Minor Project Presentation(CSE-325)

CREDIT CARD FRAUD DETECTION

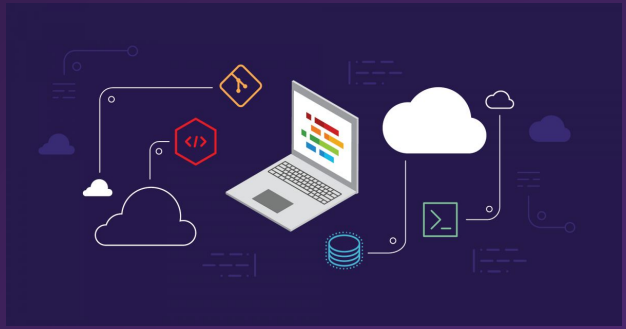
Group Members:
Nitin Gupta(20U02071)
Nimis Sharma(20U02072)

Introduction

- Methods of Credit Card Fraud
- Types of Fraud Detection Systems
- Exploration of Machine Learning Methods

Tools and Technologies Used

- Python
- Jupyter Notebook
- Kaggle
- Pandas, Matplotlib, Plotly, Seaborn
- Machine Learning Algorithms



Steps Followed

- Dataset
- Data Cleaning
- Data Visualization
- Model and Algorithm Implementation

Dataset

- We used the publicly Available Dataset which is available on kaggle.

- Train Dataset Contains 555719 Rows and 23 Columns and Test Dataset Contains 1296675 Rows and 23 Columns.

```
[ ]  # This prints the shape of dataset

     print("fraudTrain.csv Shape : " , test.shape)
     print("fraudTest.csv Shape : " , train.shape)

     fraudTrain.csv Shape :  (555719, 23)
     fraudTest.csv Shape :  (1296675, 23)
```

Dataset

```
Data columns (total 23 columns):
 #   Column                 Non-Null Count   Dtype
---  ------                 --------------   -----
 0   Unnamed: 0             555719 non-null  int64
 1   trans_date_trans_time  555719 non-null  object
 2   cc_num                 555719 non-null  int64
 3   merchant               555719 non-null  object
 4   category               555719 non-null  object
 5   amt                    555719 non-null  float64
 6   first                  555719 non-null  object
 7   last                   555719 non-null  object
 8   gender                 555719 non-null  object
 9   street                 555719 non-null  object
 10  city                   555719 non-null  object
 11  state                  555719 non-null  object
 12  zip                    555719 non-null  int64
 13  lat                    555719 non-null  float64
 14  long                   555719 non-null  float64
 15  city_pop               555719 non-null  int64
 16  job                    555719 non-null  object
 17  dob                    555719 non-null  object
 18  trans_num              555719 non-null  object
 19  unix_time              555719 non-null  int64
 20  merch_lat              555719 non-null  float64
 21  merch_long             555719 non-null  float64
 22  is_fraud               555719 non-null  int64
dtypes: float64(5), int64(6), object(12)
memory usage: 97.5+ MB
```

Test Dataset

```
Data columns (total 23 columns):
 #   Column                 Non-Null Count    Dtype
---  ------                 --------------    -----
 0   Unnamed: 0             1296675 non-null  int64
 1   trans_date_trans_time  1296675 non-null  object
 2   cc_num                 1296675 non-null  int64
 3   merchant               1296675 non-null  object
 4   category               1296675 non-null  object
 5   amt                    1296675 non-null  float64
 6   first                  1296675 non-null  object
 7   last                   1296675 non-null  object
 8   gender                 1296675 non-null  object
 9   street                 1296675 non-null  object
 10  city                   1296675 non-null  object
 11  state                  1296675 non-null  object
 12  zip                    1296675 non-null  int64
 13  lat                    1296675 non-null  float64
 14  long                   1296675 non-null  float64
 15  city_pop               1296675 non-null  int64
 16  job                    1296675 non-null  object
 17  dob                    1296675 non-null  object
 18  trans_num              1296675 non-null  object
 19  unix_time              1296675 non-null  int64
 20  merch_lat              1296675 non-null  float64
 21  merch_long             1296675 non-null  float64
 22  is_fraud               1296675 non-null  int64
dtypes: float64(5), int64(6), object(12)
memory usage: 227.5+ MB
```

Train Dataset

Information of Train and Test Dataset

Data Cleaning

```
train.drop("Unnamed: 0",axis=1,inplace=True)
test.drop("Unnamed: 0",axis=1,inplace=True)
train.head()
```

Python

	trans_date_trans_time	cc_num	merchant	category	amt	first	last	gender	street	city	...	long	city_pop	job	dob	
0	2019-01-01 00:00:18	2703186189652095	fraud_Rippin, Kub and Mann	misc_net	4.97	Jennifer	Banks	F	561 Perry Cove	Moravian Falls	...	-81.1781	3495	Psychologist, counselling	1988-03-09	0b242abb623afc578
1	2019-01-01 00:00:44	630423337322	fraud_Heller, Gutmann and Zieme	grocery_pos	107.23	Stephanie	Gill	F	43039 Riley Greens Suite 393	Orient	...	-118.2105	149	Special educational needs teacher	1978-06-21	1f76529f857473494
2	2019-01-01 00:00:51	38859492057661	fraud_Lind-Buckridge	entertainment	220.11	Edward	Sanchez	M	594 White Dale Suite 530	Malad City	...	-112.2620	4154	Nature conservation officer	1962-01-19	a1a22d70485983eac
3	2019-01-01 00:01:16	3534093764340240	fraud_Kutch, Hermiston and Farrell	gas_transport	45.00	Jeremy	White	M	9443 Cynthia Court Apt. 038	Boulder	...	-112.1138	1939	Patent attorney	1967-01-12	6b849c168bdad6f867
4	2019-01-01 00:03:06	375534208663984	fraud_Keeling-Crist	misc_pos	41.96	Tyler	Garcia	M	408 Bradley Rest	Doe Hill	...	-79.4629	99	Dance movement psychotherapist	1986-03-28	a41d7549acf907893

```
test.trans_date.head(),test.dob.head(),train.trans_date.head(),train.dob.head()
```

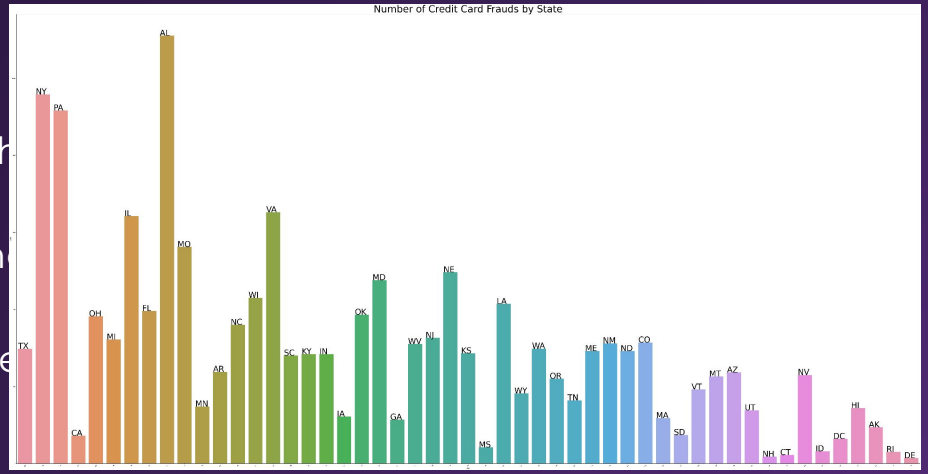
```
(0   2019-01-01
1   2019-01-01
2   2019-01-01
3   2019-01-01
4   2019-01-01
Name: trans_date, dtype: datetime64[ns],
0   1988-03-09
1   1978-06-21
2   1962-01-19
3   1967-01-12
4   1986-03-28
Name: dob, dtype: datetime64[ns],
0   2019-01-01
1   2019-01-01
2   2019-01-01
3   2019-01-01
4   2019-01-01
Name: trans_date, dtype: datetime64[ns],
0   1988-03-09
1   1978-06-21
2   1962-01-19
3   1967-01-12
4   1986-03-28
Name: dob, dtype: datetime64[ns])
```

Data Cleaning and Converting Data Types

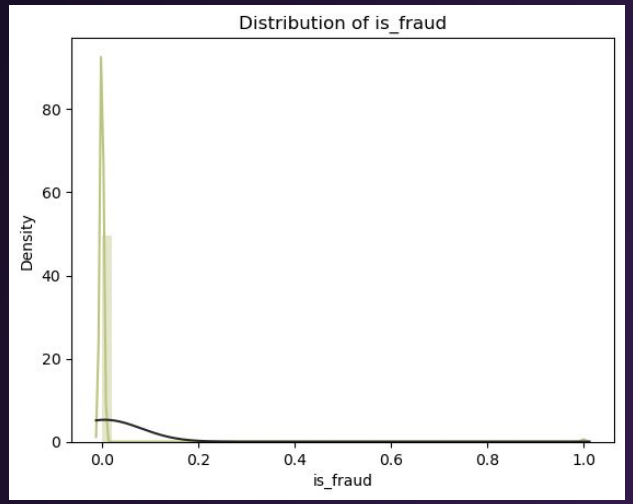
Data Visualization

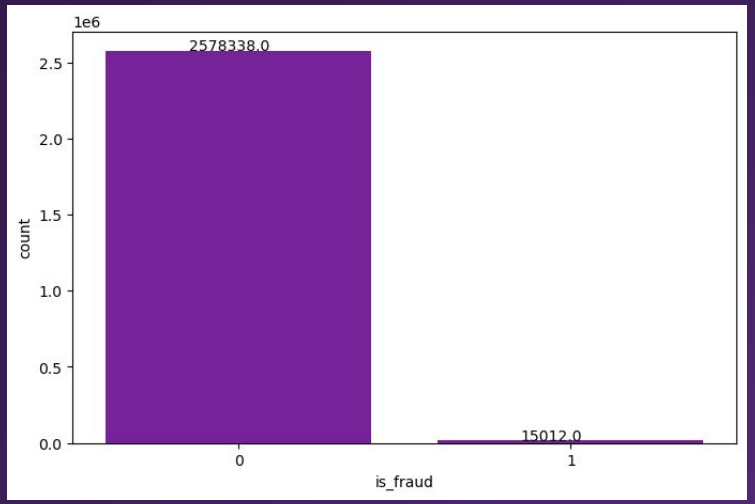
We did data Visualisation by:

- Category of
- Gender of th
- State of th
- City of the
- Job of the victim.



Upsampling the Data

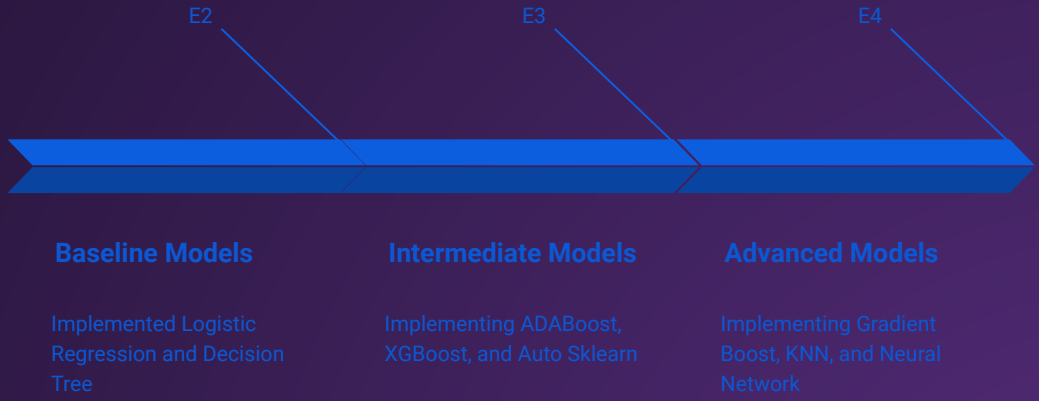




Skewness of the dataset

Models Used

- Logistic Regression
- Decision Tree
- Adaptive Boosting
- Extreme Gradient Boost
- Auto Sklearn
- Gradient Boost
- K-Nearest Neighbour
- Neural Network



Comparison of Models

Related Work and Comparative Analysis

Model Name	Geeks for Geeks	Emmanuel et al.	Pranjal Saxena	data-flair.training	Our Implementation
Logistic Regression	-	99.91 %	99.91%	-	86.66%
Decision Tree	-	99.92%	99.92%	99.96%	99.97%
Random Forest	99.95%	99.94%	-	99.92%	-
AdaBoost	-	-	-	-	99.97%
Gradient Boost	-	-	-	-	98.4%
Naive Bayes	-	98.13%,	-	-	-
XGBoost	-	-	99.95%	-	99.6%
Neural Network	-	99.94%	-	-	91.81%
KNN	-	-	99.95%	-	93.26%

Accuracy Comparison

Future Scope

- Integration with other security systems
- Real Time Fraud Detection
- Incorporating more data sources and parameters

Conclusion

- Developed comprehensive fraud detection system
- Implemented wide range of Models
- Compared them on the basis of efficiency, generalizability, explainability and innovation

Github Link:-

https://github.com/nitingupta-max/Minor_project

ANY QUESTIONS?

THANK YOU!