applications.

6.1 Feature Importance

The first explanation can be obtained using the built-in method in the Scikit-Learn [8] toolkit. Top 20 features are displayed for each model.

For **Logistic Regression**, the coefficient values are used for determining the important features which contributed to the final labelling.

	feature	feature_importance
0	loan_amnt	46.245107
1	funded_amnt	46.245107
2	funded_amnt_inv	45.612036
3	collection_recovery_fee	33.550955
4	recoveries	33.341203
5	total_rec_int	17.377571
6	installment	7.885027
7	last_pymnt_d	3.547725
8	sub_grade	2.945394
9	total_rec_late_fee	2.232988
10	total_bc_limit	1.803436
11	num_tl_op_past_12m	1.685348
12	total_bal_ex_mort	1.127820
13	revol_bal	1.113537
14	tot_hi_cred_lim	1.002995
15	bc_util	0.787912
16	il_util	0.718033
17	issue_yr	0.613442
18	mo_sin_old_rev_tl_op	0.592534
19	num_sats	0.586269

Figure 66: Feature Importance - Logistic Regression

The features *loan_amnt* - Loan Amount , *funded_amnt* - Funded Amount and *funded_amnt_inv* - Funded Amount by Investors are the top 3 contributors. Therefore, in future the borrower as well as the lender can pay more attention to these features.

For **Support Vector Machine**, the coefficient values are used for determining the important features which contributed to the final labelling.

	feature	feature_importance
0	loan_amnt	12.315249
1	funded_amnt	12.315249
2	funded_amnt_inv	12.165394
3	recoveries	10.024786
4	collection_recovery_fee	10.001478
5	total_rec_int	5.560786
6	installment	4.951436
7	sub_grade	0.524543
8	total_bal_ex_mort	0.414914
9	issue_d	0.352105
10	total_bc_limit	0.343736
11	open_acc	0.271905
12	total_rec_late_fee	0.183565
13	num_rev_accts	0.182097
14	60	0.172181
15	il_util	0.139829
16	tot_hi_cred_lim	0.109655
17	bc_util	0.104325
18	num_il_tl	0.096538
19	dti	0.082546

Figure 67: Feature Importance - Support Vector Machine

The features *loan_amnt* - Loan Amount , *funded_amnt* - Funded Amount and *funded_amnt_inv* - Funded Amount by Investors are the top 3 contributors. These results coincide with the results of Logistic Regression. The difference is visible in the 4^{th} feature.

For **Random Forest Classifier**, the information gain (entropy) is used for determining the important features which contributed to the final labelling.

	feature	feature_importance
0	recoveries	0.162925
1	collection_recovery_fee	0.132542
2	last_fico_range_low	0.113391
3	total_rec_prncp	0.110827
4	last_fico_range_high	0.101593
5	last_pymnt_amnt	0.062911
6	total_pymnt	0.052217
7	total_pymnt_inv	0.047781
8	loan_amnt	0.026855
9	funded_amnt_inv	0.024687
10	funded_amnt	0.024445
11	installment	0.022321
12	total_rec_late_fee	0.010847
13	total_rec_int	0.009672
14	last_credit_pull_d	0.007111
15	last_pymnt_d	0.006785
16	issue_d	0.006160
17	out_prncp_inv	0.005877
18	36	0.005151
19	out_prncp	0.005090

Figure 68: Feature Importance - Random Forest Classifier

The features *recoveries* - Number of recoveries in the past, *collection_recovery_fee* - Amount spent on recovery and *last_fico_range_low* - Last lower FICO range are the top 3 contributors. These results differ from the results of Logistic Regression and Support Vector Machine. The features highlighted here are important from the lender's perspective.

For **Gradient Boosting Classifier**, the information gain (entropy) is used for determining the important features which contributed to the final labelling.

	feature	feature_importance
0	recoveries	0.581102
1	last_fico_range_low	0.101985
2	last_fico_range_high	0.084538
3	total_rec_prncp	0.075457
4	last_pymnt_amnt	0.067351
5	loan_amnt	0.042281
6	funded_amnt	0.024770
7	last_pymnt_d	0.004554
8	funded_amnt_inv	0.004498
9	out_prncp_inv	0.004170
10	out_prncp	0.003058
11	issue_d	0.002792
12	last_credit_pull_d	0.001058
13	collection_recovery_fee	0.000822
14	installment	0.000392
15	next_pymnt_d	0.000195
16	total_rec_late_fee	0.000191
17	total_pymnt	0.000172
18	total_pymnt_inv	0.000088
19	fico_range_high	0.000072

Figure 69: Feature Importance - Gradient Boosting Classifier

The features *recoveries* - Number of recoveries in the past , *last_fico_range_high* - Last higher FICO range and *last_fico_range_low* - Last lower FICO range are the top 3 contributors. These results differ from the results of Logistic Regression and Support Vector Machine but are similar to Random Forest. The features highlighted here are important from the lender's perspective.
Note: The base model for SHAP and LIME is used as Random Forest.

6.2 SHAP

SHAP [6], which stands for *SHapley Additive exPlanations* uses game theory concept and approach to explain the output of machine learning models. It can provide both local and global explanations.

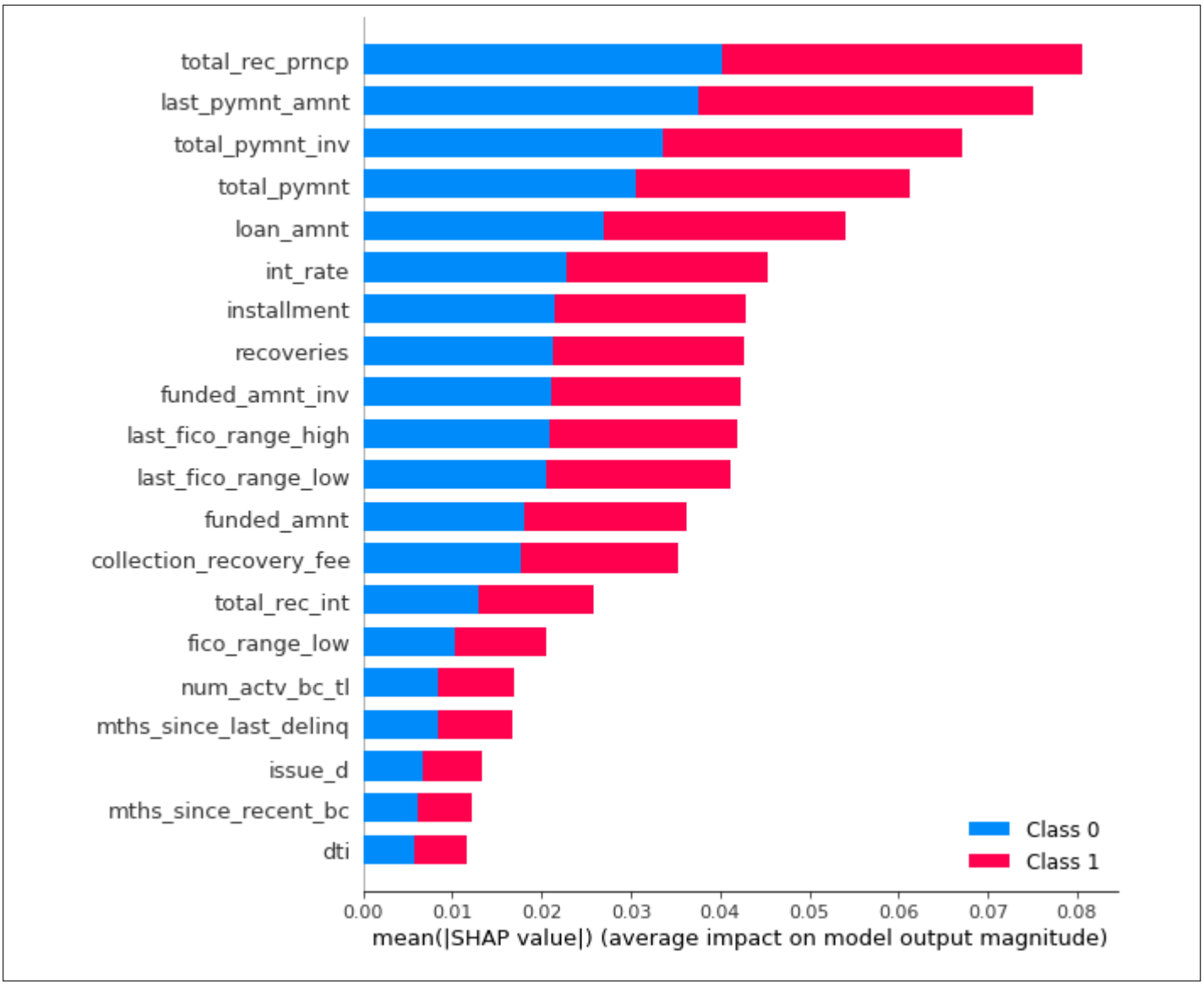


Figure 70: Global Explanation - SHAP

The explanation in Figure 70 is global and it also provides the extent to which each feature contributes to either classes.For local explanation, there are two sample outputs, one for each class. The biggest contributor to the acceptance of the particular customer is the feature *total_rec_prncp* - Total received principle amount. In this particular case there is no clear margin for any feature.

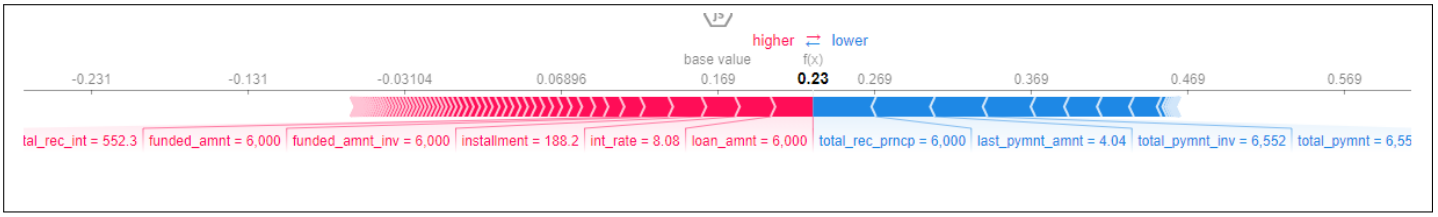


Figure 71: Local Explanation : Accepted Customer - SHAP

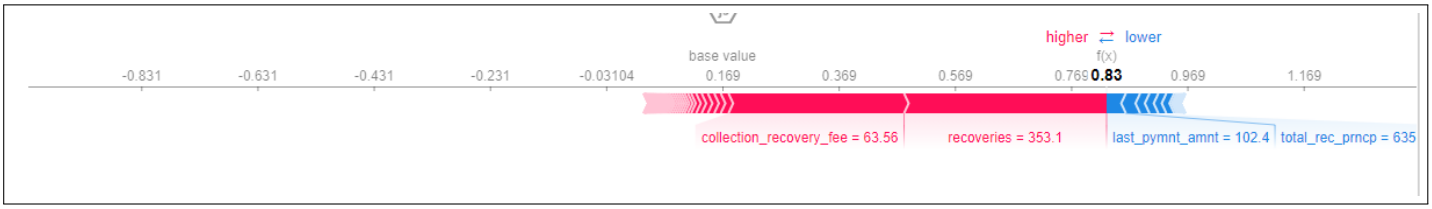


Figure 72: Local Explanation : Rejected Customer - SHAP

Unlike the acceptance case, the rejection for this customer was heavily influenced by a particular feature. The features that lead to the decline in loan application are *recoveries* - Number of recoveries in the past, *collection_recovery_fee* - Amount spent on recovery which are in line with the built-in scikit-learn feature importance.

6.3 LIME

LIME [9] stands for *Local Interpretable Model-Agnostic Explanations*. As the name suggests, this technique is used for getting local explanations. Here are two sample outputs, one for each class. Figure 73 provides a case where the application was accepted. It also depicts the features which contributed to this decision in blue and the features which had a negative impact in orange. Finally, it provides the prediction probabilities as well. Similarly, Figure 74 provides a case where the application was rejected. It also depicts the features which contributed to this decision in orange and the features which had a positive impact in blue. Finally, it provides the prediction probabilities as well.

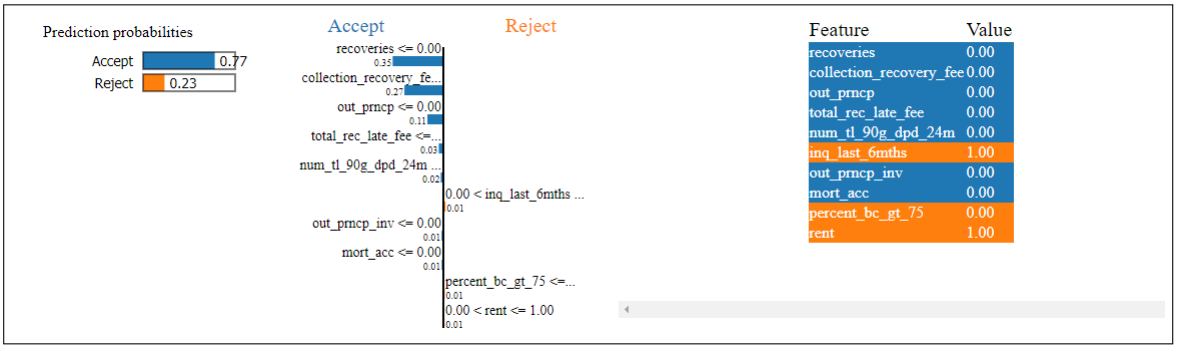


Figure 73: Local Explanation : Accepted Customer - LIME

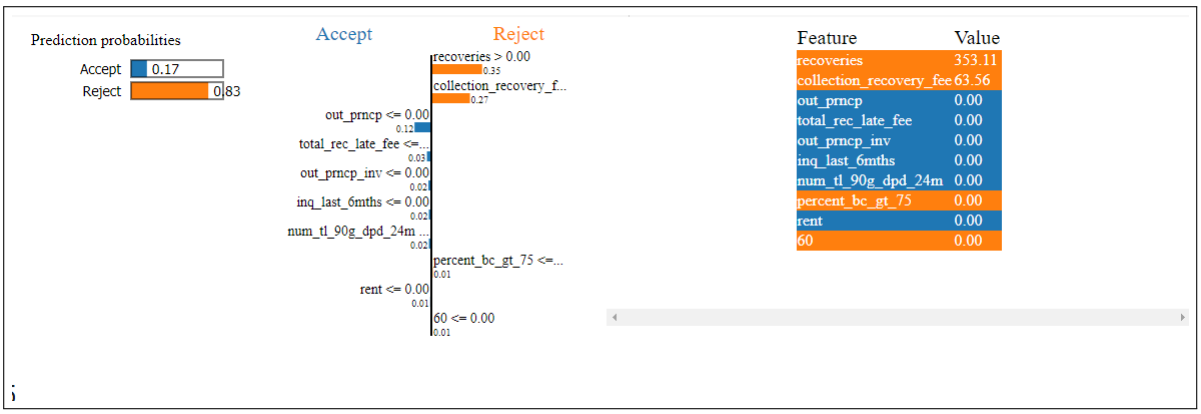


Figure 74: Local Explanation : Rejected Customer - LIME

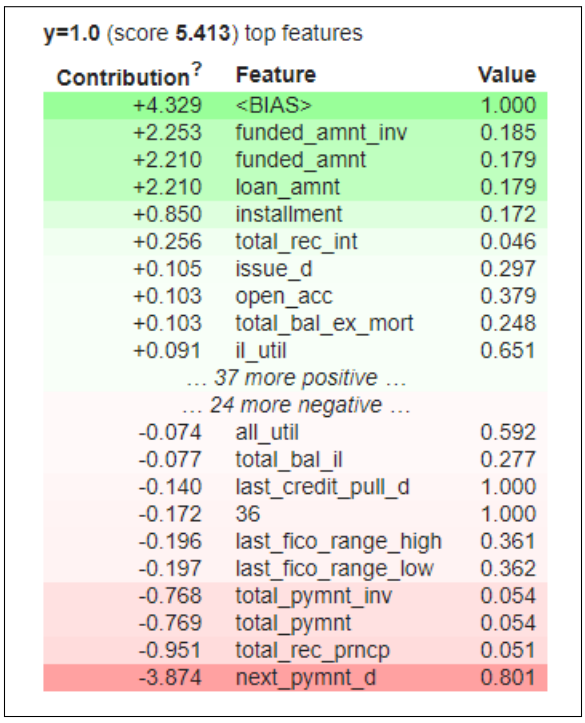
6.4 ELI5

ELI5 [4] stands for *Explain Like I'm 5*. It helps demystify the machine learning models and can provide both global and local explanations. The local explanations are provided using *explain_prediction* and the global explanation is provided using *explain_weights*. Local Explanations will be done on the sample in index 1.

y=1.0 (probability **1.000**, score **19.399**) top features

Contribution?	Feature	Value
+15.273	<BIAS>	1.000
+8.449	funded_amnt_inv	0.185
+8.300	funded_amnt	0.179
+8.300	loan_amnt	0.179
+1.639	last_pymnt_d	0.462
+1.354	installment	0.172
+0.801	total_rec_int	0.046
+0.481	mo_sin_old_rev_tl_op	0.812
+0.468	il_util	0.651
+0.440	credit_card	1.000
+0.421	num_tl_op_past_12m	0.250
	… 33 more positive …	
	… 28 more negative …	
-0.412	total_rev_hi_lim	0.142
-0.513	num_bc_sats	0.357
-1.078	last_credit_pull_d	1.000
-1.361	last_fico_range_high	0.361
-1.369	last_fico_range_low	0.362
-2.679	total_pymnt_inv	0.054
-2.683	total_pymnt	0.054
-3.289	total_rec_prncp	0.051
-13.704	next_pymnt_d	0.801

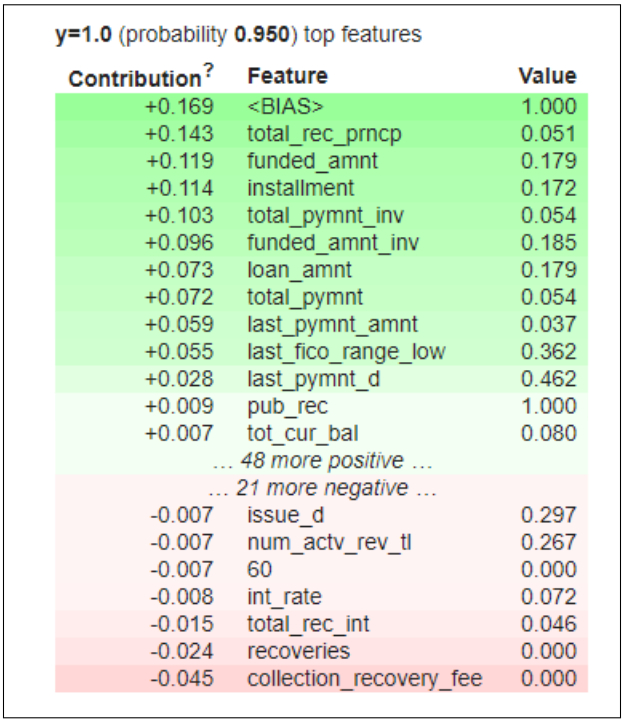
Figure 75: Local Explanation : Logistic Regression - ELI5



y=1.0 (score **5.413**) top features

Contribution[?]	Feature	Value
+4.329	<BIAS>	1.000
+2.253	funded_amnt_inv	0.185
+2.210	funded_amnt	0.179
+2.210	loan_amnt	0.179
+0.850	installment	0.172
+0.256	total_rec_int	0.046
+0.105	issue_d	0.297
+0.103	open_acc	0.379
+0.103	total_bal_ex_mort	0.248
+0.091	il_util	0.651
	... 37 more positive ...	
	... 24 more negative ...	
-0.074	all_util	0.592
-0.077	total_bal_il	0.277
-0.140	last_credit_pull_d	1.000
-0.172	36	1.000
-0.196	last_fico_range_high	0.361
-0.197	last_fico_range_low	0.362
-0.768	total_pymnt_inv	0.054
-0.769	total_pymnt	0.054
-0.951	total_rec_prncp	0.051
-3.874	next_pymnt_d	0.801

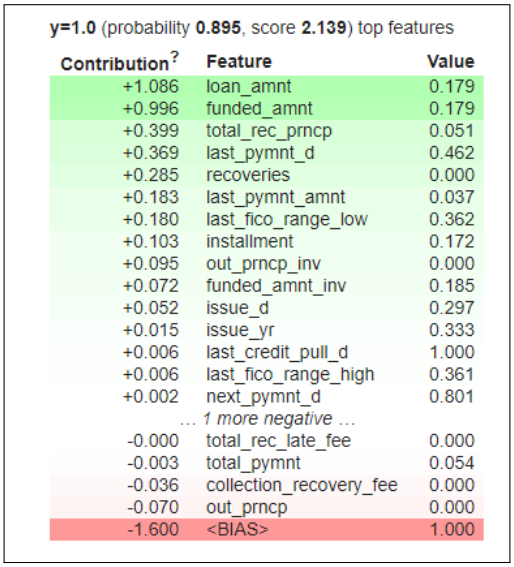
Figure 76: Local Explanation : Support Vector Machine - ELI5



y=1.0 (probability **0.950**) top features

Contribution[?]	Feature	Value
+0.169	<BIAS>	1.000
+0.143	total_rec_prncp	0.051
+0.119	funded_amnt	0.179
+0.114	installment	0.172
+0.103	total_pymnt_inv	0.054
+0.096	funded_amnt_inv	0.185
+0.073	loan_amnt	0.179
+0.072	total_pymnt	0.054
+0.059	last_pymnt_amnt	0.037
+0.055	last_fico_range_low	0.362
+0.028	last_pymnt_d	0.462
+0.009	pub_rec	1.000
+0.007	tot_cur_bal	0.080
	... 48 more positive ...	
	... 21 more negative ...	
-0.007	issue_d	0.297
-0.007	num_actv_rev_tl	0.267
-0.007	60	0.000
-0.008	int_rate	0.072
-0.015	total_rec_int	0.046
-0.024	recoveries	0.000
-0.045	collection_recovery_fee	0.000

Figure 77: Local Explanation : Random Forest - ELI5

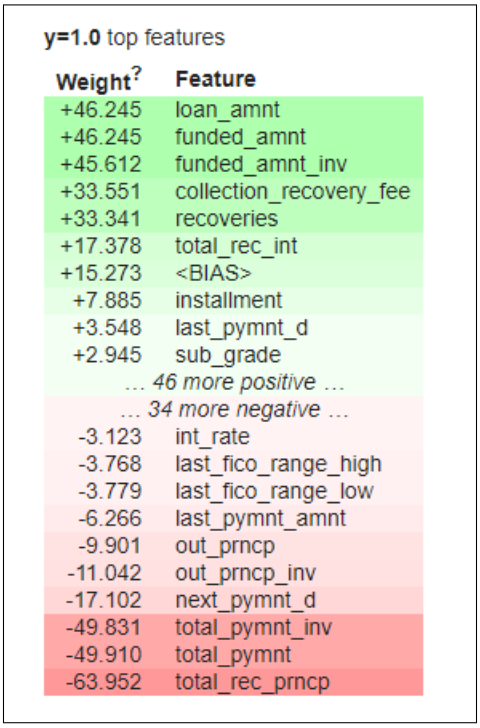


y=1.0 (probability **0.895**, score **2.139**) top features

Contribution[?]	Feature	Value
+1.086	loan_amnt	0.179
+0.996	funded_amnt	0.179
+0.399	total_rec_prncp	0.051
+0.369	last_pymnt_d	0.462
+0.285	recoveries	0.000
+0.183	last_pymnt_amnt	0.037
+0.180	last_fico_range_low	0.362
+0.103	installment	0.172
+0.095	out_prncp_inv	0.000
+0.072	funded_amnt_inv	0.185
+0.052	issue_d	0.297
+0.015	issue_yr	0.333
+0.006	last_credit_pull_d	1.000
+0.006	last_fico_range_high	0.361
+0.002	next_pymnt_d	0.801
	... 1 more negative ...	
-0.000	total_rec_late_fee	0.000
-0.003	total_pymnt	0.054
-0.036	collection_recovery_fee	0.000
-0.070	out_prncp	0.000
-1.600	<BIAS>	1.000

Figure 78: Local Explanation : Gradient Boosting Classifier - ELI5

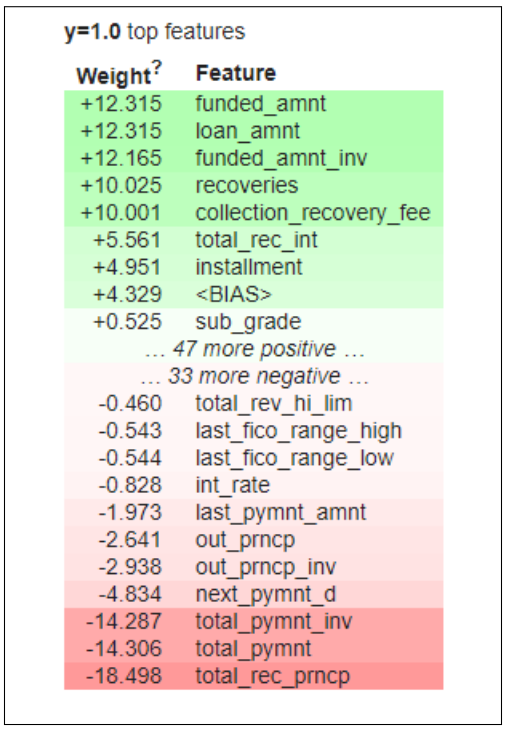
In all of the local explanations there is a clear demarcation of which features contributed the most and the features that had the least contribution for each of the models. Global explanation provides the weights for the entire model.For tree based models, Eli5 provides the ranking based on information gain (entropy).



y=1.0 top features

Weight[?]	Feature
+46.245	loan_amnt
+46.245	funded_amnt
+45.612	funded_amnt_inv
+33.551	collection_recovery_fee
+33.341	recoveries
+17.378	total_rec_int
+15.273	<BIAS>
+7.885	installment
+3.548	last_pymnt_d
+2.945	sub_grade
	... 46 more positive ...
	... 34 more negative ...
-3.123	int_rate
-3.768	last_fico_range_high
-3.779	last_fico_range_low
-6.266	last_pymnt_amnt
-9.901	out_prncp
-11.042	out_prncp_inv
-17.102	next_pymnt_d
-49.831	total_pymnt_inv
-49.910	total_pymnt
-63.952	total_rec_prncp

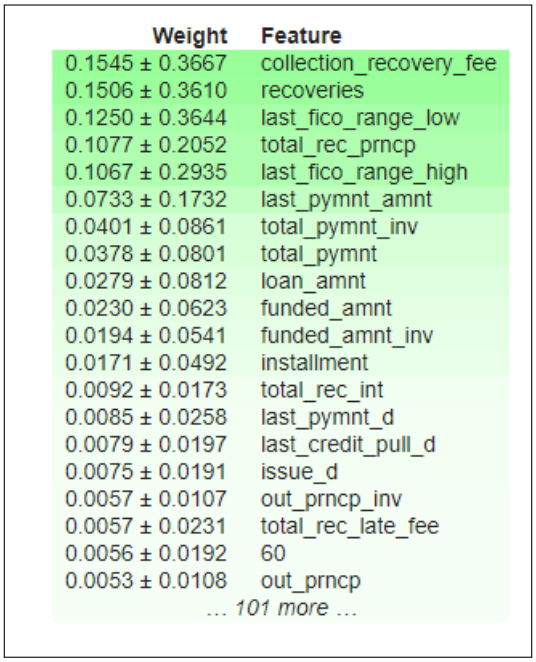
Figure 79: Global Explanation : Logistic Regression - ELI5



y=1.0 top features

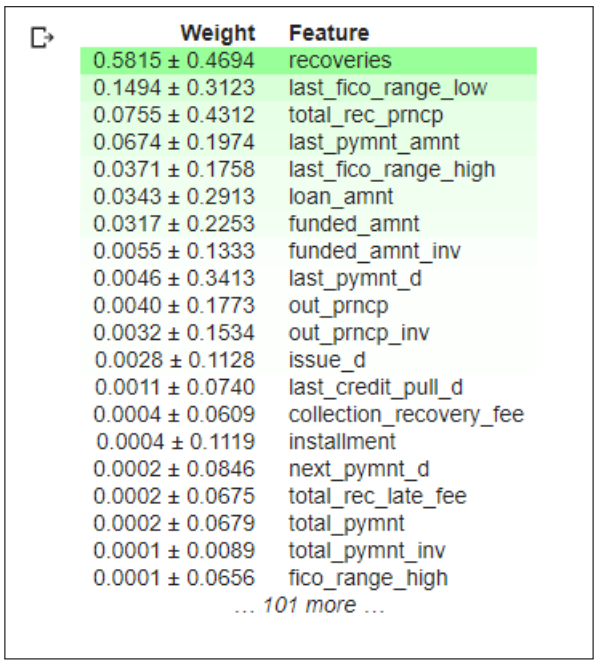
Weight[?]	Feature
+12.315	funded_amnt
+12.315	loan_amnt
+12.165	funded_amnt_inv
+10.025	recoveries
+10.001	collection_recovery_fee
+5.561	total_rec_int
+4.951	installment
+4.329	<BIAS>
+0.525	sub_grade
… 47 more positive …	
… 33 more negative …	
-0.460	total_rev_hi_lim
-0.543	last_fico_range_high
-0.544	last_fico_range_low
-0.828	int_rate
-1.973	last_pymnt_amnt
-2.641	out_prncp
-2.938	out_prncp_inv
-4.834	next_pymnt_d
-14.287	total_pymnt_inv
-14.306	total_pymnt
-18.498	total_rec_prncp

Figure 80: Global Explanation : Support Vector Machine - ELI5



Weight	Feature
0.1545 ± 0.3667	collection_recovery_fee
0.1506 ± 0.3610	recoveries
0.1250 ± 0.3644	last_fico_range_low
0.1077 ± 0.2052	total_rec_prncp
0.1067 ± 0.2935	last_fico_range_high
0.0733 ± 0.1732	last_pymnt_amnt
0.0401 ± 0.0861	total_pymnt_inv
0.0378 ± 0.0801	total_pymnt
0.0279 ± 0.0812	loan_amnt
0.0230 ± 0.0623	funded_amnt
0.0194 ± 0.0541	funded_amnt_inv
0.0171 ± 0.0492	installment
0.0092 ± 0.0173	total_rec_int
0.0085 ± 0.0258	last_pymnt_d
0.0079 ± 0.0197	last_credit_pull_d
0.0075 ± 0.0191	issue_d
0.0057 ± 0.0107	out_prncp_inv
0.0057 ± 0.0231	total_rec_late_fee
0.0056 ± 0.0192	60
0.0053 ± 0.0108	out_prncp
… 101 more …	

Figure 81: Global Explanation : Random Forest - ELI5



Weight	Feature
0.5815 ± 0.4694	recoveries
0.1494 ± 0.3123	last_fico_range_low
0.0755 ± 0.4312	total_rec_prncp
0.0674 ± 0.1974	last_pymnt_amnt
0.0371 ± 0.1758	last_fico_range_high
0.0343 ± 0.2913	loan_amnt
0.0317 ± 0.2253	funded_amnt
0.0055 ± 0.1333	funded_amnt_inv
0.0046 ± 0.3413	last_pymnt_d
0.0040 ± 0.1773	out_prncp
0.0032 ± 0.1534	out_prncp_inv
0.0028 ± 0.1128	issue_d
0.0011 ± 0.0740	last_credit_pull_d
0.0004 ± 0.0609	collection_recovery_fee
0.0004 ± 0.1119	installment
0.0002 ± 0.0846	next_pymnt_d
0.0002 ± 0.0675	total_rec_late_fee
0.0002 ± 0.0679	total_pymnt
0.0001 ± 0.0089	total_pymnt_inv
0.0001 ± 0.0656	fico_range_high
… 101 more …	

Figure 82: Global Explanation : Gradient Boosting Classifier - ELI5

6.5 PDP

PDP [7] stands for Partial Dependence Plot. As the name suggests, it shows the partial or marginal affect one or more features have on the outcome of the model.

Ethik AI [2] is used for plotting PDPs. Ethik is a python package for AI interpretability. The features that are picked to study the partial affects are based on the previous results by the other XAI techniques.

The selected features are:

- Collection Recovery Fee
- Funded Amount
- Funded Amount by Investor
- Installment
- Loan Amount
- Recoveries
- Total Payment
- Total Payment by Investor

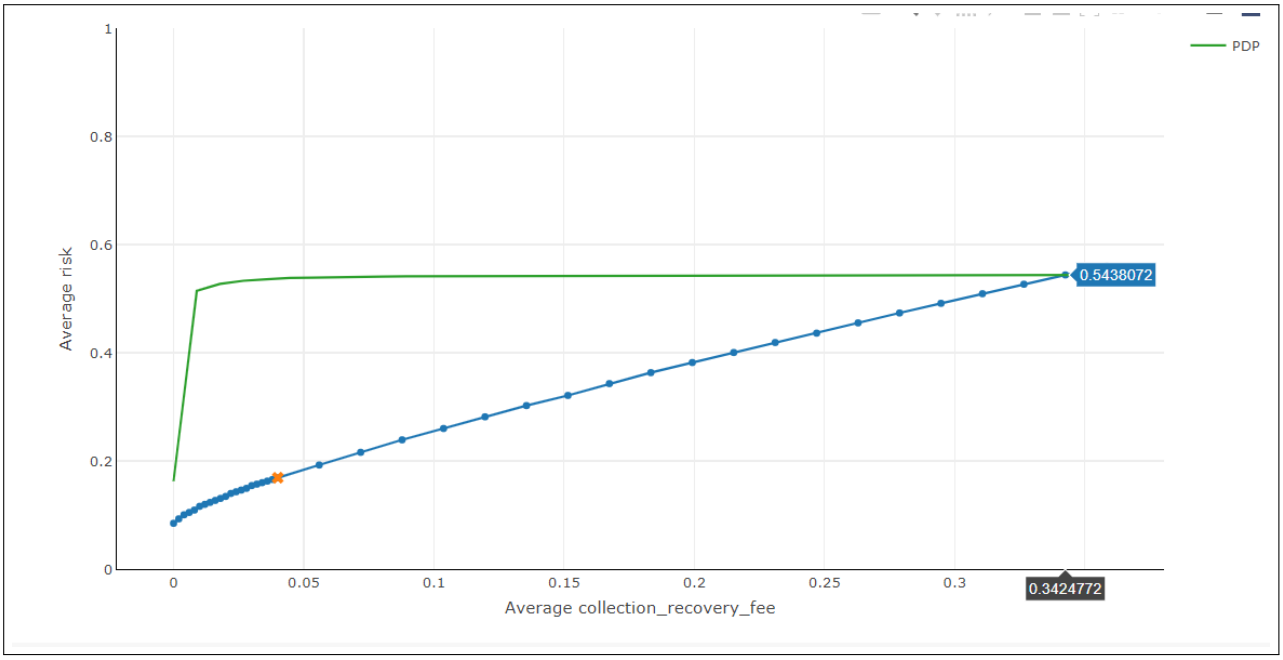


Figure 83: Collection Recovery Fee - PDP

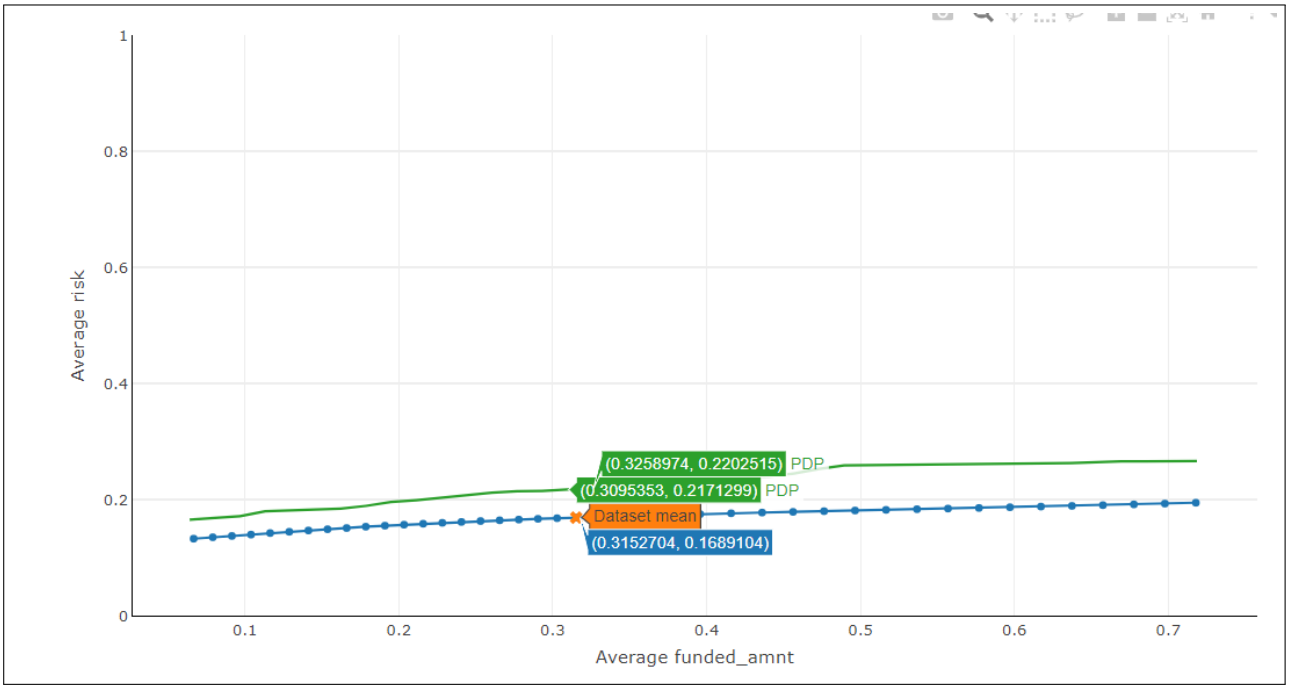


Figure 84: Funded Amount - PDP

The **blue** line displays the actual averaged data values and **green** line shows the partial dependency. The **orange** mark shows the data set mean value.

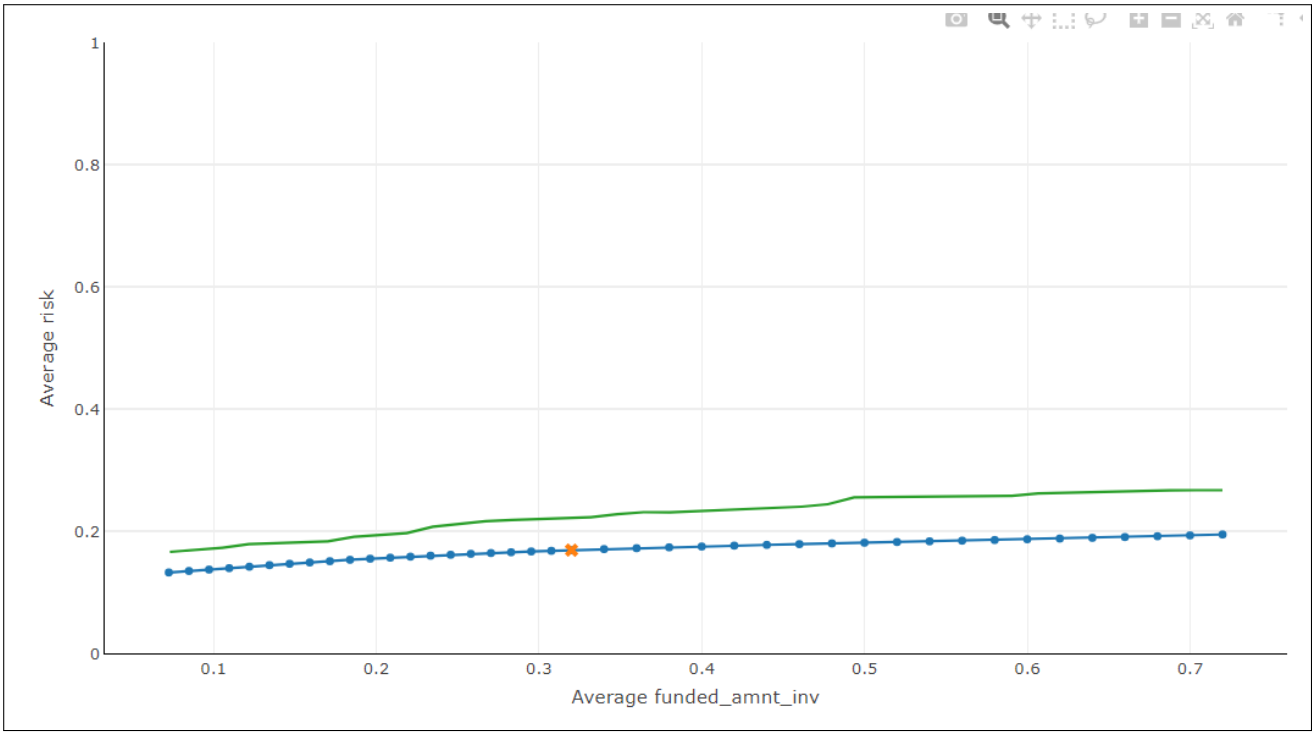


Figure 85: Funded Amount by Investor - PDP

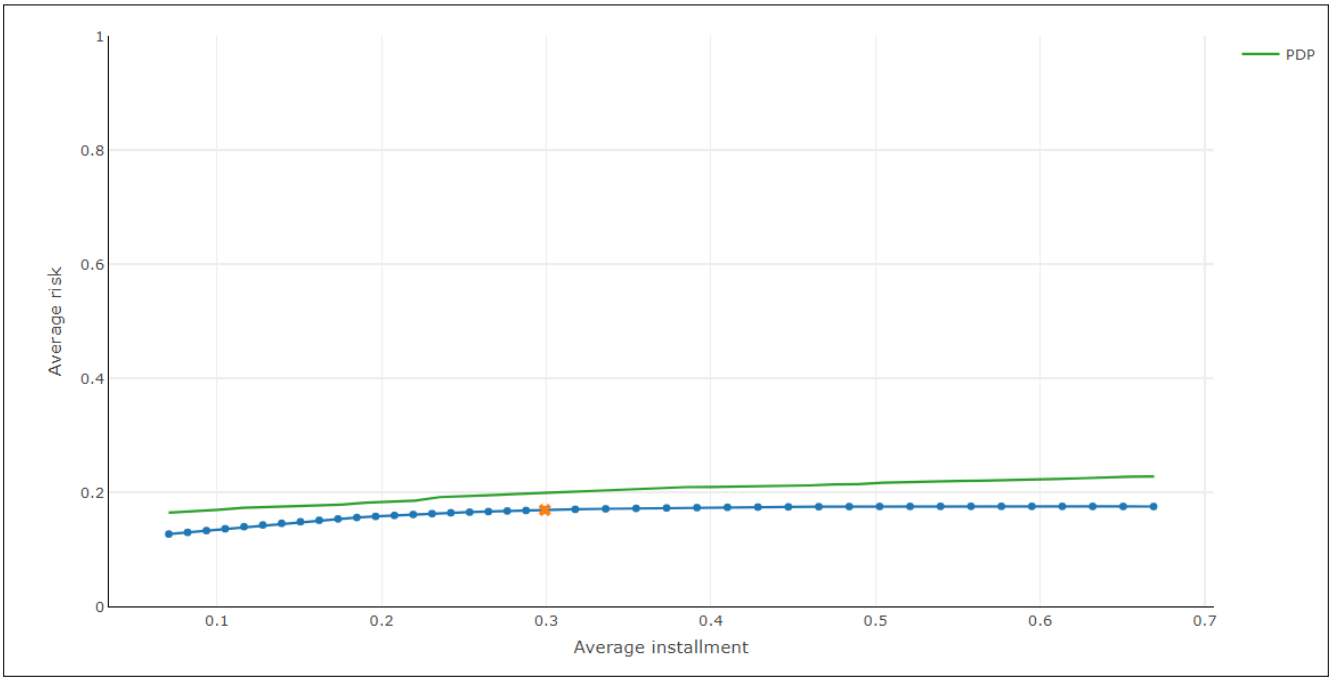


Figure 86: Installment - PDP

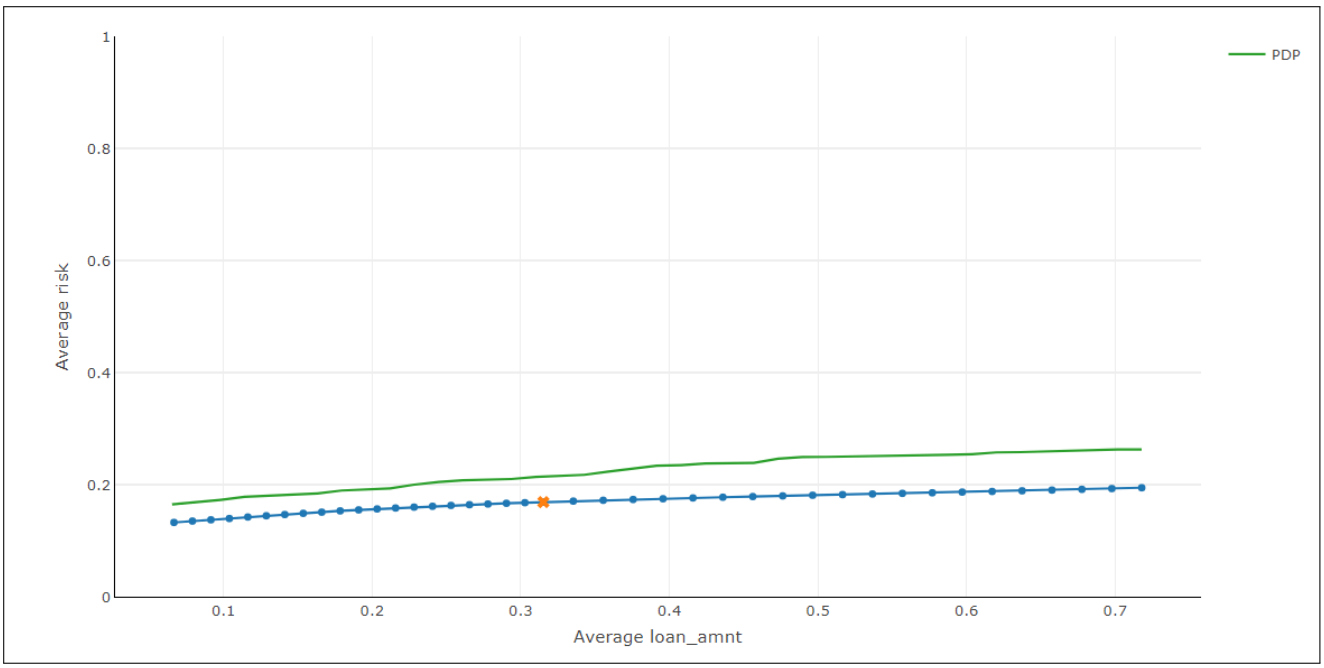


Figure 87: Loan Amount - PDP

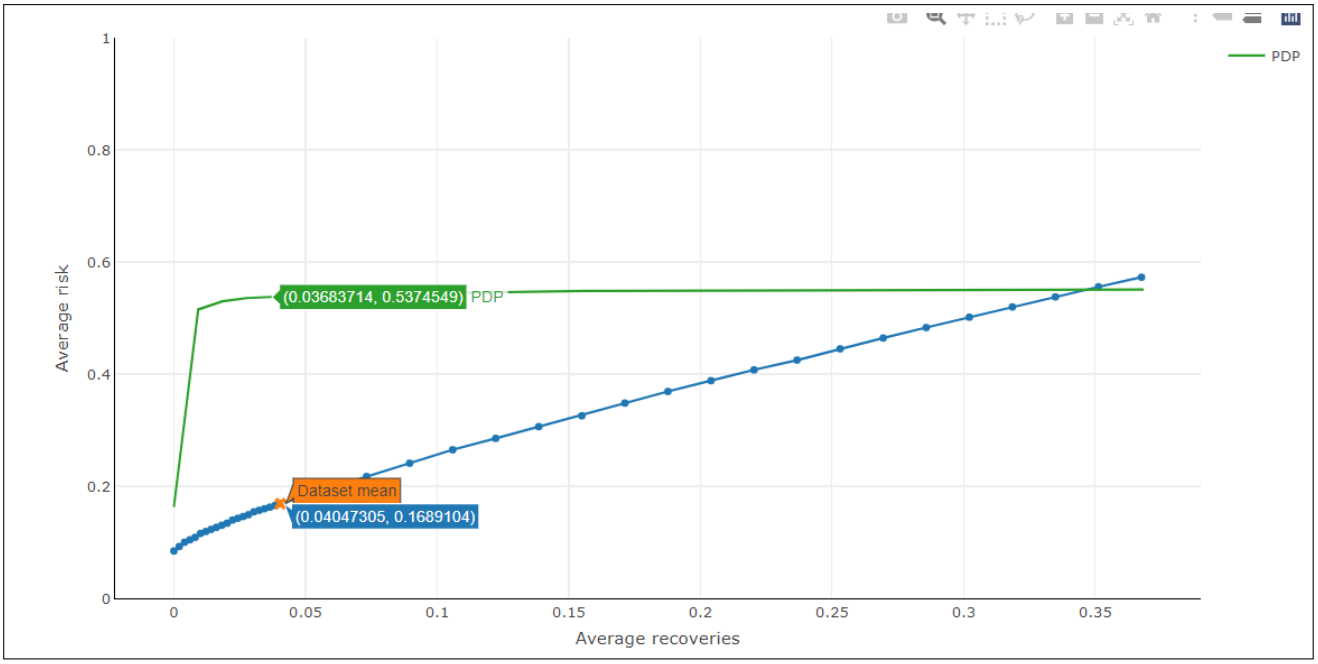


Figure 88: Recoveries - PDP

The x-axis shows the average feature value and the y-axis shows the average target value prediction

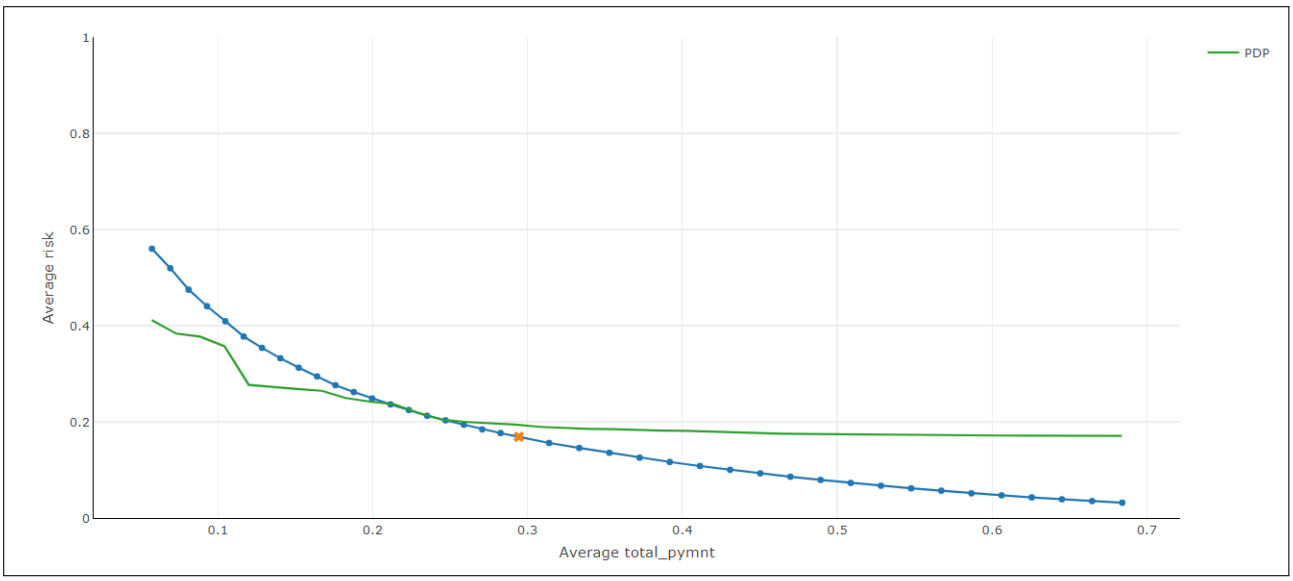


Figure 89: Total Payment - PDP

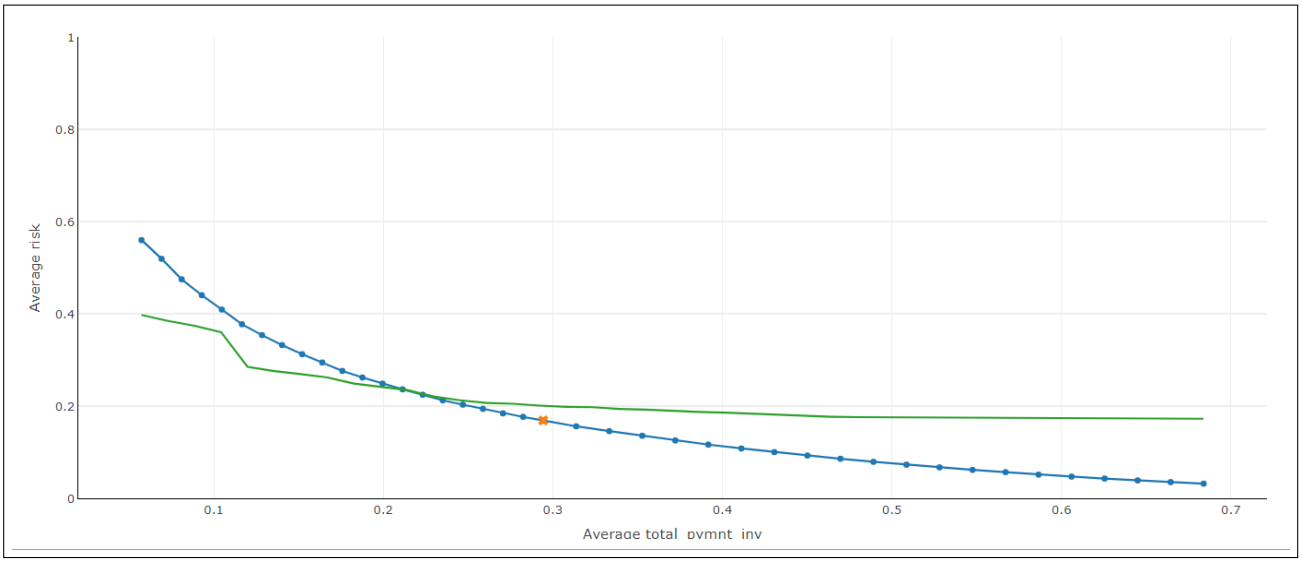


Figure 90: Total Payment by Investor - PDP

In the cases figure 89 and figure 90, as the average value of the feature decreases, the predicted value also tends to *0*.

7 Conclusion and Deployment

In order to display the results obtained to the client, the information needs to be made available that is easy to understand and interpret. **DASH** [22] by plotly is used to make interactive dashboards. It is used to make analytical web applications which can be easily understood by the business decision makers. The dashboard for CRM[13] is created using python. A predictor tool is also created which would enable the client to feed in their custom data and gain insights. It will help the client (lender or borrower) to know whether a particular application will be accepted or rejected along with the explanations related to it.

```
Loan Amount: 6000
Min FICO range: 500
Max FICO range: 600
Revolving Line Utilization: 20
Annual Income: 75000

Probability of Risky Loan :  0.81
```

Figure 91: Predictor Tool - in python

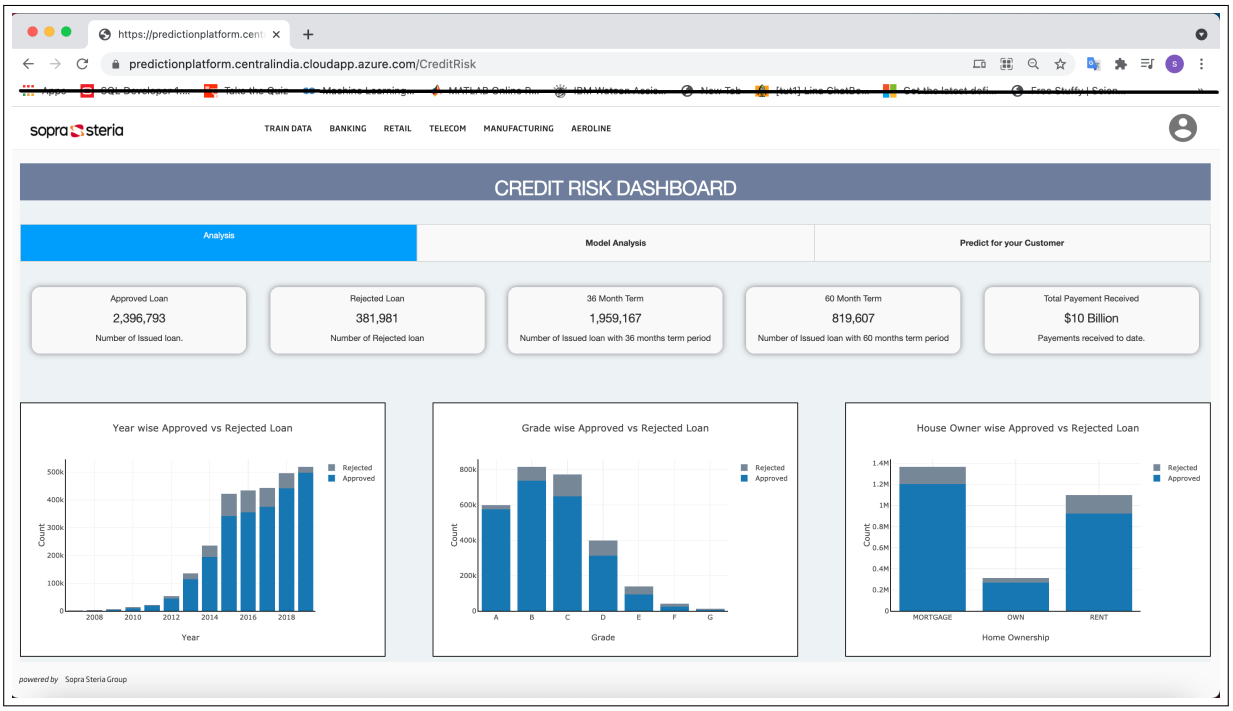


Figure 92: Dashboard Tab - Analysis

[22] https://dash.plotly.com/

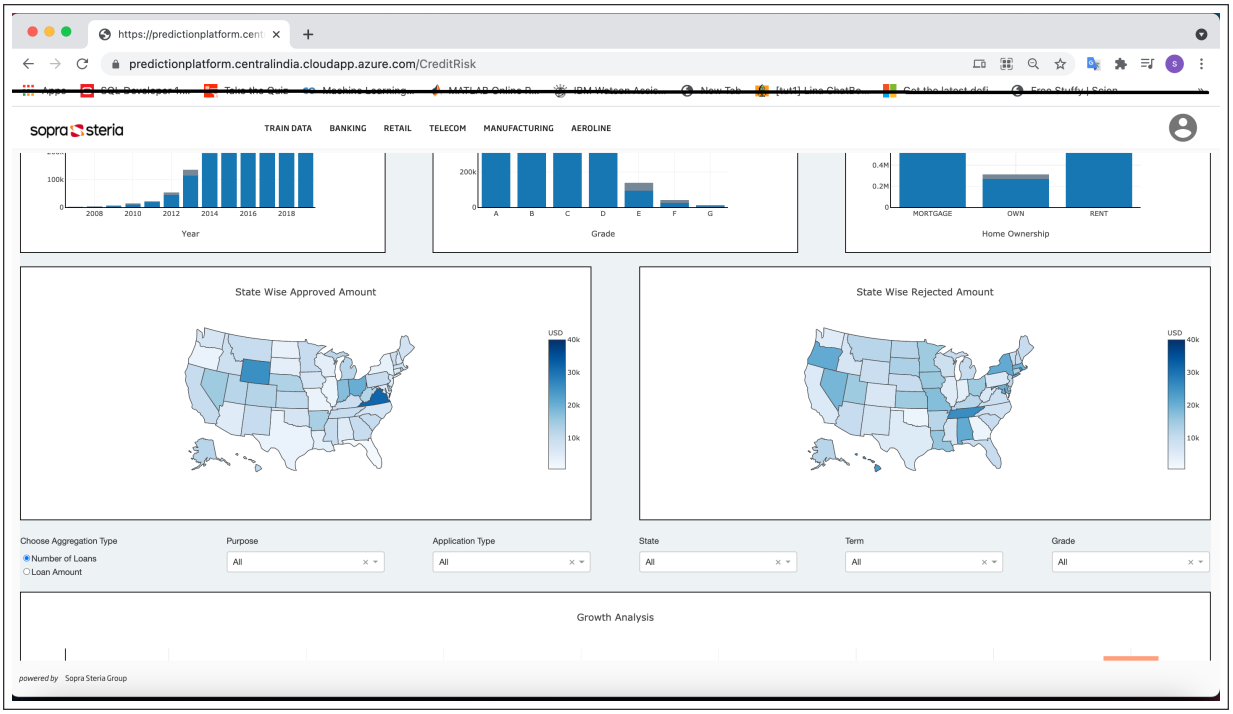


Figure 93: Dashboard Tab - Analysis

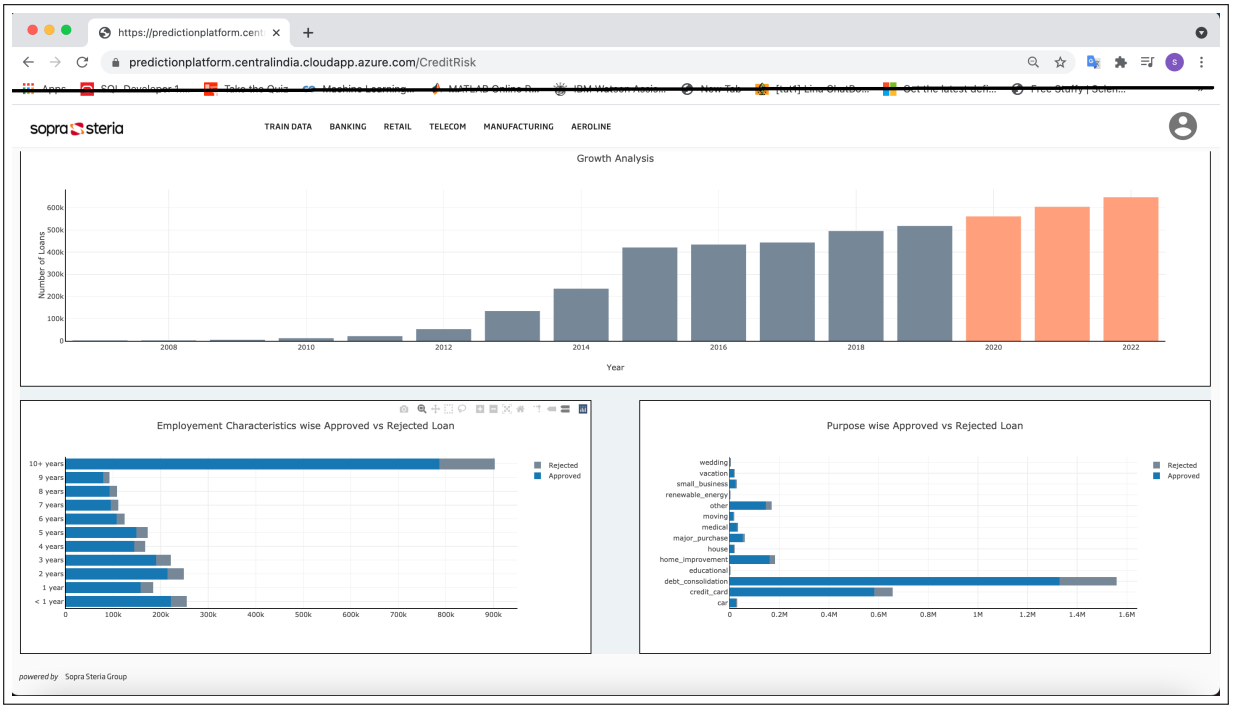


Figure 94: Dashboard Tab - Analysis

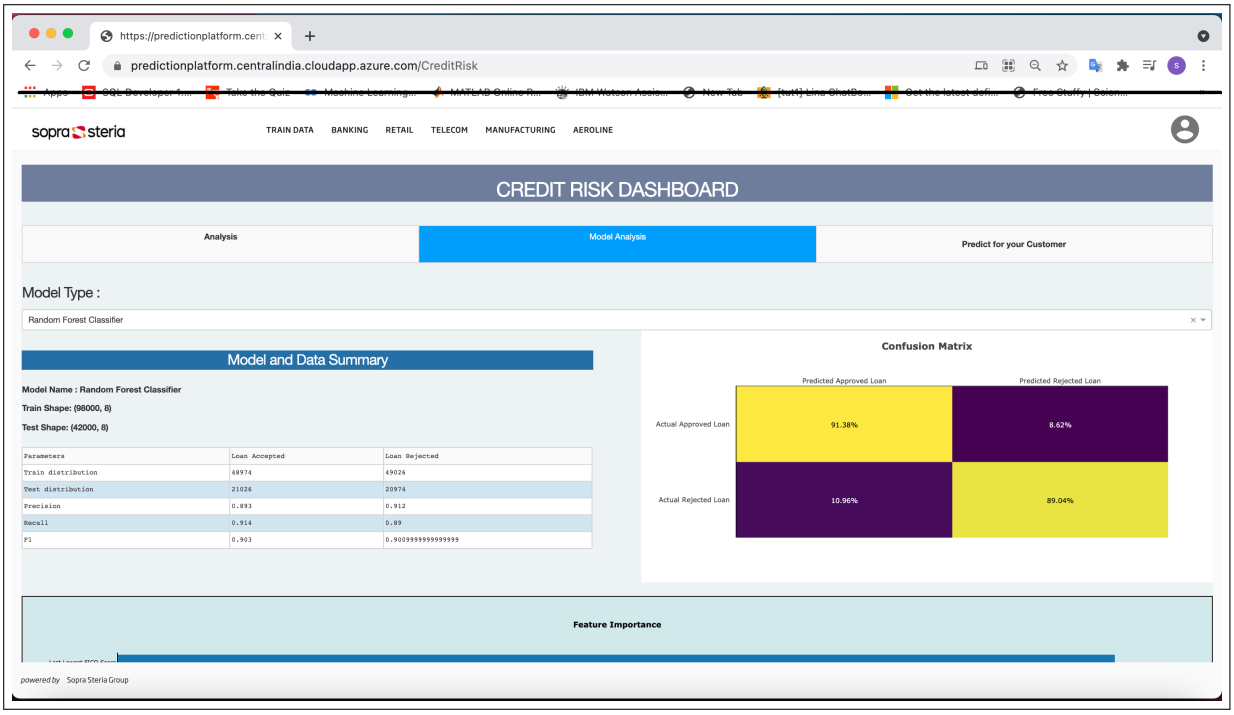


Figure 95: Dashboard Tab - Model Analysis

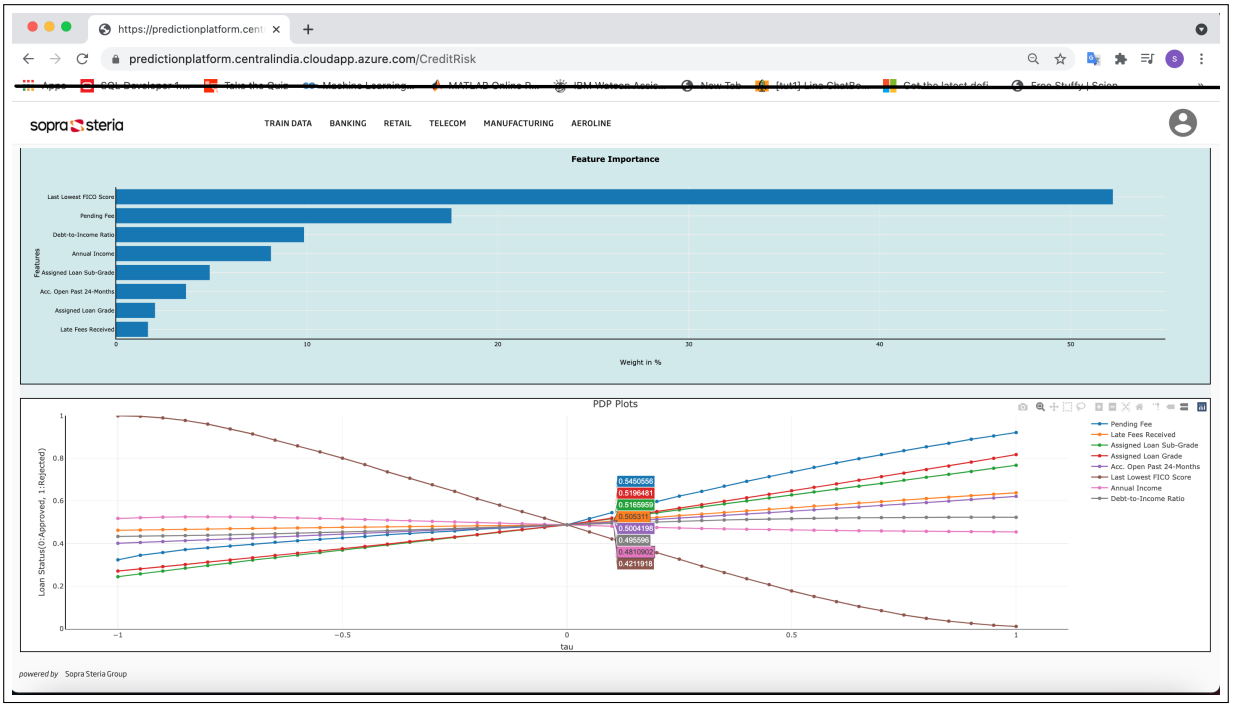


Figure 96: Dashboard Tab - Model Analysis

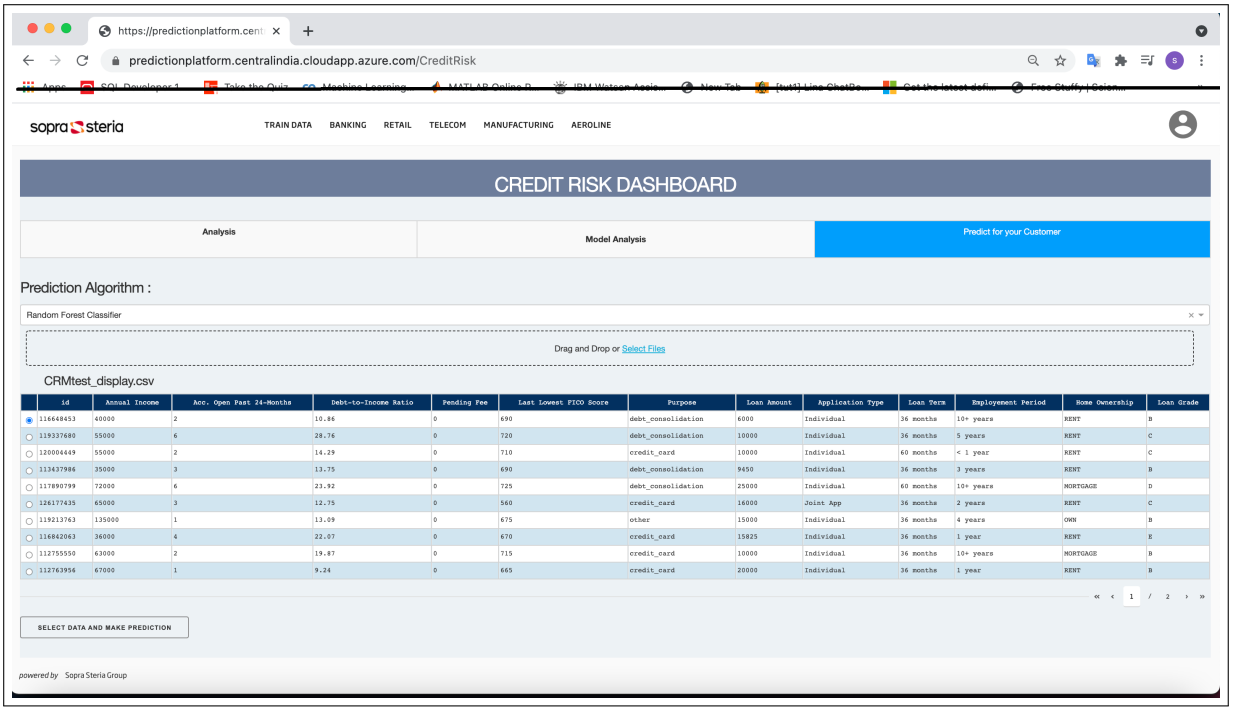


Figure 97: Dashboard Tab - Predict For Your Customer

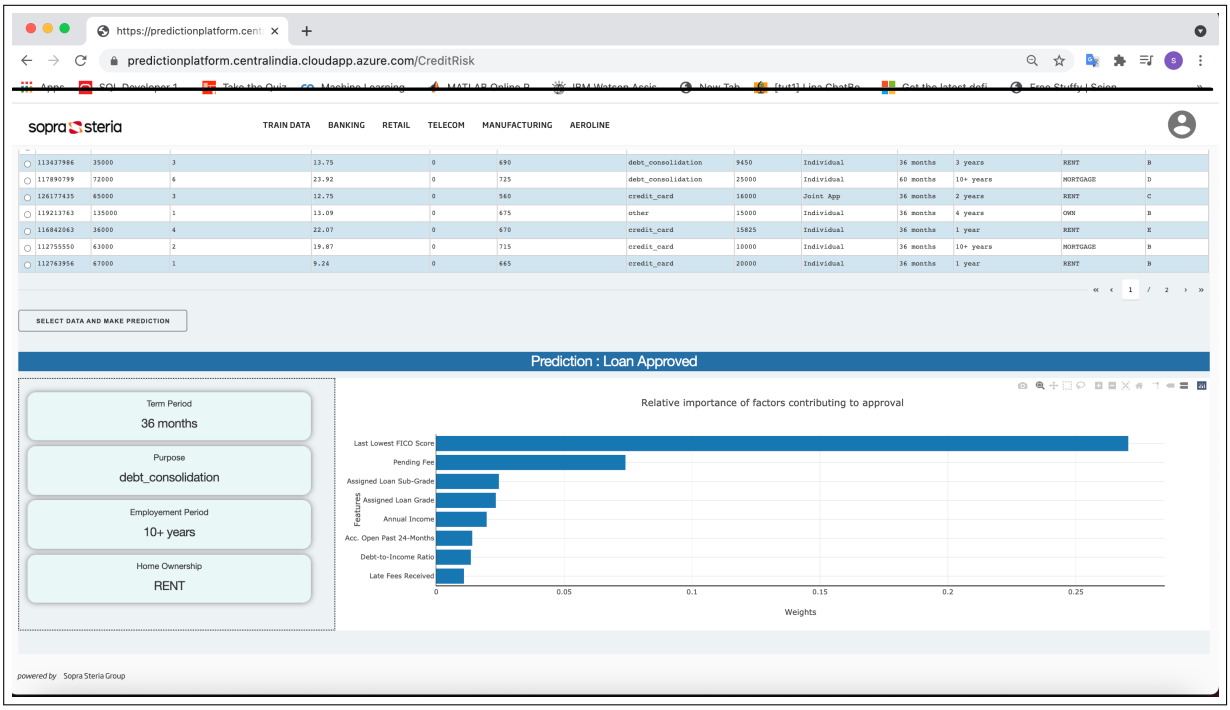


Figure 98: Dashboard Tab - Predict For Your Customer

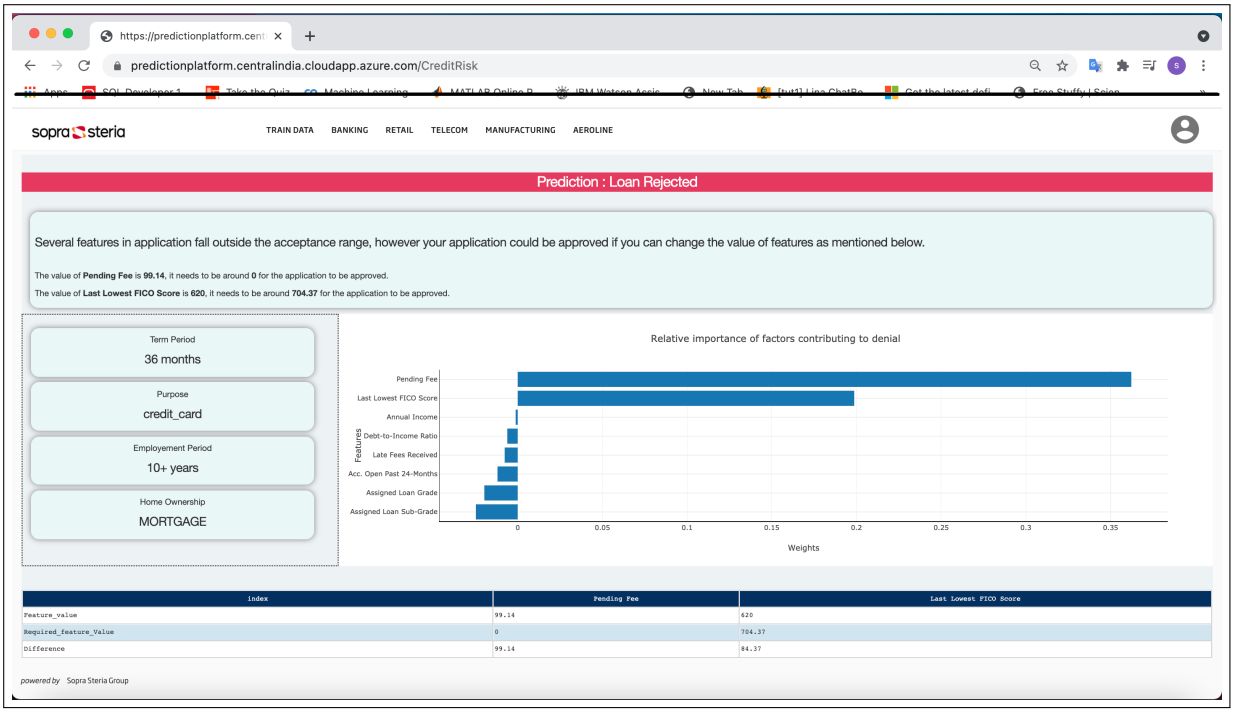


Figure 99: Dashboard Tab - Predict For Your Customer

In conclusion, this dashboard which depicts the results derived from the analytics and explainable AI will help the clients make business decisions for a smarter tomorrow.

References

[1] Dinesh Bacham and Janet Zhao. Machine learning: challenges, lessons, and opportunities in credit risk modeling. *Moody's Analytics Risk Perspectives*, 9:30–35, 2017.

[2] François Bachoc, Fabrice Gamboa, Max Halford, Jean-Michel Loubes, and Laurent Risser. Explaining machine learning models using entropic variable projection, 2020.

[3] Darren Cook. *Practical machine learning with H2O: powerful, scalable techniques for deep learning and AI.* " O'Reilly Media, Inc.", 2016.

[4] Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. Eli5: Long form question answering. In *Proceedings of ACL 2019*, 2019.

[5] Gilles Gasso. Logistic regression, 2019.

[6] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, 2(1):2522–5839, 2020.

[7] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[9] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144, 2016.

[10] Nikos Skantzos and N Castelein. Credit scoring case study in data analytics, 2016.

[11] Shan Suthaharan. Machine learning models and algorithms for big data classification. *Integr. Ser. Inf. Syst*, 36:1–12, 2016.

[12] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009.

[13] Luka Vidovic and Lei Yue. Machine learning and credit risk modelling, 2020.