CREDIT RISK ANALYTICS

SUBMITTED BY:

Vignesh Saminathan

(2147235) 6MCA – B

UNDER THE GUIDANCE OF

External Guide	**Internal Guide**
Mr. Kumar Sambhav	Dr. Rajesh Kanna R
Senior Data Analyst	Assistant Professor
Techieventures Technologies	
Private Limited	
	Computer Science Dept
	CHRIST (Deemed to be University)

A project report submitted in partial fulfilment of the requirements for the award of degree of Master of Computer Applications of
CHRIST (Deemed to be University)

May – 2023

CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

CERTIFICATE

This is to certify that the report titled **“CREDIT RISK ANALYTICS”** is a bona fide record of work done by **Vignesh Saminathan(2147235)** of CHRIST (Deemed to be University), Bengaluru, in partial fulfilment of the requirements of VI Trimester MCA during the year 2023.

Head of the Department **Faculty in charge**

Valued by:

Vignesh Saminathan - 2147235

Examination Centre
CHRIST (Deemed to be University)

Date:

OFFER LETTER

Techieventures Technologies Pvt. Ltd.
Knowledge Brick By Brick

Date: 01/02/2023

Internship Offer Letter

Vignesh Saminathan
#132, 32nd Main,
BTM Layout 1st Stage,
Bangalore - 560068

Dear **Vignesh Saminathan,**

We are pleased to confirm your acceptance of an internship position as **"Data Analyst Intern"**. Your first day of the Internship will be **06-02-2023**. The internship assignment is expected to be completed within six months of your joining. Your assignments for this position will be those described to you in your orientation with Mr. Kumar Sambhav. **The aim of Internship will be, to perform Credit Risk Analytics.**

During the internship you will be paid fixed stiffen of Rs. 6,500/- .

Please report to the Human Resources Department at **11:30am** on **06-02-2023** with the appropriate documents and completed forms.

If you have any questions, please feel free to contact us. We are very pleased that you have decided to join **Techieventures Technologies Pvt. Ltd.** We look forward to seeing you on **06-02-2023** and offer a very warm welcome.

Warm Regards

I ACCEPT THE ABOVE

Name:
Date

Kumar Sambhav
CEO/Managing Director,
Techieventures Technologies Pvt. Ltd.
Bangalore.

ACKNOWLEDGEMENTS

First and foremost, I thank the Almighty for his unfathomable grace and blessings bestowed upon me throughout every phase of this effort. I want to take this opportunity to express my gratitude to a number of people for their support throughout the course of this endeavor. I want to thank Dr. Ashok Immanuel, the department head and program coordinator, for allowing me to work on this project. They have continuously offered me guidance and inspiration, which has considerably boosted my motivation.

Dr. Rajesh Kanna R, who served as my project mentor, has been a constant source of inspiration and encouragement during the entire project. His guidance and expertise have tremendously helped me develop my ideas and skills.

Additionally, I am really appreciative to Techieventures Technologies Private Limited, India for providing me with this excellent opportunity to work on a real-world business project.

Finally, I want to thank my family and friends for their unwavering encouragement and support. Their unending support and encouragement have greatly aided my motivation and focus.

Finally, I want to express my profound gratitude to everyone who has helped me along this process. They consistently supported me, which helped me complete the task within the specified time.

Signature

Vignesh Saminathan(2147235)

LIST OF FIGURES

TABLE OF CONTENTS

ABSTRACT

A wide range of financial services are provided by banks, including different kinds of banking facilities, credit cards, loans, insurance, wealth management, and more. By examining a dataset of transactions that determine whether or not a person is offered a loan, this credit card data analysis and visualization project seeks to shed light on the typical usage of credit cards. Loan_amnt, int_rate, member_id, and other pertinent data are included in the data. To find trends and patterns in the data, the project makes use of a variety of data analysis techniques, including data cleaning, exploration, and modeling. Additionally, to present the results in an understandable and simple way, the project makes use of data visualization tools like bar charts, scatter plots, and AUC curves. The analysis is focused on determining the Probability of Default across data for more than 800,000 consumer loans that the company issued from 2007 to 2015. The knowledge gained from this project can assist businesses in making educated decisions regarding loan decisions, risk management, and product development

1. INTRODUCTION

1.1 PROBLEM DESCRIPTION

The "Credit Risk Modeling" refers to data-driven risk models which calculate the chances of a borrower who defaults on a loan (or credit card). The failure to repay the loan taken, the amount he/she owes at the time, and the amount the lender would lose from the outstanding amount.

Demographic Data

Existing Relationship

Payment Information

Term

Credit Variables

The probability of Default is used to score each customer to assess his/her likelihood of Default.

This credit score can be built internally by a bank, or banks can use the score of credit bureaus.

Credit Bureaus collect individuals' credit information from various banks and sell it as a credit report. They also release credit scores.

1.2 EXISTING SYSTEM

Banking is built on the idea of profiting by loaning money to those that require money. Banks then collect interest on the payments which the Borrower makes in order to pay back the money they borrowed. The likely event that some borrowers will default on their loans, that is, fail to make their payments, results in a financial loss for the bank. In the application process for new loans, banks assess the potential Borrower's creditworthiness. As a measure of creditworthiness, some assessments are made on the probability of Default for the potential borrowers. The risk that the credit assessment of the borrowers is too modest is called credit

risk. Credit risk modeling is quite an active research field. The probability of Default is a crucial figure in the daily operation of any credit institute, as it is used as a measure of credit risk in internal and external reporting. The credit risk assessments made by banks are
commonly referred to as credit rating models. This thesis uses various statistical methods as models for credit rating models.

1.3 PROJECT SCOPE AND APPLICABILITY

PURPOSE

This project was done in cooperation with a corporate bank, which supplied the necessary data resources. The project aimed to see whether the cooperating corporate bank used heuristic credit rating models. Validation of credit ratings is complex compared to regular modeling, where no actual or observed rating can be compared with the predicted credit rating to measure the prediction error. Some validation methods are available, but no single measure can be used to decide whether one model is better than the other. It is thus necessary to consider numerous measures simultaneously to draw some conclusions on model performance. This has a clear disadvantage as it might be debatable whether one model is better. In order to address this problem, an attempt was made to combine the available measures into a single measure. As missing values are frequently apparent in many modeling variables, some thoughts are made on how to solve that problem. The problem regarding a small sample of data is dealt with. The general purpose of this project is to see how it is possible to construct credit rating models. Particular emphasis is made on the practical methods a bank in the corporate banking sector
could use to develop a new credit rating model.

SCOPE

Credit risk modeling is a broad field. This project attempts to create a working model to understand better and shed light on the various subjects of credit risk modeling. We are providing a fundamental understanding of credit risk modeling and finding ways to perform the model.

APPLICABILITY

We divided the data into two periods because we assume that some data are available when we need to build Expected Loss models, and some data comes from applications after. Later, we investigate whether our applications after building the Probability of Default (PD) model have similar characteristics to those we used to build the PD model.

1.4 OVERVIEW OF THE PROJECT

Credit risk modeling is a statistical method for estimating the probability that a borrower would stop making payments on a loan. The Borrower's default probability (PD) is calculated using previous data on borrower behavior, including repayment history, income, and credit score. Banks and other financial institutions frequently utilize credit risk models to guide lending choices, set interest rates, and establish credit limits.

2. SYSTEM ANALYSIS AND REQUIREMENTS

2.1 PROBLEM DEFINITION

A significant risk to the lender is the amount they can recover should the Borrower Default on their obligation. In Default, lenders rarely receive the total value of their legal claim. The fraction of principal recovered has been shown to vary with whether there are other claims on the Borrower's assets, the type and condition of the Borrower's assets, economic conditions, geography, and other factors.

2.1 REQUIREMENTS SPECIFICATION

Objective	Credit Risk Modeling
Tools	Python
Editor	Jupyter Notebook
Backend	PostgreSQL

2.1.1 HARDWARE REQUIREMENTS

RAM	2 GB
CPU	Intel Core i3 or Higher
Hard Disk Size	80 GB

2.1.1 SOFTWARE REQUIREMENTS

Objective	Credit Risk Modeling
Tools	Python
Editor	Jupyter Notebook
Backend	PostgreSQL

2.2.1 FUNCTIONAL REQUIREMENTS

The functional requirements for a credit card analysis project using Pyspark and Tableau may

include

The various packages and modules required for the system are:

- **Numpy: To perform a wide range of operations related to math on arrays and has an enormous library for**

  high-level math functions that operate on these arrays and matrices.

- **Pandas: To analyze big data and make relevant conclusions based on Stats theory. Allows to make big messy**

  data readable and relevant.

- **From sklearn.model_selection import train_test_split: To split data arrays into two subsets for training and**

  testing purposes.

- **matplotlib.pyplot: To make plots that allow us to visually represent and show the data in charts and add**

  features to control line styles, font properties, and formatting axes.

- **Seaborn: To explore and understand the data. Its plotting functions operate on data frames and arrays containing whole datasets and internally perform semantic mapping and statistical aggregation to produce**

  informative plots.

  Data Sources: The credit card analysis project requires access to transactional data.

2.2.2 NON-FUNCTIONAL REQUIREMENTS

For the Credit risk modeling analysis project, the non-functional needs could be:

1. Performance: The project must be planned to work well, promptly processing enormous amounts of data.

   Additionally, it must manage multiple users at once and have a responsive user experience.

2. Reliability: The project must be trustworthy to guarantee that the credit card transactional data is correctly processed

   and examined and that the outcomes are consistent.
3. Scalability: The project should be scalable, allowing for larger datasets and more intricate Analysis with little

   adverse effect on reliability and performance.
4. Maintainability: To enable future improvements and bug fixes, the project should be manageable, with

   well-structured code, documentation, and version control.
5. Usability: The project should be simple, with an intuitive user interface and clear visualizations that give

   non-technical stakeholders valuable insights.
6. Accessibility: To ensure stakeholders can access the insights and reports when needed, the project should be available with little downtime and disruption.

2.3 SYSTEM REQUIREMENTS

2.3.1 USER CHARACTERISTICS

There are many consumers of credit risk modeling, each with particular preferences. Credit risk modeling's primary user attributes include some of the following:

Data scientists: In most cases, data scientists are the primary consumers of credit risk modeling. They are responsible for creating and using the models and are skilled in programming,
machine learning, and statistical modeling.

Credit analysts: Credit analysts often determine a borrower's creditworthiness and make lending choices. They employ credit risk models to assess the risk of Default and establish the terms and conditions of loans.

Risk managers are in charge of directing the organization's risk management procedure. Credit risk models are used to determine and control credit risk exposure and to manage and improve the risk-reward tradeoff in the lending portfolio.

Regulators: Regulators are responsible for ensuring that financial institutions abide by all applicable laws and have enough capital to support their lending operations. They evaluate the capital sufficiency of financial institutions and pinpoint potential risk areas using credit risk models.

Investors: Because credit risk models shed light on investment portfolios' risk and return characteristics, investors are drawn to them. They assess the risk of Default in fixed-income instruments and the creditworthiness of issuers using credit risk models.

Credit analysts, data scientists, risk managers, regulators, and investors are just a few of the many users for whom credit risk modeling is an essential tool. Despite the differences in each user group's characteristics, all rely on credit risk models to assess their exposure to credit risk and make wise lending and investment decisions.

2.3.2 SOFTWARE REQUIREMENTS

The software and requirements for a credit card analysis project using Pyspark and Tableau may include the following:

The various packages and modules required for the system are:

- **Numpy: To perform a wide range of operations related to math on arrays and has an enormous library for**

  high-level math functions that operate on these arrays and matrices.
- **Pandas: To analyze big data and make relevant conclusions based on Stats theory. Allows to make big messy**

  data readable and relevant.
- **From sklearn.model_selection import train_test_split: To split data arrays into two subsets for training and**

  testing purposes.
- **matplotlib.pyplot: To make plots that allow us to visually represent and show the data in charts and add**

  features to control line styles, font properties, and formatting axes.
- **Seaborn: To explore and understand the data. Its plotting functions operate on data frames and arrays containing whole datasets and internally perform semantic mapping and statistical aggregation to produce**

  informative plots.

Data Sources: The credit card analysis project requires access to transactional data.

1.**HARDWARE REQUIREMENTS**

1. Processor: For effective data processing, a multi-core processor with a clock speed of at least 2.5 GHz is advised.
2. Memory (RAM): A minimum of 16 GB RAM is advised for handling large datasets and detailed studies.
3. Storage: Ample storage space is needed to keep the data, preliminary findings, and software. It has advised to have 500 GB or more of free disc space.
4. GPU: Although not required, a GPU is advised for quicker data visualization and rendering. GPUs from NVIDIA GeForce or AMD Radeon are advised.

5. Operating System (OS): Windows, Linux, or macOS can all be used to run the project.

2.3.2 HARDWARE REQUIREMENTS

The hardware requirements may vary depending on the size of the dataset and the complexity of the analysis. However, some general hardware requirements may include:

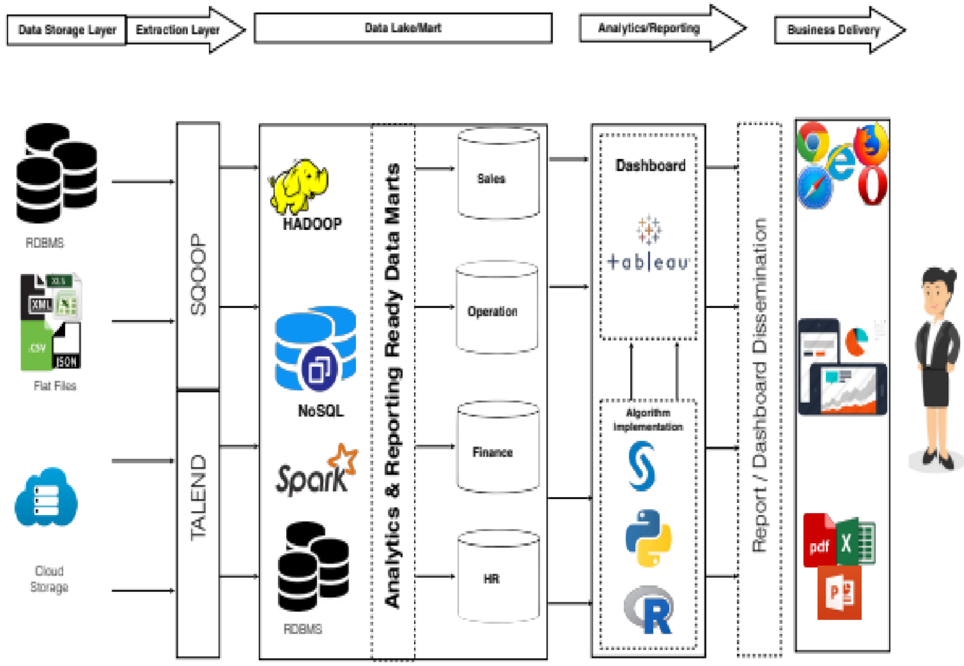
1. Processor: A multi-core processor with a clock speed of at least 2.5 GHz is recommended to ensure efficient data processing.
2. Memory (RAM): A minimum of 16 GB RAM is recommended to handle large datasets and complex analyses.
3. Storage: Sufficient storage space is required to store the data, intermediate results, and software. At least 500 GB of free disk space is recommended.
4. Graphics Processing Unit (GPU): A GPU is optional but recommended for faster data visualization and rendering. NVIDIA GeForce or AMD Radeon GPUs are recommended.
5. Network: A stable and reliable internet connection is required for data access, software updates, and collaboration.
6. Operating System (OS): The project can be executed on Windows, Linux, or macOS. It is important to note that these hardware requirements are general recommendations, and the actual requirements may vary depending on the specific needs of the project. It is recommended to consult with a technical expert to determine the optimal hardware configuration for the project

2.3.4 CONSTRAINTS

The constraints for a credit card analysis project using Pyspark and Tableau may include the following:

1. Data Quality: The credit card transactional data quality is critical to the project's success. If the data is complete, accurate, and consistent, the results of the Analysis will be reliable. Data cleansing and normalization may be required to ensure that the data is high quality.
2. Data Privacy: Credit card transactional data contains sensitive information, such as customer names, card numbers, and transaction details. The project must comply with data privacy regulations, such as GDPR and HIPAA, to protect the confidentiality of the data.
3. Time Constraints: The project may have time constraints, such as a deadline for completing the Analysis or launching a marketing campaign. It is essential to manage time effectively to ensure the project is completed on time.
4. Resource Constraints: The project may have resource constraints, such as a limited budget or staff. It is essential to manage resources effectively to ensure the project is completed within budget and meets its objectives.
5. Technical Expertise: The project requires technical expertise in Pyspark, Tableau, and data analytics. Finding qualified personnel with the necessary skills and knowledge to complete the project successfully may be challenging.
6. Scalability: The project may need to scale to handle larger datasets or more complex analyses in the future. It is essential to design the project with scalability to ensure it can accommodate future growth.

3. SYSTEM DESIGN



3.1 **SYSTEM ARCHITECTURE**

The system architecture of a credit card analysis project using Pyspark and Tableau may include the following components:

1. Data Sources: The project will source transactional data from various sources such as databases, spreadsheets, and

   cloud storage.

2. Data Ingestion Layer: The data ingestion layer will use Pyspark to ingest the data and perform initial data cleaning

   and preprocessing.

3. Data Transformation Layer: The data transformation layer will use Pyspark to transform the data into a usable format

   suitable for Analysis. This will include filtering, aggregating, and joining the data.

4. Data Analysis Layer: The data analysis layer will use Pyspark's machine learning and statistical libraries to identify

   patterns and trends in the data. The analysis results will be used to generate insights for business decision-making.

5. Data Visualization Layer: The data visualization layer will use Tableau's interactive and dynamic dashboards, charts,

   and graphs to visualize the insights and results of the data analysis.
6. Deployment Layer: The deployment layer will deploy the project to a cloud platform, such as AWS or Azure, or a

   local server for easy access and collaboration.
7. Security Layer: The security layer will ensure the security and integrity of the data with access controls, encryption,

   and compliance with data privacy regulations.
8. Scalability Layer: The scalability layer will allow the project to handle larger datasets and more complex analyses by

   adding resources such as CPUs, GPUs, and memory.
9. Maintenance Layer: The maintenance layer will ensure the project is maintainable, with well-structured code, documentation, and version control.

3.2 MODULE DESIGN

Several variables include the data sources, model choice, feature engineering, and performance assessment. Here are some general guidelines to bear in mind:

Define the scope and objectives: Specifying the module's scope and objectives before starting the design process is necessary. This entails deciding which credit risk categories should be modeled, as well as the target audience and
performance criteria employed for assessment.

Collect information: To model credit risk, information on the borrower's financial situation, credit history, and other relevant factors is needed. Various internal and external sources, including credit bureaus, financial documents, loan application forms, etc., can be used to obtain the data.

Data preprocessing: To ensure the data is suitable for analysis, it may be necessary to clean, transform, and normalize it. This could entail encoding categorical variables, scaling numerical features, and handling missing data.

After preprocessing the data, the next step is identifying pertinent characteristics that may be utilized to create a predictive model. This process is known as feature engineering. Increasing the variables' ability to predict outcomes may entail establishing new variables, choosing crucial variables, and changing existing variables.

Model selection: After the features have been retrieved, a suitable model needs to be chosen. Standard credit risk modeling models include logistic regression, decision trees, random forests, and neural networks. Multiple models may be assessed throughout the model selection process, and their performance may be compared using cross-validation or other methods.

3.3 DATABASE DESIGN

The database design consists of the following tables:

LoanStatNew	Description
addr_state	The state provided by the borrower in the loan application
annual_inc	The self-reported annual income provided by the borrower during registration.
annual_inc_joint	The combined self-reported annual income provided by the co-borrowers during registration
application_type	Indicates whether the loan is an individual application or a joint application with two co-borrowers
collection_recovery_ fee	post charge off collection fee
collections_12_mths _ex_med	Number of collections in 12 months excluding medical collections

delinq_2yrs	The number of 30+ days past-due incidences of delinquency in the borrower's credit file for the past 2 years
Desc	Loan description provided by the borrower
Dti	A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income.
dti_joint	A ratio calculated using the co-borrowers' total monthly payments on the total debt obligations, excluding mortgages and the requested LC loan, divided by the co-borrowers' combined self-reported monthly income
earliest_cr_line	The month the borrower's earliest reported credit line was opened
emp_length	Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years.
emp_title	The job title supplied by the Borrower when applying for the loan.*
fico_range_high	The upper boundary range the borrower's FICO at loan origination belongs to.
fico_range_low	The lower boundary range the borrower's FICO at loan origination belongs to.
funded_amnt	The total amount committed to that loan at that point in time.
funded_amnt_inv	The total amount committed by investors for that loan at that point in time.
Grade	LC assigned loan grade
home_ownership	The home ownership status provided by the borrower during registration. Our values are: RENT, OWN, MORTGAGE, OTHER.
Id	A unique LC assigned ID for the loan listing.
initial_list_status	The initial listing status of the loan. Possible values are – W, F
inq_last_6mths	The number of inquiries in past 6 months (excluding auto and mortgage inquiries)
Installment	The monthly payment owed by the borrower if the loan originates.
int_rate	Interest Rate on the loan
is_inc_v	Indicates if income was verified by LC, not verified, or if the income source was verified

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 belongs to.
last_fico_range_low	The lower boundary range the borrower's last FICO pulled belongs to.
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. If at some point in time, the credit department reduces the loan amount, then it will be reflected in this value.
loan_status	Current status of the loan
member_id	A unique LC assigned Id for the borrower member.
mths_since_last_del inq	The number of months since the borrower's last delinquency.
mths_since_last_ma jor_derog	Months since most recent 90-day or worse rating
mths_since_last_rec ord	The number of months since the last public record.
next_pymnt_d	Next scheduled payment date
open_acc	The number of open credit lines in the borrower's credit file.
out_prncp	Remaining outstanding principal for total amount funded
out_prncp_inv	Remaining outstanding principal for portion of total amount funded by investors
policy_code	publicly available policy_code=1 new products not publicly available policy_code=2
pub_rec	Number of derogatory public records
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_util	Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
sub_grade	LC assigned loan subgrade

Term	The number of payments on the loan. Values are in months and can be either 36 or 60.
Title	The loan title provided by the borrower
total_acc	The total number of credit lines currently in the borrower's credit file
total_pymnt	Payments received to date for total amount funded
total_pymnt_inv	Payments received to date for portion of total amount funded by investors
total_rec_int	Interest received to date
total_rec_late_fee	Late fees received to date
total_rec_prncp	Principal received to date
url	URL for the LC page with listing data.
verified_status_joint	Indicates if the co-borrowers' joint income was verified by LC, not verified, or if the income source was verified
zip_code	The first 3 numbers of the zip code provided by the borrower in the loan application.
open_acc_6m	Number of open trades in last 6 months
open_il_6m	Number of currently active installment trades
open_il_12m	Number of installment accounts opened in past 12 months
open_il_24m	Number of installment accounts opened in past 24 months
mths_since_rcnt_il	Months since most recent installment accounts opened
total_bal_il	Total current balance of all installment accounts
il_util	Ratio of total current balance to high credit/credit limit on all install acct
open_rv_12m	Number of revolving trades opened in past 12 months
open_rv_24m	Number of revolving trades opened in past 24 months
max_bal_bc	Maximum current balance owed on all revolving accounts
all_util	Balance to credit limit on all trades
total_rev_hi_lim	Total revolving high credit/credit limit
inq_fi	Number of personal finance inquiries
total_cu_tl	Number of finance trades
inq_last_12m	Number of credit inquiries in past 12 months

acc_now_delinq	The number of accounts on which the borrower is now delinquent.

3.3 DATA DESCRIPTION

The presented data is sourced from the dataset containing all available data for over 800,000 consumer loans issued from 2007 to 2015 by Company X Data prominent US peer-to-peer lending company.

tot_coll_amt	Total collection amounts ever owed
tot_cur_bal	Total current balance of all accounts

We divided the data into two periods because we assume that some data are available when we need to build Expected Loss models, and some data comes from applications after. Later, we investigate whether the applications we have after building the Probability of Default (PD) model have similar characteristics to those we used to build the PD model.

```
In [11]: loan_data
```

Out[11]:

	sl_no	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	grade	...	total_bal_il	il_util	open_r
0	0.0	1077501.0	1296599.0	5000.0	5000.0	4975.0	36 months	10.65	162.87	B	...	None	None	
1	1.0	1077430.0	1314167.0	2500.0	2500.0	2500.0	60 months	15.27	59.83	C	...	None	None	
2	2.0	1077175.0	1313524.0	2400.0	2400.0	2400.0	36 months	15.96	84.33	C	...	None	None	
3	3.0	1076863.0	1277178.0	10000.0	10000.0	10000.0	36 months	13.49	339.31	C	...	None	None	
4	4.0	1075358.0	1311748.0	3000.0	3000.0	3000.0	60 months	12.69	67.79	B	...	None	None	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
466280	466280.0	8598660.0	1440975.0	18400.0	18400.0	18400.0	60 months	14.47	432.64	C	...	None	None	

Fig 3.1 Table with sample data

4.IMPLEMENTATION

4.1 IMPLEMENTATION APPROACHES

Credit risk modeling implementation calls for a methodical strategy that includes multiple steps. Here are a few typical methods for credit risk modeling implementation:

Identify the Issue: The first step is to identify the issue you are attempting to resolve. This entails figuring out the business goals you want to accomplish and the credit risk you seek to model.

Data Collection: The following stage is to collect pertinent data. This comprises internal and external data, such as economic statistics, demographic information, and previous loan performance.

After the information has been acquired, it must be prepared for modeling. This covers handling missing data, altering variables, and cleaning up the data.

Model Selection: The following step is to choose a model suited for the current issue. This entails picking a model

appropriate for the data and business objectives, such as logistic regression, decision trees, or neural networks.

Model development is necessary when a model has been chosen. The model must be trained on the data, and its parameters adjusted to achieve the best performance.

Model validation is necessary to ensure the model is correct and trustworthy after its development. This entails putting the model to the test on a hold-out sample of data and contrasting the predictions with the results.

Model Implementation: If the model exhibits good performance on the validation data, the business can use it. This entails incorporating the model into decision-making procedures, such as credit scoring or loan underwriting.

Model monitoring is necessary to ensure the model keeps working well over time once it has been used. This entails

monitoring its effectiveness and periodically reviewing its parameters.

Applying credit risk modeling necessitates a systematic strategy that includes data preparation and collection, model selection and development, model validation and implementation, and performance tracking over time. This strategy can assist in ensuring the model's accuracy, dependability, and effectiveness in reducing credit risk.

4.2 CODING STANDARDS & CONVENTIONS

Coding conventions are style guidelines for programming. They typically cover Naming and declaration rules for variables and functions, Rules for using white space, indentation and comments, and Programming practices and principles. The Coding conventions secure quality: Improves code readability and Makes code maintenance easier. Coding conventions can

be documented rules for teams or your coding practice.

1. **VARIABLE NAMES**

The application uses camelCase for identifier names (variables and functions). All names start with a letter. At the bottom of

this page, you will find a more comprehensive discussion about naming rules.

1. **SPACES AROUND OPERATORS**

Always put spaces around operators (= + - * /) and after commas.

1. **STATEMENT RULES**

General rules for simple statements: Always end a simple statement with a semicolon. General rules for complex (compound) statements: Put the opening bracket at the end of the first line, use one space before the opening bracket, Put the

closing bracket on a new line without leading spaces, and Do not end a complex statement with a semicolon.

1. **OBJECT RULES**

General rules for object definitions:

1. Place the opening bracket on the same line as the object name.
2. Use a colon plus one space between each property and its value.
3. Use quotes around string values, not around numeric values.
4. Do not add a comma after the last property-value pair.
5. Place the closing bracket on a new line without leading spaces.

Furthermore, Always end an object definition with a semicolon.

Department of Computer Science (Christ Deemed to be University)

1. **LINE LENGTH**

For readability, avoid lines longer than 80 characters. If a statement does not fit on one line, the best place to break it is after

an operator or a comma.

1. **NAMING CONVENTIONS**

Always use the same naming convention for all your code. For example, Variable and function names are written as camelCase. Global variables are written in UPPERCASE (We do not, but it's pretty standard), and Constants (like PI) are written in UPPERCASE. Underscores: Many programmers prefer to use underscores (date_of_birth), especially in SQL databases. Underscores are often used in PHP documentation. PascalCase: PascalCase is often preferred by C programmers and camelCase: camelCase is used by

Python libraries. Do not start names with a $ sign. It will put you in conflict with many Python library names.

1. **PERFORMANCE**

Computers do not use coding conventions. Most rules have little impact on the execution of programs. Indentation and extra spaces are not significant in small scripts. For code in development, readability should be preferred. Larger production scripts should be minified.

Enable will make it easier for you and other developers to work on the project over time.

4.2 CODING DETAILS

Some of the coding details are mentioned below.

1. NumPy and Pandas are two examples of Python's many modules for data cleaning and preprocessing. These

   packages can deal with outliers, delete missing values, and format data for modeling.

2. Model Selection: Scikit-learn is only two of the many model selection libraries available in Python. These libraries

   can be used to choose the best models for the given problem and data and

3. Model Fitting:
4. Model Validation:

Python offers tools, including data preparation and cleaning, model choice, model fitting, and model validation. The above examples give a good foundation for using Python in credit risk modeling. However, numerous other libraries and methods can be applied depending on the particular requirements of the issue.

Imports

Data Preparation

connection of backend to postgreSQL

In [1]:
```
! pip install psycopg2-binary
```
```
Requirement already satisfied: psycopg2-binary in c:\users\vignesh\anaconda3\lib\site-packages (2.9.6)
```

In [2]:
```
import psycopg2 as ps
```

In [3]:
```
pip install pandas.io.sql.read_sql
```
```
Note: you may need to restart the kernel to use updated packages.
ERROR: Could not find a version that satisfies the requirement pandas.io.sql.read_sql (from versions: none)
ERROR: No matching distribution found for pandas.io.sql.read_sql
```

In [4]:
```
import pandas.io.sql as sqlio
```

In [5]:
```
conn2=ps.connect(dbname="loan_db",

               user="postgres",password= "Vigvig01",host="localhost",

                port="5432")
```

In [6]:
```
sql=""" SELECT * FROM pg_catalog.pg_tables """
```

In [7]:
```
sql="""SELECT * FROM loan_details """
```

In [8]:
```
loan_data_backup=sqlio.read_sql_query(sql,conn2)

loan_data_backup
```
```
C:\Users\Vignesh\anaconda3\lib\site-packages\pandas\io\sql.py:762: UserWarning: pandas only support SQLAlchemy connectable(e
ngine/connection) ordatabase string URI or sqlite3 DBAPI2 connectionother DBAPI2 objects are not tested, please consider usi
ng SQLAlchemy
  warnings.warn(
```

Out[8]:

	sl_no	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	grade	...	total_bal_il	il_util	open_rv_1
0	0.0	1077501.0	1296599.0	5000.0	5000.0	4975.0	36 months	10.65	162.87	B	...	None	None	N
1	1.0	1077430.0	1314167.0	2500.0	2500.0	2500.0	60 months	15.27	59.83	C	...	None	None	N
2	2.0	1077175.0	1313524.0	2400.0	2400.0	2400.0	36 months	15.96	84.33	C	...	None	None	N
3	3.0	1076863.0	1277178.0	10000.0	10000.0	10000.0	36 months	13.49	339.31	C	...	None	None	N
4	4.0	1075358.0	1311748.0	3000.0	3000.0	3000.0	60 months	12.69	67.79	B	...	None	None	N
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
466280	466280.0	8598660.0	1440975.0	18400.0	18400.0	18400.0	60 months	14.47	432.64	C	...	None	None	N

Import Libraries

In [9]:
```
import numpy as np
import pandas as pd
```

Reading and Pre-processing

Explore Data

In [12]:
```
pd.options.display.max_columns = None
#pd.options.display.max_rows = None
# Sets the pandas dataframe options to display all columns/ rows.
```

In [13]:
```
loan_data
```

Out[13]:

	sl_no	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	grade	sub_grade	
0	0.0	1077501.0	1296599.0	5000.0	5000.0	4975.0	36 months	10.65	162.87	B	B2	
1	1.0	1077430.0	1314167.0	2500.0	2500.0	2500.0	60 months	15.27	59.83	C	C4	
2	2.0	1077175.0	1313524.0	2400.0	2400.0	2400.0	36 months	15.96	84.33	C	C5	
3	3.0	1076863.0	1277178.0	10000.0	10000.0	10000.0	36 months	13.49	339.31	C	C1	RES
4	4.0	1075358.0	1311748.0	3000.0	3000.0	3000.0	60 months	12.69	67.79	B	B5	
...	...	...	...	...	...	...	...	...	...	...	...	
466280	466280.0	8598660.0	1440975.0	18400.0	18400.0	18400.0	60 months	14.47	432.64	C	C2	
466281	466281.0	9684700.0	11536848.0	22000.0	22000.0	22000.0	60 months	19.97	582.50	D	D5	Inte (Par

Preprocessing few continuous variables

In [18]:
```
loan_data['emp_length'].unique()
# Displays unique values of a column.
```

Out[18]:
```
array(['10+ years', '< 1 year', '1 year', '3 years', '8 years', '9 years',
       '4 years', '5 years', '6 years', '2 years', '7 years', None],
      dtype=object)
```

In [19]:
```
loan_data['emp_length_int'] = loan_data['emp_length'].str.replace('\+ years', '')
loan_data['emp_length_int'] = loan_data['emp_length_int'].str.replace('< 1 year', str(0))
loan_data['emp_length_int'] = loan_data['emp_length_int'].str.replace('n/a',  str(0))
loan_data['emp_length_int'] = loan_data['emp_length_int'].str.replace(' years', '')
loan_data['emp_length_int'] = loan_data['emp_length_int'].str.replace(' year', '')
# We store the preprocessed 'employment length' variable in a new variable called 'employment length int',
# We assign the new 'employment length int' to be equal to the 'employment length' variable with the string '+ y
# replaced with nothing. Next, we replace the whole string 'less than 1 year' with the string '0'.
# Then, we replace the 'n/a' string with the string '0'. Then, we replace the string 'space years' with nothing.
# Finally, we replace the string 'space year' with nothing.
```

```
C:\Users\Vignesh\AppData\Local\Temp\ipykernel_4216\3553965977.py:1: FutureWarning: The default value of regex wi
om True to False in a future version.
  loan_data['emp_length_int'] = loan_data['emp_length'].str.replace('\+ years', '')
```

Preprocessing few discrete variables

In [41]:
```
loan_data.info()
# Displays column names, complete (non-missing) cases per column,
# and datatype per column.
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 466285 entries, 0 to 466284
Data columns (total 80 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   sl_no                        466285 non-null  float64
 1   id                           466285 non-null  float64
 2   member_id                    466285 non-null  float64
 3   loan_amnt                    466285 non-null  float64
 4   funded_amnt                  466285 non-null  float64
 5   funded_amnt_inv              466285 non-null  float64
 6   term                         466285 non-null  object
 7   int_rate                     466285 non-null  float64
 8   installment                  466285 non-null  float64
 9   grade                        466285 non-null  object
 10  sub_grade                    466285 non-null  object
 11  emp_title                    438697 non-null  object
 12  emp_length                   445277 non-null  object
 13  home_ownership               466285 non-null  object
 14  annual_inc                   466281 non-null  float64
 15  verification_status          466285 non-null  object
 16  issue_d                      466285 non-null  object
 17  loan_status                  466285 non-null  object
 18  pymnt_plan                   466285 non-null  object
 19  url                          466285 non-null  object
```

Statistical Analysis

4.2 SCREENSHOTS

Screenshots of some of the Analysis is presented below

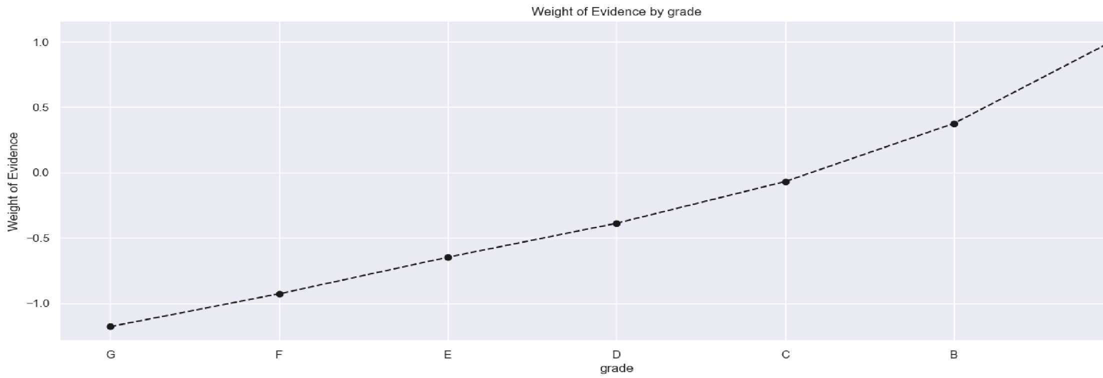


Fig 4.1 Analysis Graph-1

Fig 4.2 Analysis Graph-2

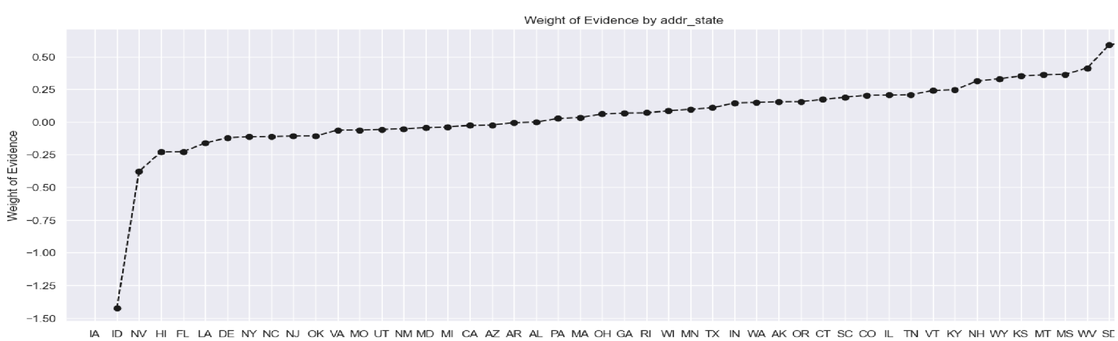


Fig 4.3 Analysis Graph-3

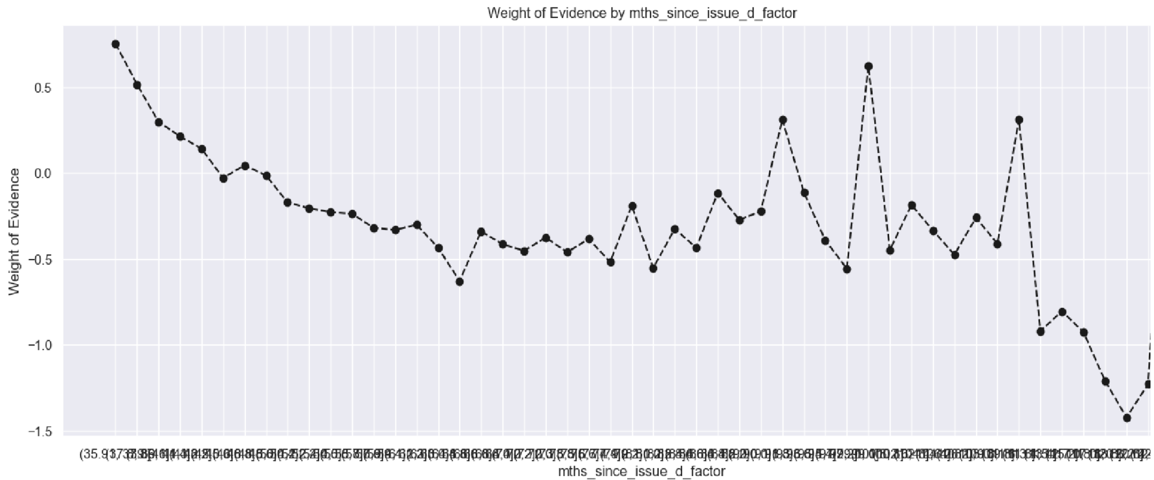


Fig 4.3 Analysis Graph-4

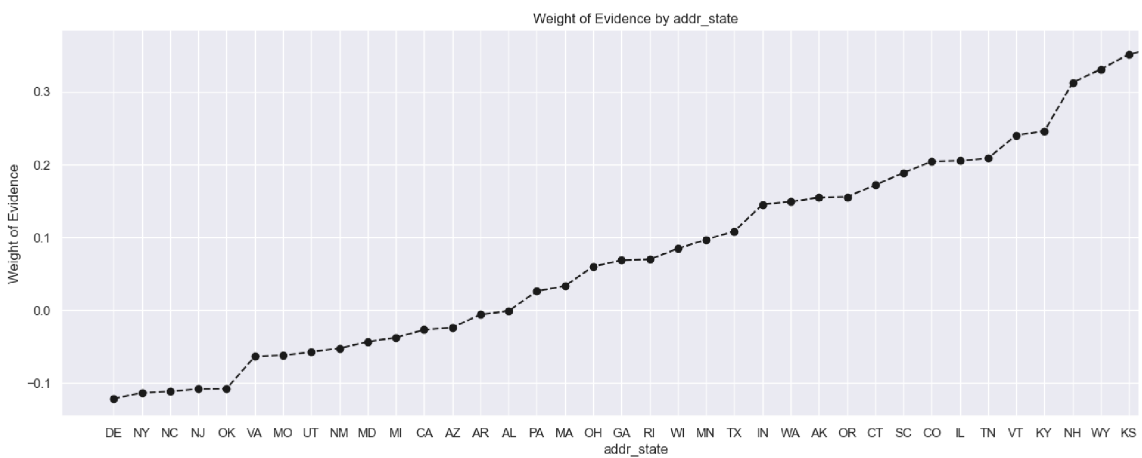


Fig 4.6 Analysis Graph-5

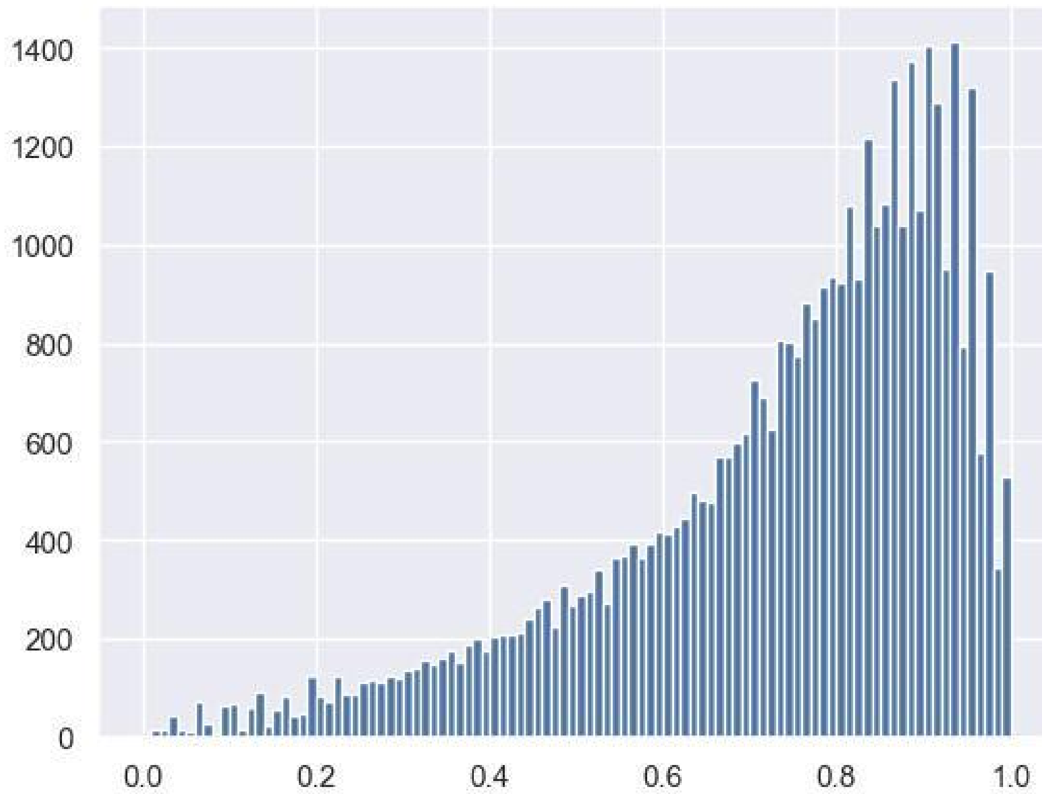


Fig 4.8 Analysis Graph-6

Solution Description

In general, when giving credit, a creditor, also known as a lender, provides goods and services to a debtor, who can be referred to as a borrower, based on the trust that the borrower will repay the lender at some point and the future in exchange for providing the goods or services to the borrower. In the present, the lender receives a payment interest.

The most common type of loan provided by lenders to borrowers is money. Credit cards and home ownership loans are two outstanding examples of credit lenders provide.

So when a bank gives you a credit card, you can spend as much money as you want up to a specific limit called a credit limit. This money is not yours, and you are required to repay the respective sum to the bank. Typically, they have to repay with interest, which is how the bank makes a profit. When someone signs a homeownership loan, the borrowed money from a bank is against the equity in a residence that the person wants to acquire; the residence covers the debt in case the person fails to repay.

The bank has the right to take ownership of the residence, sell it and satisfy the remaining debt. Another example of credit is asset financing that commercial banks offer to companies; firms use asset financing to obtain the equipment they need, usually paying a regular charge for utilizing it over an agreed period instead of paying the total price of the equipment upfront.

The likelihood that a borrower would not repay their loan to the lender is called credit risk. In this case, the lender would not receive the owed principal. Moreover, they would not be paid the interest and suffer a substantial loss.

In addition, the lender will likely have to sustain substantial costs to recover outstanding debt. These costs are called collection costs. The event of a borrower being unable to make the required payments to repay their debt is called default. To protect themselves against borrower defaults, lenders must assess the credit risk associated with each borrower very well. One way for lenders to diminish losses due to borrower defaults is to require collateral that would cover the outstanding debt.

Another way for the lender to increase the price of lending the funds is the interest rate for borrowers with high credit risk, and this is commonly known as risk-based pricing.

No matter which strategy the borrower decides to pursue to diminish losses due, the borrower defaults. What is most important is to be able to estimate the credit risk of each borrower precisely as possible.

Let us take a few moments to provide a noteworthy example from the recent past. It is essential to know that lenders' inability or failure to estimate borrowers' probability of default can have grave consequences for lenders and society.

Lending to borrowers with a high probability of default is one of the main reasons for severe financial crises, such as the global financial crisis in 2008.

It was the most severe financial crisis since the Times of the Great Depression in the 1930s. The main factor that led to the 2008 financial crisis was the high default rates of subprime home mortgages

in the United States. First, low-interest rates encourage financial institution lenders to increase mortgage lending. Banks were willing to finance 100 percent or more of the value of a new home. High mortgage approval rates increased the demand for homes, which led to increased housing prices because the value of homes had increased.

Many homeowners borrowed money and used their homes as a guarantee to the bank.

However, many of them were with relatively high credit risk subprime and, at some point, could not continue repaying their debt and defaulted—consequently, financial instruments based on mortgages, such as mortgage-backed securities, lost value.

Thus, big banks holding these instruments absorbed huge losses.

Some of them, like Lehman Brothers and Bear Stearns, went bankrupt, while the government bailed out many others.The following assumptions are made to prove that the model is a good fit. The model is linear in its regression parameters (β)

The model's residuals follow a normal distribution, and the expected value (mean for the loss calculation) of the residuals is zero. Time series data are assumed to be correlated. The variance of the residuals is constant to test the homoscedasticity.

The variables have no high correlation, and the model is stable, As proved in the last graph.

Estimating the Accuracy of the Model

In [54]:
```
tr = 0.5
# We create a new column with an indicator,
# where every observation that has predicted probability greater than the threshold has a value of 1,
# and every observation that has predicted probability lower than the threshold has a value of 0.
df_actual_predicted_probs['y_hat_test_lgd_stage_1'] = np.where(df_actual_predicted_probs['y_hat_test_proba_lgd_stage_1'] > tr
```

In [55]:
```
pd.crosstab(df_actual_predicted_probs['lgd_targets_stage_1_test'], df_actual_predicted_probs['y_hat_test_lgd_stage_1'], rowna
# Creates a cross-table where the actual values are displayed by rows and the predicted values by columns.
# This table is known as a Confusion Matrix.
```

Out[55]:

Predicted	0	1
Actual		
0	993	2769
1	687	4199

In [56]:
```
pd.crosstab(df_actual_predicted_probs['lgd_targets_stage_1_test'], df_actual_predicted_probs['y_hat_test_lgd_stage_1'], rowna
# Here we divide each value of the table by the total number of observations,
# thus getting percentages, or, rates.
```

Out[56]:

Predicted Actual	0	1
0	0.114824	0.320190
1	0.079440	0.485546

```
In [57]: (pd.crosstab(df_actual_predicted_probs['lgd_targets_stage_1_test'], df_actual_predicted_probs['y_hat_test_lgd_stage_1'], rowr
         # Here we calculate Accuracy of the model, which is the sum of the diagonal rates.

Out[57]: 0.6003700277520814

In [58]: from sklearn.metrics import roc_curve, roc_auc_score

In [59]: fpr, tpr, thresholds = roc_curve(df_actual_predicted_probs['lgd_targets_stage_1_test'], df_actual_predicted_probs['y_hat_test
         # Returns the Receiver Operating Characteristic (ROC) Curve from a set of actual values and their predicted probabilities.
         # As a result, we get three arrays: the false positive rates, the true positive rates, and the thresholds.
         # we store each of the three arrays in a separate variable.

In [60]: plt.plot(fpr, tpr)
         # We plot the false positive rate along the x-axis and the true positive rate along the y-axis,
         # thus plotting the ROC curve.
         plt.plot(fpr, fpr, linestyle = '--', color = 'k')
         # We plot a seconary diagonal line, with dashed line style and black color.
         plt.xlabel('False positive rate')
         # We name the x-axis "False positive rate".
         plt.ylabel('True positive rate')
         # We name the x-axis "True positive rate".
         plt.title('ROC curve')
         # We name the graph "ROC curve".

Out[60]: Text(0.5, 1.0, 'ROC curve')
```

Fig 4.9 Solution Figure-1

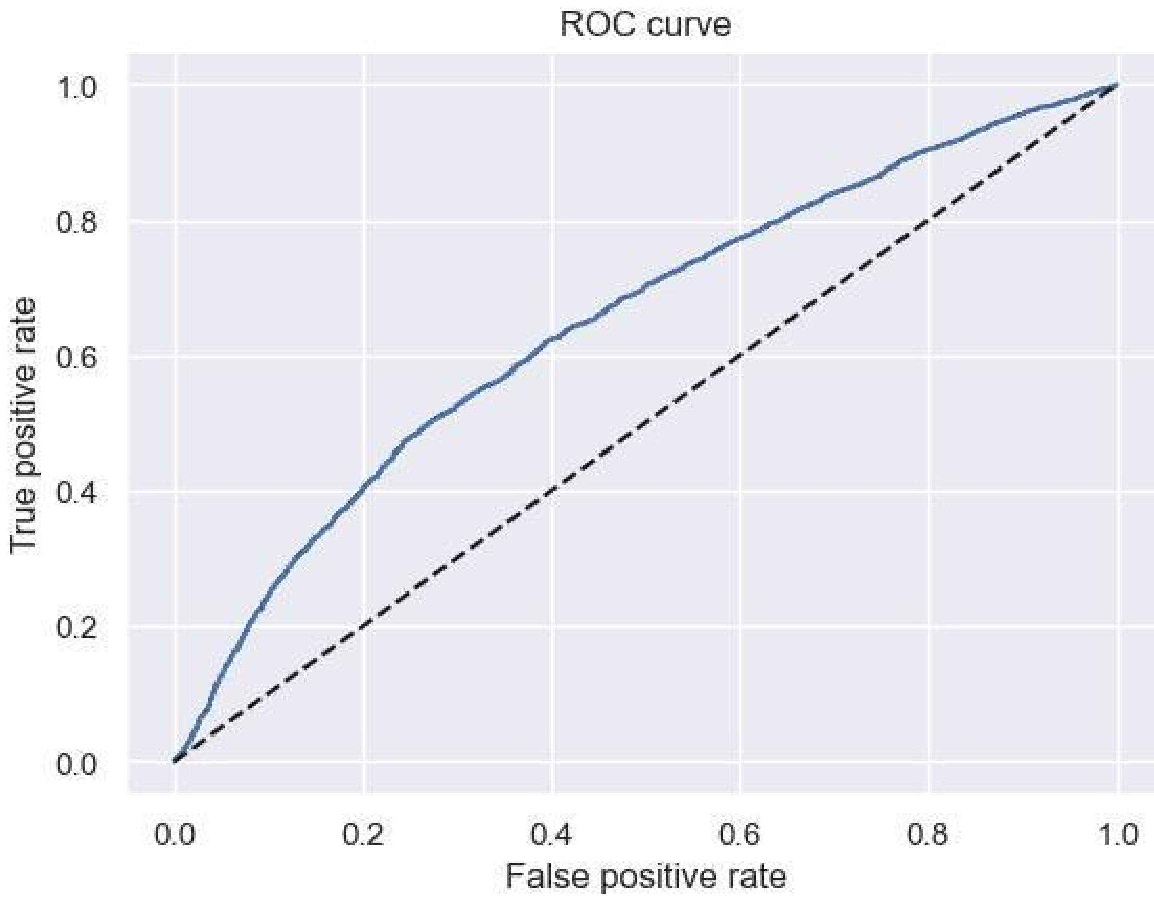


Fig 4.10 Solution Figure-2

5.TESTING

5.1 TEST CASES

1. **Test Cases for Data Quality:** This group of test cases assesses the reliability and validity of the data used in the credit risk model. These checks could involve looking for outliers, inconsistent data, and missing data.

2. **Test Cases for Model Calibration:** This group of test cases assesses the model's calibration. These tests entail comparing the probabilities of Default projected by the model to the default rates seen in historical data.

3. **Test Cases for Model Validation**: This group of test cases assesses the model's performance in various validation circumstances. These tests involve assessing the model's performance using data not from the sample or various periods.

TEST REPORT

Test report for credit risk modeling. This report aims to assess the model's accuracy and dependability and ensure it complies with the institution's and regulatory authorities' standards. The following are some essential elements that must be present in a test report for credit risk modeling:

Procedures for Testing: All facets of the model, including data quality, model choice, validation,and regulatory compliance, should be tested thoroughly.

Results: The findings from the testing processes should be presented in this section. The outcomes should thoroughly evaluate the model's precision and dependability and any potential areas for model development.

In conclusion, a test report for credit risk modeling is a vital record that assesses the precision and dependability of the credit risk model. The report must be thorough and address every model component, such as the choice of the model, validation, and legal compliance. The report should include suggestions for enhancing the model in light of the findings of the testing procedures. Financial organizations can efficiently manage credit risk and adhere to regulatory standards with a well-written test report.

6.CONCLUSIONS

6.1 DESIGN AND IMPLEMENTATION ISSUES

Design and implementation issues are critical considerations in credit risk modeling. Below are some of the basic design and implementation issues that should be addressed:

Data Quality: For credit risk modeling, data quality is crucial. The accuracy and quality of the data utilized in the modeling process can significantly affect how well the model performs. Therefore, it is essential to guarantee that the data is accurate, timely, and complete.

Model Choice: There are many different types of complicated credit risk modelsavailable ,including structural, reduced-form, and machine learning-based models. It is choosing the suitable model that suits the institution's requirements and the portfolio's properties.

Model Validation: Model validation is an essential phase in the process of modeling credit risk. Validation ensures that the model operates as anticipated and that its outputs are trustworthy. It is essential to validate the model and change it as required continuously.

Portfolio Make-Up: The portfolio's make-up can significantly affect how credit risk is modeled. When creating a credit risk model, it is crucial to consider the portfolio's diversification and concentration risk.

Economic Context: Credit risk modeling can be impacted by the economic context. When creating a credit risk model, it is critical to consider economic cycles, interest rates, and inflation.

Regulatory regulations: When calculating credit risk, regulatory regulations are crucial. It is crucial to make sure the model conforms with legal standards. In order to accurately predict credit risk, transparency and explain ability are essential. Models should be transparent, and stakeholders should know the model's assumptions and constraints.

6.2 ADVANTAGES

- Greater Accuracy: Compared to conventional methods, data science techniques are more accurate and efficient at analyzing vast amounts of data. As a result, credit risk projections become more precise.
- Early Risk Detection: By identifying borrowers most likely to default early in the credit cycle, data science-based credit risk models can help lenders take corrective action before the problem worsens.
- Customization: By using data processing techniques, lenders can create credit risk models that are specifically suited to the requirements of their business, producing more accurate risk assessments.
- Real-time Updates: Data science makes it possible to monitor a borrower's credit risk continuously and to update the models as soon as new information becomes available.

6.3 LIMITATIONS

1. Limitations of Data Science Credit Risk Modeling:
2. Data Availability: The accuracy of the credit risk models is greatly influenced by the quality and amount of the data utilized in the development. There might only sometimes be enough pertinent data to allow for precise modeling.
3. Complex Modeling: Credit risk modeling calls for sophisticated statistical and machine learning techniques, which can be challenging to implement.
4. Excessive Reliance on Historical Data: Credit risk models frequently rely on historical data, which could not indicate future patterns or events.
5. Ethical Issues: The application approaches in credit risk modeling raise ethical issues, such as the possibility of bias or discrimination against particular borrower groups.

6.4 FUTURE ENHANCEMENTS

Credit risk modeling has a bright future because of data science, machine learning, and artificial intelligence developments.

Big Data: The accessibility of substantial volumes of data from numerous sources, including social media, internet searches, and transaction data, may result in more thorough credit risk models that can include the complete financial profile of a borrower.

Artificial intelligence (AI) and machine learning algorithms: can assist lenders in automating the credit risk assessment process, resulting in a more rapid, accurate, and objective decision-making process.

Real-time Monitoring: By tracking borrower activity and credit ratings in real time, lenders may be better able to predict and respond to credit risk, leading to a more proactive approach to risk management.

Blockchain: By utilizing blockchain technology, a credit scoring system that is more transparent and secure might be created, lowering the possibility of fraud and enhancing the accuracy of credit risk models.

Alternative Credit Scoring: Including non-financial data and other alternative credit scoring techniques may result in more inclusive credit scoring models that make credit available to borrowers without standard credit histories.

Explainable AI techniques: could be used to find and reduce potential biases in the models as there is a greater emphasis on moral and open credit risk modeling.

7. REFERENCES

1. Baesens, B. 2003. "Developing Intelligent Systems for Credit Scoring Using Machine Learning
Techniques." Ph.D. diss., K.U.Leuven, Belgium.

2. Cespedes, Credit Risk Modeling and Basel II, *Algo Research Quarterly* 5(no. 1): Spring 2002. Dwyer D.W., The Moody's KMV EDF RiskCalc. 3.1: Model Next-Generation Technology for Predicting Private Firm Credit Risk. Moody's KMV. 2004.

3. Engelmann, Hayden, and Tasche. "Measuring the Discriminative Power of Rating Systems." Deutsche Bundesbank. 2003.

4. Hand, D. and R. J. Till. "A Simple Generalization of the Area under the ROC Curve to Multiple Class Classification Problems." *Machine Learning* 45, no.2(2001):171–86

5. Benzschawel, T., T. Wu and A. Haroon, 2012, "Simulating Correlated Default Rates and Recovery Values", *Journal of Credit Risk* (under review).