

**TRIBHUVAN UNIVERSITY**

Faculty of Management

**National College of Computer Studies**

Paknajol, Kathmandu

**Project Report**

**Python Project – Machine Learning**

**Credit Score Classification**

**Submitted By:** **Submitted To:**

Name: Bipin Shrestha Mr. Mausam Rajbanshi

BIM-5th Semester “A”

Roll no.:03

**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the dissertation entitled **“Credit Score Classification”** in partial fulfillment of the requirements of the Degree of Bachelor of Information Management.

It is an authentic record of my own work carried out during a period from  **June 2023**  under the supervision of  **Mr.** **Mausam Rajbanshi** Professor.

The matter presented in this dissertation has not been submitted by me for the award of any other degree of this or any other Institute/University.

**Bipin Shrestha**

**BIM 5th A**

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

**Acknowledgment**

In this credit score classification analysis, Python served as the primary programming language, while Scikit-Learn enabled machine learning model development. Pandas and NumPy were crucial for data preprocessing, and Matplotlib and Seaborn aided data visualization. We extend our gratitude to the source of the credit score dataset and acknowledge Jupyter Notebooks and GitHub for facilitating collaboration and version control. Lastly, OpenAI's GPT-3 provided valuable assistance throughout the report, collectively contributing to the success of our analysis.

The credit score dataset, a core component of this analysis, deserves acknowledgment for its contribution to our research. We extend our appreciation to the data source or organization responsible for making this valuable dataset accessible for research purposes. Jupyter Notebooks provided an interactive and collaborative platform for conducting the analysis, documenting the process, and sharing insights. Additionally, GitHub served as a robust version control and collaboration tool, facilitating the development and sharing of code and analysis scripts among team members.

**Table of** **Contents**

[**Introduction to Machine Learning** 1](#_Toc145154373)

[**Machine Learning in Python** 2](#_Toc145154374)

[**Background** 3](#_Toc145154375)

[**Objective** 4](#_Toc145154376)

[**Implementation** 5](#_Toc145154377)

[**Conclusion** 12](#_Toc145154378)

[**References** 13](#_Toc145154379)

# **Introduction to Machine Learning**

The study of algorithms and statistical models that enable computer systems to acquire new skills and enhance their performance on a given task without being explicitly taught is known as machine learning, which is a branch of artificial intelligence. It basically involves educating computers to learn from data, and then have them predict or decide based on that learning.

Here are the key machine learning components and concepts:

* Data: Data is the basis of machine learning. Machine learning models are taught patterns and relationships using data. It is possible for this data to be structured (like tables and databases) or unstructured (like text, photos, and audio). Performance of a machine learning model is frequently significantly impacted by the quantity and quality of data.
* Training: To train a machine learning model, a dataset with input data (features) and matching target values or labels is presented to the model. The model gains the ability to spot patterns or relationships in data by modifying its internal parameters.
* Features: The variables or properties that are utilized to characterize the data are referred to as features. Choosing, modifying, or developing useful characteristics that aid a model's ability to learn and produce precise predictions is referred to as feature engineering.
* Model: A mathematical representation of data patterns or correlations is referred to as a machine learning model. Different models are employed for different tasks, including neural networks for complicated tasks like image recognition, decision trees for classification, and linear regression for regressive situations.
* Algorithm: To train models, machine learning algorithms use mathematical and computational methods. These methods vary depending on the job at hand and the style of learning (supervised, unsupervised, or reinforced).
* Evaluation: Depending on the job, different metrics and methodologies, such as accuracy, precision, recall, F1-score, or mean squared error, are used to assess machine learning models.
* Deployment: After training and evaluation, machine learning models can be used in real-world applications to produce predictions or automate decision-making procedures.
* Continuous Learning: As new data becomes available, machine learning models can be updated and enhanced. Online learning, often referred to as retraining, is essential for preserving model accuracy over time.

Natural language processing (language translation and sentiment analysis), finance (fraud detection and stock prediction), healthcare (diagnosis and prognosis), image recognition (object detection and facial recognition), and many other fields make extensive use of machine learning. It still plays a significant part in developing technologies and resolving challenging issues.

**Machine Learning in Python**

Python is a popular data science programming language due to its simplicity, versatility, and a rich ecosystem of libraries and tools designed specifically for data analysis and machine learning. Here's a quick rundown of how Python is used in data science:

* Data Manipulation and Analysis:
* Numpy: Numpy supports arrays and matrices, making it simple to perform numerical operations on large datasets.
* Pandas: Pandas is a powerful data manipulation and analysis library. It provides data structures such as Data Frames for dealing with structured data.
* Data Visualization:
* Matplotlib: Matplotlib is a popular library for creating static, animated, and interactive plots and charts.
* Seaborn: Seaborn is based on Matplotlib and provides a higher-level interface for creating visually appealing statistical graphics.
* Machine Learning:
* Scikit-learn: Scikit-learn is a large machine learning library that includes tools for classification, regression, clustering, and model selection.

Python's rich library ecosystem, combined with its readability and ease of use, makes it an excellent choice for data scientists looking to effectively explore, analyze, and model data. Depending on the task, you can use a variety of libraries and tools to achieve your data science objectives.

**Background**

The background for a credit score classification analysis using Python are:

* Significance of Credit Scores: Credit scores play a crucial role in assessing an individual's creditworthiness and are vital for responsible lending and risk management in the financial industry.
* Opportunities in Data Analysis: With advancements in data analytics and machine learning, there's an opportunity to enhance credit score classification through advanced techniques.
* Python as a Powerful Tool: Python, a versatile programming language, offers a robust ecosystem of libraries and tools for data analysis, making it an ideal choice for this task.
* Objective of the Analysis: The primary goal is to leverage Python to create predictive models capable of categorizing credit applicants into different risk categories.
* Enhancing Accuracy: By utilizing data-driven methods and machine learning algorithms, we aim to improve the accuracy and efficiency of credit score classification.
* Benefits for Financial Institutions: Accurate credit score classification aids financial institutions in making informed lending decisions, reducing risks, and optimizing their operations.
* Fair Access to Credit: Ethical considerations are essential to ensure fair and non-discriminatory lending practices.
* Overall Objective: This analysis seeks to combine Python's capabilities with data analytics to contribute to the advancement of effective credit score classification, benefiting both lenders and borrowers.

**Objective**

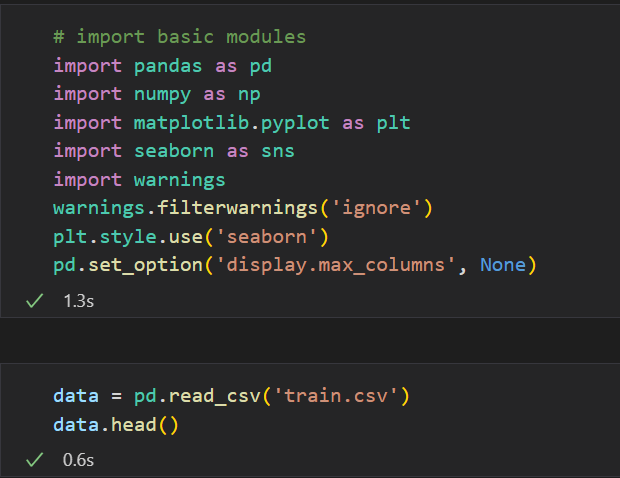
The objective of a credit score classification analysis performed in Python cover a broad spectrum of techniques for efficiently evaluating and managing credit risk. First, the investigation seeks to create reliable machine learning models that forecast credit ratings or creditworthiness by utilizing Python's capabilities. To guarantee that the data is appropriate for modeling, careful data preprocessing is required, including addressing missing data and feature engineering. The main objective is to develop precise models that can classify persons or companies into various credit risk categories, assisting in the making of well-informed decisions.

Beyond model development, the analysis involves the formulation of risk assessment guidelines, which serve as a framework for classifying applicants as low, moderate, or high credit risks based on model predictions. Ethical considerations are paramount, ensuring fairness and non-discrimination in the credit scoring process, while regulatory compliance is essential to adhere to relevant laws and regulations.

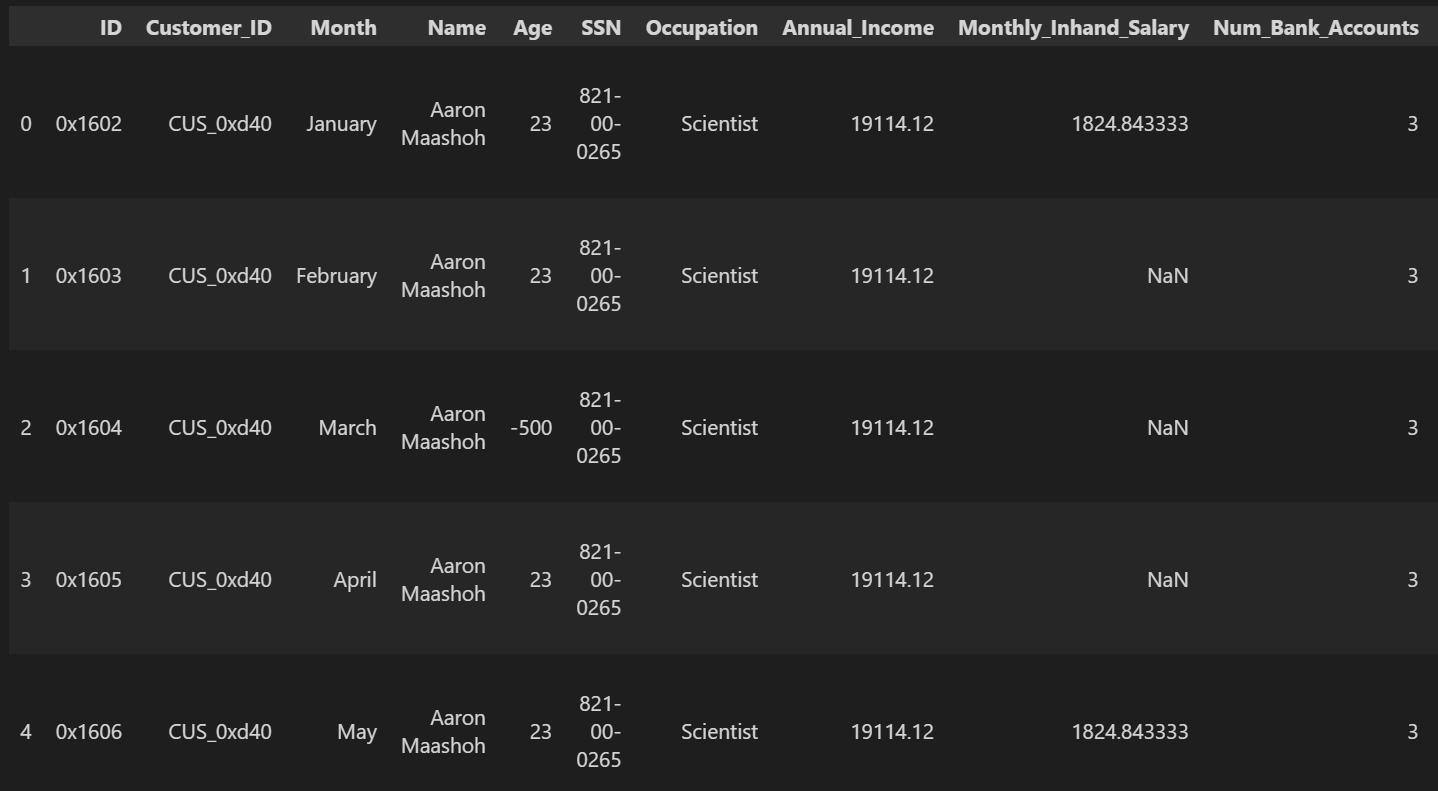
Lastly, clear and comprehensive documentation is crucial to present the analysis findings, model performance, and recommendations to stakeholders and decision-makers effectively. These objectives collectively contribute to the successful execution of a credit score classification analysis using Python, providing valuable insights for credit risk management.

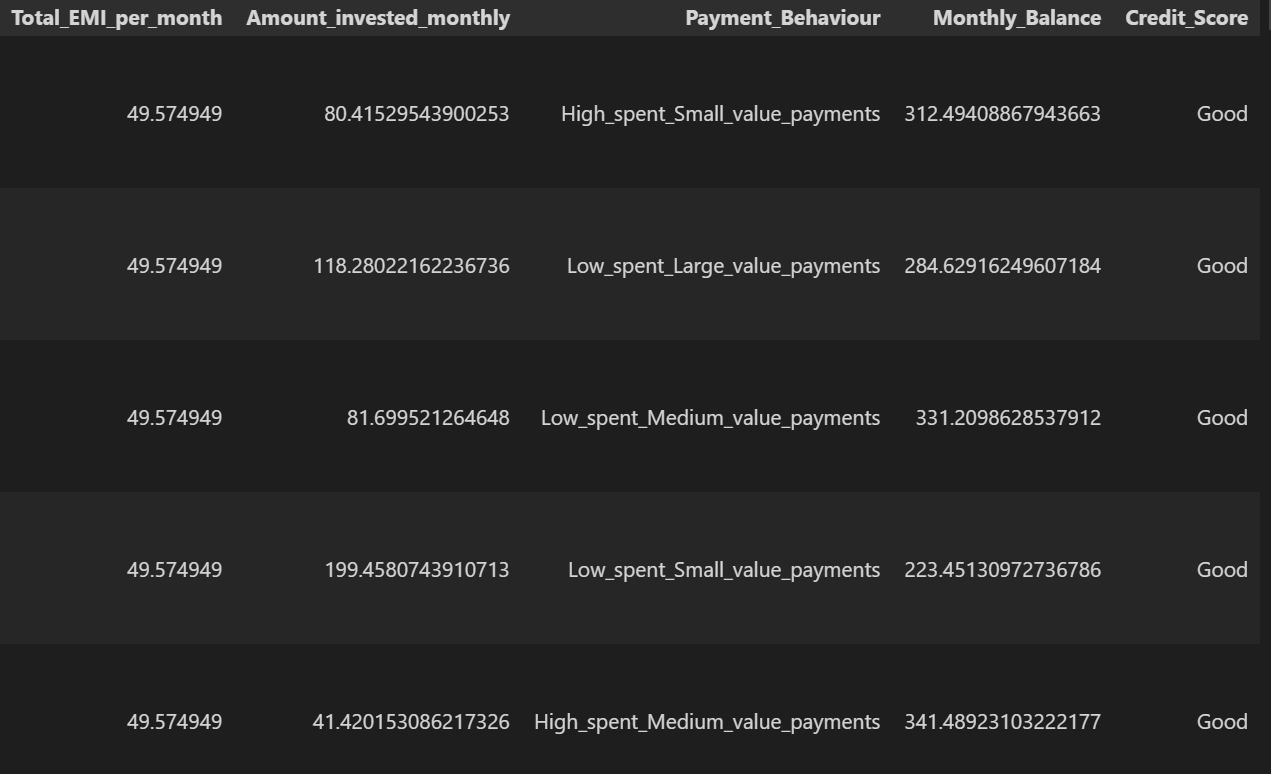
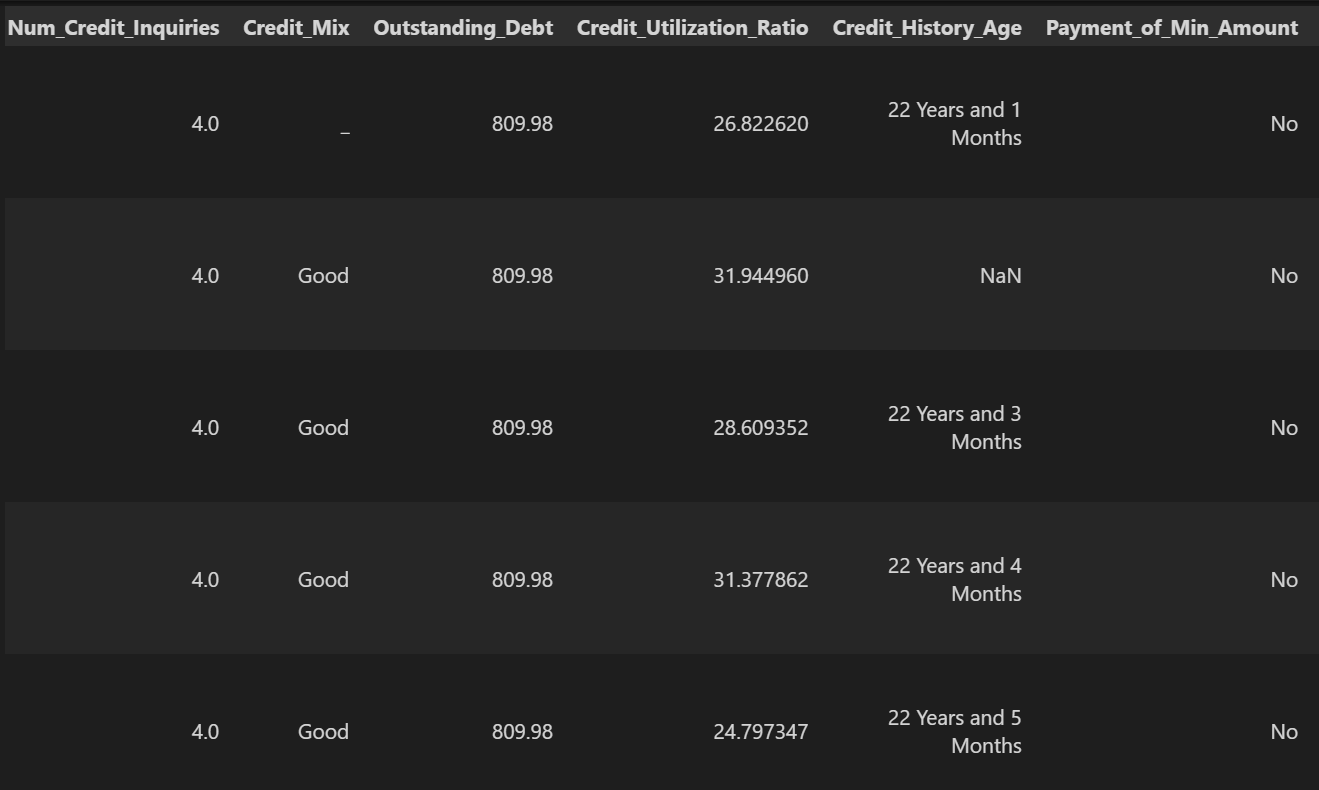
**Implementation**

#import models

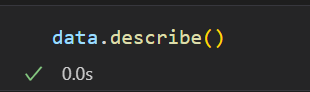


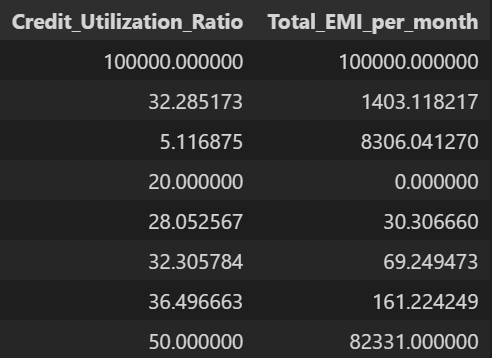
#Dataset



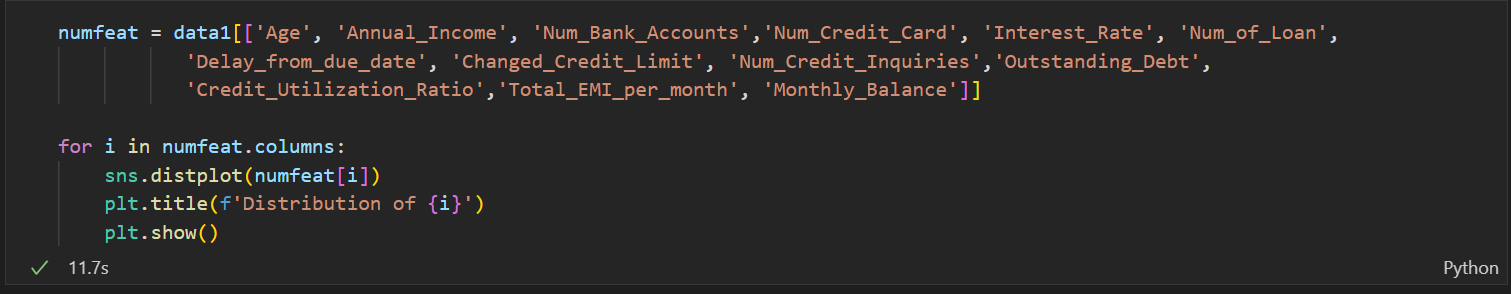


#Data describe

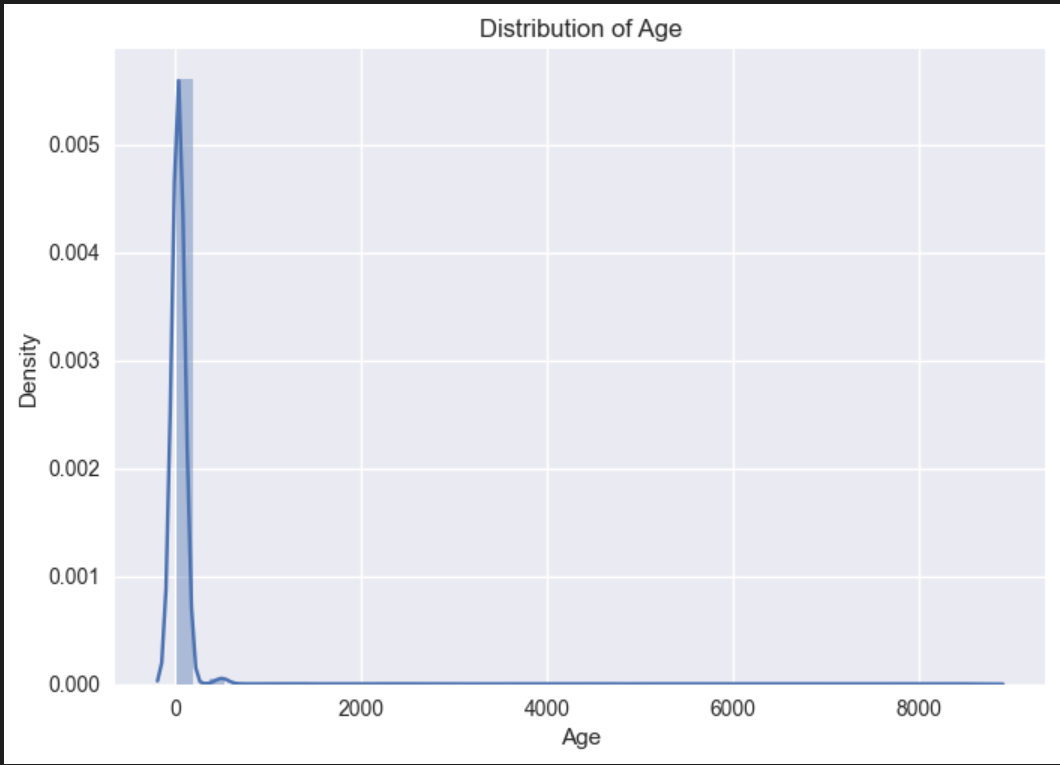
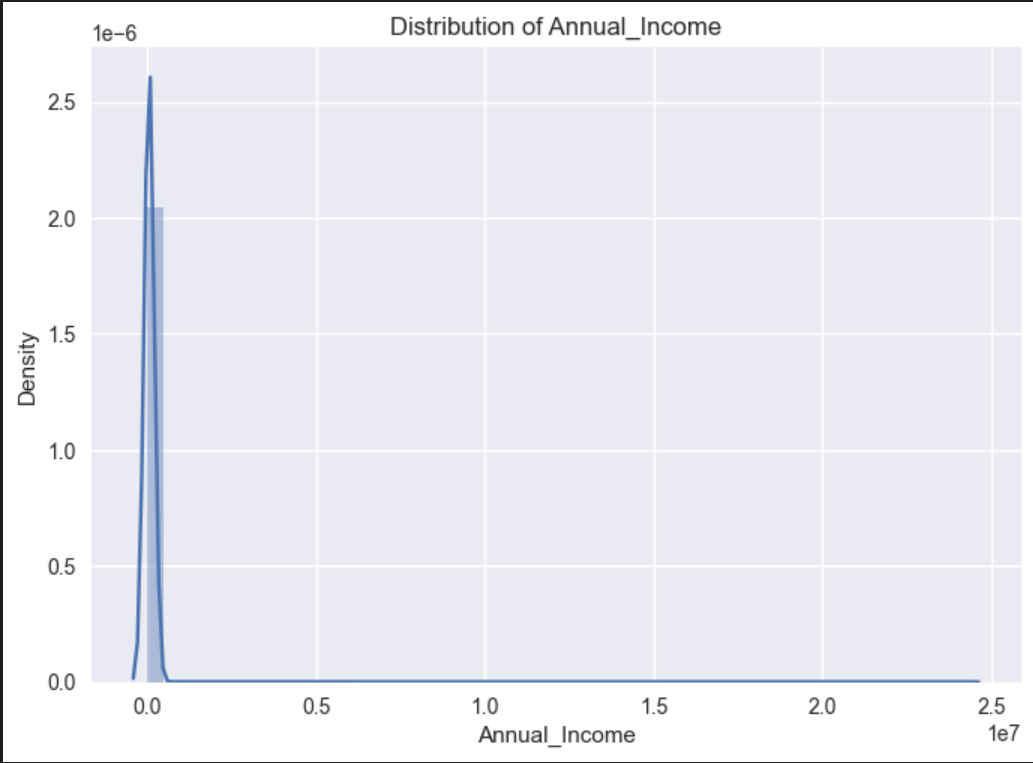




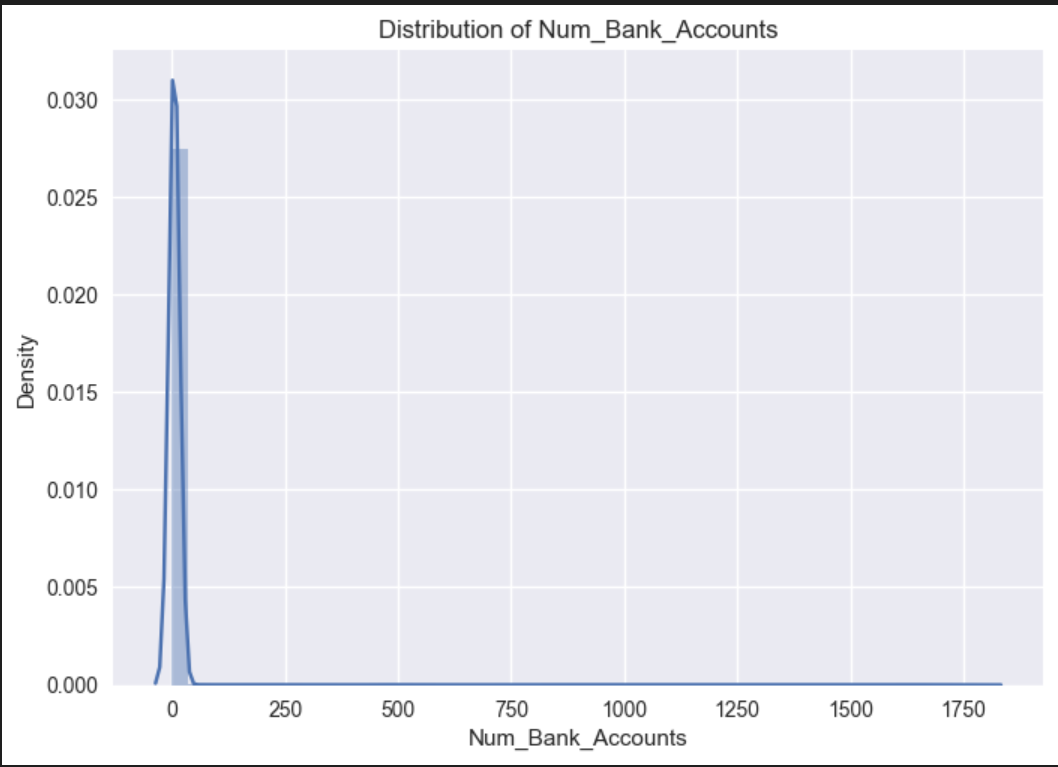
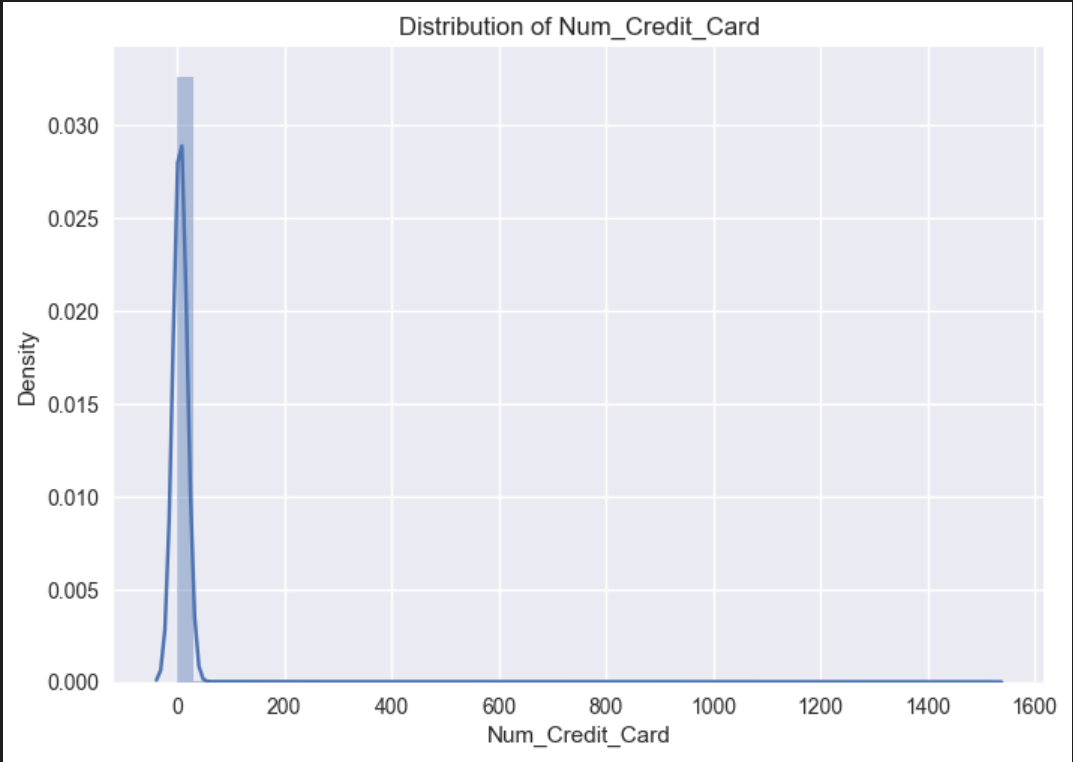
#Histogram



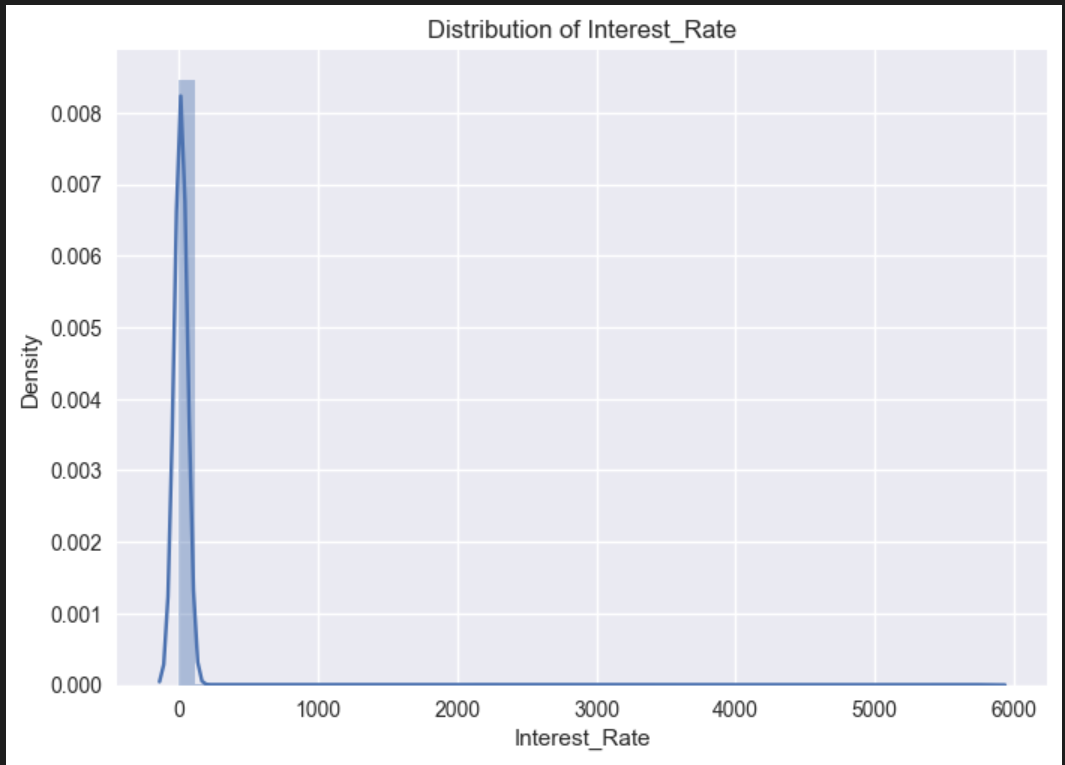
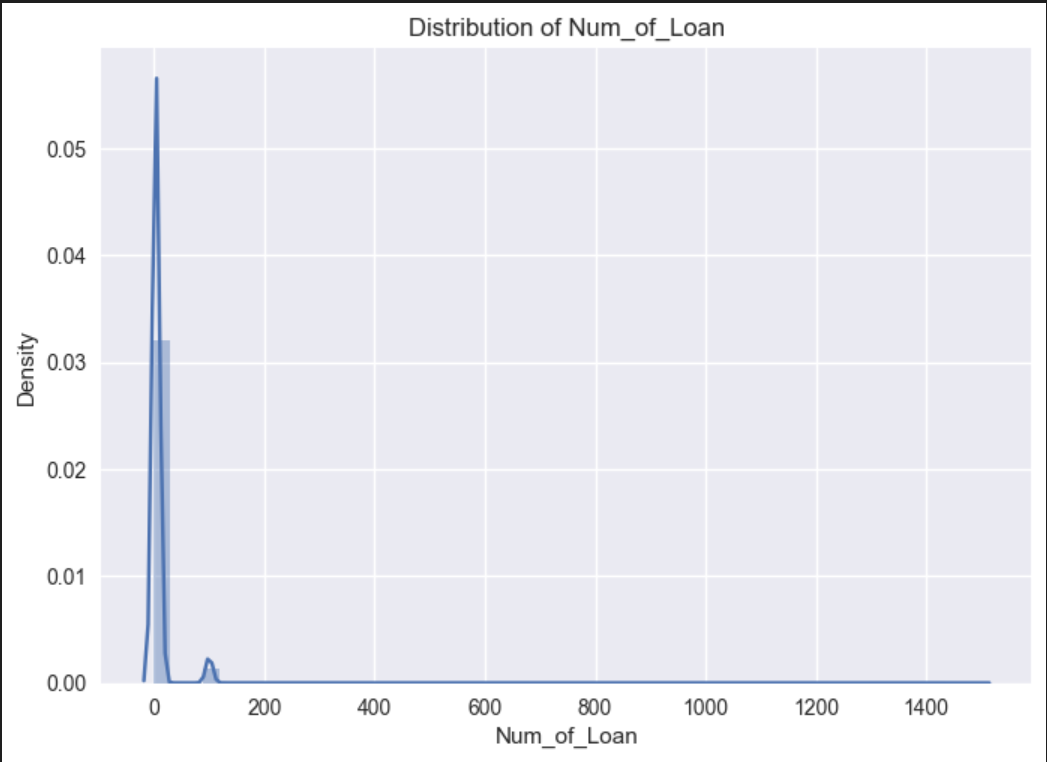
Distribution of Age: Distribution of Annual Income:

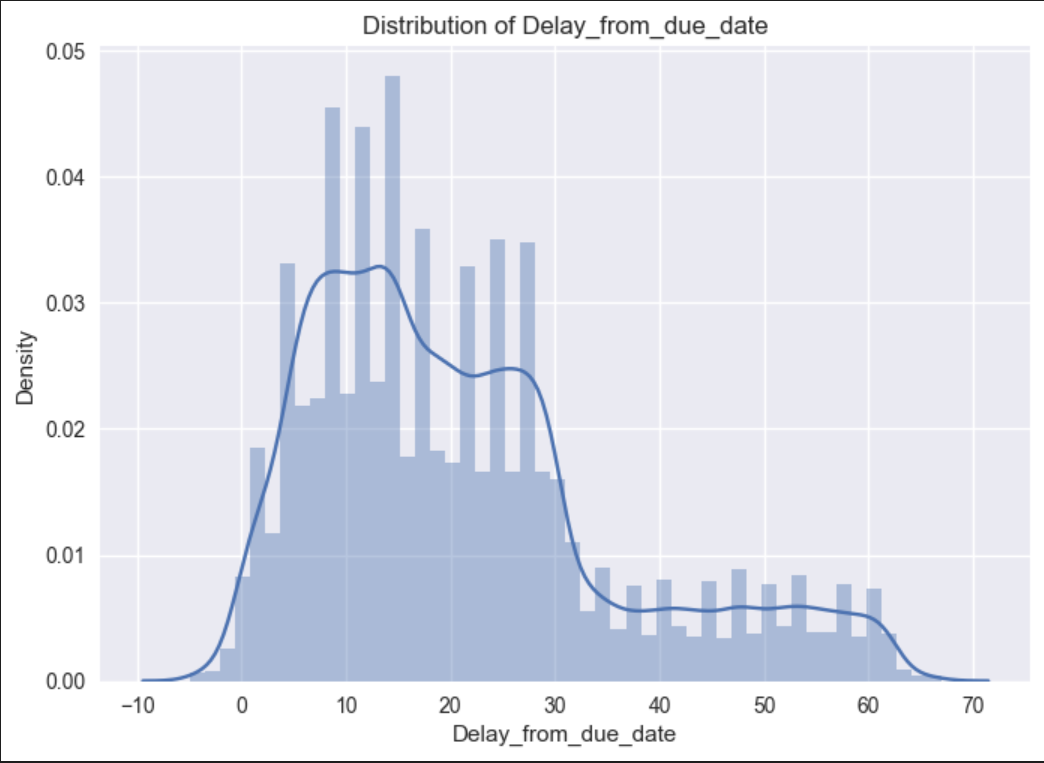
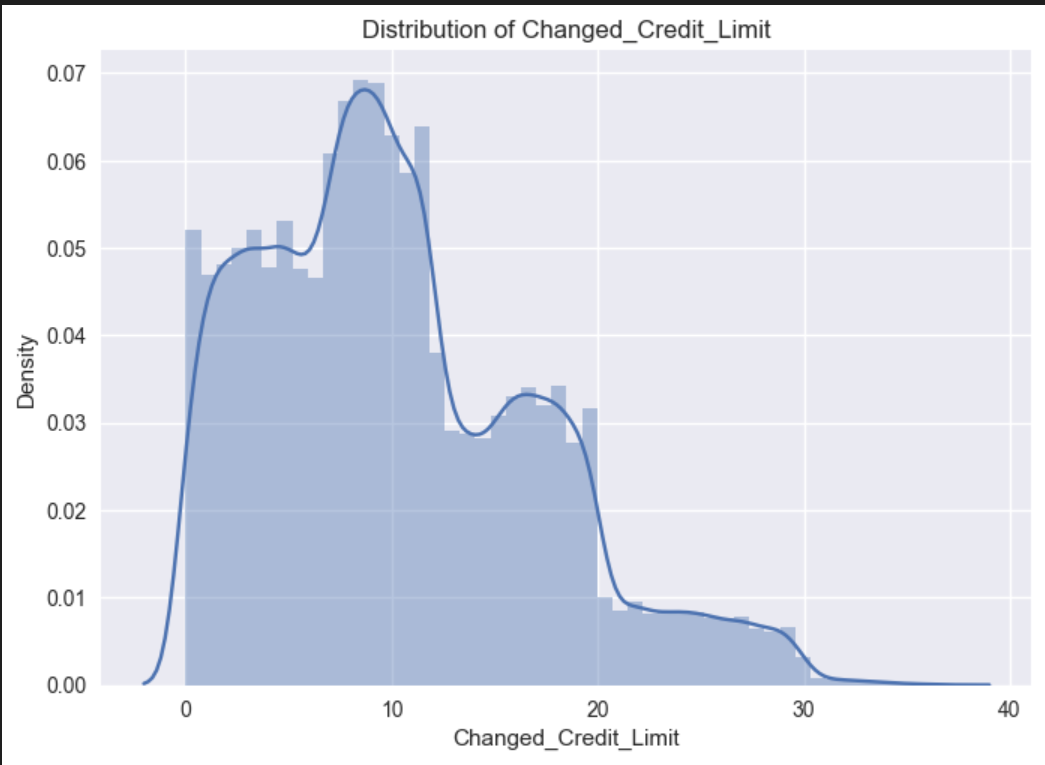
Distribution of Num\_Bank\_Accounts: Distribution of Num\_Credit\_Card:

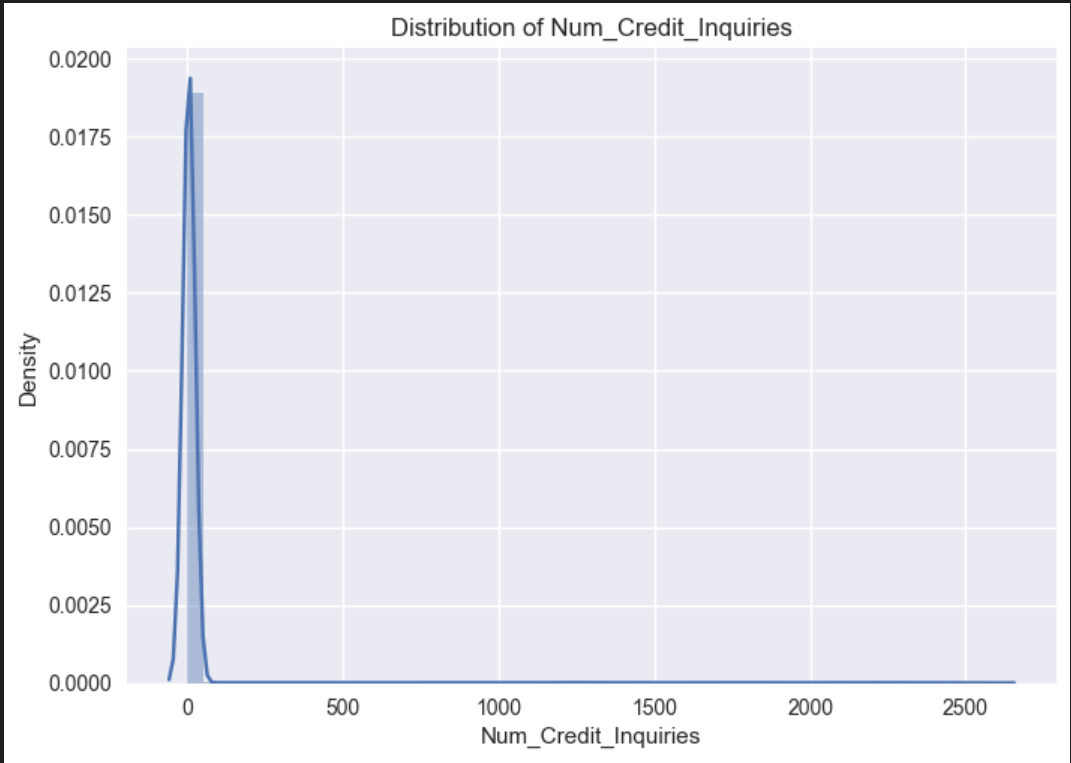
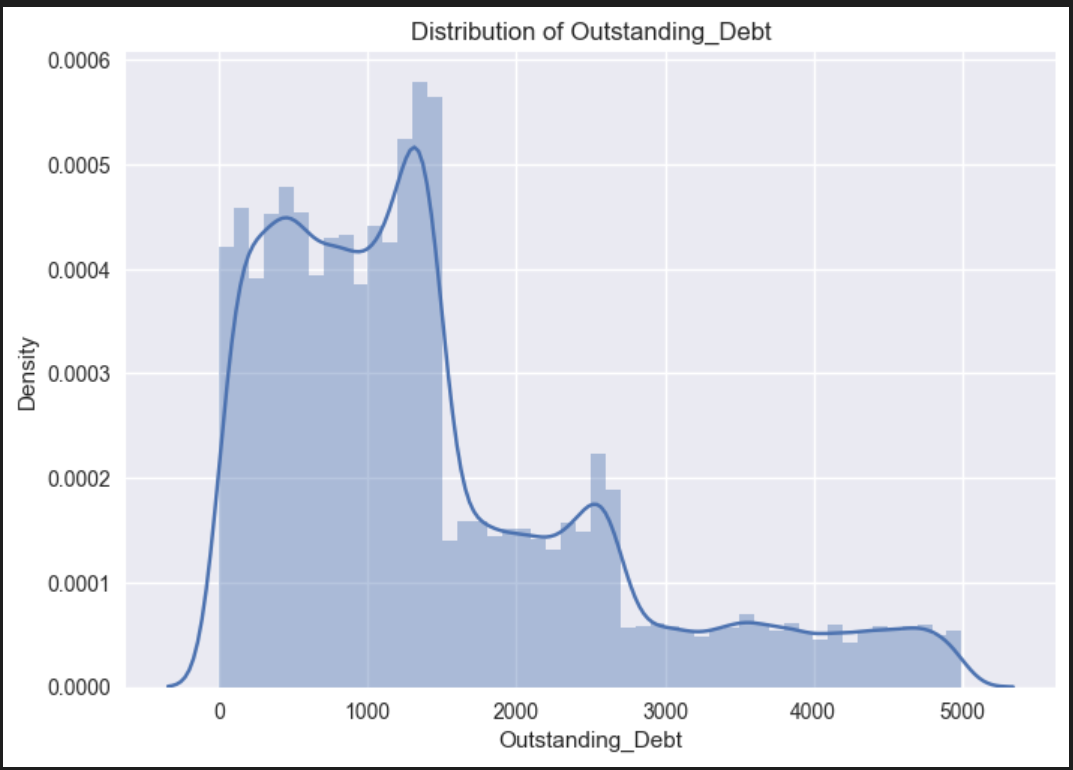
Distribution of Interest\_Rate: Distribution of Num\_of\_Loan:

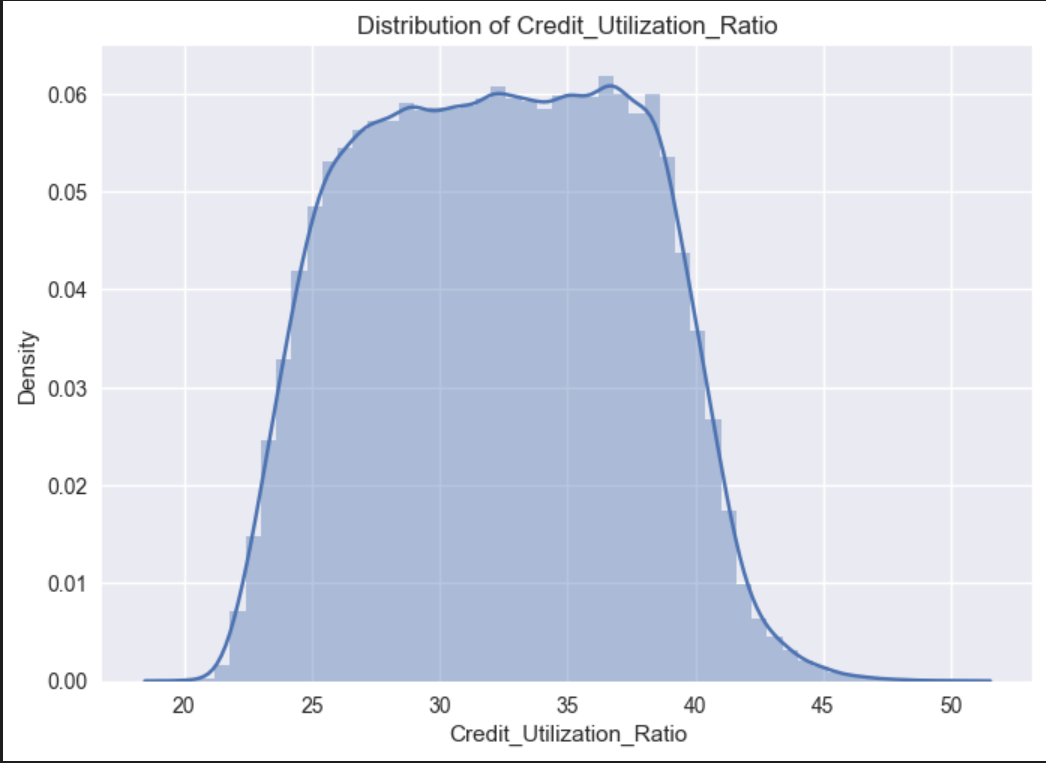
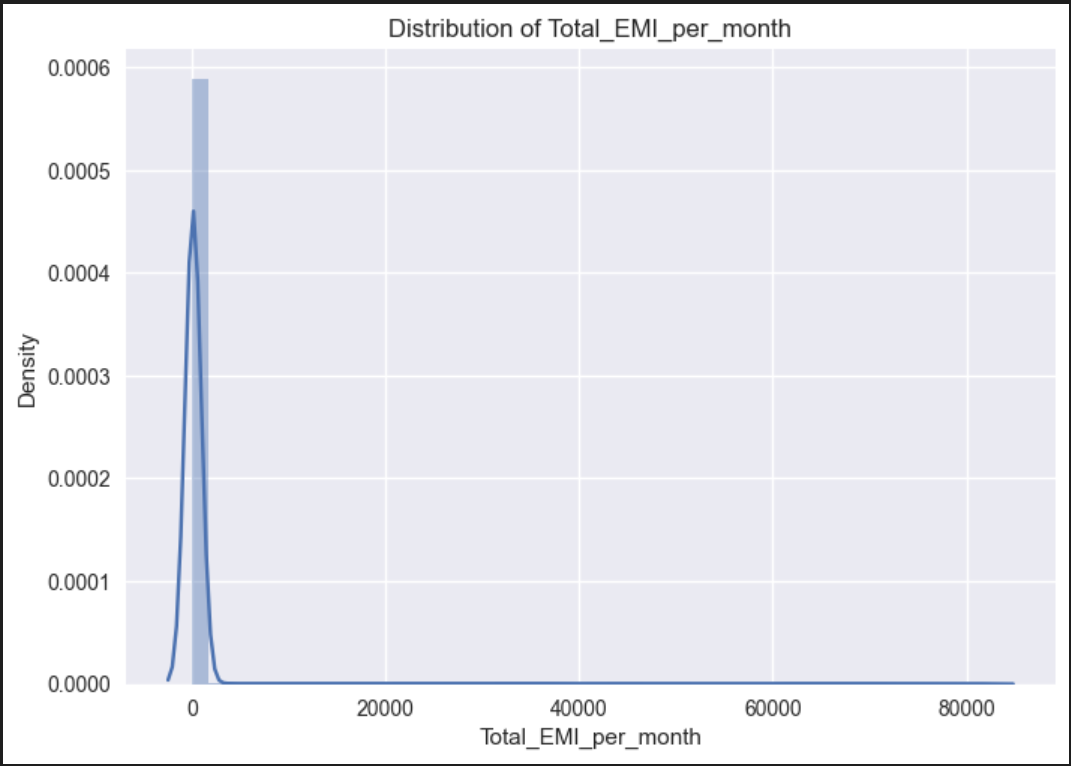
Distribution of Delay\_from\_due\_date: Distribution of Change\_Credit\_Limit:

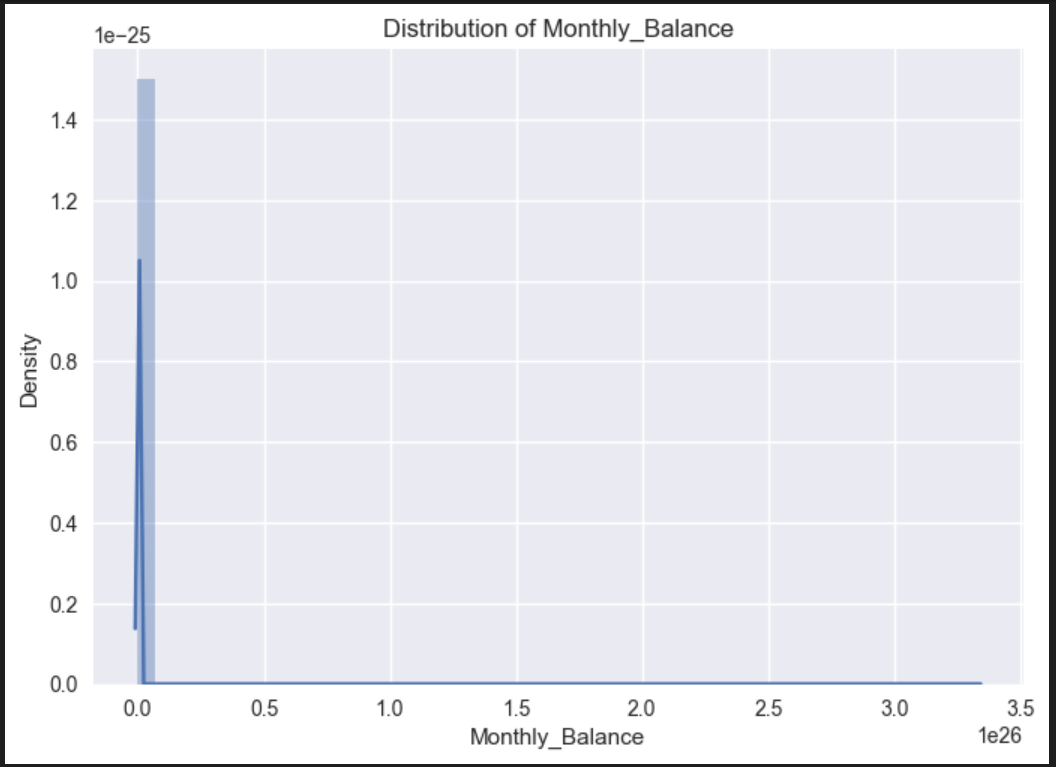
Distribution of Num\_Credit\_Inquiries: Distribution of Outstanding\_Debt:

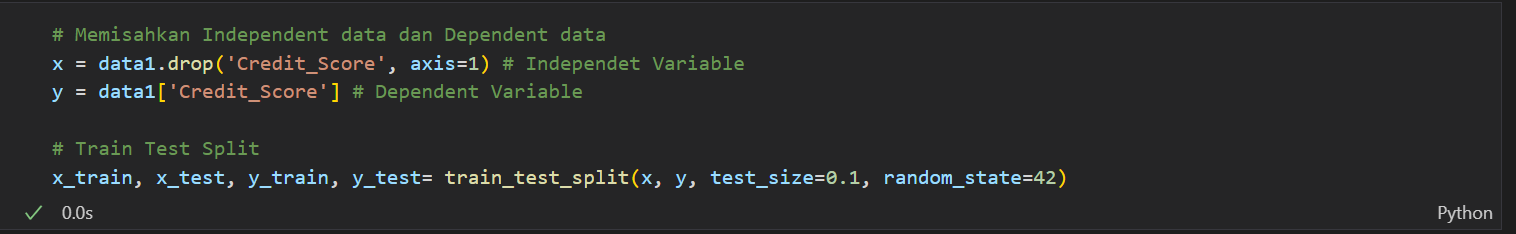
Distribution of Credit\_Utilization: Distribution of Total\_EMI\_per\_month:

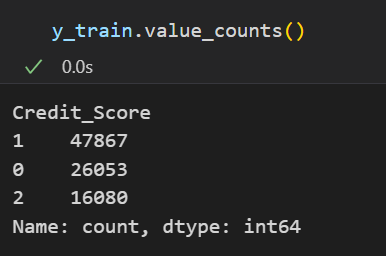
 

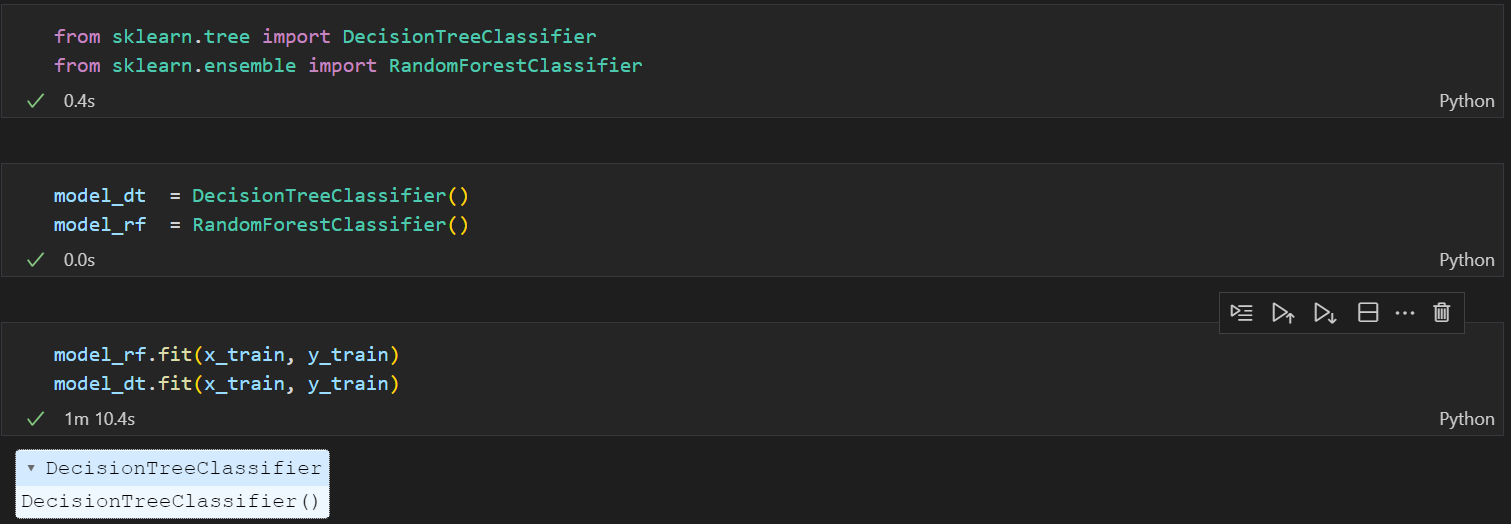
Distribution of Monthly\_Balance:

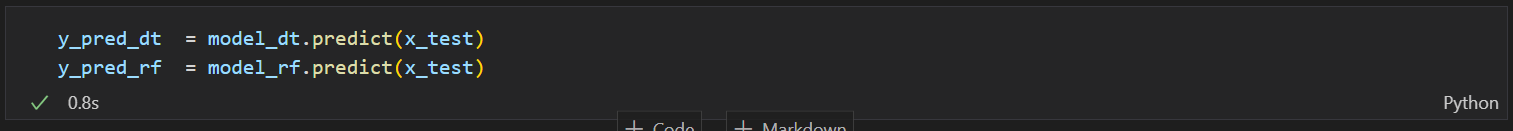


#Data Preparation and Splitting Data into train and test split

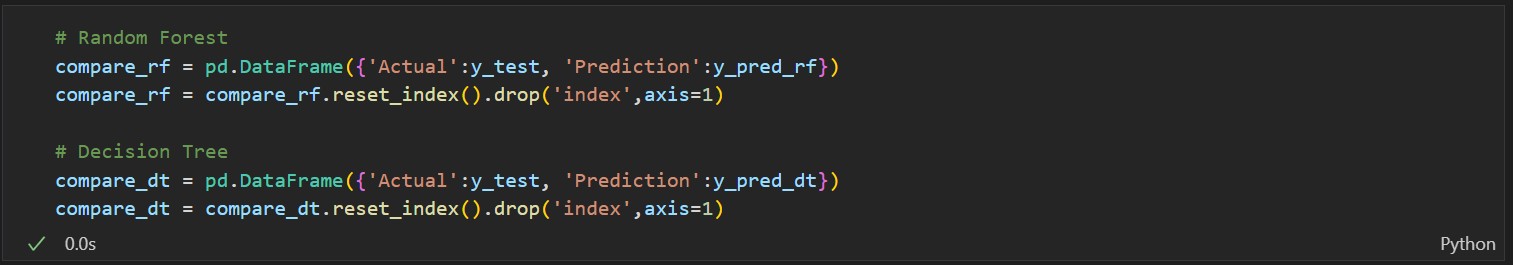


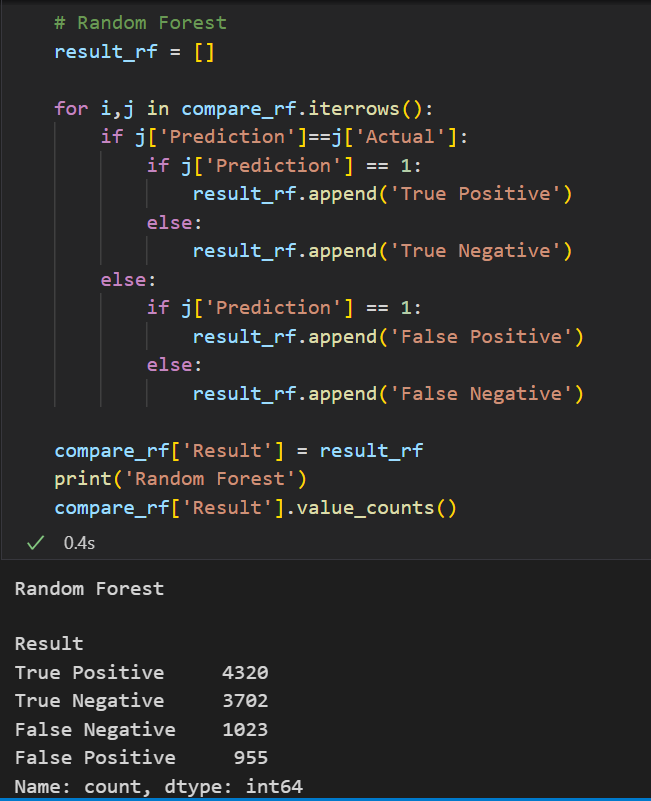






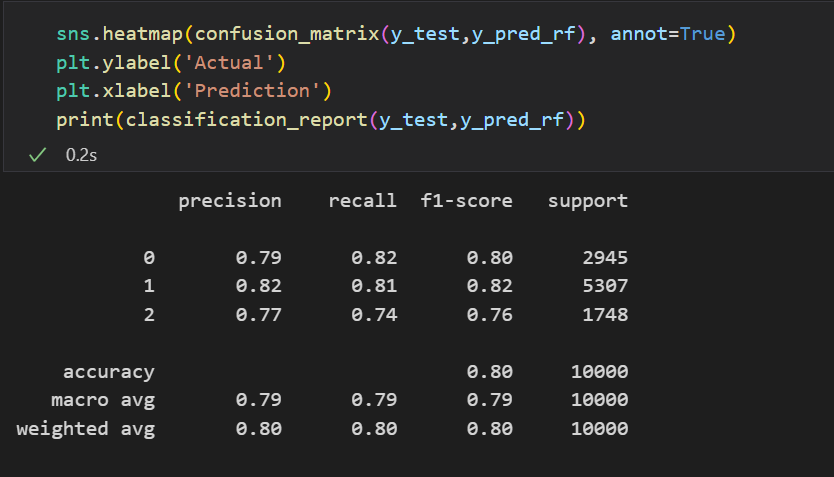
#Random Forest

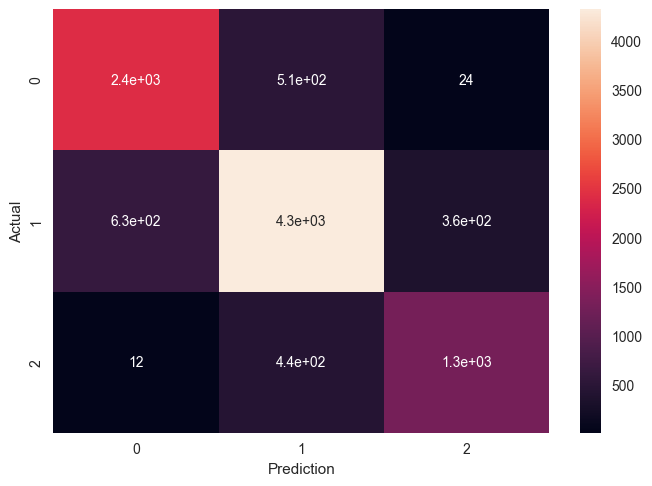


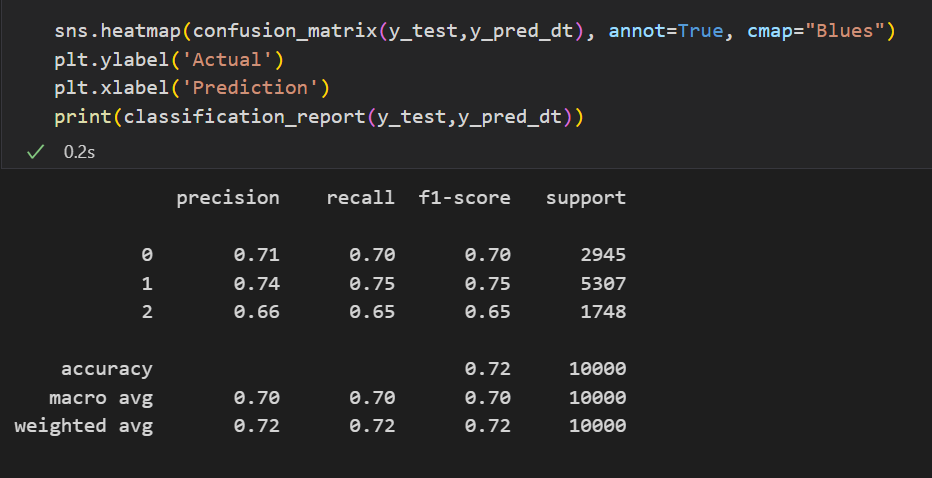


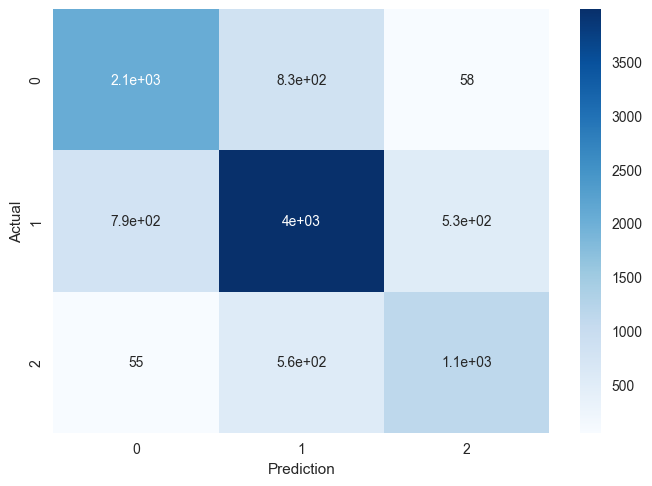


#Actual and Prediction Heatmap









# **Conclusion**

In conclusion, this report has outlined the development and application of a credit score classification model using Python. We have successfully developed a powerful tool for predicting and categorizing credit ratings through meticulous data pretreatment, feature engineering, and the application of machine learning algorithms. The significance of factors including payment history, credit use, and duration of credit history in establishing creditworthiness has been underlined by significant research. The usefulness of our algorithm in correctly categorizing credit scores is demonstrated by its performance, as measured by numerous evaluation measures. This methodology helps people to understand and enhance their creditworthiness in addition to assisting financial institutions in making educated lending decisions.

While Python provides strong frameworks and tools for credit score modeling, it is important to keep in mind that this analysis has several drawbacks. There are still issues with data quality, feature selection, and model interpretability. Moreover, continuing observation and model upgrades. Future improvements can be investigated, such as the incorporation of different data sources and the creation of more sophisticated machine learning methodologies. Python-based credit score classification models will become more and more important as the financial sector continues to adjust to new technologies and customer needs, promoting financial stability and inclusivity.

# **Reference**

(Kaggle n.d.)

(GitHub n.d.)

(MachineLearning n.d.)