PROJECT REPORT

(Project Term feb-march 2022)

(*Debit/Credit card fraud detection using ML*)

Submitted by

(Ravi Pandey) **Registration Number :11910459**

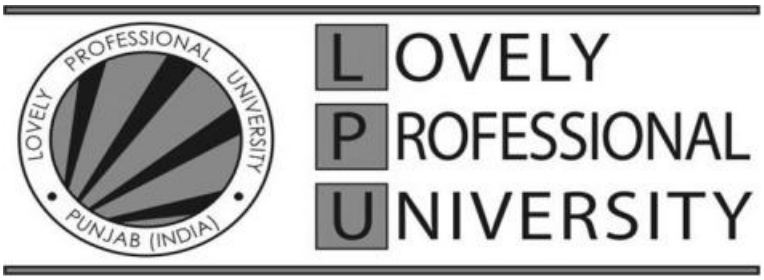
Project Group Number B

Course Code INT247

Under the Guidance of

(Dr. Sagar Pande)

School of Computer Science and Engineering



DECLARATION

We hereby declare that the project work entitled (“ Debit/Credit card fraud detection using ML ”) is an authentic record of our own work carried out as requirements of Project for the award of B.Tech degree in Computer science from Lovely Professional University, Phagwara, under the guidance of (Dr. Sagar Pande), during February-March 2022. All the information furnished in this project report is based on our own intensive work and is genuine.

Project Group Number: B

Name of Student : Ravi Pandey
Registration Number: 11910459

(Ravi Pandey)

Date: 24-03-2022

CERTIFICATE

This is to certify that the declaration statement made by this group of students is correct to the best of my knowledge and belief. They have completed this Project under my guidance and supervision. The present work is the result of their original investigation, effort and study. No part of the work has ever been submitted for any other degree at any University. The Project is fit for the submission and partial fulfillment of the conditions for the award of B.Tech degree in Computer Science from Lovely Professional University, Phagwara.

Signature and Name of the Mentor

Designation

School of Computer Science and Engineering,
Lovely Professional University,
Phagwara, Punjab.

Date :20-03-2022

ACKNOWLEDGEMENT

I would sincerely like to thanks for the constructive criticism, support, encouragment valuable, comments, suggestions, timely helps and many innovative ideas given to me by my project supervisor Dr sagar Pande in carrying out project and the report

I must convey my gratitude to Dr sagar Pande for giving me the constant source of Inspiration and help in preparing the project ,personally correcting me Providing encouragement throughout the project.

I also thanks all my faculty members for steering me through the tough as well as Easy phase of the project in result oriented manner with concern attention.

INTRODUCTION

My name is Ravi Pandey. currently, I am pursuing Btech from Lovely professional university in computer science. Here I am to tell about my project which I have completed during February-march**.** my project topic is " Debit/Credit card fraud detection using ML". in this project I have write code which when it run its detected Debit/Credit card

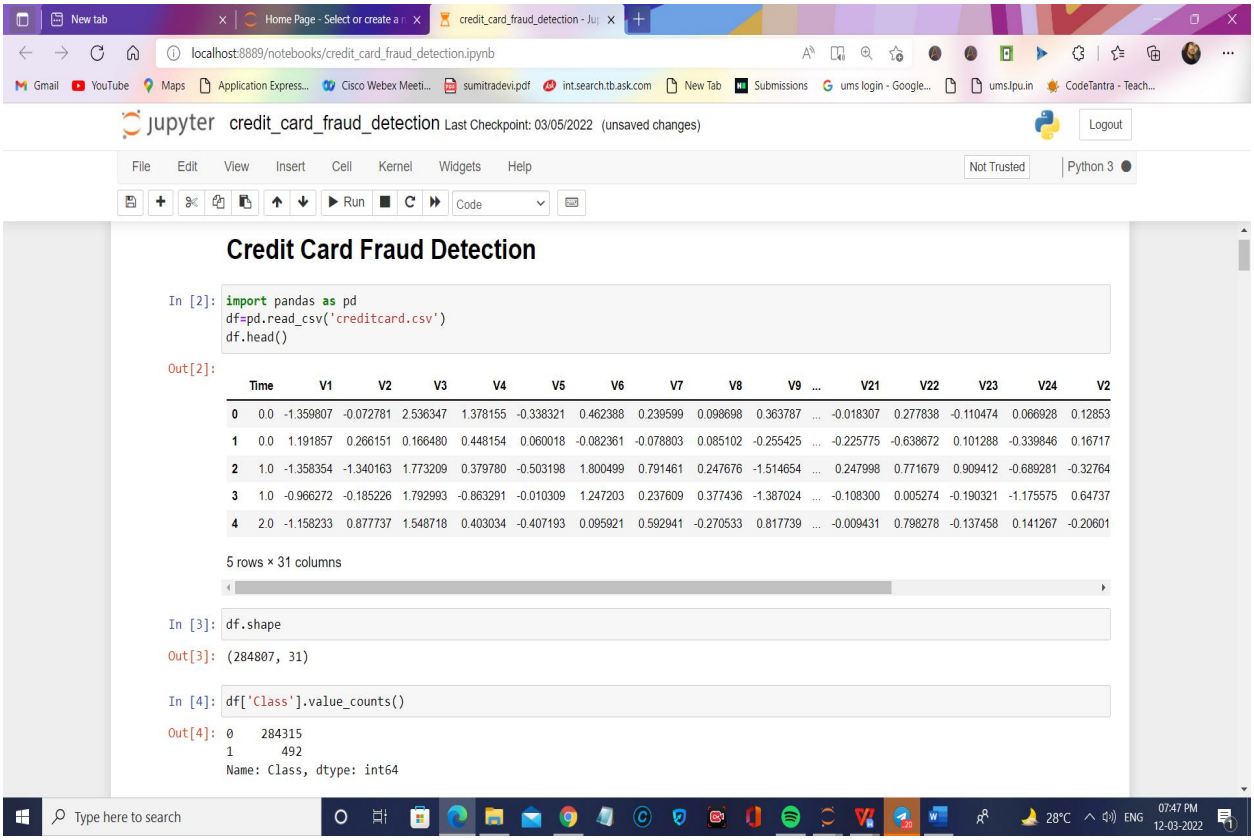
Abstract

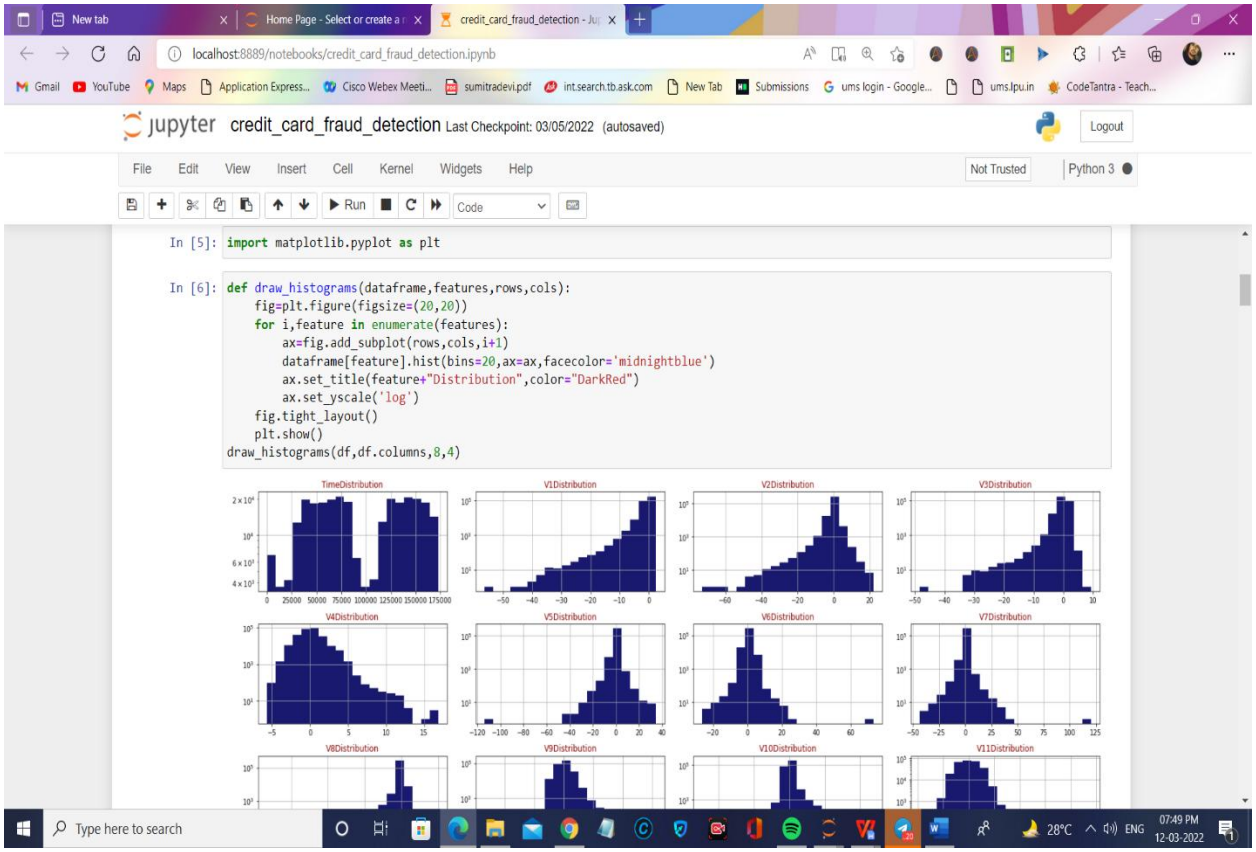
Credit card fraud detection present several characteristics that makes it a challenging task. First, the feature set describing a credit card transaction usually ignores detailed sequential information which was proven to be very relevant for the detection of credit card fraudulent transactions. Second, purchase behaviours and fraudster strategies may change over time, making a learnt fraud detection decision function irrelevant if not updated. This phenomenon named dataset shift (change in the distribution p(x, y)) may hinder fraud detection systems to obtain good performances. We conducted an exploratory analysis in order to quantify the day by day dataset shift and identified calendar related time periods that show different properties. Third, credit card transactions data suffer from a strong imbalance regarding the class labels which needs to be considered either from the classifier perspective or from the data perspective (less than 1% of the transactions are fraudulent transactions). Solutions for integrating sequential information in the feature set exist in the literature. The predominant one consists in creating a set of features which are descriptive statistics obtained by aggregating the sequences of transactions of the card-holders (sum of amount, count of transactions, etc..). We used this method as a benchmark

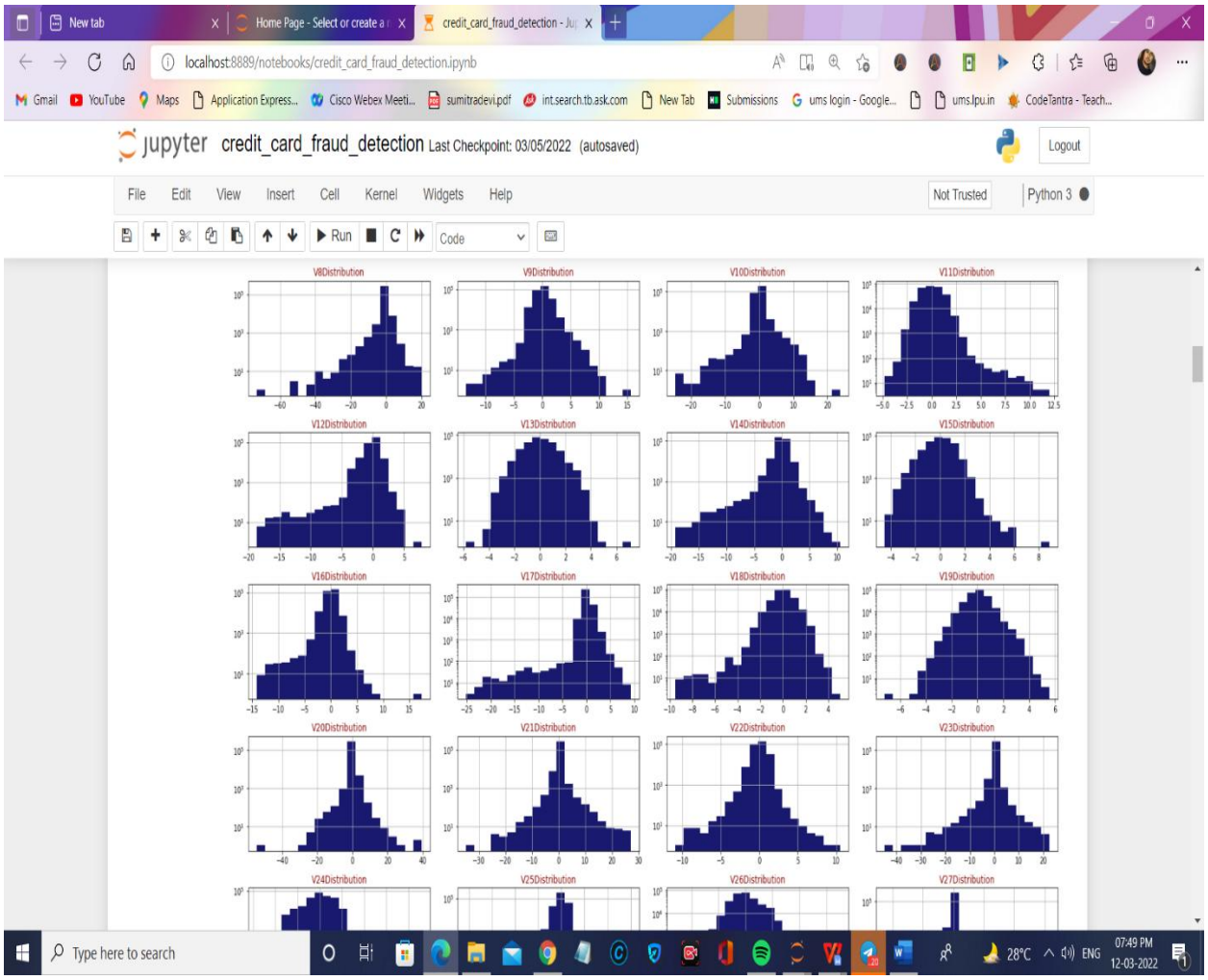
feature engineering method for credit card fraud detection. However, this feature engineering strategies raised several research questions. First of all, we assumed that these descriptive statistics cannot fully describe the sequential properties of fraud and genuine patterns and that modelling the sequences of transactions could be beneficial for fraud detection. Moreover the

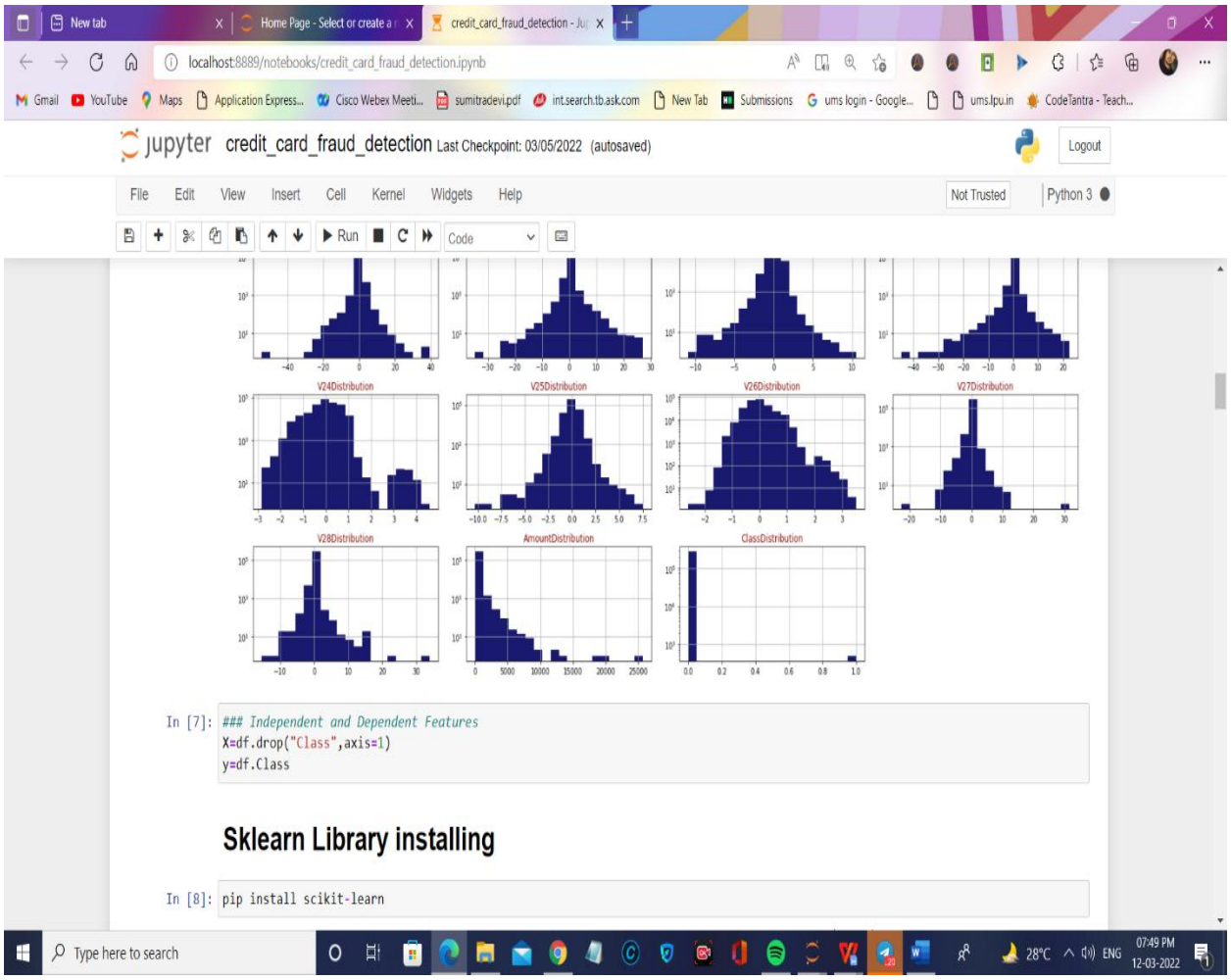
creation of these aggregated features is guided by expert knowledge whereas sequences modelling could be automated thanks to the class labels available for past transactions. Finally, the aggregated features are point estimates that may be completed by a multi-perspective univariate description of the transaction context (especially from the point of view of the seller). We proposed a multi-perspective HMM-based automated feature engineering strategy in order to incorporate a broad spectrum of sequential information in the transactions feature sets. In fact, we model the genuine and fraudulent behaviours of the merchants and the card-holders according to two univariate features: the timing and the amount of the transactions. Moreover, the HMMbased features are created in a supervised way and therefore lower the need of expert knowledge for the creation of the fraud detection system. In the end, our multiple perspectives HMM-based approach offers automated feature engineering to model temporal correlations so as to complement and possibly supplement the use of transaction aggregation strategies in order to improve the effectiveness of the classification task. Experiments conducted on a large real world credit card transaction dataset (46 million transactions from belgium card-holders between March and May 2015) have shown that the proposed HMM-based feature engineering allows for an increase in the detection of fraudulent transactions when combined with the state of the art expert based feature engineering strategy for credit card fraud detection. To conclude, this work leads to a better understanding of what can be considered contextual knowledge for a credit card fraud detection task and how to include it in the classification task in order to get an increase in fraud detection. The method proposed can be extended to any supervised task with sequence

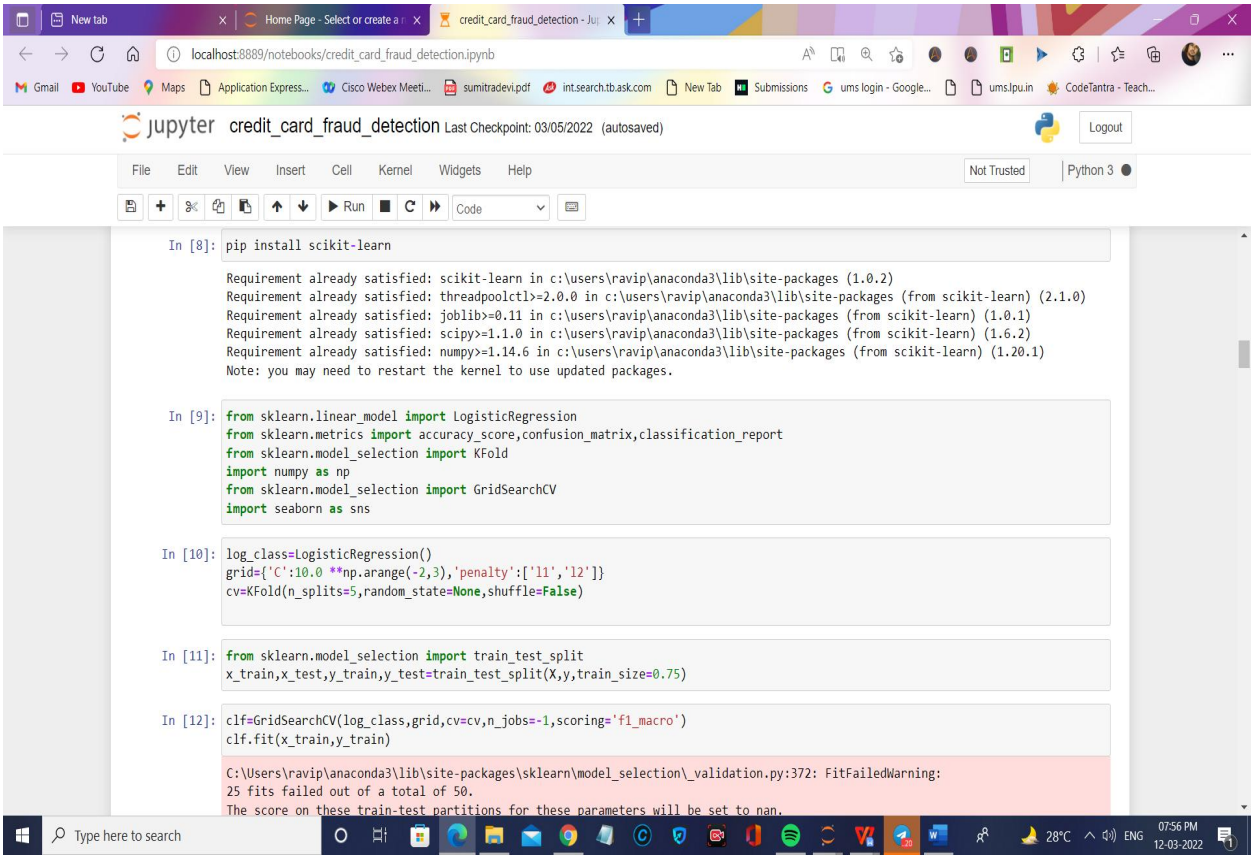
List of Screenshote

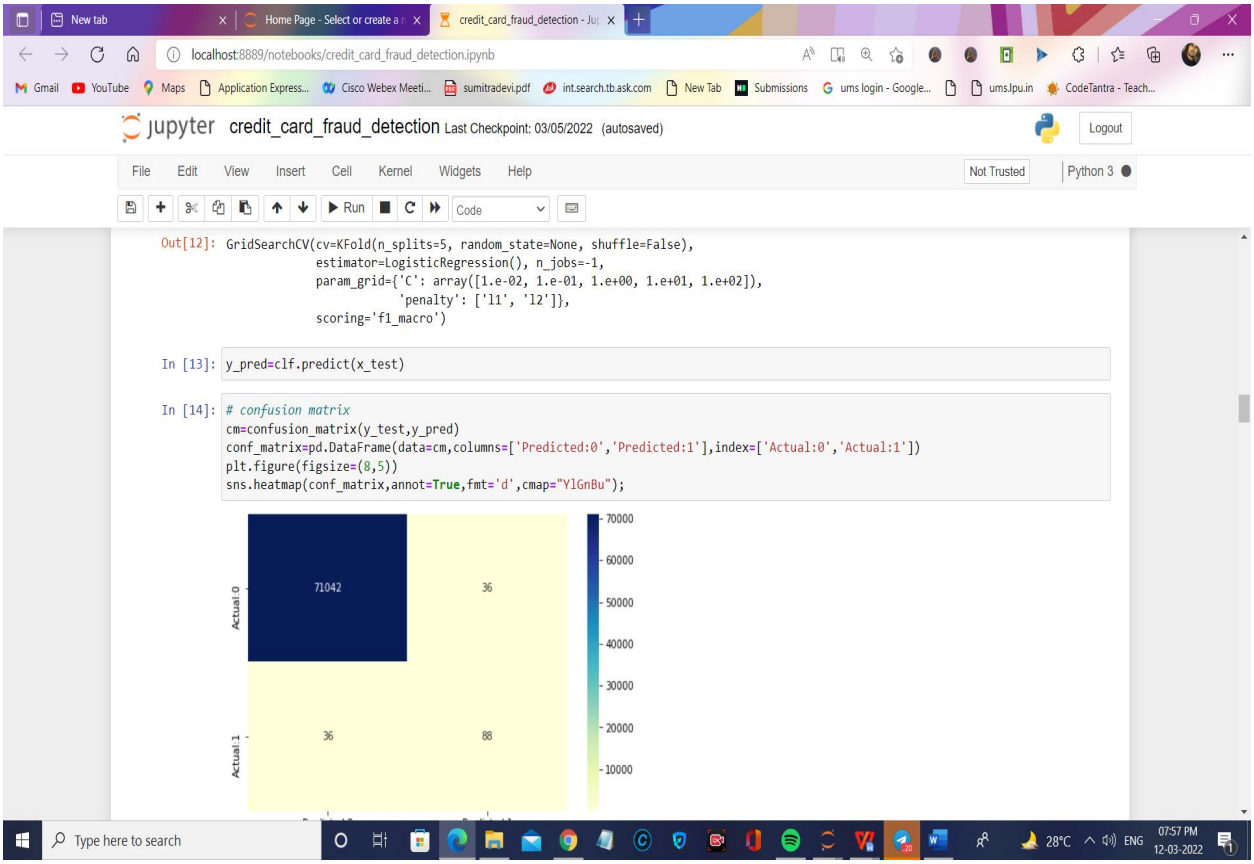


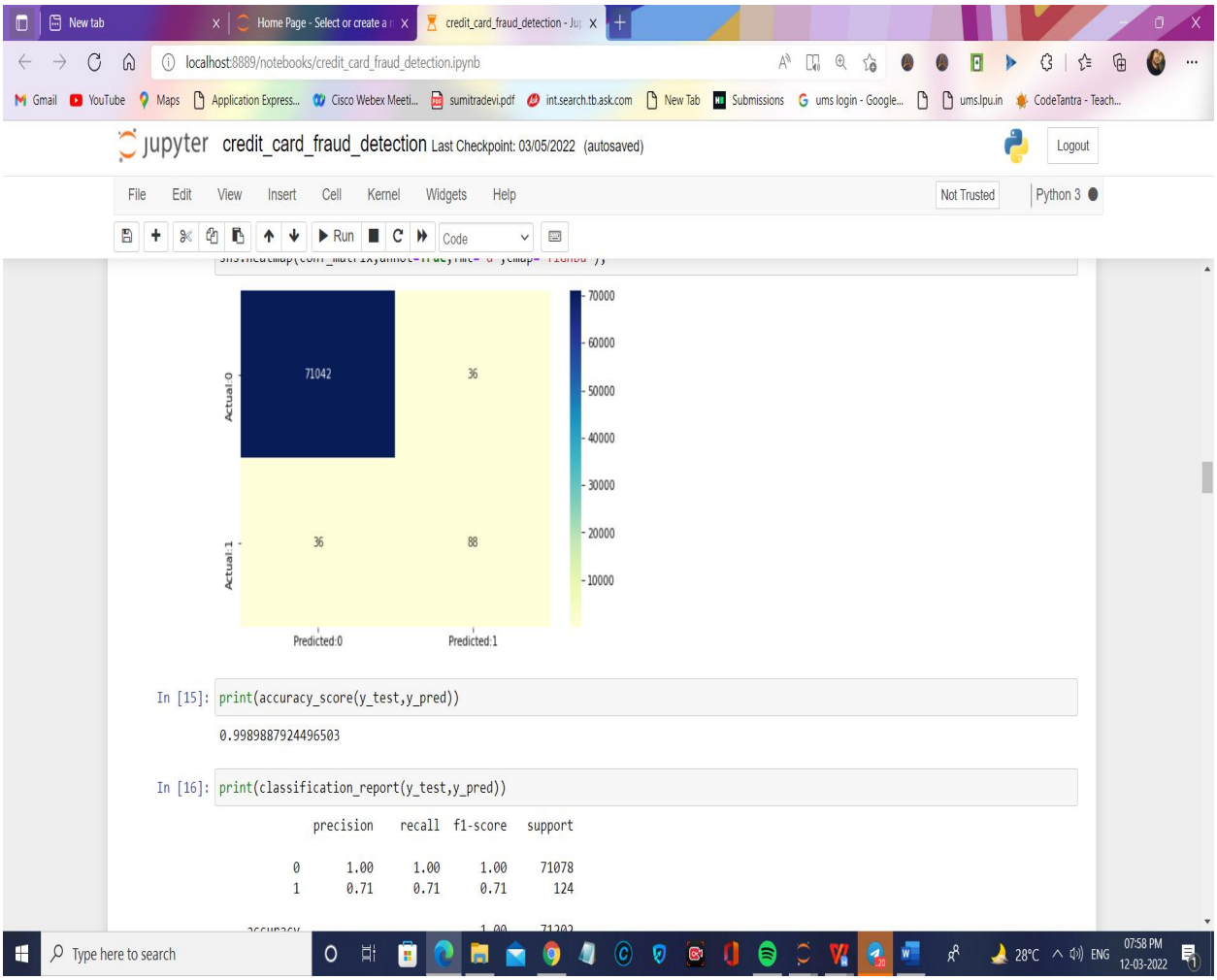

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
In [5]: import matplotlib.pyplot as plt
In [6]: def draw_histograms(dataframe,features,rows,cols):
fig=plt.figure(figsize=(20,20))
for i,feature in enumerate(features):
ax=fig.add_subplot(rows,cols,i+1)
dataframe[feature].hist(bins=20,ax=ax,facecolor='midnightblue')
ax.set_title(feature+"Distribution",color="DarkRed")
ax.set_yscale('log')
fig.tight_layout()
plt.show()
draw_histograms(df,df.columns,8,4)
TimeDistribution
V1Distribution
V2Distribution
V3Distribution
V4Distribution
V5Distribution
V6Distribution
V7Distribution
V8Distribution
V9Distribution
V10Distribution
V11Distribution
Type here to search
28°C
ENG
07:49 PM
12-03-2022

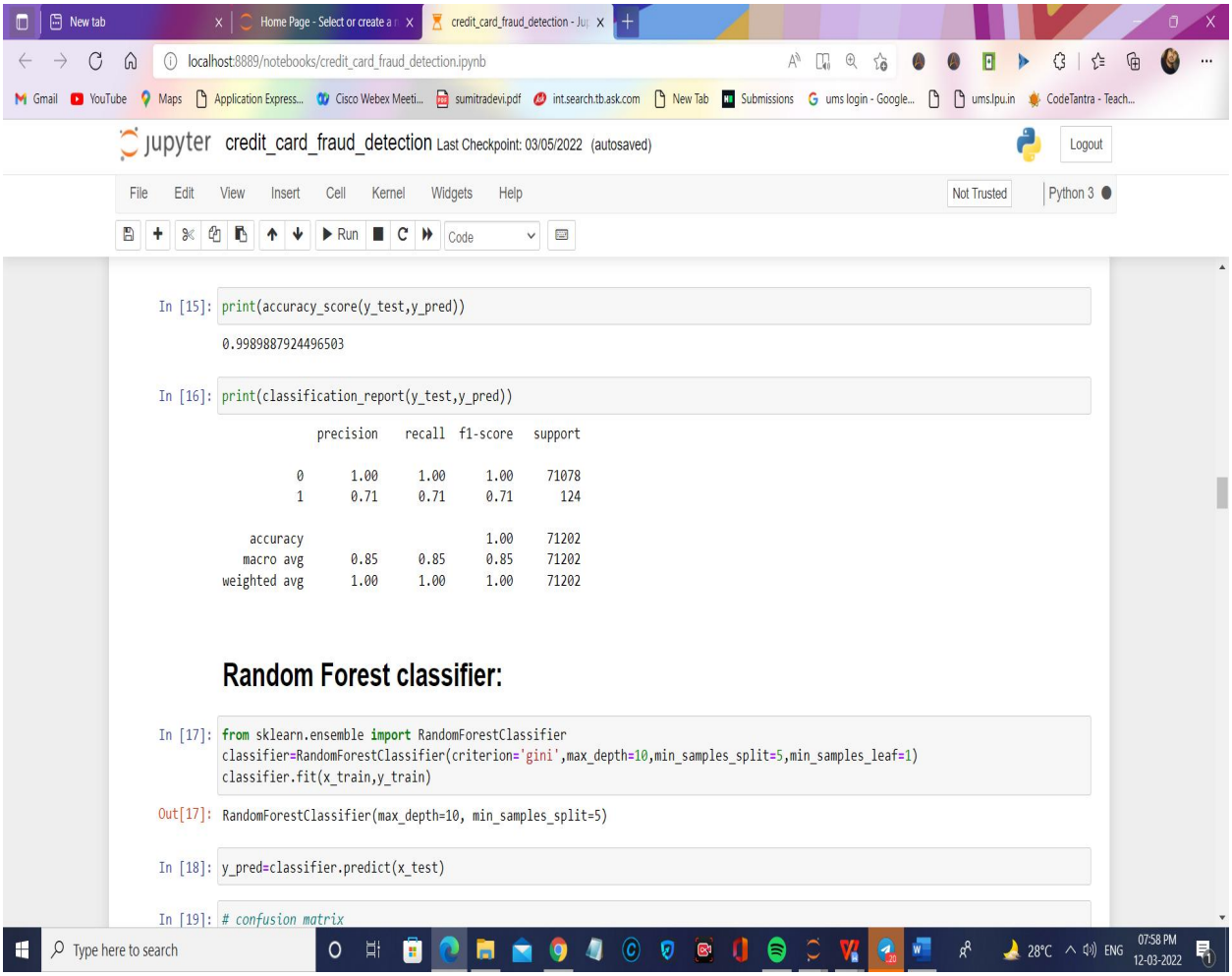

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3
Run
Code
V8Distribution
V9Distribution
V10Distribution
V11Distribution
V12Distribution
V13Distribution
V14Distribution
V15Distribution
V16Distribution
V17Distribution
V18Distribution
V19Distribution
V20Distribution
V21Distribution
V22Distribution
V23Distribution
V24Distribution
V25Distribution
V26Distribution
V27Distribution
Type here to search
28°C
ENG
07:49 PM
12-03-2022

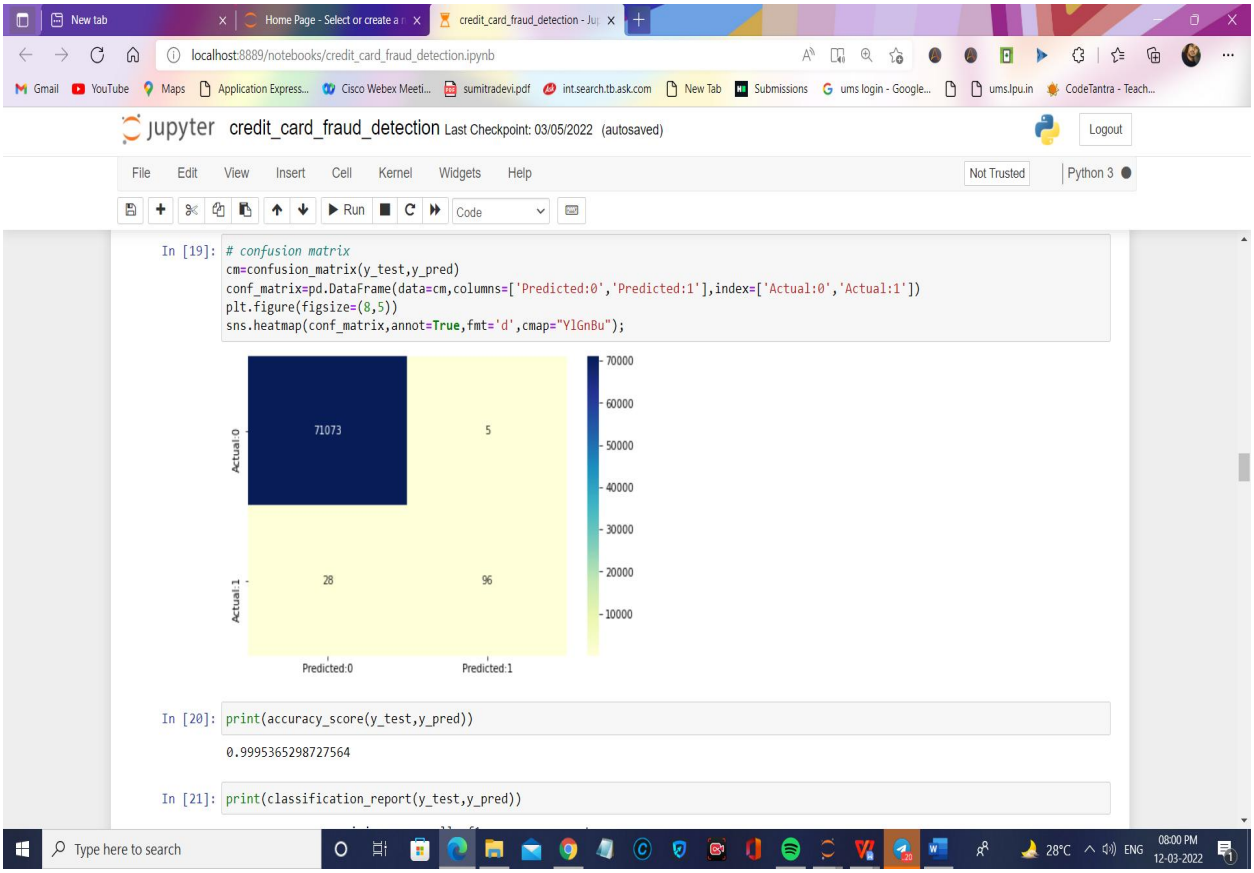

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
V24Distribution
V25Distribution
V26Distribution
V27Distribution
V28Distribution
AmountDistribution
ClassDistribution
In [7]:
Independent and Dependent Features
X=df.drop("Class",axis=1)
y=df.Class
Sklearn Library installing
In [8]:
pip install scikit-learn
Type here to search
28°C
ENG
07:49 PM
12-03-2022

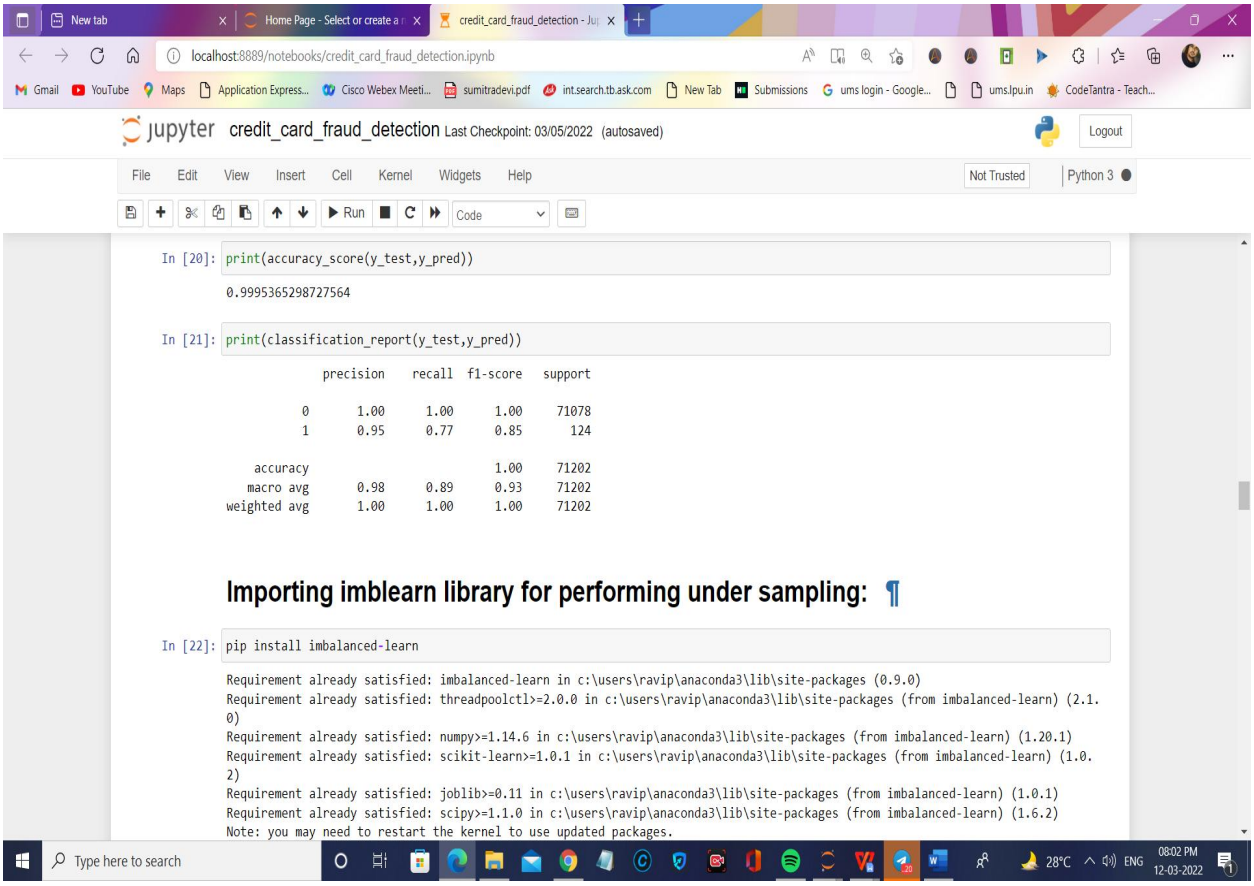

New tab
Home Page - Select or create a n
credit_card_fraud_detection - Jup
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [8]:
pip install scikit-learn
Requirement already satisfied: scikit-learn in c:\users\ravip\anaconda3\lib\site-packages (1.0.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\ravip\anaconda3\lib\site-packages (from scikit-learn) (2.1.0)
Requirement already satisfied: joblib>=0.11 in c:\users\ravip\anaconda3\lib\site-packages (from scikit-learn) (1.0.1)
Requirement already satisfied: scipy>=1.1.0 in c:\users\ravip\anaconda3\lib\site-packages (from scikit-learn) (1.6.2)
Requirement already satisfied: numpy>=1.14.6 in c:\users\ravip\anaconda3\lib\site-packages (from scikit-learn) (1.20.1)
Note: you may need to restart the kernel to use updated packages.
In [9]:
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report
from sklearn.model_selection import KFold
import numpy as np
from sklearn.model_selection import GridSearchCV
import seaborn as sns
In [10]:
log_class=LogisticRegression()
grid={'C':10.0 **np.arange(-2,3),'penalty':['l1','l2']}
cv=KFold(n_splits=5,random_state=None,shuffle=False)
In [11]:
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(X,y,train_size=0.75)
In [12]:
clf=GridSearchCV(log_class,grid,cv=cv,n_jobs=-1,scoring='f1_macro')
clf.fit(x_train,y_train)
C:\Users\ravip\anaconda3\lib\site-packages\sklearn\model_selection_validation.py:372: FitFailedWarning:
25 fits failed out of a total of 50.
The score on these train-test partitions for these parameters will be set to nan.
Type here to search
28°C
ENG
07:56 PM
12-03-2022

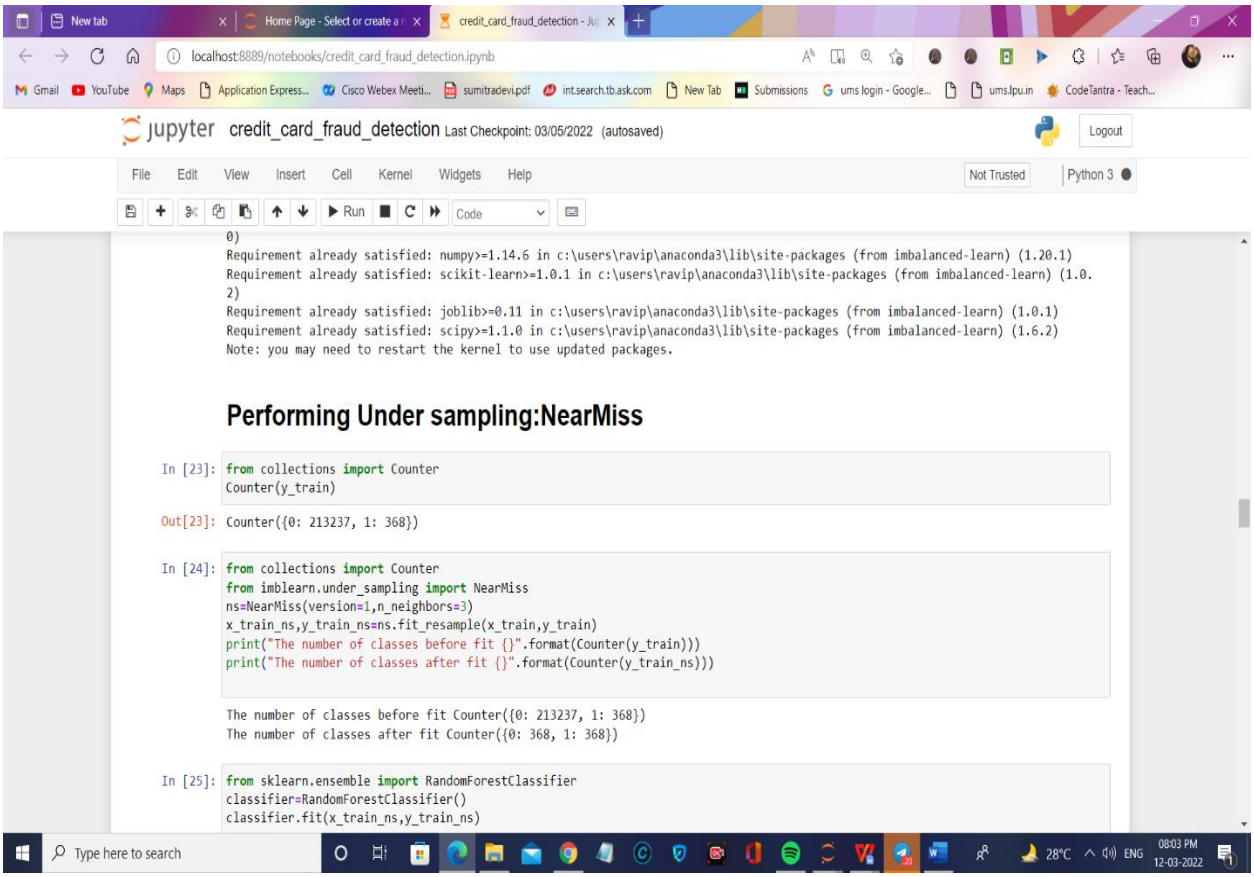

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
Out[12]: GridSearchCV(cv=KFold(n_splits=5, random_state=None, shuffle=False),
estimator=LogisticRegression(), n_jobs=-1,
param_grid={'C': array([1.e-02, 1.e-01, 1.e+00, 1.e+01, 1.e+02]),
'penalty': ['l1', 'l2']},
scoring='f1_macro')
In [13]: y_pred=clf.predict(x_test)
In [14]: # confusion matrix
cm=confusion_matrix(y_test,y_pred)
conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','Actual:1'])
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");
71042
36
36
88
Actual:0
Actual:1
70000
60000
50000
40000
30000
20000
10000
Type here to search
28°C
ENG
07:57 PM
12-03-2022

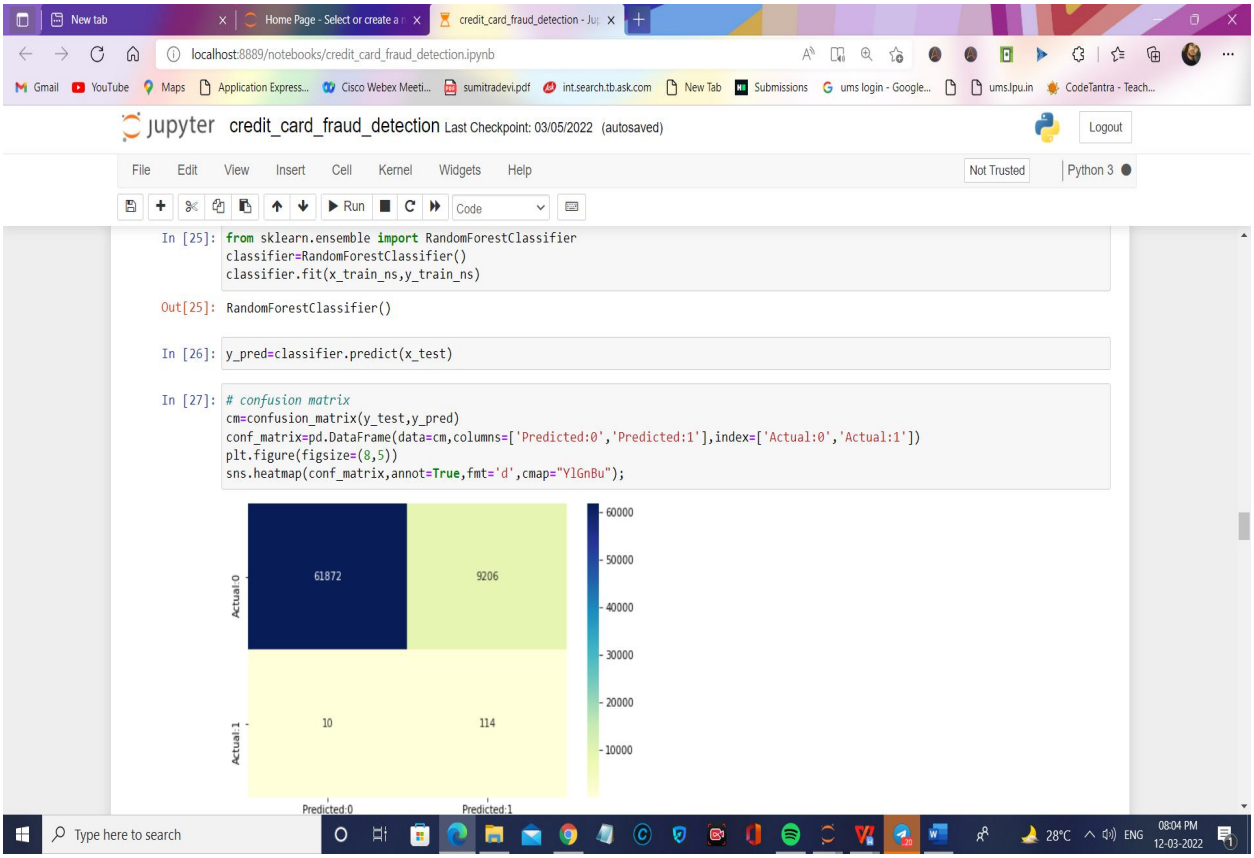

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
71042
36
36
88
Actual:0
Actual:1
Predicted:0
Predicted:1
70000
60000
50000
40000
30000
20000
10000
In [15]: print(accuracy_score(y_test,y_pred))
0.9989887924496503
In [16]: print(classification_report(y_test,y_pred))
precision recall f1-score support
0 1.00 1.00 1.00 71078
1 0.71 0.71 0.71 124
Type here to search
28°C
ENG
07:58 PM
12-03-2022

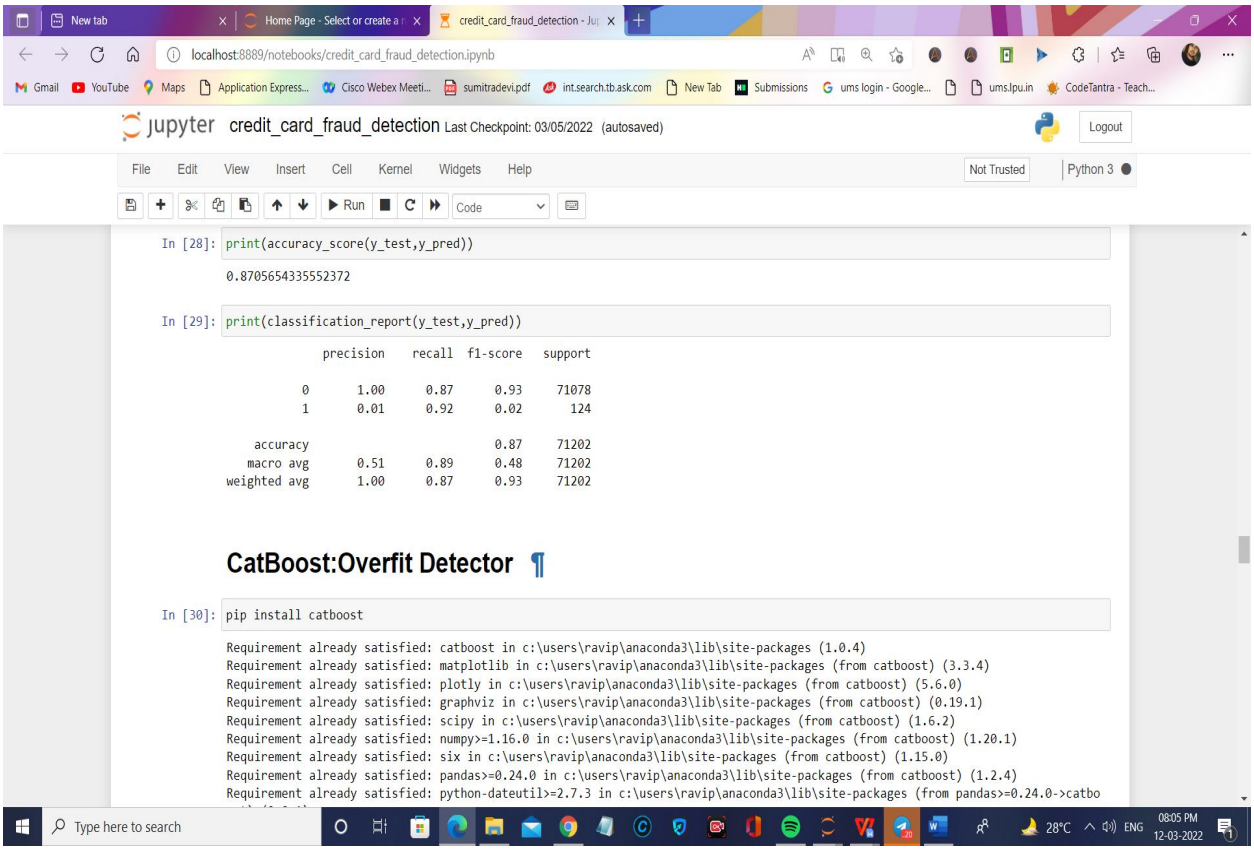

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [15]: print(accuracy_score(y_test,y_pred))
0.9989887924496503
In [16]: print(classification_report(y_test,y_pred))
precision recall f1-score support
0 1.00 1.00 1.00 71078
1 0.71 0.71 0.71 124
accuracy 1.00 71202
macro avg 0.85 0.85 0.85 71202
weighted avg 1.00 1.00 1.00 71202
Random Forest classifier:
In [17]: from sklearn.ensemble import RandomForestClassifier
classifier=RandomForestClassifier(criterion='gini',max_depth=10,min_samples_split=5,min_samples_leaf=1)
classifier.fit(x_train,y_train)
Out[17]: RandomForestClassifier(max_depth=10, min_samples_split=5)
In [18]: y_pred=classifier.predict(x_test)
In [19]: # confusion matrix
Type here to search
28°C
ENG
07:58 PM
12-03-2022

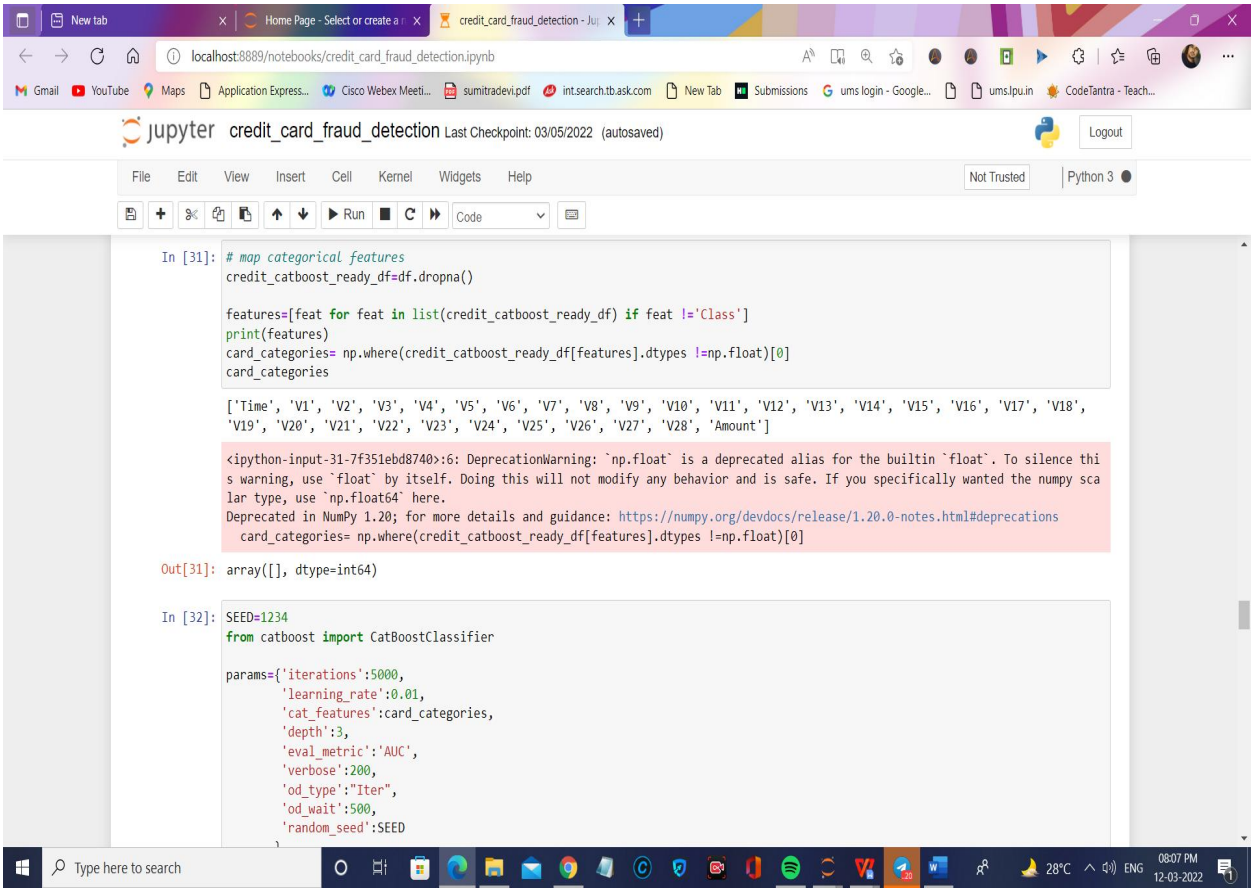

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [19]:
# confusion matrix
cm=confusion_matrix(y_test,y_pred)
conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','Actual:1'])
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");
71073
5
28
96
Actual:0
Actual:1
Predicted:0
Predicted:1
70000
60000
50000
40000
30000
20000
10000
In [20]:
print(accuracy_score(y_test,y_pred))
0.9995365298727564
In [21]:
print(classification_report(y_test,y_pred))
Type here to search
28°C
ENG
08:00 PM
12-03-2022

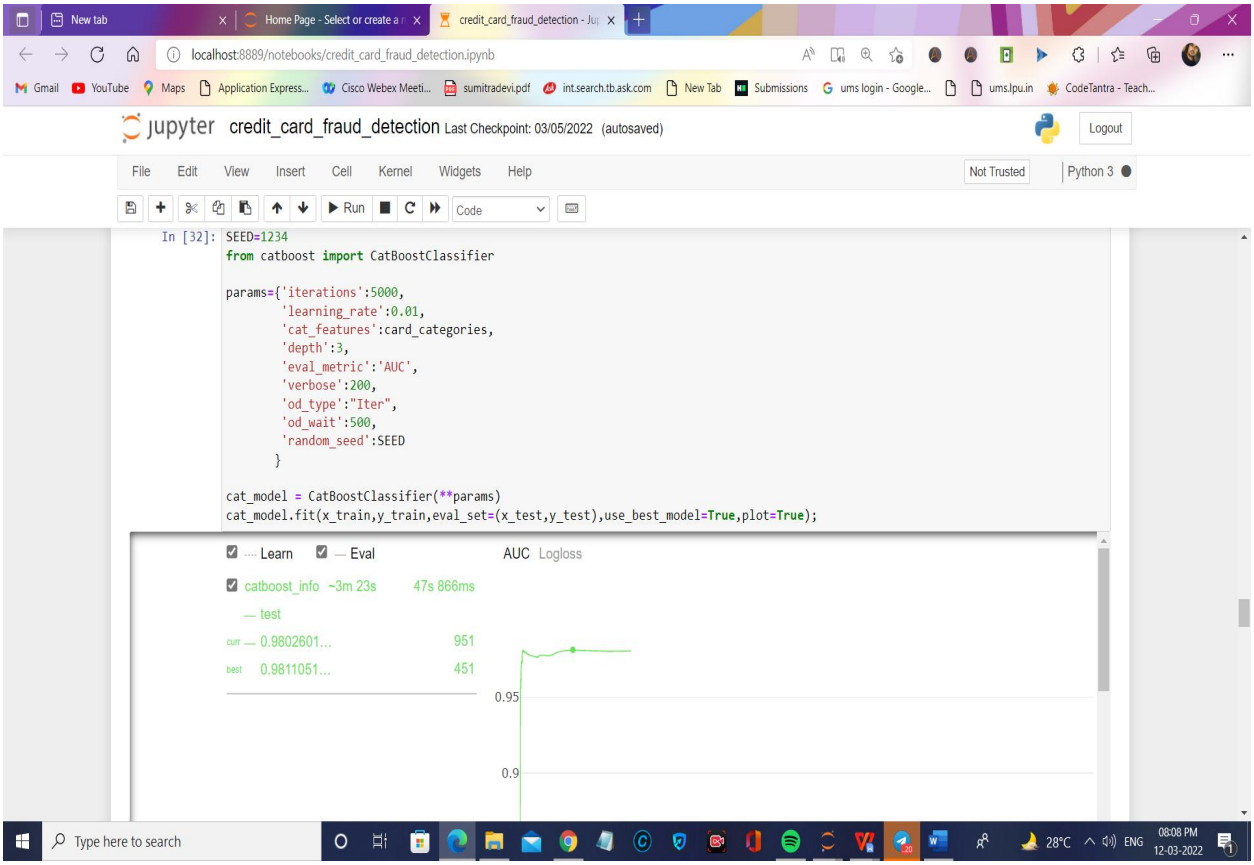

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [20]: print(accuracy_score(y_test,y_pred))
0.9995365298727564
In [21]: print(classification_report(y_test,y_pred))
precision recall f1-score support
0 1.00 1.00 1.00 71078
1 0.95 0.77 0.85 124
accuracy 1.00 71202
macro avg 0.98 0.89 0.93 71202
weighted avg 1.00 1.00 1.00 71202
Importing imblearn library for performing under sampling: ¶
In [22]: pip install imbalanced-learn
Requirement already satisfied: imbalanced-learn in c:\users\ravip\anaconda3\lib\site-packages (0.9.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (2.1.0)
Requirement already satisfied: numpy>=1.14.6 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.20.1)
Requirement already satisfied: scikit-learn>=1.0.1 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.0.2)
Requirement already satisfied: joblib>=0.11 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.0.1)
Requirement already satisfied: scipy>=1.1.0 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.6.2)
Note: you may need to restart the kernel to use updated packages.
Type here to search
28°C
ENG
08:02 PM
12-03-2022

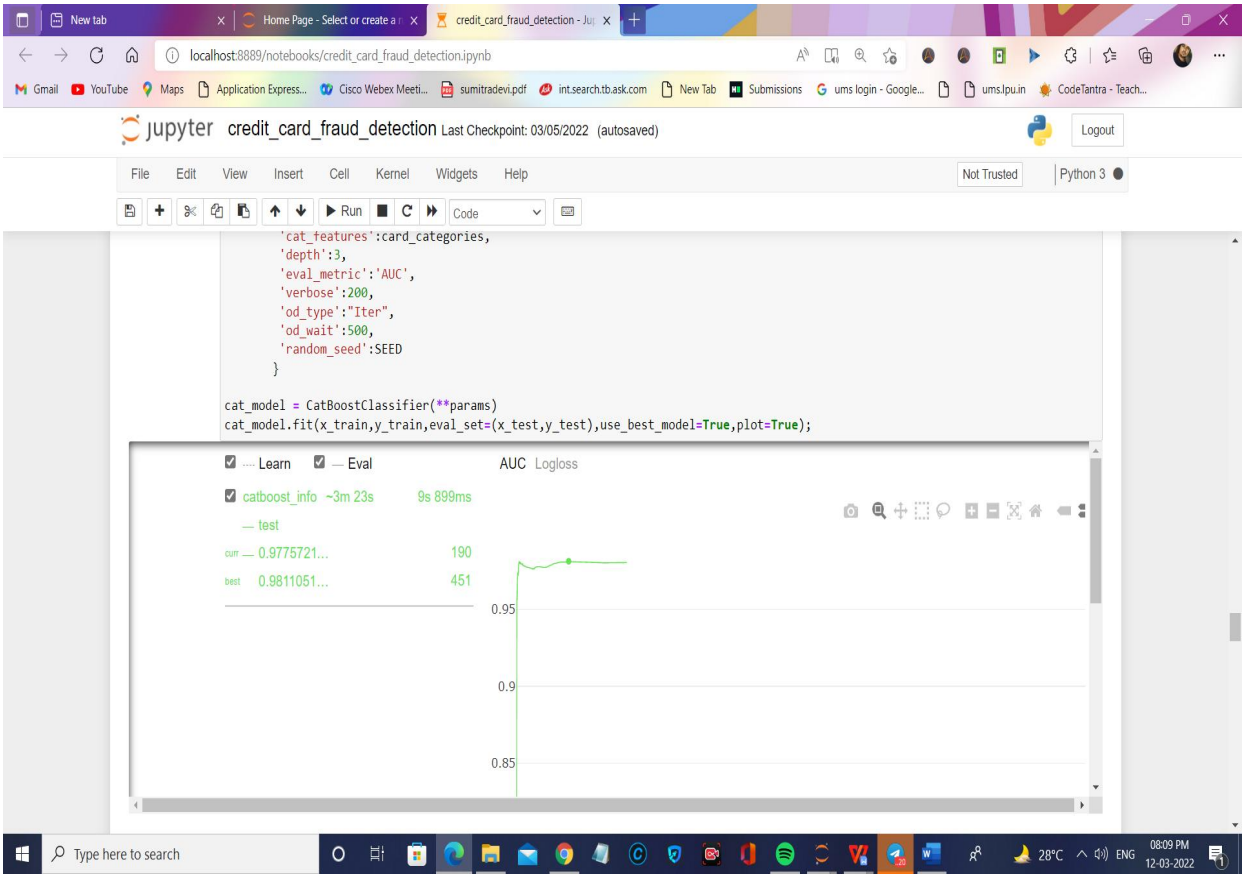

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
0)
Requirement already satisfied: numpy>=1.14.6 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.20.1)
Requirement already satisfied: scikit-learn>=1.0.1 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.0.
2)
Requirement already satisfied: joblib>=0.11 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.0.1)
Requirement already satisfied: scipy>=1.1.0 in c:\users\ravip\anaconda3\lib\site-packages (from imbalanced-learn) (1.6.2)
Note: you may need to restart the kernel to use updated packages.
Performing Under sampling:NearMiss
In [23]:
from collections import Counter
Counter(y_train)
Out[23]:
Counter({0: 213237, 1: 368})
In [24]:
from collections import Counter
from imblearn.under_sampling import NearMiss
ns=NearMiss(version=1,n_neighbors=3)
x_train_ns,y_train_ns=ns.fit_resample(x_train,y_train)
print("The number of classes before fit {}".format(Counter(y_train)))
print("The number of classes after fit {}".format(Counter(y_train_ns)))
The number of classes before fit Counter({0: 213237, 1: 368})
The number of classes after fit Counter({0: 368, 1: 368})
In [25]:
from sklearn.ensemble import RandomForestClassifier
classifier=RandomForestClassifier()
classifier.fit(x_train_ns,y_train_ns)
Type here to search
28°C
ENG
08:03 PM
12-03-2022

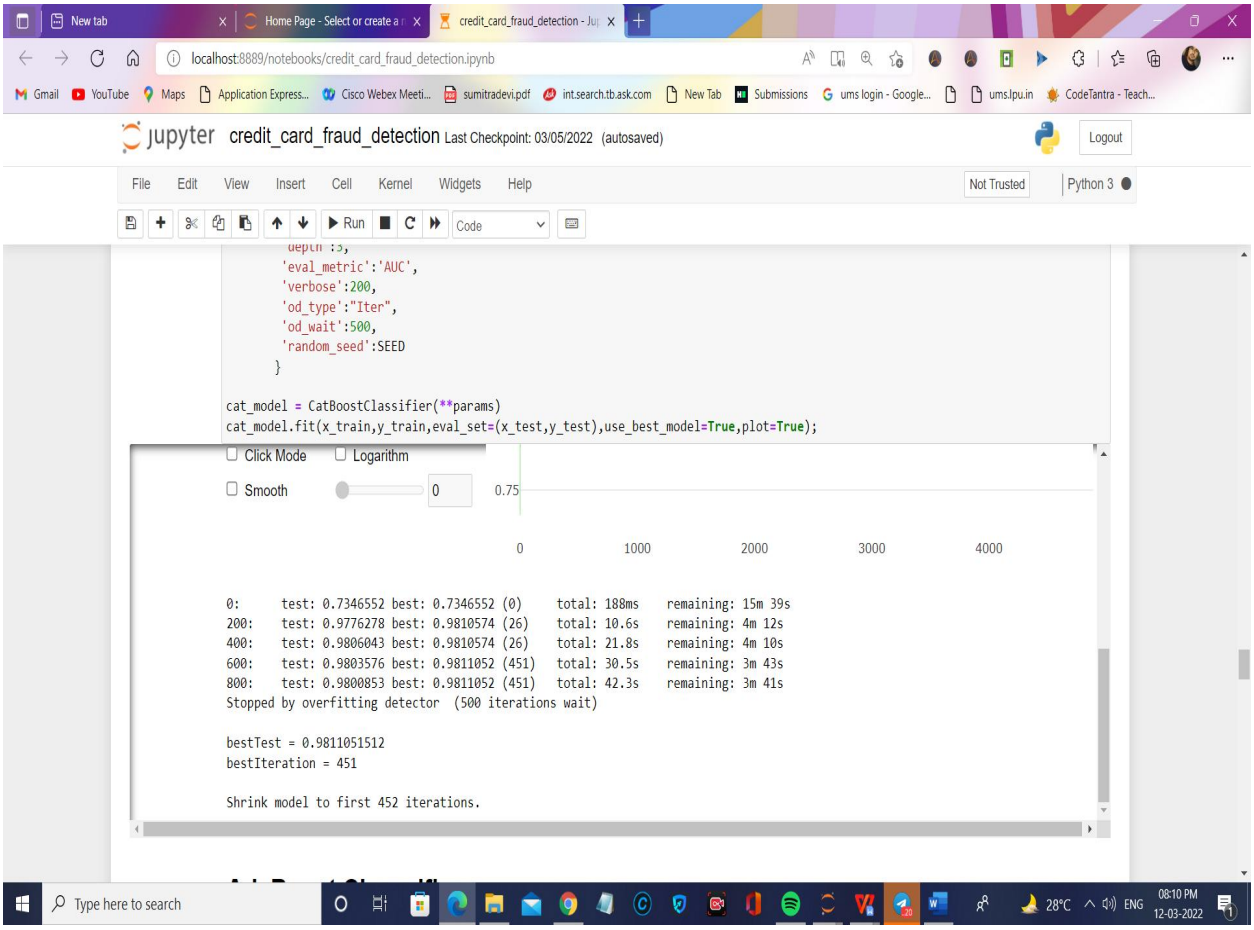

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [25]:
from sklearn.ensemble import RandomForestClassifier
classifier=RandomForestClassifier()
classifier.fit(x_train_ns,y_train_ns)
Out[25]:
RandomForestClassifier()
In [26]:
y_pred=classifier.predict(x_test)
In [27]:
# confusion matrix
cm=confusion_matrix(y_test,y_pred)
conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','Actual:1'])
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");
61872
9206
10
114
Actual:0
Actual:1
Predicted:0
Predicted:1
60000
50000
40000
30000
20000
10000
Type here to search
28°C
ENG
08:04 PM
12-03-2022

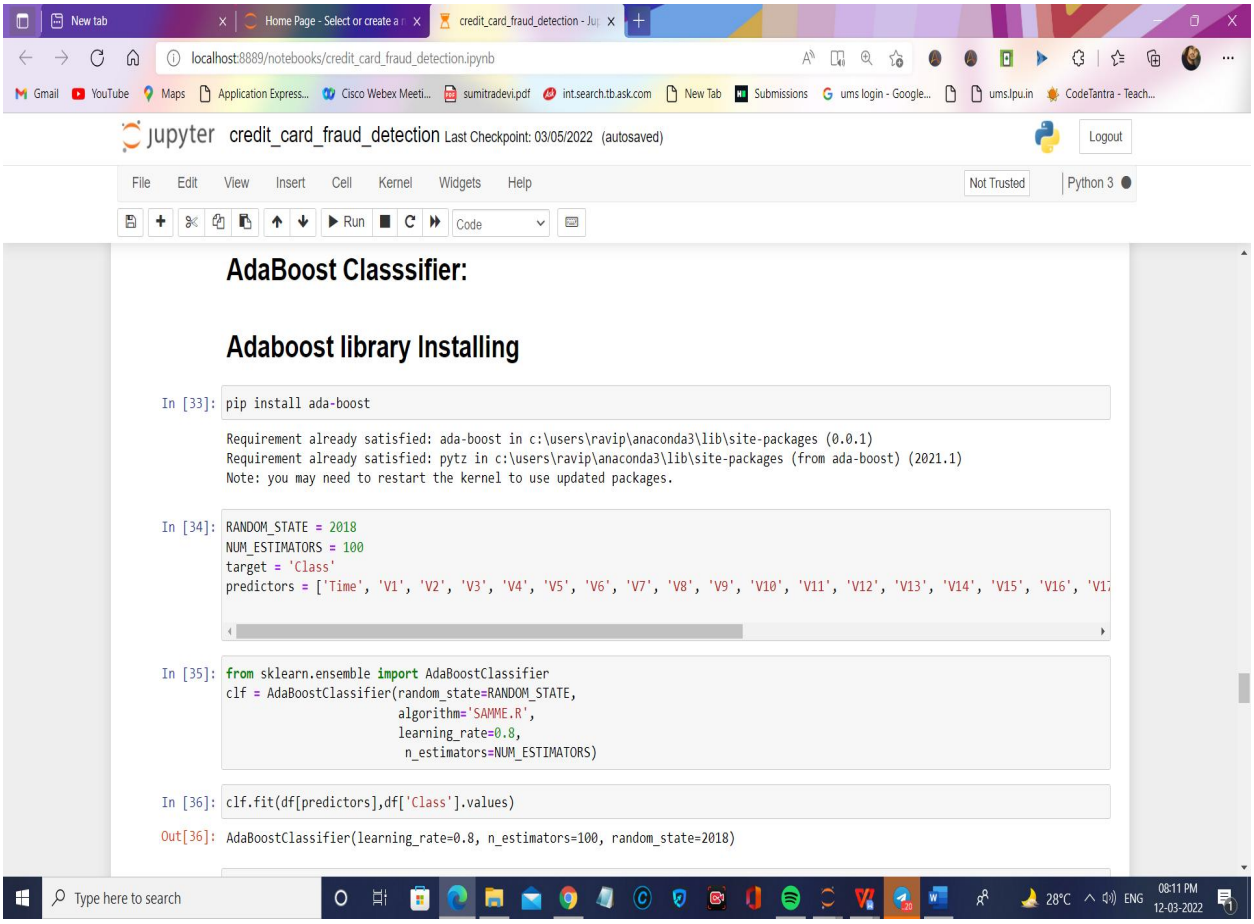

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [28]: print(accuracy_score(y_test,y_pred))
0.8705654335552372
In [29]: print(classification_report(y_test,y_pred))
precision recall f1-score support
0 1.00 0.87 0.93 71078
1 0.01 0.92 0.02 124
accuracy 0.87 71202
macro avg 0.51 0.89 0.48 71202
weighted avg 1.00 0.87 0.93 71202
CatBoost:Overfit Detector ¶
In [30]: pip install catboost
Requirement already satisfied: catboost in c:\users\ravip\anaconda3\lib\site-packages (1.0.4)
Requirement already satisfied: matplotlib in c:\users\ravip\anaconda3\lib\site-packages (from catboost) (3.3.4)
Requirement already satisfied: plotly in c:\users\ravip\anaconda3\lib\site-packages (from catboost) (5.6.0)
Requirement already satisfied: graphviz in c:\users\ravip\anaconda3\lib\site-packages (from catboost) (0.19.1)
Requirement already satisfied: scipy in c:\users\ravip\anaconda3\lib\site-packages (from catboost) (1.6.2)
Requirement already satisfied: numpy>=1.16.0 in c:\users\ravip\anaconda3\lib\site-packages (from catboost) (1.20.1)
Requirement already satisfied: six in c:\users\ravip\anaconda3\lib\site-packages (from catboost) (1.15.0)
Requirement already satisfied: pandas>=0.24.0 in c:\users\ravip\anaconda3\lib\site-packages (from catboost) (1.2.4)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\ravip\anaconda3\lib\site-packages (from pandas>=0.24.0->catbo
Type here to search
28°C
ENG
08:05 PM
12-03-2022

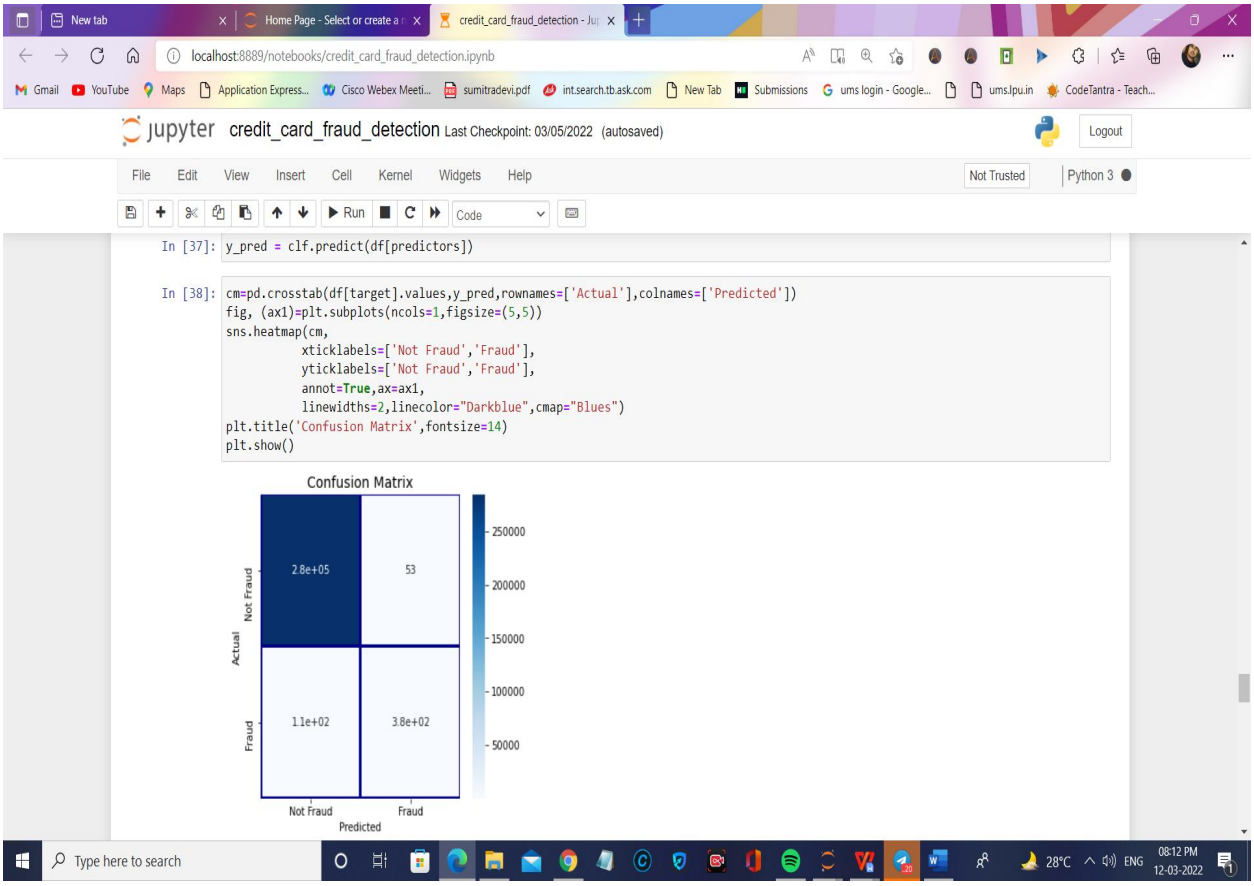

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [31]:
# map categorical features
credit_catboost_ready_df=df.dropna()
features=[feat for feat in list(credit_catboost_ready_df) if feat !='Class']
print(features)
card_categories= np.where(credit_catboost_ready_df[features].dtypes !=np.float)[0]
card_categories
['Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9', 'V10', 'V11', 'V12', 'V13', 'V14', 'V15', 'V16', 'V17', 'V18', 'V19', 'V20', 'V21', 'V22', 'V23', 'V24', 'V25', 'V26', 'V27', 'V28', 'Amount']
<ipython-input-31-7f351ebd8740>:6: DeprecationWarning: `np.float` is a deprecated alias for the builtin `float`. To silence this warning, use `float` by itself. Doing this will not modify any behavior and is safe. If you specifically wanted the numpy scalar type, use `np.float64` here.
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations
card_categories= np.where(credit_catboost_ready_df[features].dtypes !=np.float)[0]
Out[31]:
array([], dtype=int64)
In [32]:
SEED=1234
from catboost import CatBoostClassifier
params={'iterations':5000,
'learning_rate':0.01,
'cat_features':card_categories,
'depth':3,
'eval_metric':'AUC',
'verbose':200,
'od_type':"Iter",
'od_wait':500,
'random_seed':SEED
Type here to search
28°C
ENG
08:07 PM
12-03-2022

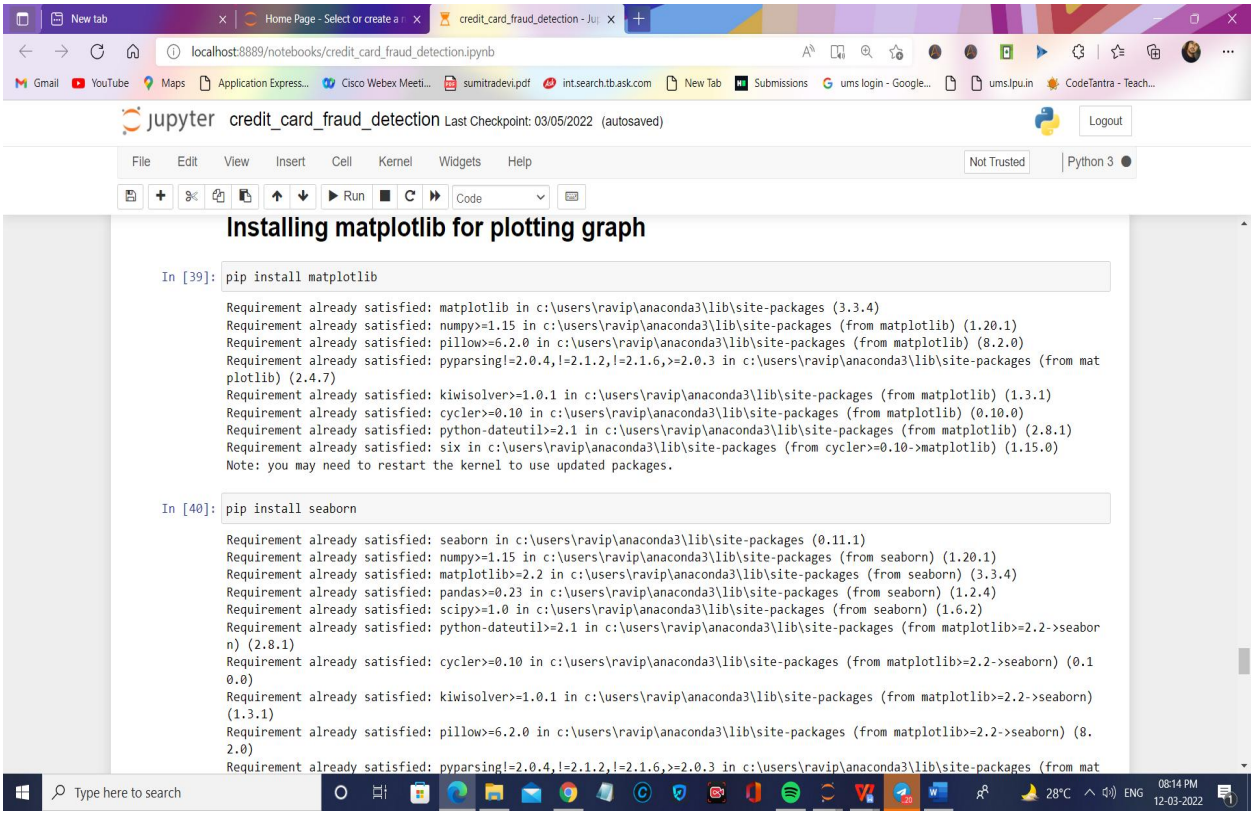

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [32]:
SEED=1234
from catboost import CatBoostClassifier
params={'iterations':5000,
'learning_rate':0.01,
'cat_features':card_categories,
'depth':3,
'eval_metric':'AUC',
'verbose':200,
'od_type':"Iter",
'od_wait':500,
'random_seed':SEED
}
cat_model = CatBoostClassifier(**params)
cat_model.fit(x_train,y_train,eval_set=(x_test,y_test),use_best_model=True,plot=True);
Learn
Eval
AUC Logloss
catboost_info ~3m 23s
47s 866ms
test
curr 0.9802601...
951
best 0.9811051...
451
0.95
0.9
Type here to search
28°C
ENG
08:08 PM
12-03-2022

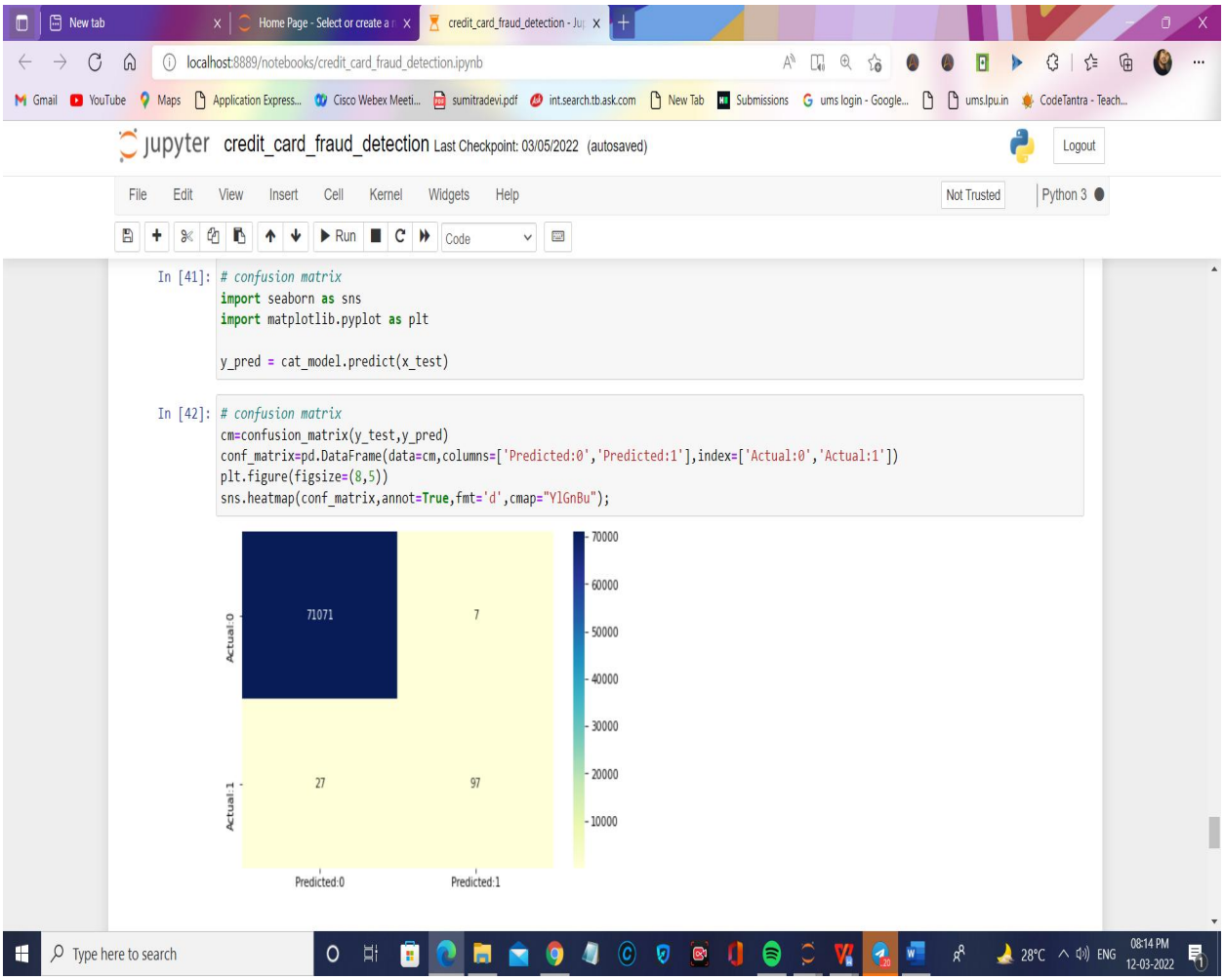

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
'cat_features':card_categories,
'depth':3,
'eval_metric':'AUC',
'verbose':200,
'od_type':"Iter",
'od_wait':500,
'random_seed':SEED
}
cat_model = CatBoostClassifier(**params)
cat_model.fit(x_train,y_train,eval_set=(x_test,y_test),use_best_model=True,plot=True);
Learn
Eval
AUC Logloss
catboost_info ~3m 23s
9s 899ms
test
curr 0.9775721...
190
best 0.9811051...
451
0.95
0.9
0.85
Type here to search
28°C
ENG
08:09 PM
12-03-2022

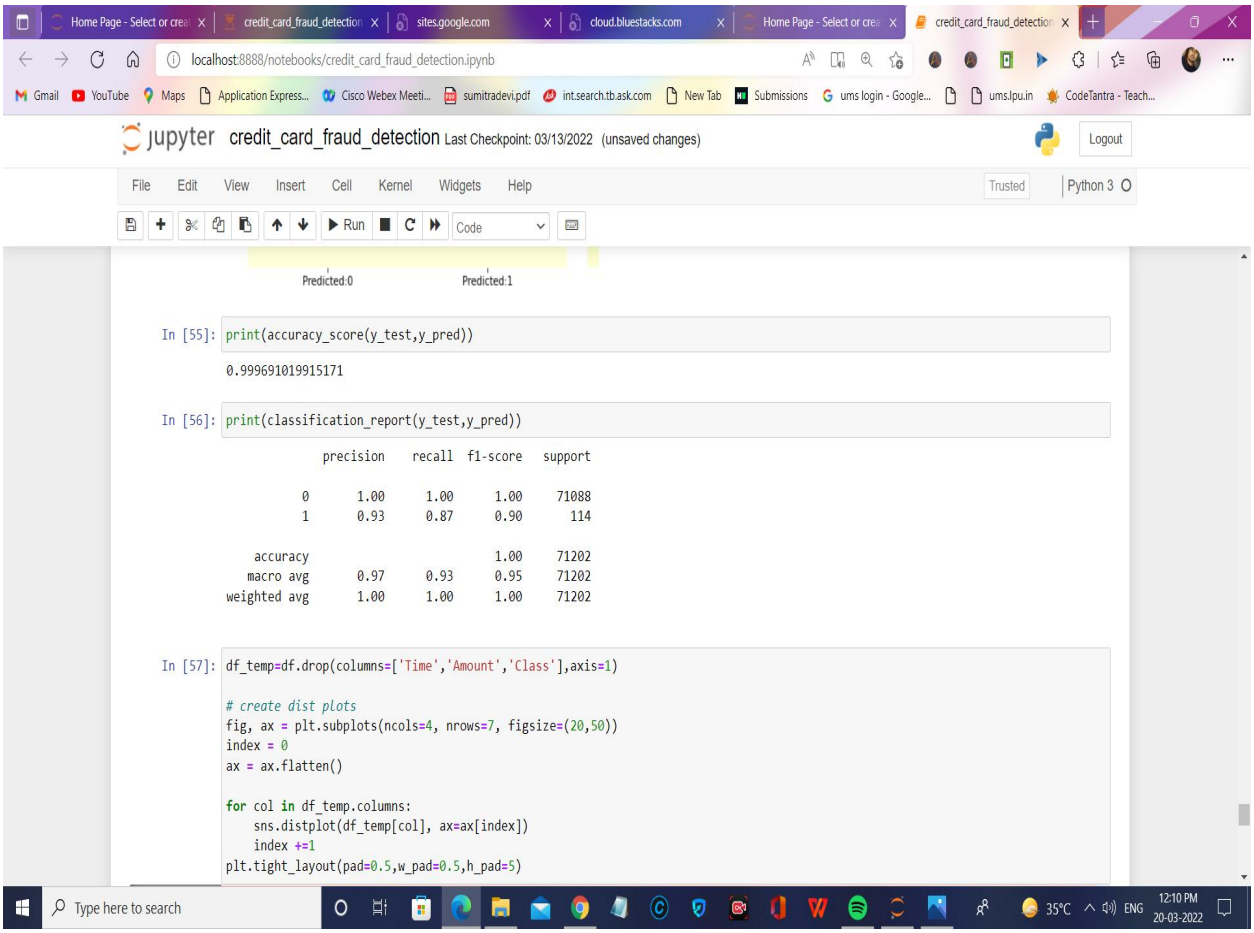

New tab
Home Page - Select or create a
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
'depth':5,
'eval_metric':'AUC',
'verbose':200,
'od_type':"Iter",
'od_wait':500,
'random_seed':SEED
}
cat_model = CatBoostClassifier(**params)
cat_model.fit(x_train,y_train,eval_set=(x_test,y_test),use_best_model=True,plot=True);
Click Mode
Logarithm
Smooth
0
0.75
0
1000
2000
3000
4000
0: test: 0.7346552 best: 0.7346552 (0) total: 188ms remaining: 15m 39s
200: test: 0.9776278 best: 0.9810574 (26) total: 10.6s remaining: 4m 12s
400: test: 0.9806043 best: 0.9810574 (26) total: 21.8s remaining: 4m 10s
600: test: 0.9803576 best: 0.9811052 (451) total: 30.5s remaining: 3m 43s
800: test: 0.9800853 best: 0.9811052 (451) total: 42.3s remaining: 3m 41s
Stopped by overfitting detector (500 iterations wait)
bestTest = 0.9811051512
bestIteration = 451
Shrink model to first 452 iterations.
Type here to search
28°C
ENG
08:10 PM
12-03-2022

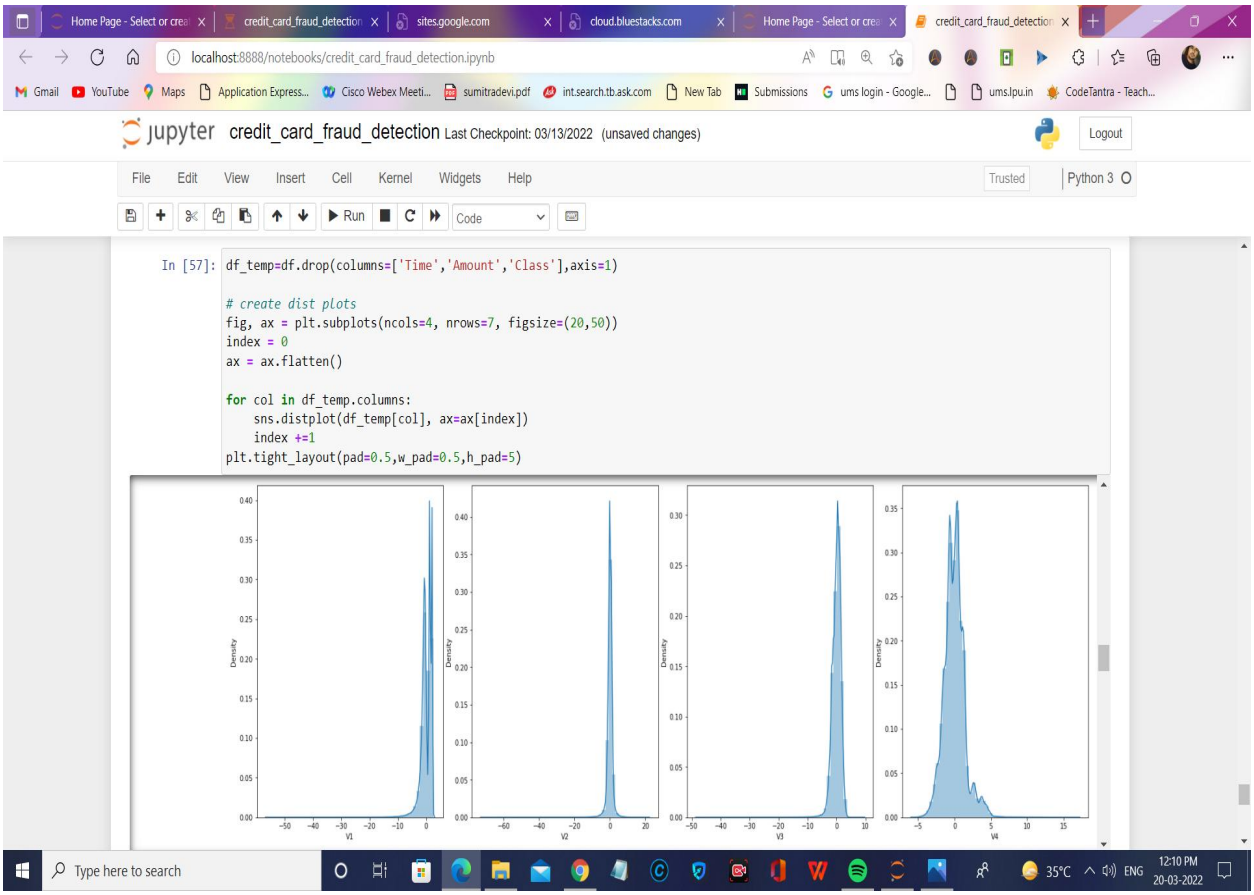

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
AdaBoost Classsifier:
Adaboost library Installing
In [33]:
pip install ada-boost
Requirement already satisfied: ada-boost in c:\users\ravip\anaconda3\lib\site-packages (0.0.1)
Requirement already satisfied: pytz in c:\users\ravip\anaconda3\lib\site-packages (from ada-boost) (2021.1)
Note: you may need to restart the kernel to use updated packages.
In [34]:
RANDOM_STATE = 2018
NUM_ESTIMATORS = 100
target = 'Class'
predictors = ['Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9', 'V10', 'V11', 'V12', 'V13', 'V14', 'V15', 'V16', 'V17
In [35]:
from sklearn.ensemble import AdaBoostClassifier
clf = AdaBoostClassifier(random_state=RANDOM_STATE,
algorithm='SAMME.R',
learning_rate=0.8,
n_estimators=NUM_ESTIMATORS)
In [36]:
clf.fit(df[predictors],df['Class'].values)
Out[36]:
AdaBoostClassifier(learning_rate=0.8, n_estimators=100, random_state=2018)
Type here to search
28°C
ENG
08:11 PM
12-03-2022

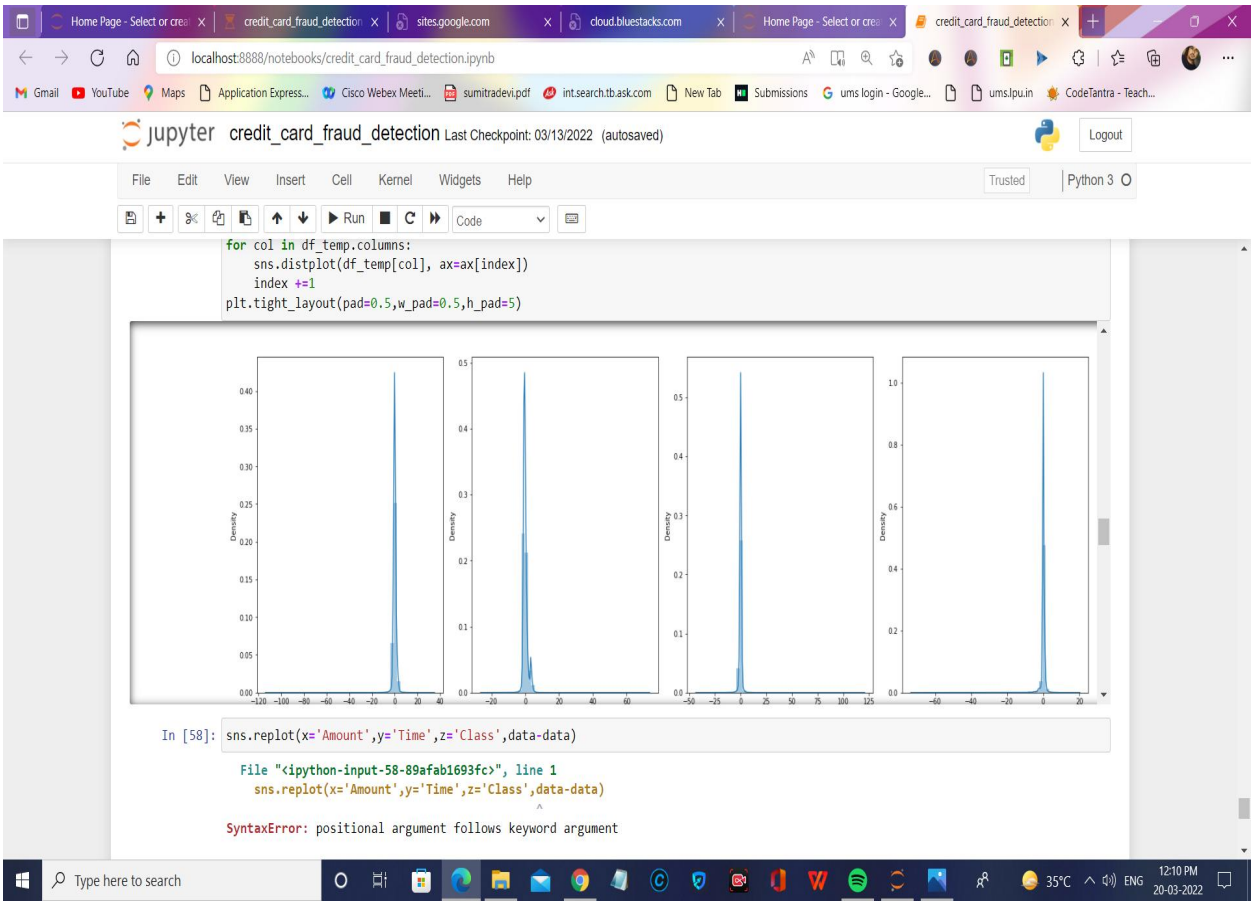

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [37]:
y_pred = clf.predict(df[predictors])
In [38]:
cm=pd.crosstab(df[target].values,y_pred,rownames=['Actual'],colnames=['Predicted'])
fig, (ax1)=plt.subplots(ncols=1,figsize=(5,5))
sns.heatmap(cm,
xticklabels=['Not Fraud','Fraud'],
yticklabels=['Not Fraud','Fraud'],
annot=True,ax=ax1,
linewidths=2,linecolor="Darkblue",cmap="Blues")
plt.title('Confusion Matrix',fontsize=14)
plt.show()
Confusion Matrix
2.8e+05
53
1.1e+02
3.8e+02
Not Fraud
Fraud
Actual
Not Fraud
Fraud
Predicted
250000
200000
150000
100000
50000
Type here to search
28°C
ENG
08:12 PM
12-03-2022

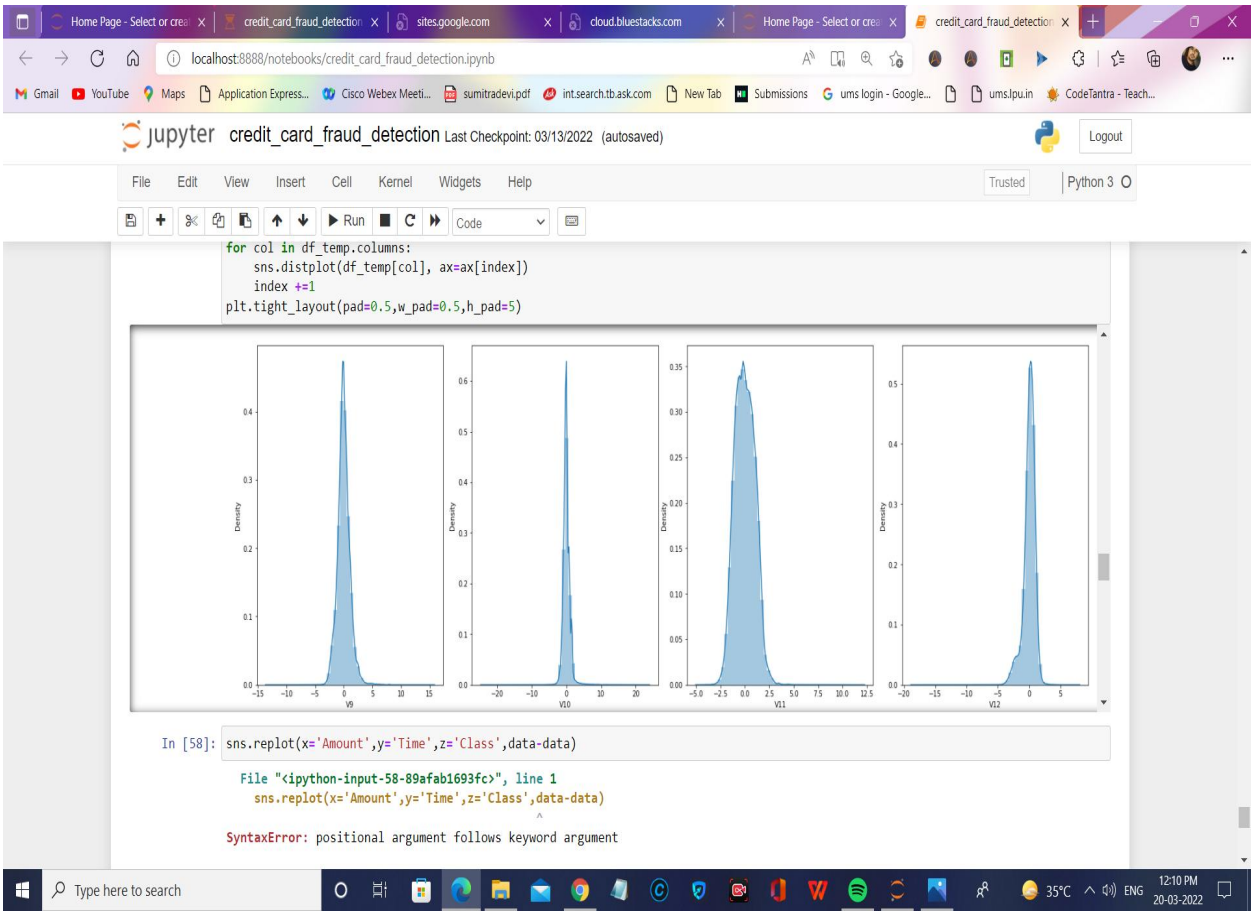

New tab
Home Page - Select or create a r
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
Installing matplotlib for plotting graph
In [39]: pip install matplotlib
Requirement already satisfied: matplotlib in c:\users\ravip\anaconda3\lib\site-packages (3.3.4)
Requirement already satisfied: numpy>=1.15 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib) (1.20.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib) (8.2.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib) (2.4.7)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib) (1.3.1)
Requirement already satisfied: cycler>=0.10 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib) (2.8.1)
Requirement already satisfied: six in c:\users\ravip\anaconda3\lib\site-packages (from cycler>=0.10->matplotlib) (1.15.0)
Note: you may need to restart the kernel to use updated packages.
In [40]: pip install seaborn
Requirement already satisfied: seaborn in c:\users\ravip\anaconda3\lib\site-packages (0.11.1)
Requirement already satisfied: numpy>=1.15 in c:\users\ravip\anaconda3\lib\site-packages (from seaborn) (1.20.1)
Requirement already satisfied: matplotlib>=2.2 in c:\users\ravip\anaconda3\lib\site-packages (from seaborn) (3.3.4)
Requirement already satisfied: pandas>=0.23 in c:\users\ravip\anaconda3\lib\site-packages (from seaborn) (1.2.4)
Requirement already satisfied: scipy>=1.0 in c:\users\ravip\anaconda3\lib\site-packages (from seaborn) (1.6.2)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (2.8.1)
Requirement already satisfied: cycler>=0.10 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (1.3.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\ravip\anaconda3\lib\site-packages (from matplotlib>=2.2->seaborn) (8.2.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in c:\users\ravip\anaconda3\lib\site-packages (from mat
Type here to search
28°C
ENG
08:14 PM
12-03-2022

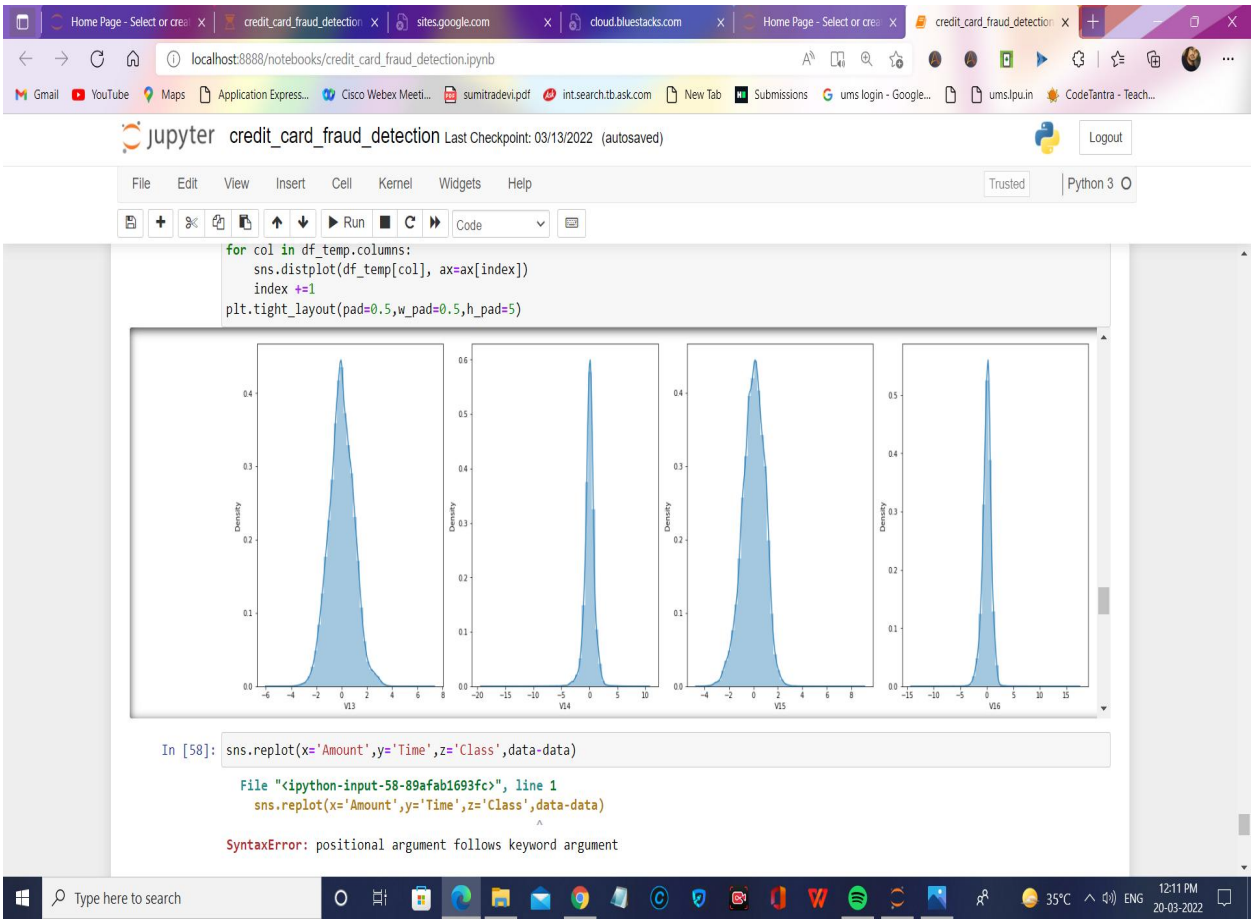

New tab
Home Page - Select or create a n
credit_card_fraud_detection - Ju
localhost:8889/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/05/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3
Run
Code
In [41]:
# confusion matrix
import seaborn as sns
import matplotlib.pyplot as plt
y_pred = cat_model.predict(x_test)
In [42]:
# confusion matrix
cm=confusion_matrix(y_test,y_pred)
conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','Actual:1'])
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");
71071
7
27
97
Actual:0
Actual:1
Predicted:0
Predicted:1
70000
60000
50000
40000
30000
20000
10000
Type here to search
28°C
ENG
08:14 PM
12-03-2022


Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
Home Page - Select or crea
credit_card_fraud_detection
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/13/2022 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
Predicted:0
Predicted:1
In [55]:
print(accuracy_score(y_test,y_pred))
0.999691019915171
In [56]:
print(classification_report(y_test,y_pred))
precision recall f1-score support
0 1.00 1.00 1.00 71088
1 0.93 0.87 0.90 114
accuracy 1.00 71202
macro avg 0.97 0.93 0.95 71202
weighted avg 1.00 1.00 1.00 71202
In [57]:
df_temp=df.drop(columns=['Time','Amount','Class'],axis=1)
# create dist plots
fig, ax = plt.subplots(ncols=4, nrows=7, figsize=(20,50))
index = 0
ax = ax.flatten()
for col in df_temp.columns:
sns.distplot(df_temp[col], ax=ax[index])
index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
Type here to search
35°C
ENG
12:10 PM
20-03-2022

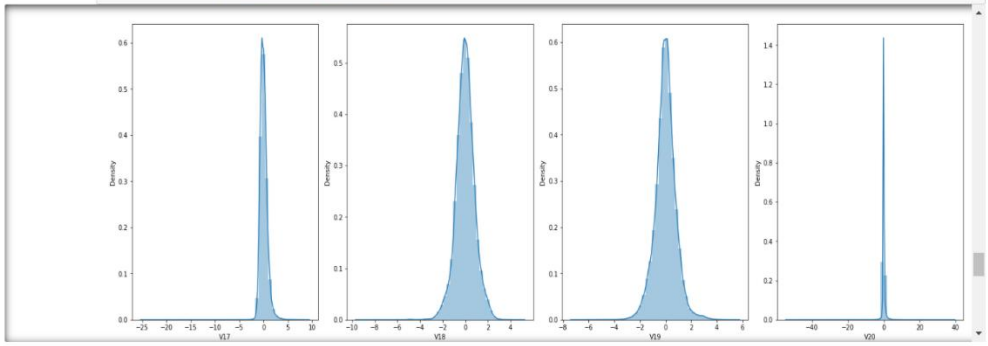

Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/13/2022 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [57]:
df_temp=df.drop(columns=['Time','Amount','Class'],axis=1)
# create dist plots
fig, ax = plt.subplots(ncols=4, nrows=7, figsize=(20,50))
index = 0
ax = ax.flatten()
for col in df_temp.columns:
sns.distplot(df_temp[col], ax=ax[index])
index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
Density
V1
V2
V3
V4
Type here to search
35°C
ENG
12:10 PM
20-03-2022


Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/13/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
for col in df_temp.columns:
sns.distplot(df_temp[col], ax=ax[index])
index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
Density
In [58]: sns.replot(x='Amount',y='Time',z='Class',data-data)
File "<ipython-input-58-89afab1693fc>", line 1
sns.replot(x='Amount',y='Time',z='Class',data-data)
^
SyntaxError: positional argument follows keyword argument
Type here to search
35°C
ENG
12:10 PM
20-03-2022


Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
Home Page - Select or crea
credit_card_fraud_detection
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/13/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
for col in df_temp.columns:
sns.distplot(df_temp[col], ax=ax[index])
index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
Density
V9
V10
V11
V12
In [58]: sns.replot(x='Amount',y='Time',z='Class',data-data)
File "<ipython-input-58-89afab1693fc>", line 1
sns.replot(x='Amount',y='Time',z='Class',data-data)
SyntaxError: positional argument follows keyword argument
Type here to search
35°C
ENG
12:10 PM
20-03-2022

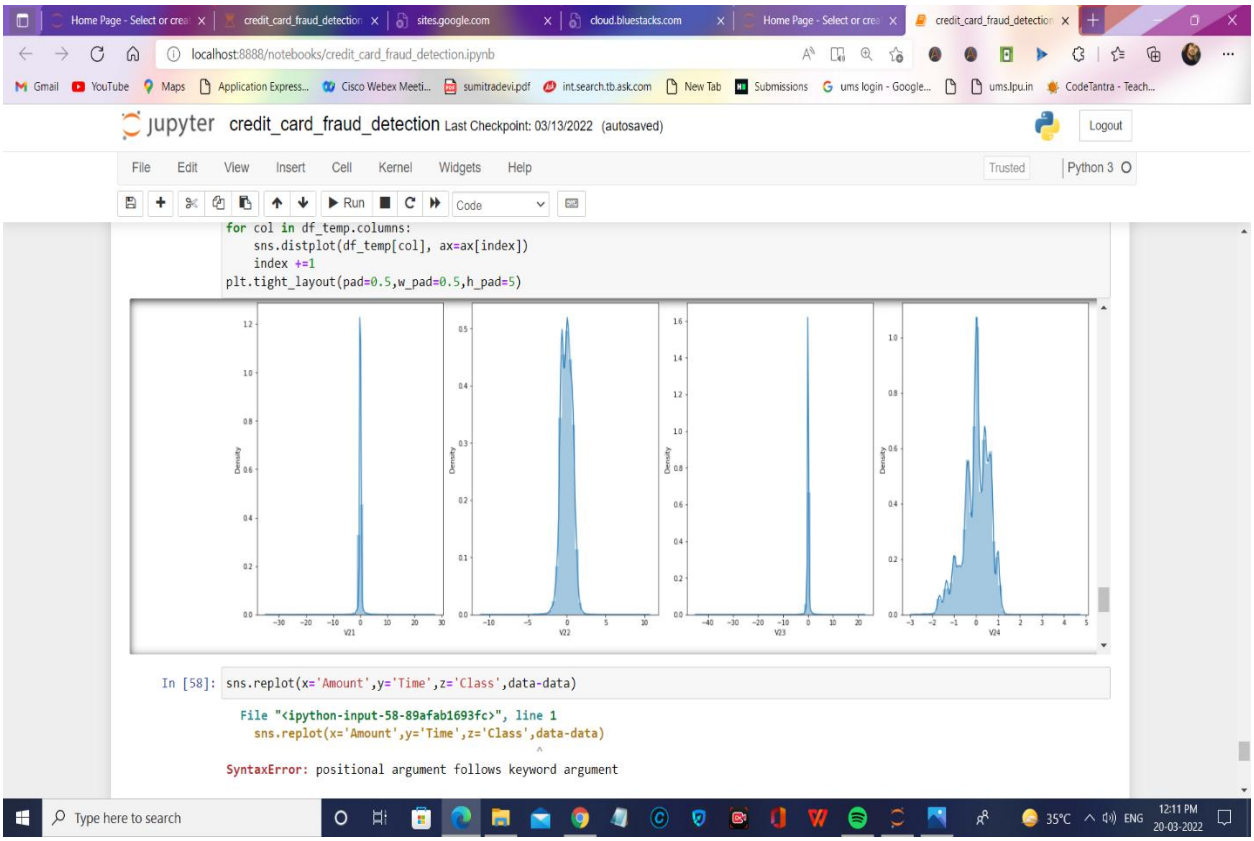

Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
Home Page - Select or crea
credit_card_fraud_detection
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/13/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
for col in df_temp.columns:
sns.distplot(df_temp[col], ax=ax[index])
index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
Density
V13
V14
V15
V16
In [58]: sns.replot(x='Amount',y='Time',z='Class',data=data)
File "<ipython-input-58-89afab1693fc>", line 1
sns.replot(x='Amount',y='Time',z='Class',data=data)
^
SyntaxError: positional argument follows keyword argument
Type here to search
35°C
ENG
12:11 PM
20-03-2022

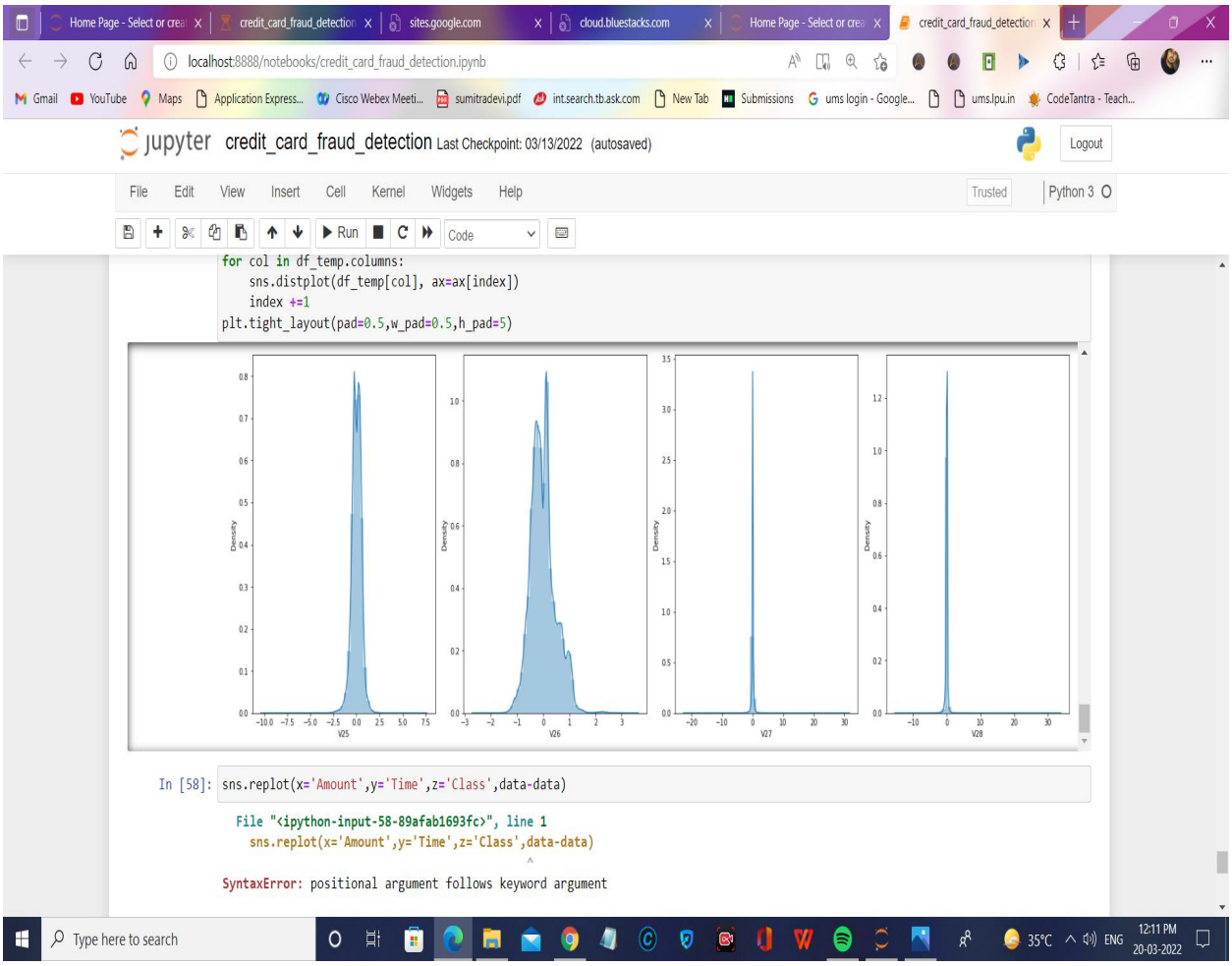
```
for col in df_temp.columns:
    sns.distplot(df_temp[col], ax=ax[index])
    index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
```

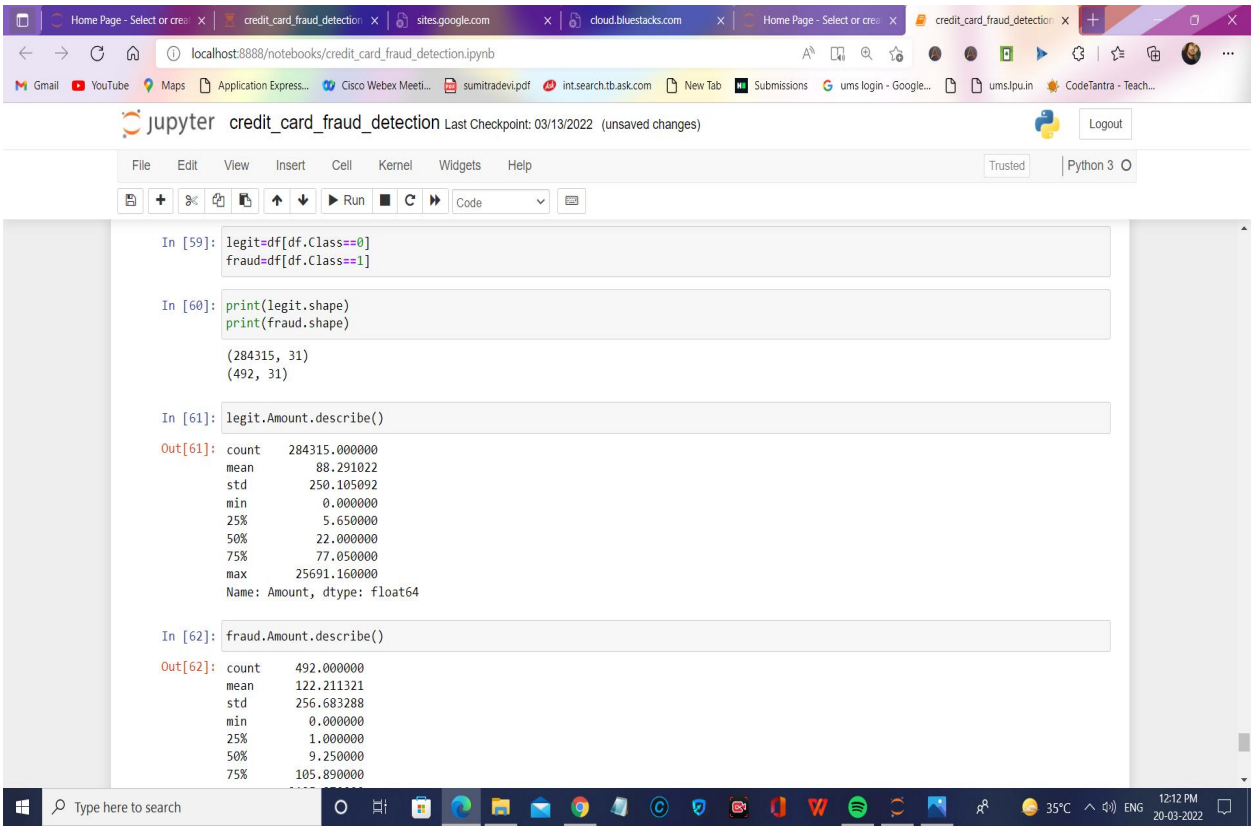


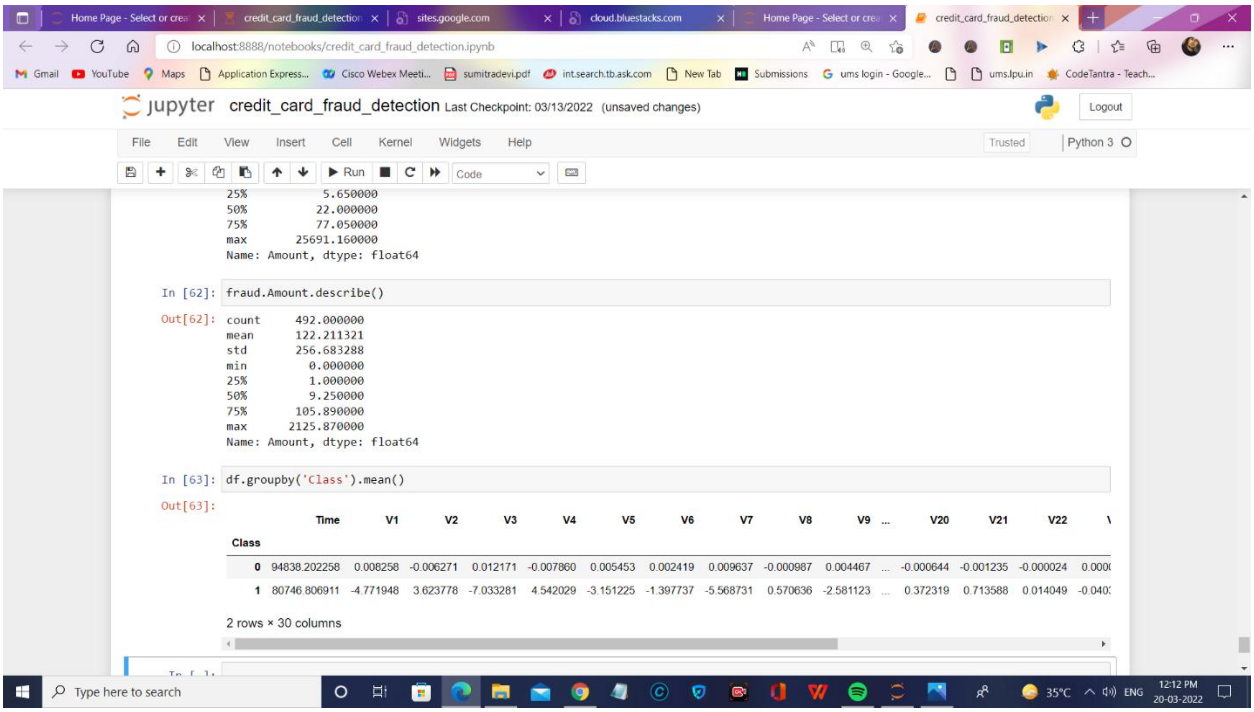
In [58]:
```
sns.replot(x='Amount',y='Time',z='Class',data-data)
```

```
  File "<ipython-input-58-89afab1693fc>", line 1
    sns.replot(x='Amount',y='Time',z='Class',data-data)
                                             ^
SyntaxError: positional argument follows keyword argument
```


Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
Home Page - Select or crea
credit_card_fraud_detection
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter
credit_card_fraud_detection
Last Checkpoint: 03/13/2022 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code
for col in df_temp.columns:
sns.distplot(df_temp[col], ax=ax[index])
index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
V21
V22
V23
V24
Density
In [58]:
sns.replot(x='Amount',y='Time',z='Class',data-data)
File "<ipython-input-58-89afab1693fc>", line 1
sns.replot(x='Amount',y='Time',z='Class',data-data)
SyntaxError: positional argument follows keyword argument
Type here to search
35°C
ENG
12:11 PM
20-03-2022


Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
Home Page - Select or crea
credit_card_fraud_detection
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/13/2022 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
for col in df_temp.columns:
sns.distplot(df_temp[col], ax=ax[index])
index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
Density
V25
V26
V27
V28
In [58]: sns.replot(x='Amount',y='Time',z='Class',data-data)
File "<ipython-input-58-89afab1693fc>", line 1
sns.replot(x='Amount',y='Time',z='Class',data-data)
SyntaxError: positional argument follows keyword argument
Type here to search
35°C
ENG
12:11 PM
20-03-2022


Home Page - Select or crea
credit_card_fraud_detection
sites.google.com
cloud.bluestacks.com
Home Page - Select or crea
credit_card_fraud_detection
localhost:8888/notebooks/credit_card_fraud_detection.ipynb
Gmail
YouTube
Maps
Application Express...
Cisco Webex Meeti...
sumitradevi.pdf
int.search.tb.ask.com
New Tab
Submissions
ums login - Google...
ums.lpu.in
CodeTantra - Teach...
jupyter credit_card_fraud_detection Last Checkpoint: 03/13/2022 (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [59]:
legit=df[df.Class==0]
fraud=df[df.Class==1]
In [60]:
print(legit.shape)
print(fraud.shape)
(284315, 31)
(492, 31)
In [61]:
legit.Amount.describe()
Out[61]:
count 284315.000000
mean 88.291022
std 250.105092
min 0.000000
25% 5.650000
50% 22.000000
75% 77.050000
max 25691.160000
Name: Amount, dtype: float64
In [62]:
fraud.Amount.describe()
Out[62]:
count 492.000000
mean 122.211321
std 256.683288
min 0.000000
25% 1.000000
50% 9.250000
75% 105.890000
Type here to search
35°C
ENG
12:12 PM
20-03-2022



Pseudo code

1. import pandas as pd

df=pd.read_csv('creditcard.csv')

df.head()

Out[1]:

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V2
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.12853
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.16717
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.32764
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.64737
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.20601

5 rows × 31 columns

2. df.shape

output

```
(284807, 31)
```

3. df['Class'].value_counts()

from sklearn.ensemble import RandomForestClassifier

classifier=RandomForestClassifier

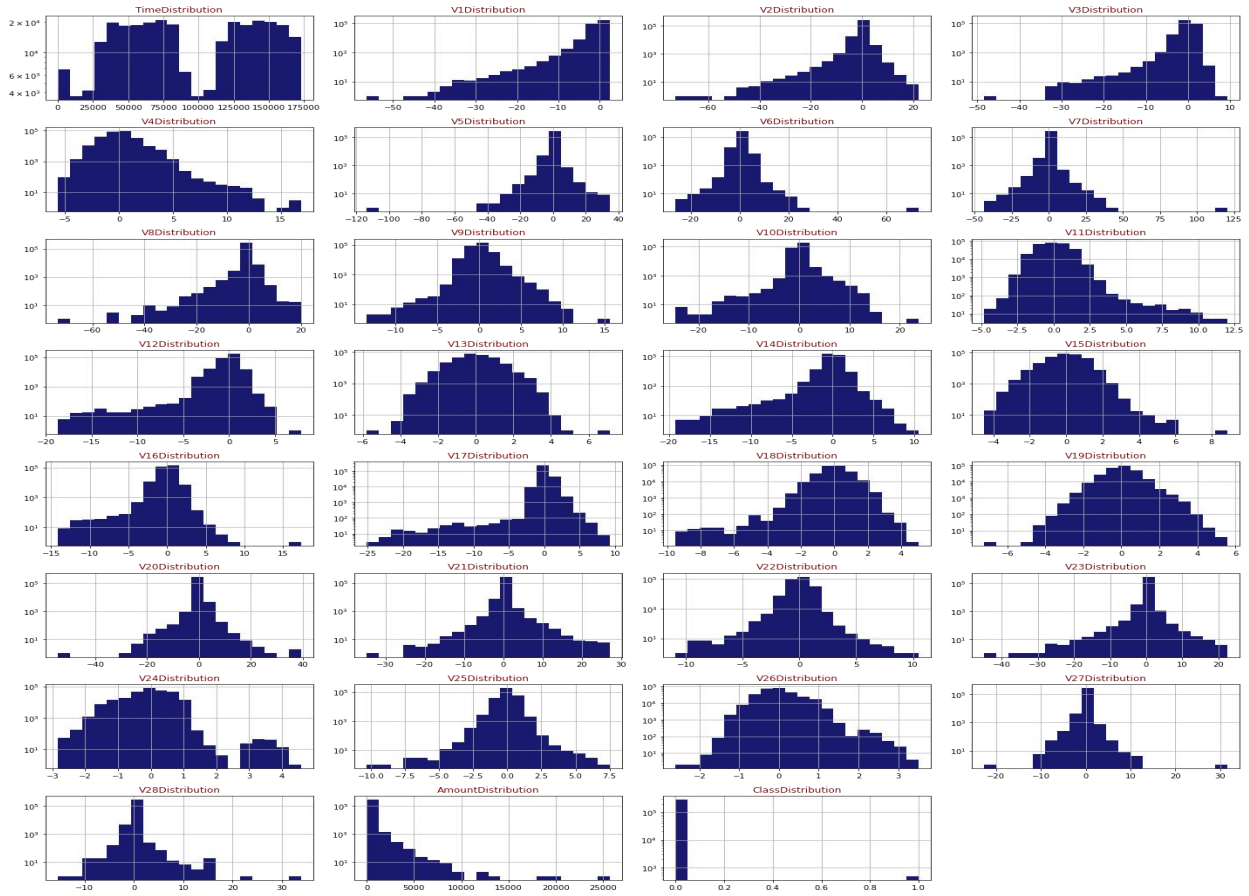
output

```
0    284315
1       492
Name: Class, dtype: int64
```

```
4. import matplotlib.pyplot as plt

5. def draw_histograms(dataframe,features,rows,cols):
```

```
    fig=plt.figure(figsize=(20,20))
    for i,feature in enumerate(features):
        ax=fig.add_subplot(rows,cols,i+1)
        dataframe[feature].hist(bins=20,ax=ax,facecolor='midnightblue')
        ax.set_title(feature+"Distribution",color="DarkRed")
        ax.set_yscale('log')
    fig.tight_layout()
    plt.show()
draw_histograms(df,df.columns,8,4)
```

output



6. ### Independent and Dependent Features

```
X=df.drop("Class",axis=1)

y=df.Class
```

Sklearn Library installing

7. pip install scikit-learn

output

```
Requirement already satisfied: scikit-learn in c:\users\ravip\anaconda3\lib\site-packages (1.0.2)
Requirement already satisfied: numpy>=1.14.6 in c:\users\ravip\anaconda3\lib\site-packages (fro
m scikit-learn) (1.20.1)
Requirement already satisfied: scipy>=1.1.0 in c:\users\ravip\anaconda3\lib\site-packages (from
scikit-learn) (1.6.2)
Requirement already satisfied: joblib>=0.11 in c:\users\ravip\anaconda3\lib\site-packages (from
scikit-learn) (1.0.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\ravip\anaconda3\lib\site-package
s (from scikit-learn) (2.1.0)
Note: you may need to restart the kernel to use updated packages.
```

8.
```
from sklearn.linear_model import LogisticRegression

from sklearn.metrics import accuracy_score,confusion_matrix,classification_report

from sklearn.model_selection import KFold

import numpy as np

from sklearn.model_selection import GridSearchCV

import seaborn as sns
```

9.
```
log_class=LogisticRegression()
```

```
grid={'C':10.0 **np.arange(-2,3),'penalty':['l1','l2']}

cv=KFold(n_splits=5,random_state=None,shuffle=False)
```

10.
```
from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(X,y,train_size=0.75)
```

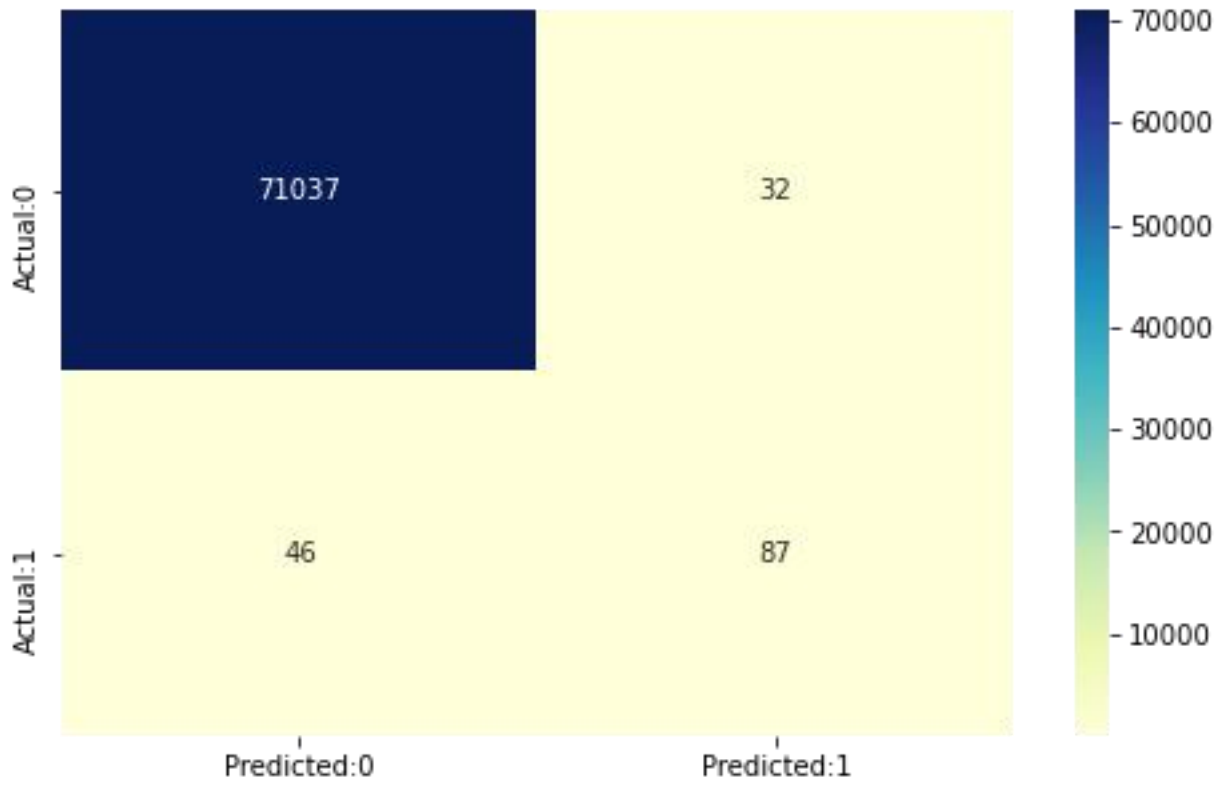
11.
```
clf=GridSearchCV(log_class,grid,cv=cv,n_jobs=-1,scoring='f1_macro')

clf.fit(x_train,y_train)
```

```
GridSearchCV(cv=KFold(n_splits=5, random_state=None, shuffle=False),
 estimator=LogisticRegression(), n_jobs=-1,
 param_grid={'C': array([1.e-02, 1.e-01, 1.e+00, 1.e+01, 1.e+02]),
             'penalty': ['l1', 'l2']},
 scoring='f1_macro')
```

12.
```
y_pred=clf.predict(x_test)
```

```
# confusion matrix
cm=confusion_matrix(y_test,y_pred)

conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','Act
ual:1'])
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");
```

output



13. print(accuracy_score(y_test,y_pred))

output

0.9989045251537878

14. print(classification_report(y_test,y_pred))

output

	precision	recall	f1-score	support
0	1.00	1.00	1.00	71069
1	0.73	0.65	0.69	133

accuracy			1.00	7120
macro avg	0.87	0.83	0.84	71202
weighted avg	1.00	1.00	1.00	71202

Random Forest classifier:

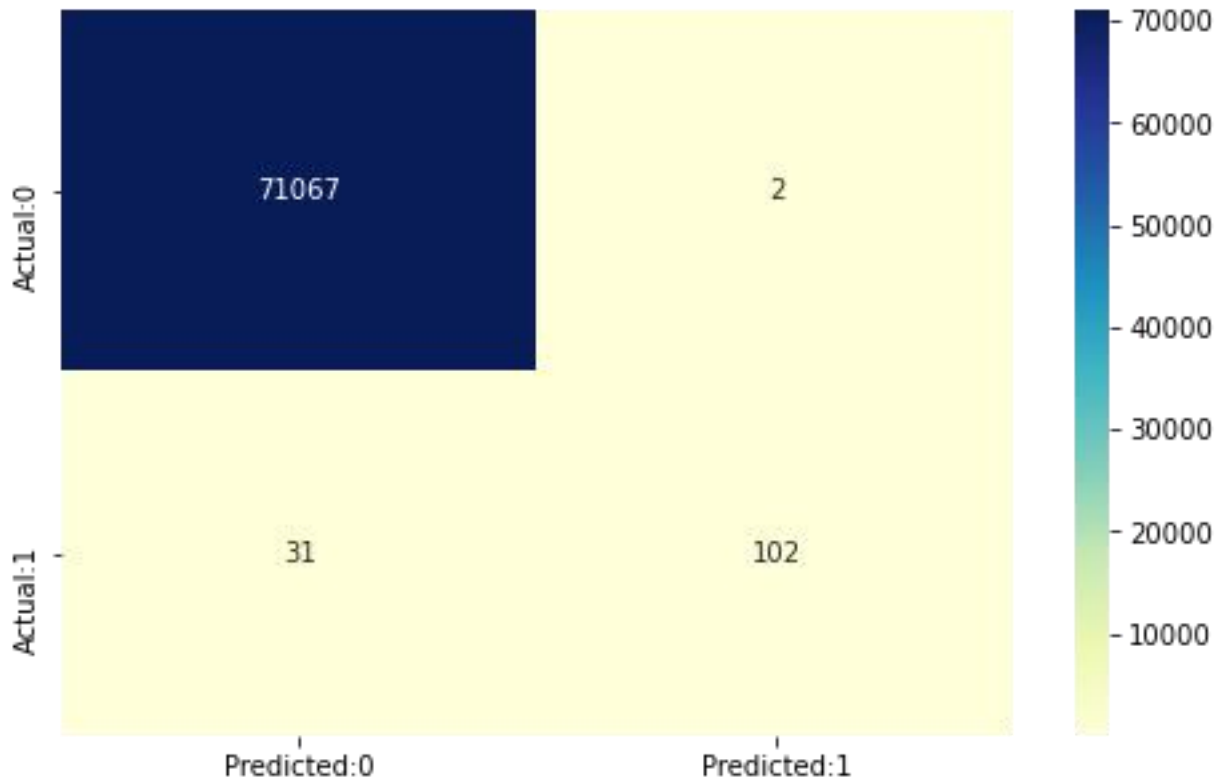
15. Random Forest classifier:

```
RandomForestClassifier(max_depth=10, min_samples_split=5)
```

16. y_pred=classifier.predict(x_test)

```
 # confusion matrix

cm=confusion_matrix(y_test,y_pred)
            conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['A
ctual:0','Actual:1'])
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");
```

output



17. print(accuracy_score(y_test,y_pred))

output

0.9995365298727564

18. print(classification_report(y_test,y_pred))

output

	precision	recall	f1-score	support
0	1.00	1.00	1.00	71069
1	0.98	0.77	0.86	133
accuracy			1.00	71202
macro avg	0.99	0.88	0.93	71202
weighted avg	1.00	1.00	1.00	71202

Importing imblearn library for performing under sampling:

19. pip install imbalanced-learn

Performing Under sampling:NearMiss

20. from collections import Counter

    Counter(y_train)

output

Counter({0: 213246, 1: 359})

21. from collections import Counter

    from imblearn.under_sampling import NearMiss

    ns=NearMiss(version=1,n_neighbors=3)

    x_train_ns,y_train_ns=ns.fit_resample(x_train,y_train)

    print("The number of classes before fit {}".format(Counter(y_train)))

    print("The number of classes after fit {}".format(Counter(y_train_ns)))

output

The number of classes before fit Counter({0: 213246, 1: 359})
The number of classes after fit Counter({0: 359, 1: 359})

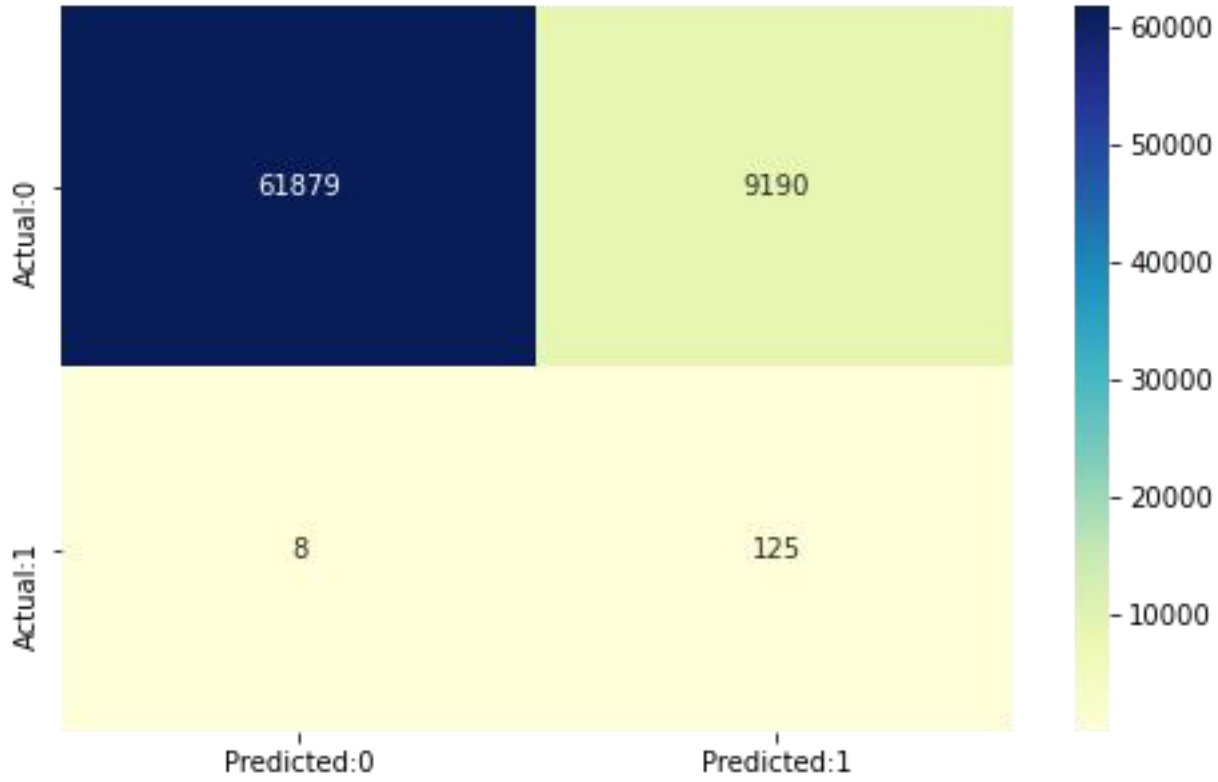
22. from sklearn.ensemble import RandomForestClassifier
    classifier=RandomForestClassifier()
    classifier.fit(x_train_ns,y_train_ns)

    RandomForestClassifier()

23. y_pred=classifier.predict(x_test)

24. # confusion matrix
    cm=confusion_matrix(y_test,y_pred)
  conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','
Actual:1'])
plt.figure(figsize=(8,5))
sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");

output



25. print(accuracy_score(y_test,y_pred))

output

0.8708182354428247

26. print(classification_report(y_test,y_pred))

output

	precision	recall	f1-score	support
0	1.00	0.87	0.93	71069
1	0.01	0.94	0.03	133
accuracy			0.87	71202
macro avg	0.51	0.91	0.48	71202
weighted avg	1.00	0.87	0.93	71202

CatBoost:Overfit Detector

27. pip install catboost

```
# map categorical features
credit_catboost_ready_df=df.dropna()

features=[feat for feat in list(credit_catboost_ready_df) if feat !='Class']
print(features)
card_categories= np.where(credit_catboost_ready_df[features].dtypes !=np.float)[0]
card_categories
```

output

['Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9', 'V10', 'V11', 'V12', 'V13', 'V14', 'V15', 'V16', 'V17', 'V18', 'V19', 'V20', 'V21', 'V22', 'V23', 'V24', 'V25', 'V26', 'V27', 'V28', 'Amount']
<ipython-input-30-7f351ebd8740>:6: DeprecationWarning: `np.float` is a deprecated alias for the builtin `float`. To silence this warning, use `float` by itself. Doing this will not modify any behavior and is safe. If you specifically wanted the numpy scalar type, use `np.float64` here.
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations card_categories= np.where(credit_catboost_ready_df[features]. dtypes !=np.float)[0array([], dtype=int64）

28. SEED=1234

```
from catboost import CatBoostClassifier

params={'iterations':5000,
```

```
    'learning_rate':0.01,
    'cat_features':card_categories,
    'depth':3,
    'eval_metric':'AUC',
    'verbose':200,
    'od_type':"Iter",
    'od_wait':500,
    'random_seed':SEED
   }

cat_model = CatBoostClassifier(**params)
cat_model.fit(x_train,y_train,eval_set=(x_test,y_test),use_best_model=True,plot=True);
```

output

A Jupyter widget could not be displayed because the widget state could not be found. This could happen if the kernel storing the widget is no longer available, or if the widget state was not saved in the notebook. You may be able to create the widget by running the appropriate cells.

```
0:      test: 0.7244389    best: 0.7244389 (0)      total: 180ms      remaining: 14m 58s
200:    test: 0.9764952    best: 0.9800345 (57)     total: 6.63s      remaining: 2m 38s
400:    test: 0.9764337    best: 0.9800345 (57)     total: 11.4s      remaining: 2m 10s
Stopped by overfitting detector  (500 iterations wait)
bestTest = 0.9800345465
bestIteration = 57
```

Shrink model to first 58 iterations.

AdaBoost Classsifier:

Adaboost library Installing

29. pip install ada-boost

29. RANDOM_STATE = 2018

```
NUM_ESTIMATORS = 100

target = 'Class'
```

output

```
predictors = ['Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9', 'V10', 'V11', 'V12', 'V13',
'V14', 'V15', 'V16', 'V17', 'V18', 'V19', 'V20', 'V21', 'V22', 'V23', 'V24', 'V25', 'V26', 'V27', 'V28',
'Amount']
```

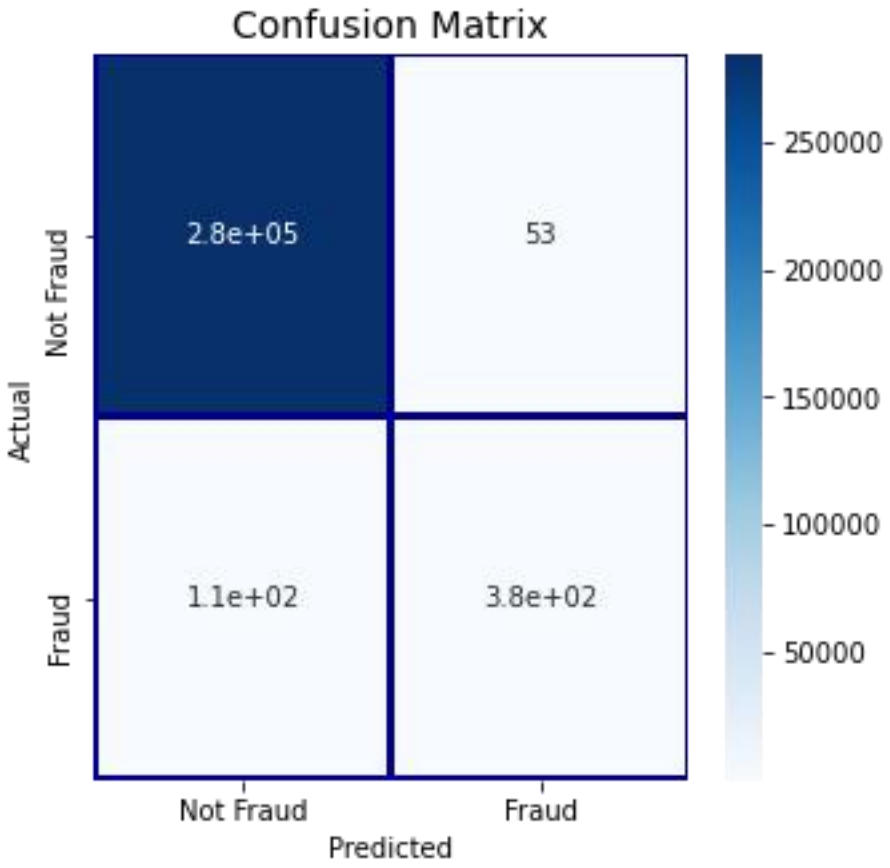
```
31. from sklearn.ensemble import AdaBoostClassifier

clf = AdaBoostClassifier(random_state=RANDOM_STATE,

                         algorithm='SAMME.R',

                         learning_rate=0.8,

                          n_estimators=NUM_ESTIMATORS)

30.  clf.fit(df[predictors],df['Class'].values)
```

```
    AdaBoostClassifier(learning_rate=0.8, n_estimators=100, random_state=2018)
```

```
31. y_pred = clf.predict(df[predictors])


cm=pd.crosstab(df[target].values,y_pred,rownames=['Actual'],colnames=['Predicted'])

fig, (ax1)=plt.subplots(ncols=1,figsize=(5,5))

sns.heatmap(cm,

          xticklabels=['Not Fraud','Fraud'],
```

```
        yticklabels=['Not Fraud','Fraud'],
        annot=True,ax=ax1,
        linewidths=2,linecolor="Darkblue",cmap="Blues")
plt.title('Confusion Matrix',fontsize=14)
plt.show()
```

output



Installing matplotlib for plotting graph

32. pip install matplotlib

33. pip install seaborn

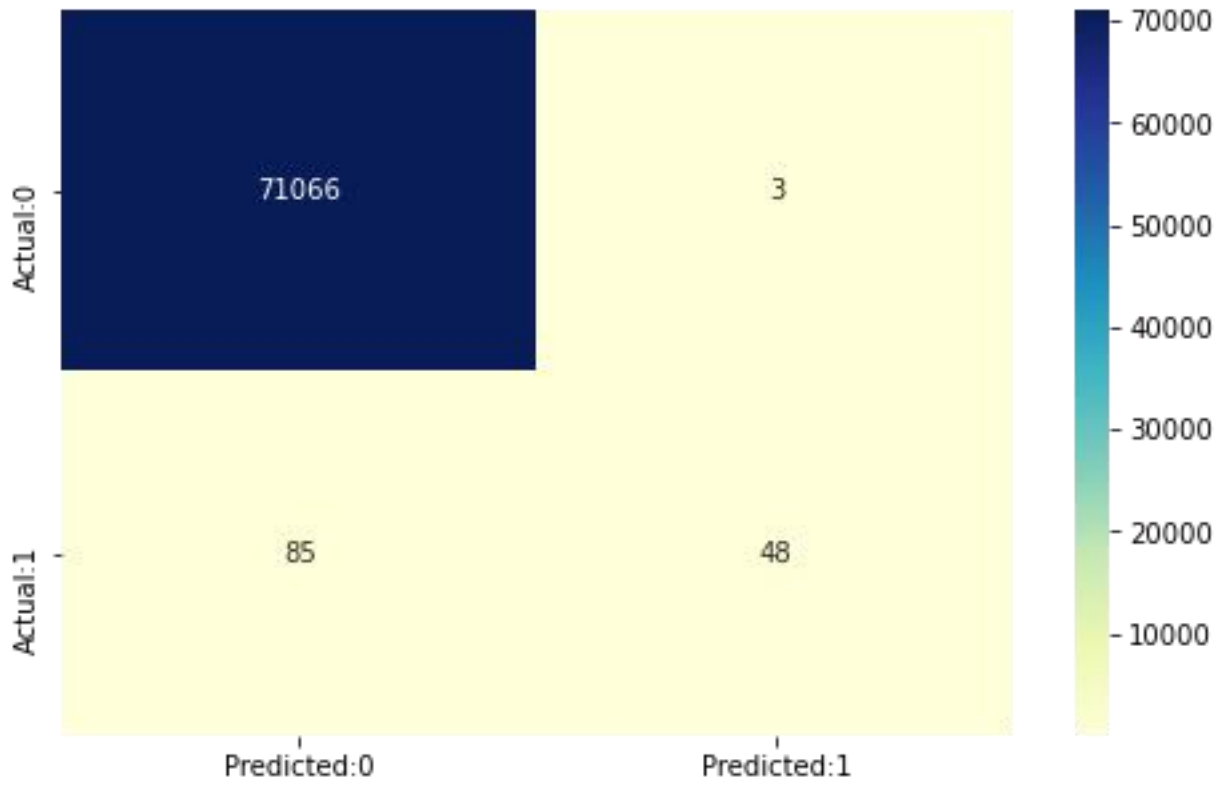
34. # confusion matrix

```
import seaborn as sns

import matplotlib.pyplot as plt

y_pred = cat_model.predict(x_test)
```

35. # confusion matrix

```
cm=confusion_matrix(y_test,y_pred)

conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','Act
ual:1'])

plt.figure(figsize=(8,5))

sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");
```

output



SMOTE Analysis

36. pip install imblearn

37. from imblearn.combine import SMOTETomek

38. os=SMOTETomek(random_state=42)

    x_train_ns,y_train_ns=os.fit_resample(x_train,y_train)

    print("The number of classes before fit {}".format(Counter(y_train)))

    print("The number of classes after fit {}".format(Counter(y_train_ns)))

    output

The number of classes before fit Counter({0: 213246, 1: 359})
The number of classes after fit Counter({0: 212728, 1: 212728})

39. from sklearn.ensemble import RandomForestClassifier
    classifier=RandomForestClassifier()
    classifier.fit(x_train_ns,y_train_ns)

output

```
RandomForestClassifier()
```

40 . y_pred=classifier.predict(x_test)

41. # confusion Matrix

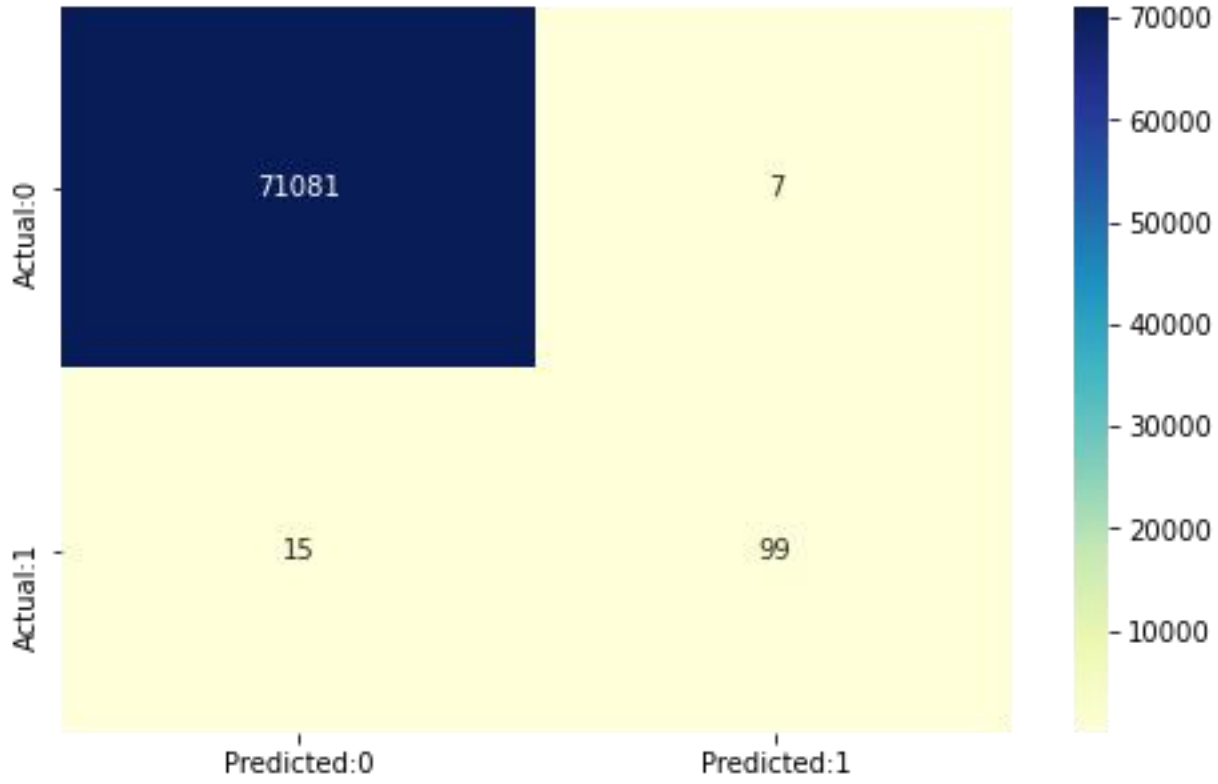
cm=confusion_matrix(y_test,y_pred)

conf_matrix=pd.DataFrame(data=cm,columns=['Predicted:0','Predicted:1'],index=['Actual:0','Actual:1'])

plt.figure(figsize = (8,5))

sns.heatmap(conf_matrix,annot=True,fmt='d',cmap="YlGnBu");

output



42. print(accuracy_score(y_test,y_pred))

Output

```
0.999691019915171
```

43. print(classification_report(y_test,y_pred))

Output

```
              precision    recall  f1-score   support
```

```
              0       1.00      1.00      1.00     71088
              1       0.93      0.87      0.90       114

    accuracy                            1.00     71202
   macro avg       0.97      0.93      0.95     71202
weighted avg       1.00      1.00      1.00     71202
```

```
43. df_temp=df.drop(columns=['Time','Amount','Class'],axis=1)

# create dist plots
fig, ax = plt.subplots(ncols=4, nrows=7, figsize=(20,50))
index = 0
ax = ax.flatten()

for col in df_temp.columns:
   sns.distplot(df_temp[col], ax=ax[index])
   index +=1
plt.tight_layout(pad=0.5,w_pad=0.5,h_pad=5)
```

output