CREDIT CARD APPROVAL SYSTEM
TERM REPORT

by

Pratyush Priyam
&
Thota Rahul

Section: KM118
Roll Numbers: RKM118B35 & RKM118A10

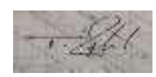
Department of Intelligent Systems,
School of Computer Science Engineering,
Lovely Professional University, Jalandhar

November 2022

STUDENT DECLARATION

This is to declare that this report has been written by us. No part of the report is copied from other sources. All information included from other sources have been duly acknowledged. We aver that if any part of the report is found to be copied, we shall take full responsibility for it.

Pratyush Priyam
RKM118B35



Thota Rahul
RKM118A10

Place : Lovely Professional University, Jalandhar
Date: 08th November 2022

TABLE OF CONTENTS

BONAFIDE CERTIFICATE

Certified that this project report "Credit Card Approval System using machine learning" is the Bonafede work of Pratyush Priyam and Thota Rahul who carried out the project work under my supervision.

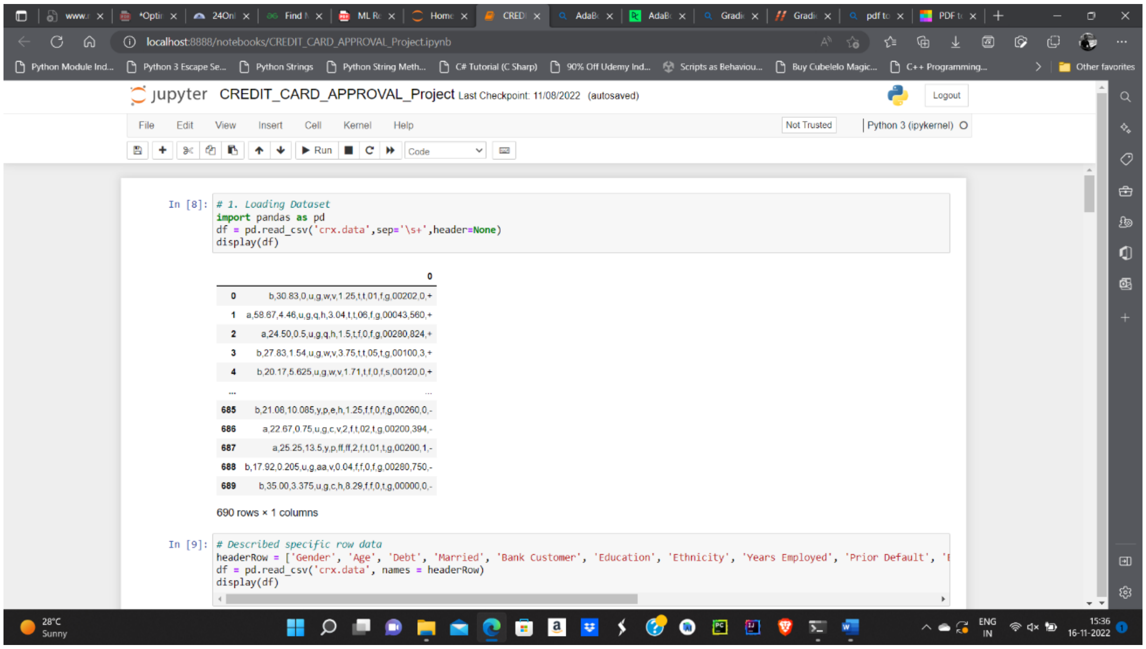
Dr. Dhanpratap Singh
Associate professor
25706
Intelligence System 1

INTRODUCTION

In this project, we will try to make a Credit Card Approval System using Machine Learning via python.

The correct assessment for credit card approval is very important for banks and organisations who lend a credit card to the people. The recent years have seen a huge growth in credit cards and loans. The exact judgement of person to be approved for credit cards allows the organisations to minimize losses and the same time make suitable credit arrangements as per requirement. Due to the huge growth in the number of applicants, there is a need for a more sophisticated method to automate the process and speed it up.

Credit card approval can be beneficial for organisations that lend credit cards, and due to increase in a huge number of the applicant, there is need to automate the task and classify the applicants into if they are eligible for a credit card or not. This helps to avoid organisation losses by avoiding potential defaulters. Here we are not just looking into bank balance but into their personal attributes like gender, married, age, Occupation etc. This can also help cut down the weekslong process into few days. This gives benefit by cutting down costs on credit analysis and faster credit decisions.



CREDIT_CARD_APPROVAL_Project Last Checkpoint: 11/08/2022 (autosaved)

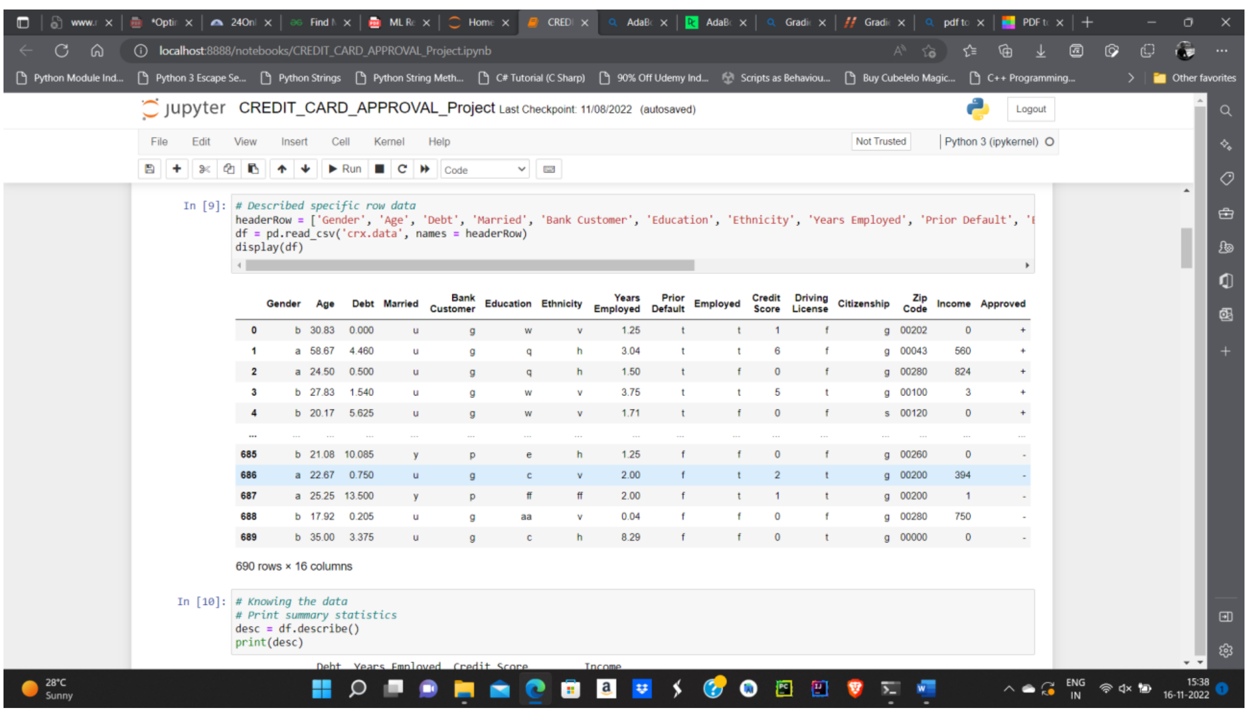
In [8]:
```
# 1. Loading Dataset
import pandas as pd
df = pd.read_csv('crx.data',sep='\s+',header=None)
display(df)
```

	0
0	b,30.83,0,u,g,w,v,1.25,t,t,01,f,g,00202,0,+
1	a,58.67,4.46,u,g,q,h,3.04,t,t,06,f,g,00043,560,+
2	a,24.50,0.5,u,g,q,h,1.5,t,f,0,f,g,00280,824,+
3	b,27.83,1.54,u,g,w,v,3.75,t,t,05,t,g,00100,3,+
4	b,20.17,5.625,u,g,w,v,1.71,t,f,0,f,s,00120,0,+
...	...
685	b,21.08,10.085,y,p,e,h,1.25,f,f,0,f,g,00260,0,-
686	a,22.67,0.75,u,g,c,v,2,f,t,02,t,g,00200,394,-
687	a,25.25,13.5,y,p,ff,ff,2,f,t,01,t,g,00200,1,-
688	b,17.92,0.205,u,g,aa,v,0.04,f,f,0,f,g,00280,750,-
689	b,35.00,3.375,u,g,c,h,8.29,f,f,0,t,g,00000,0,-

690 rows × 1 columns

In [9]:
```
# Described specific row data
headerRow = ['Gender', 'Age', 'Debt', 'Married', 'Bank Customer', 'Education', 'Ethnicity', 'Years Employed', 'Prior Default', 'E
df = pd.read_csv('crx.data', names = headerRow)
display(df)
```

Describing Data



CREDIT_CARD_APPROVAL_Project Last Checkpoint: 11/08/2022 (autosaved)

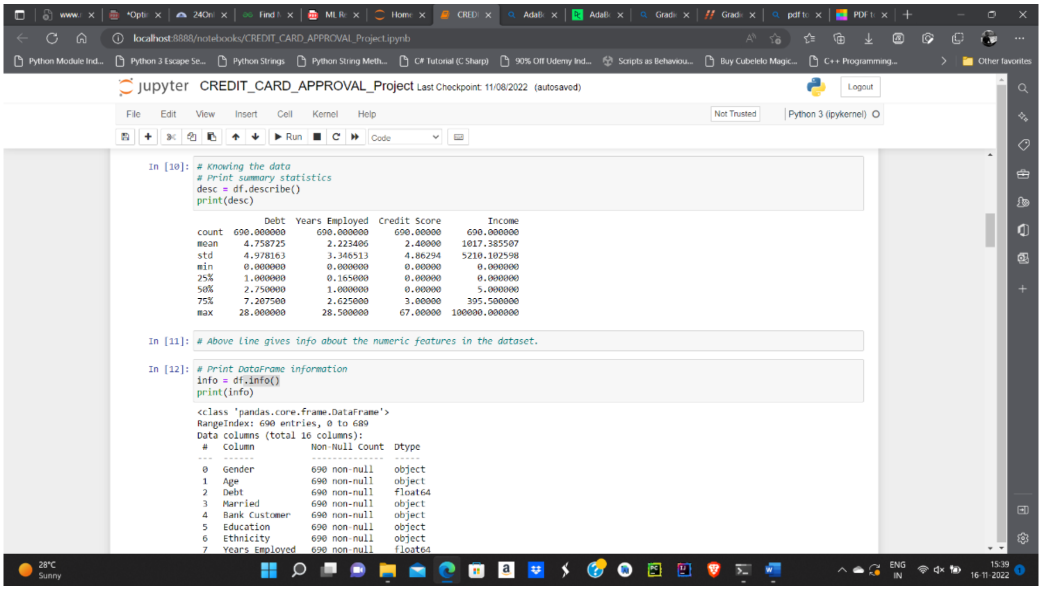
In [9]:
```
# Described specific row data
headerRow = ['Gender', 'Age', 'Debt', 'Married', 'Bank Customer', 'Education', 'Ethnicity', 'Years Employed', 'Prior Default', 'E
df = pd.read_csv('crx.data', names = headerRow)
display(df)
```

	Gender	Age	Debt	Married	Bank Customer	Education	Ethnicity	Years Employed	Prior Default	Employed	Credit Score	Driving License	Citizenship	Zip Code	Income	Approved
0	b	30.83	0.000	u	g	w	v	1.25	t	t	1	f	g	00202	0	+
1	a	58.67	4.460	u	g	q	h	3.04	t	t	6	f	g	00043	560	+
2	a	24.50	0.500	u	g	q	h	1.50	t	f	0	f	g	00280	824	+
3	b	27.83	1.540	u	g	w	v	3.75	t	t	5	t	g	00100	3	+
4	b	20.17	5.625	u	g	w	v	1.71	t	f	0	f	s	00120	0	+
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
685	b	21.08	10.085	y	p	e	h	1.25	f	f	0	f	g	00260	0	-
686	a	22.67	0.750	u	g	c	v	2.00	f	t	2	t	g	00200	394	-
687	a	25.25	13.500	y	p	ff	ff	2.00	f	t	1	t	g	00200	1	-
688	b	17.92	0.205	u	g	aa	v	0.04	f	f	0	f	g	00280	750	-
689	b	35.00	3.375	u	g	c	h	8.29	f	f	0	t	g	00000	0	-

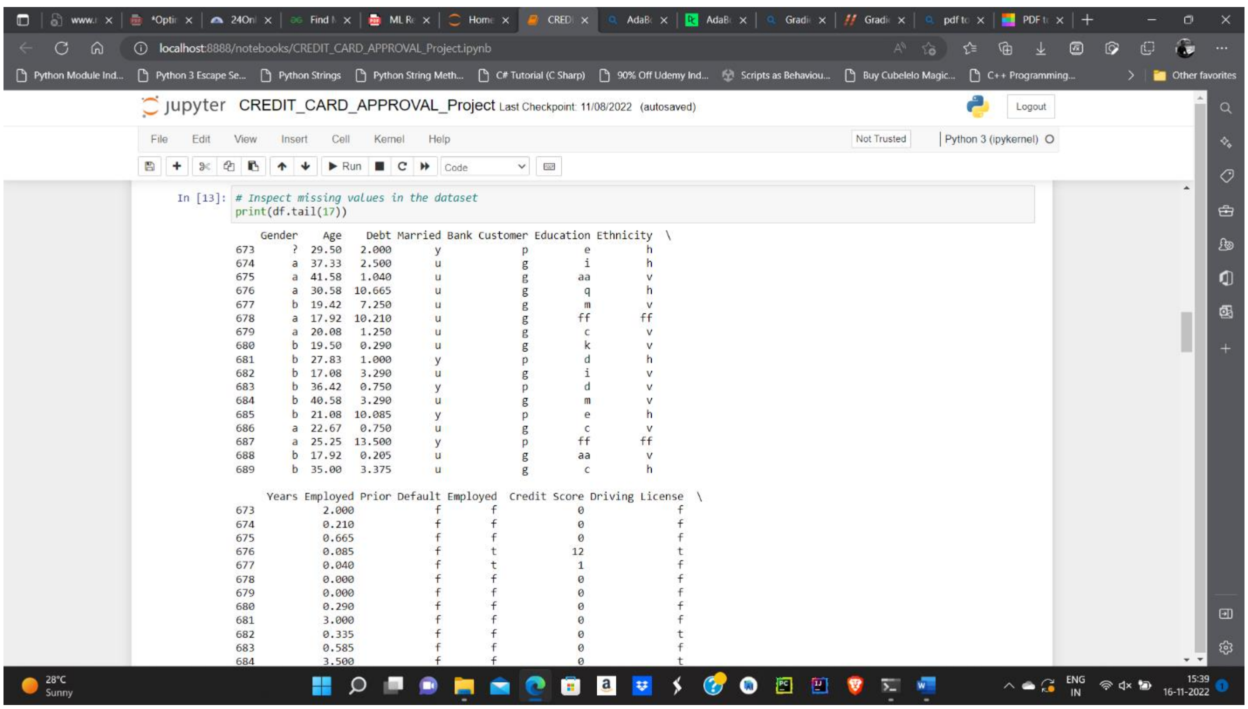
690 rows × 16 columns

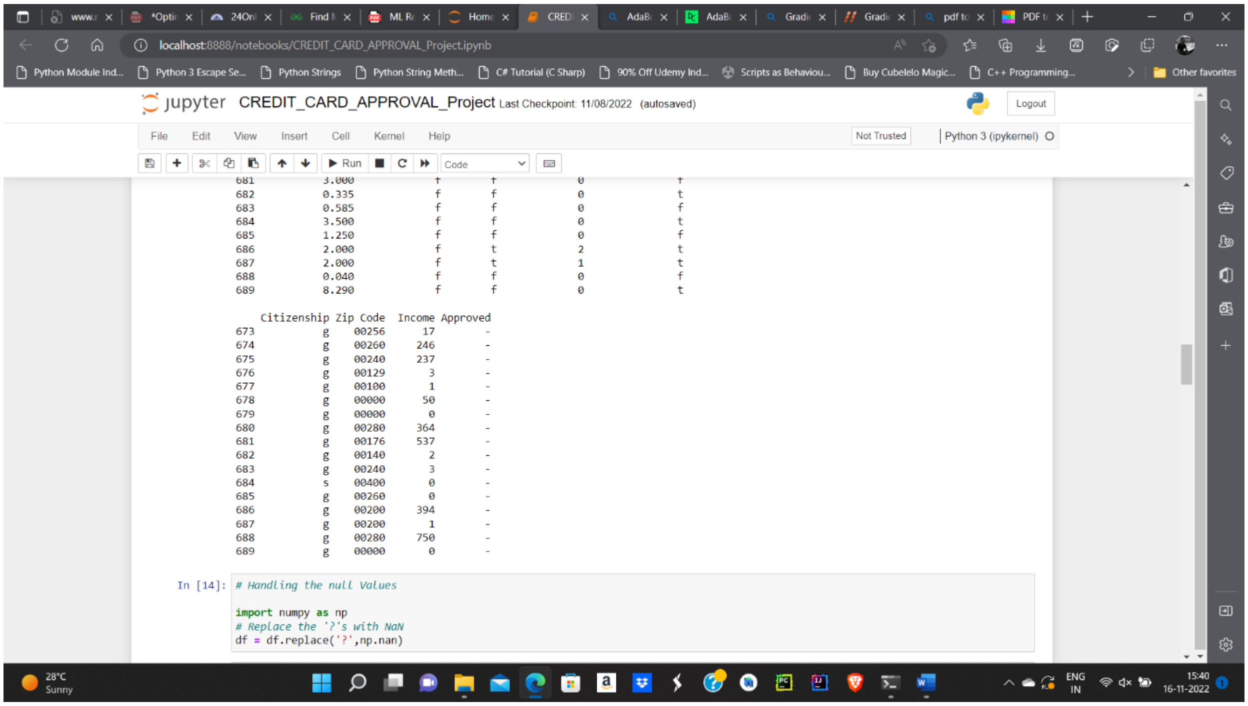
In [10]:
```
# Knowing the data
# Print summary statistics
desc = df.describe()
print(desc)
```

Knowing Data

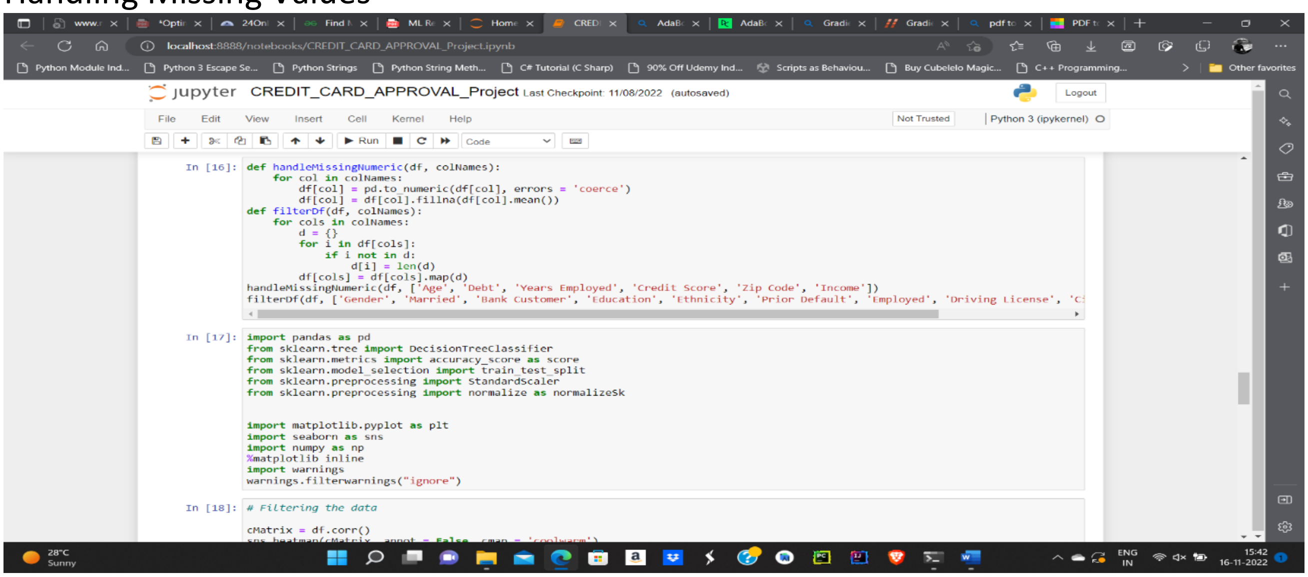


Inspecting data and finding null values

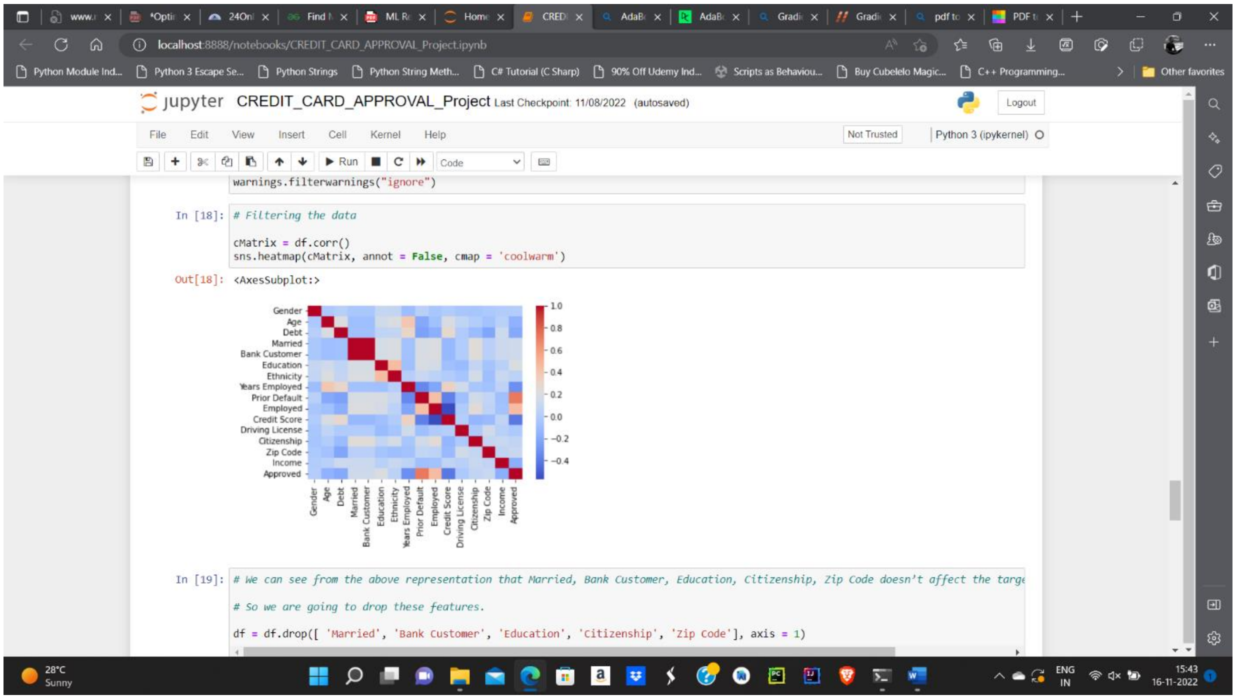


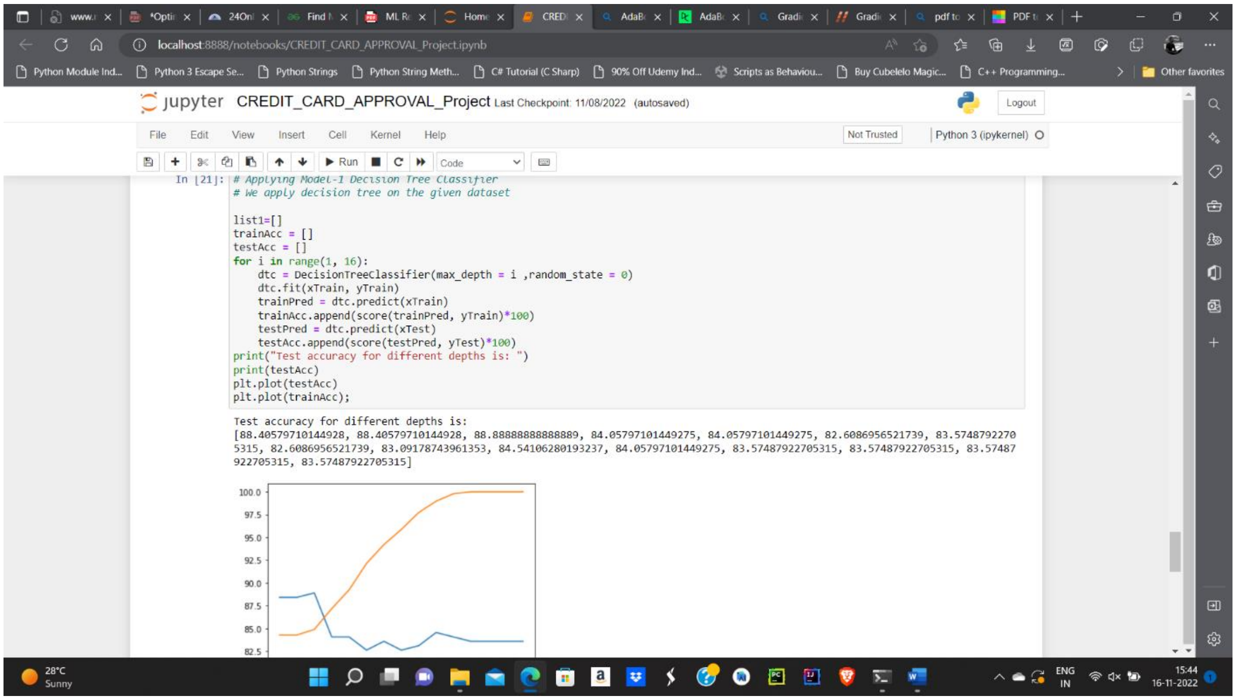


Handling Missing Values

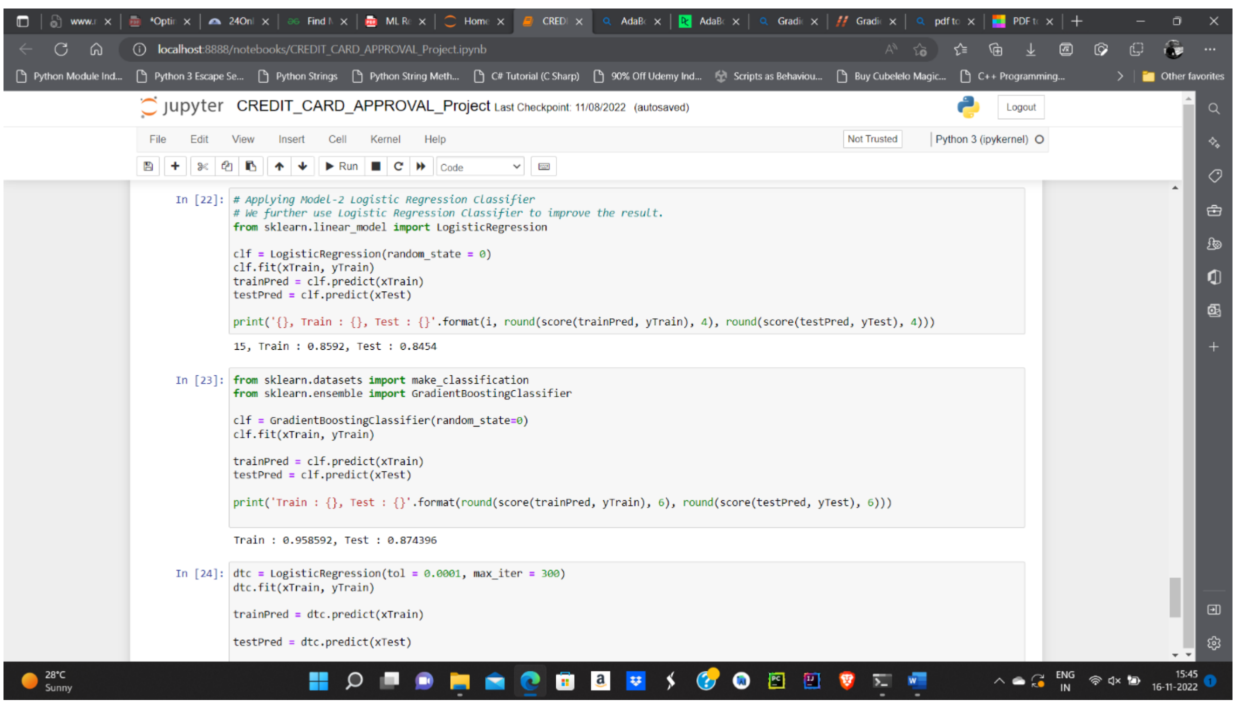


Correlation Matrix and heatmap

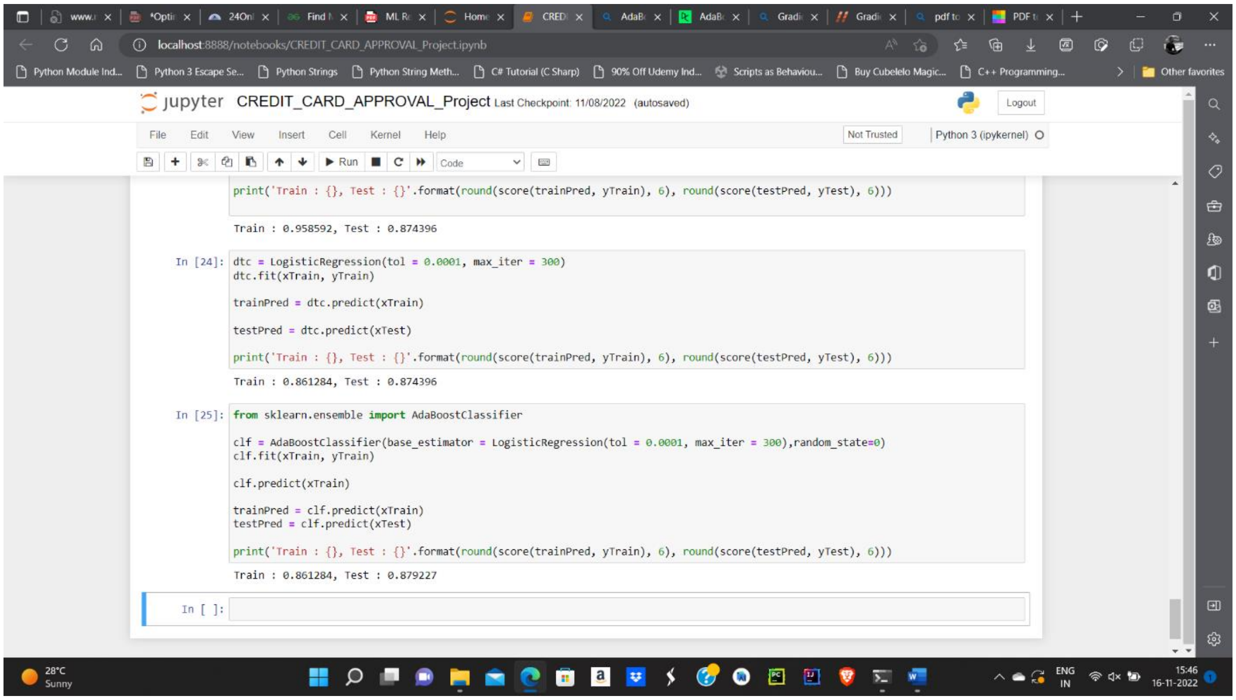




Decision tree Classifier



Logistic regression and Gradient Boosting classifier



Algorithms used

Logistic regression:

It is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval, or ratio-level independent variables.

Decision tree classifier:

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules, and each leaf node represents the outcome.

Gradient Boosting classifier:

Gradient boosting is a method standing out for its prediction speed and accuracy, particularly with large and complex datasets. From Kaggle competitions to machine learning solutions for business, this algorithm has produced the best results. We already know that errors play a major role in any machine learning algorithm There are mainly two types of errors, bias and variance error. Gradient boost algorithm helps us minimise bias error of the model

Adaboost classifier:

An Adaboost classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same data set but where the weight of in correctly classified instances r registered such that subsequent classifiers focus more on difficult cases

CONCLUSION

With ever increasing number of people who are actively using credit cards in today's world and focussing on the sheer exponential difference in the number of human employees that check and grant credit cards to customers, it is the need of the moment to introduce more reliable and sustainable means of technology that could take over this work efficiently

In these hard times, this Machine Learning project could be used to judge if multiple people make the appropriate cut to receive the credit card helping in not only reducing the human burden and mental agony but also making this process exponentially quicker.

Logistic regression:

Train Accuracy: 0.8592

Test Accuracy : 0.8454

Gradient Boosting Classifier:

Train Accuracy : 0.958592

Test Accuracy : 0.874396

AdaBoost Classifier:

Train Accuracy: 0.861284%

Test Accuracy

BIBLIOGRAPHY

- https://www.kaggle.com/datasets/rikdifos/credit-card-approval-prediction
- https://medium.datadriveninvestor.com/predicting-credit-card-approvals-using-mltechniques9cd8eaeb5b8c
- https://www.youtube.com/watch?v=kO0dnOucoWc

PROJECT GITHUB LINK

https://github.com/PratyushPriyam/CreditCardApproval_Project.git