**MACHINE LEARNING PROJECT**

**Project Title: Credit Card Fraud Detection**

**Problem Description**

Fraud detection systems (FDS) mainly perform two tasks:

1. Real-time detection
2. Posterior detection

The second task manages the largest volume of transactions. It aims at predicting if a transaction is fraudulent based on its characteristics and the past transactions of the cardholder. Yet, in posterior detection, verification often takes days, so new payments on the card become available before a decision is taken.

**Our motivation : Posterior fraud detection**

**Datasets Briefing**

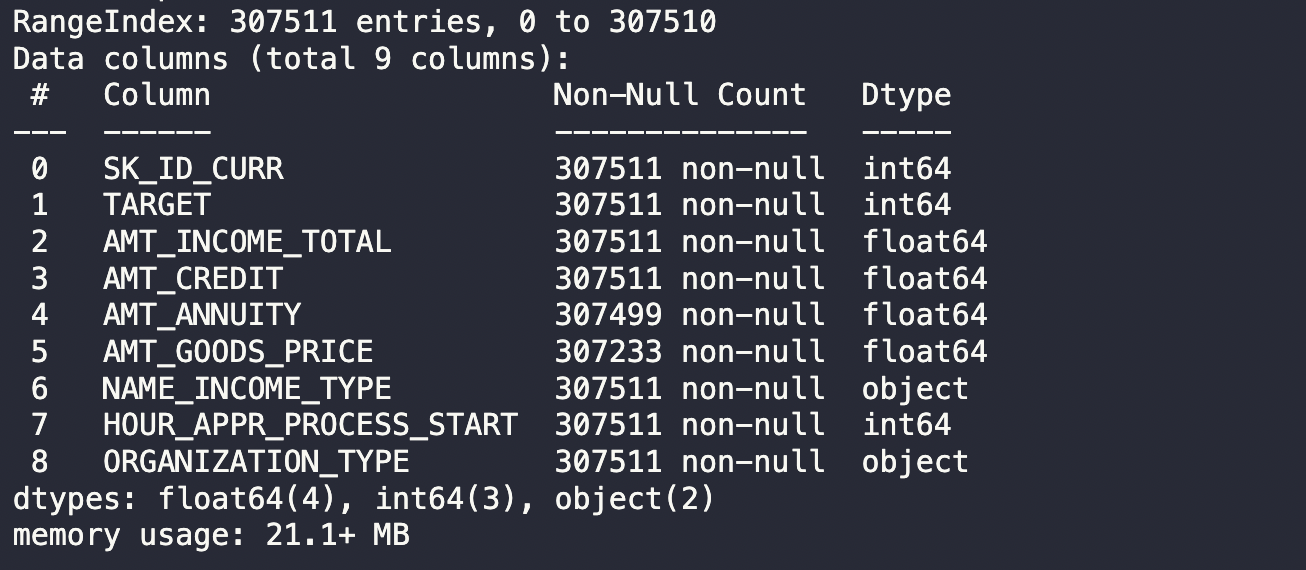
1. The Main dataset used for this project is taken from Kaggle. The dataset contains 307511 rows and 122 columns.
2. The other two Datasets used for this project is taken from GitHub. The dataset are encrypted and are used to train the model.

[Datasets](https://github.com/rajpatel8/Datasets)

**Data Pre-processing**

The dataset used for this project is highly imbalanced with only 0.17% of the data being fraudulent. The dataset is further trimmed to 307511 rows and 9 columns.

The 9 columns are:



**Keywords**

• Posterior detection • highly imbalanced

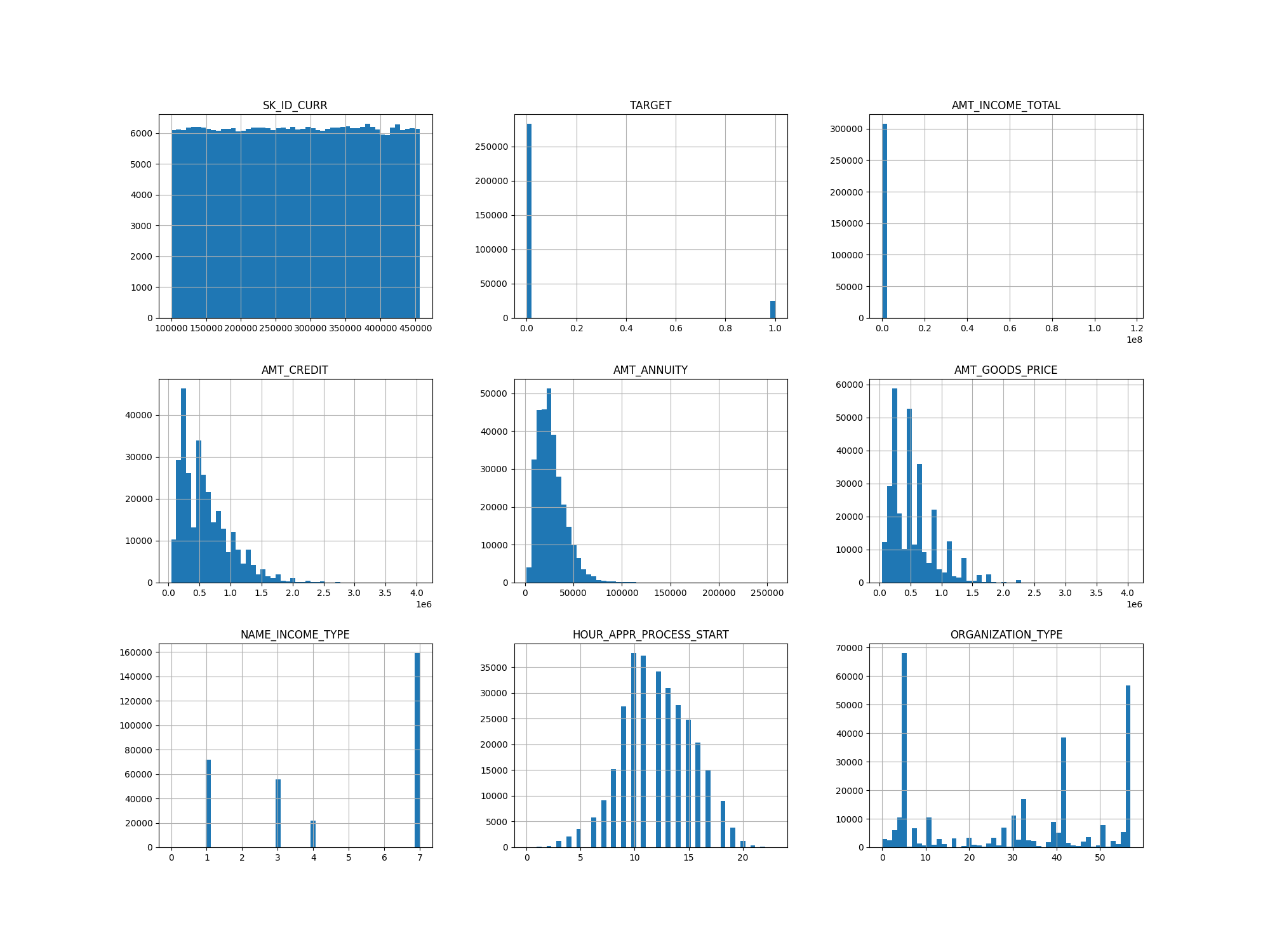
• Random Forest Classifier •Logistic Regression

•K-Neighbours Classifier •Confusion Matrix

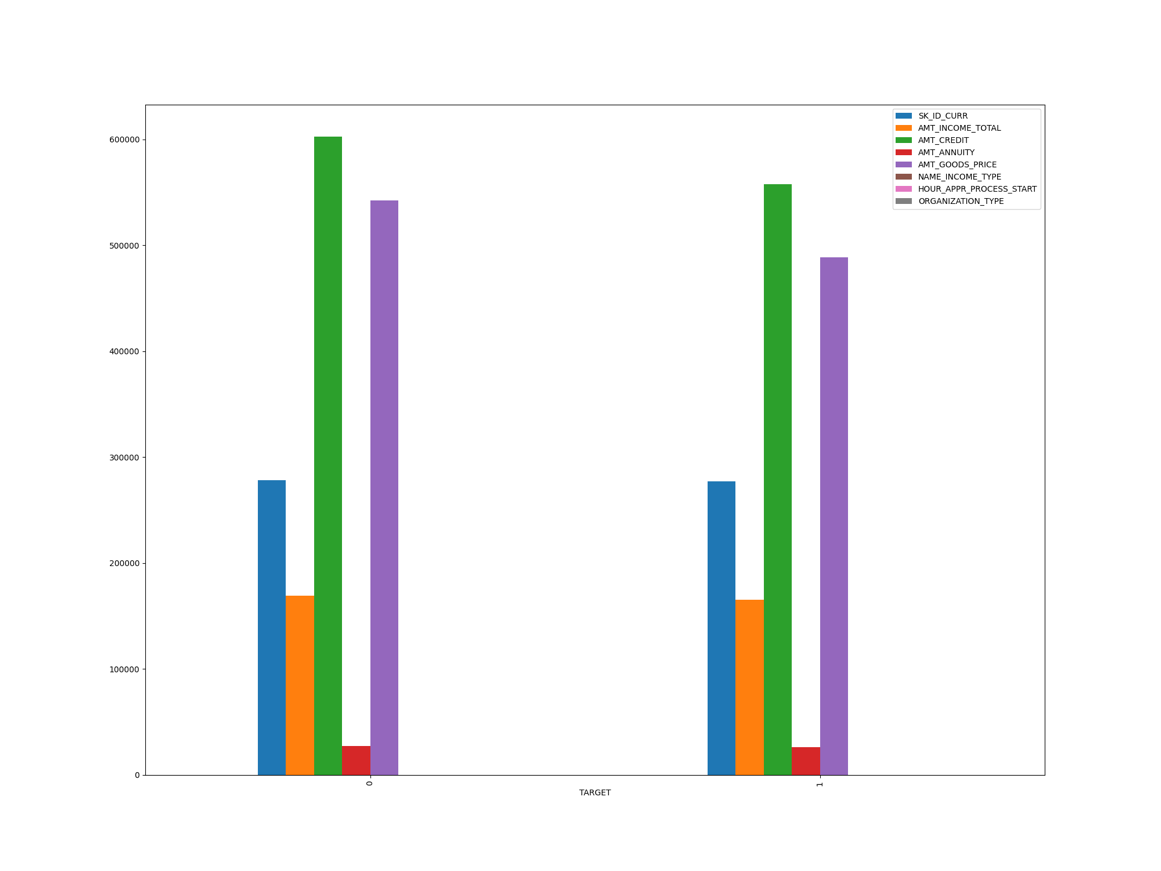
**Methods used for Data Pre-processing**

1. Correlation matrix for numerical data
2. One-hot encoding for categorical data
3. StandardScaler for numerical data
4. Label Encoding for categorical data

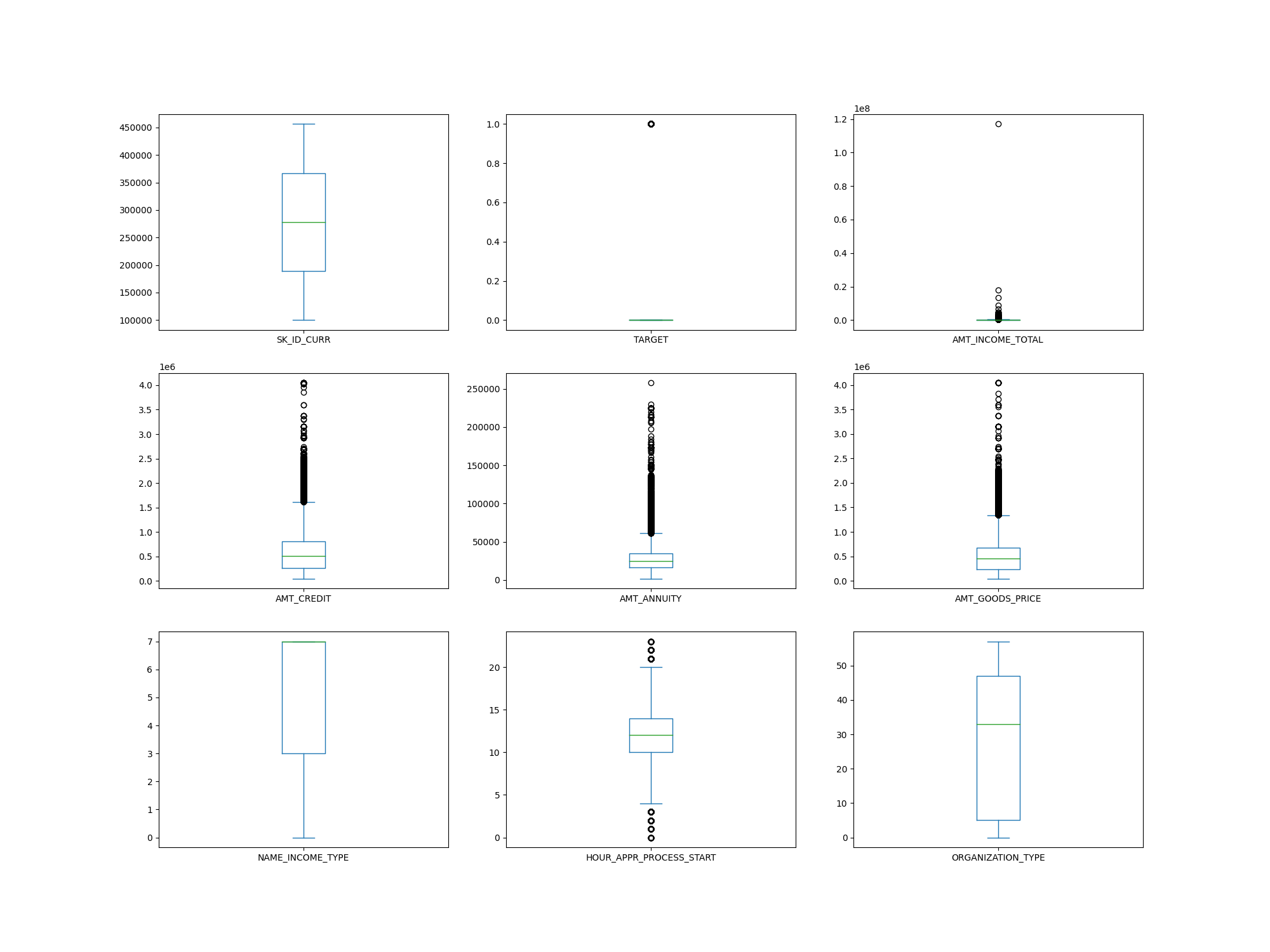
**Statistical summary of all attributes**



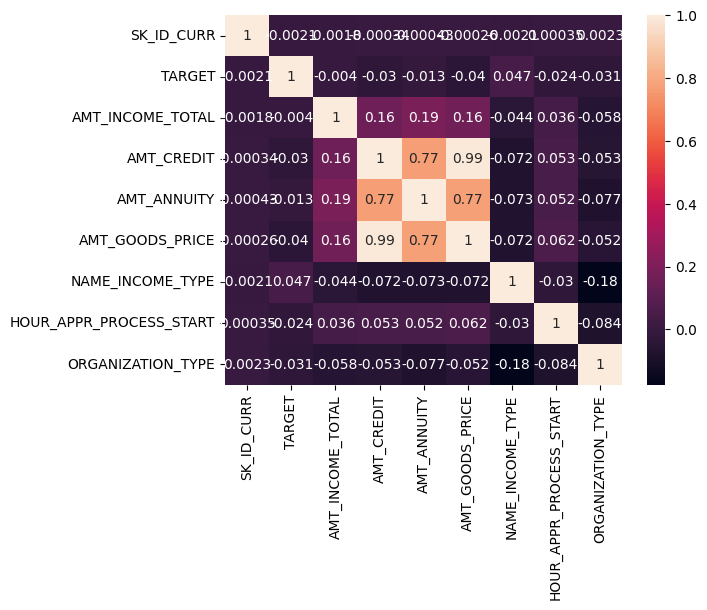
**Data Visualization**



**Checking Outliners**



**Data Corelation**



**Python Packages used for:**

* Data Pre-processing:
  + Pandas
  + Numpy
  + Matplotlib
  + Seaborn
  + Scikit-learn
* Data Visualization:
  + Matplotlib
  + Seaborn
* Model Building:
  + Scikit-learn:
    - RandomForestClassifier
    - LogisticRegression
    - SVC, LinearSVC
    - train\_test\_split
    - LabelEncoder
* Model Evaluation:
  + Scikit-learn:
    - confusion\_matrix
    - accuracy\_score
    - f1\_score
    - classification\_report
    - roc\_curve
    - roc\_auc\_score

**Learning Algorithms**

Out of all the supervised Machine Learning Algorithm. There were the 3 algorithms which performed well on our dataset.

1. Random Forest Classifier: Result mean 0.919268
2. Logistic Regression: Result mean 0.919261
3. K-Neighbours Classifier: Result mean 0.914260

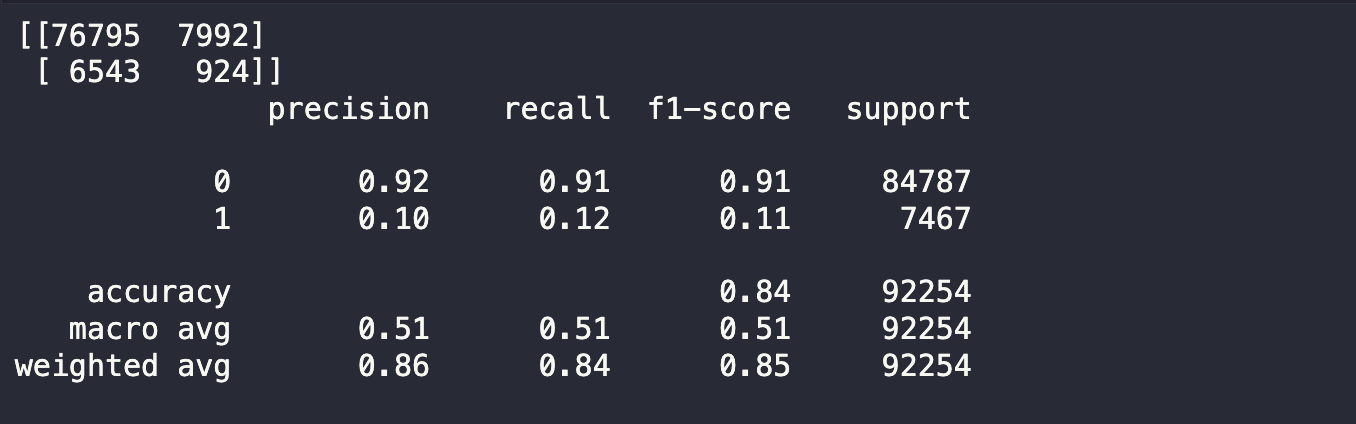
Chart

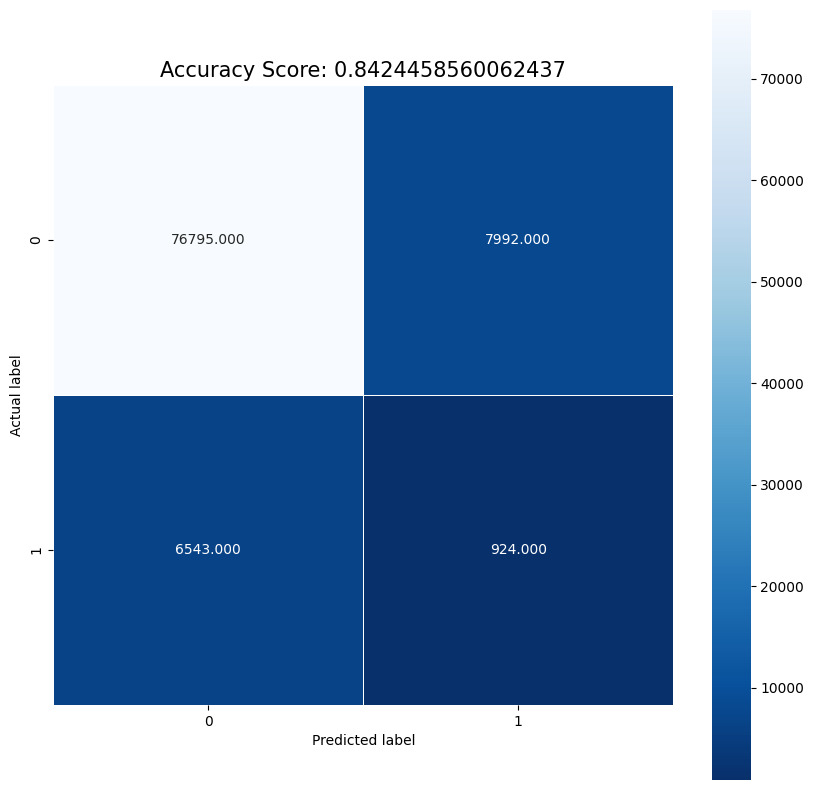
Description automatically generated with low confidence

**Result and Conclusion**

* Model Evaluation :

Confusion Matrix





* Overfit Under Fit curve

Chart, line chart

Description automatically generated

**Team Members**

* Harshit Raj – AM.EN.U4CSE20349
* Patel Rajkumar – AM.EN.U4CSE20349
* Prashant Kumar N – AM.EN.U4CSE20352
* Rupesh Kumar Tailor – AM.EN.U4CSE20357