Comsats University Islamabad,

Lahore Campus

PROJECT

Project Title:

Machine Learning Algorithms

Group Members:

Aimen Umar (SP23-BAI-008)

Malaika Nadeem (SP23-BAI-024)

Submitted to:

Dr.M. Aksam Iftikhar

1. Introduction

The dataset is for a **Credit Card Fraud Detection System**, sourced from **Kaggle**. It is highly imbalanced, with **0.17%** fraudulent transactions and **99.83%** non-fraudulent transactions. (0: non-fraud, 1: fraud). The objective is to identify fraudulent transactions while addressing challenges posed by the class imbalance.

Link: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud?resource=download

2. Methodology

Preprocessing Steps

1. **Handling Missing Data**:
    - The **SimpleImputer** is used to fill any missing values, ensuring no data points are lost during analysis.
2. **Feature Selection**:
    - **Mutual Information**: Selected features that are most relevant to predicting fraud, reducing irrelevant data and improving efficiency.
3. **Balancing the Dataset**:
    - The dataset is highly imbalanced (0.17% fraud). SMOTE oversampled the minority class to create a more balanced dataset, helping models learn patterns for fraud better.
4. **Feature Scaling**:
    - **Standard Scaling** is applied to normalize the features, ensuring that all algorithms perform optimally, especially those sensitive to feature magnitudes, such as SVM.

Algorithms

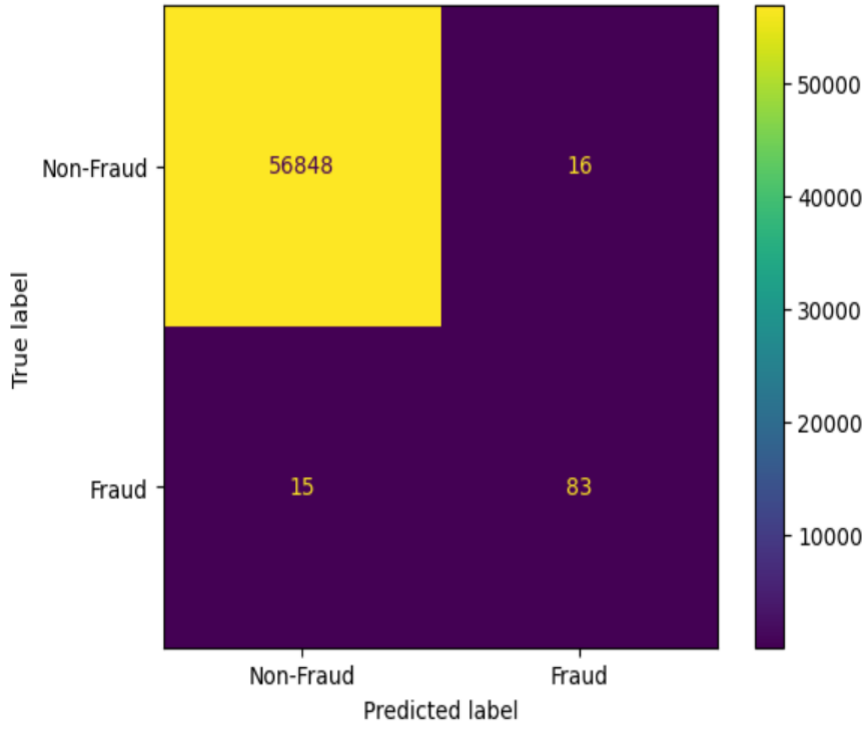
1. **Random Forest**
2. **XGBoost**
3. **LightGBM**
4. **SVM**

Optimization

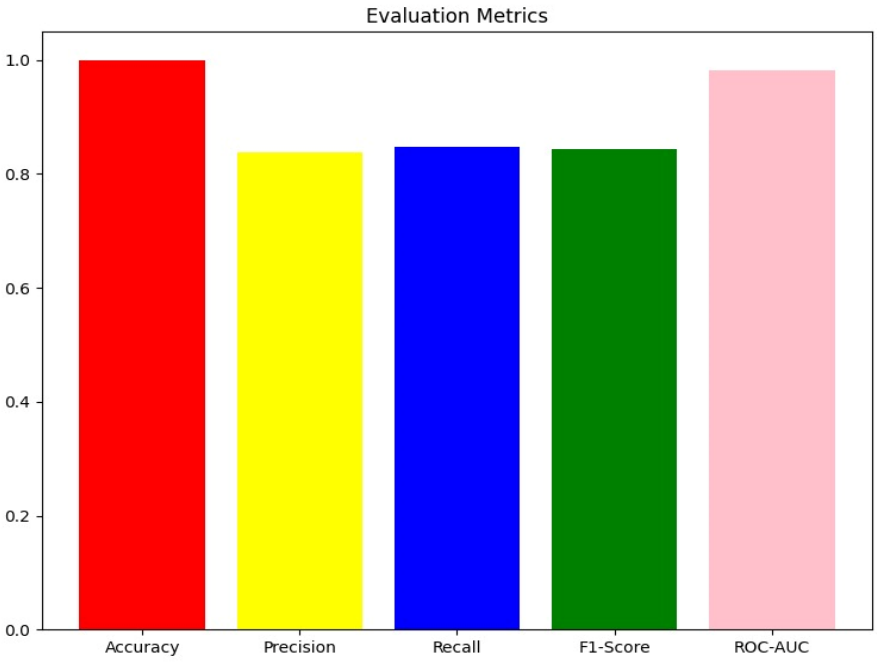
- **Grid Search**: Exhaustive hyperparameter search.
- **Random Search**: Faster, randomized hyperparameter sampling.

XGBOOST

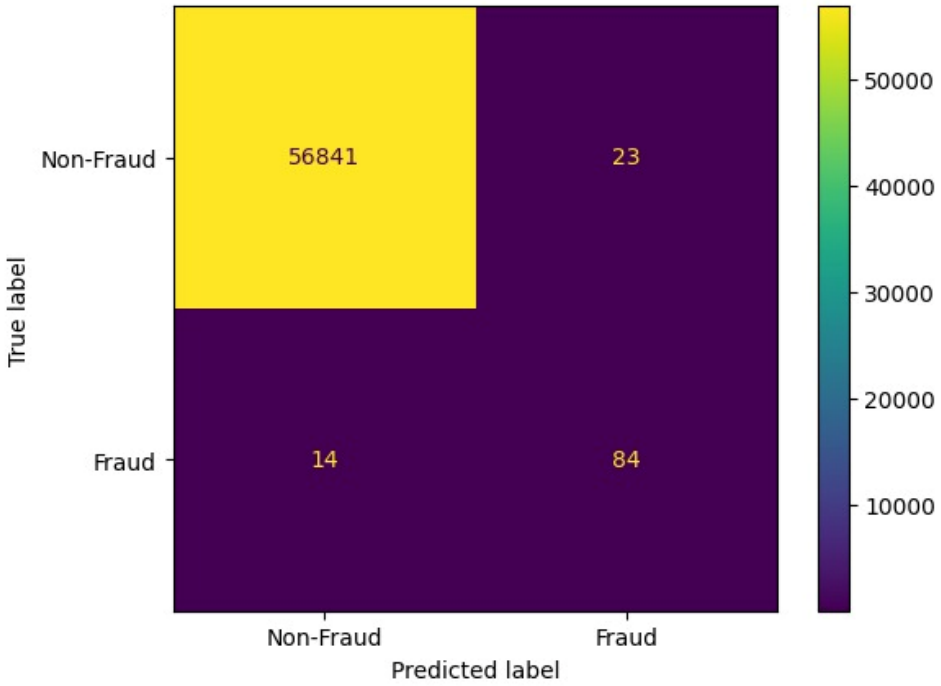
1-Confusion Matrix after Random Search:



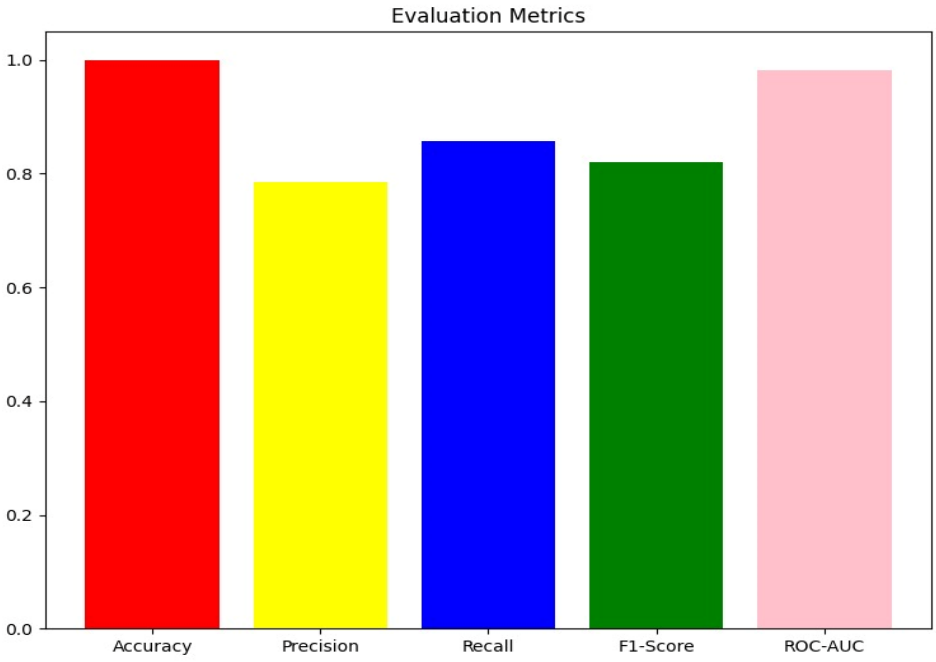
2-Bar Chart after Random Search



3- Confusion Matrix after Grid Search

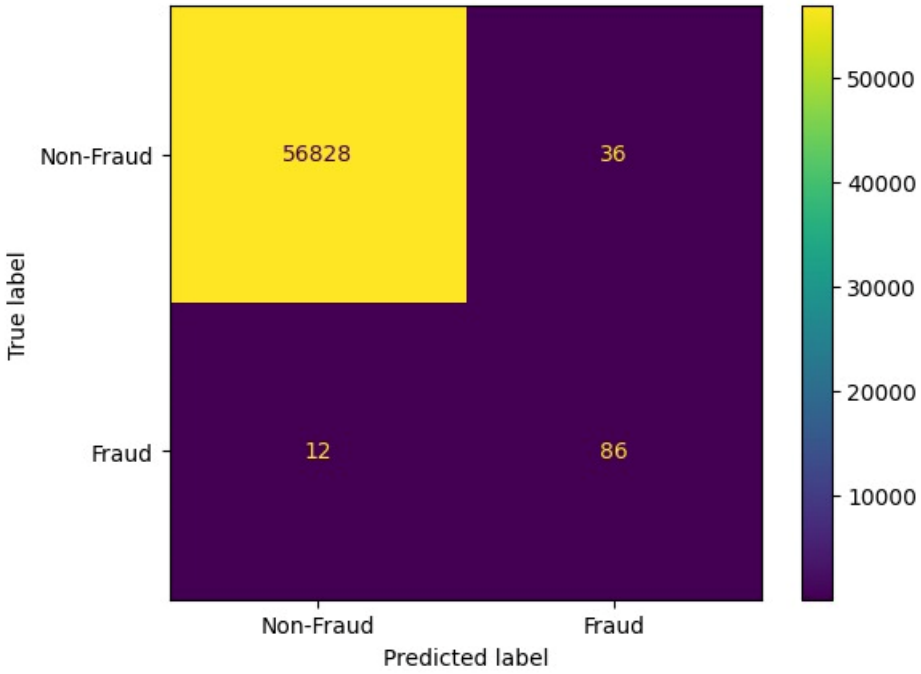


4- Bar Chart after Grid Search

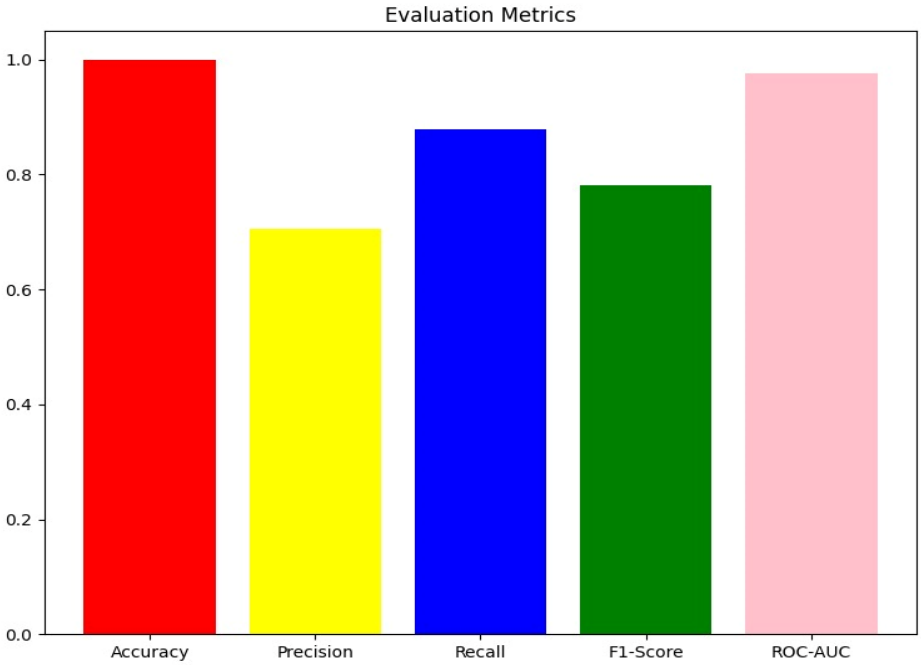


RANDOM FOREST

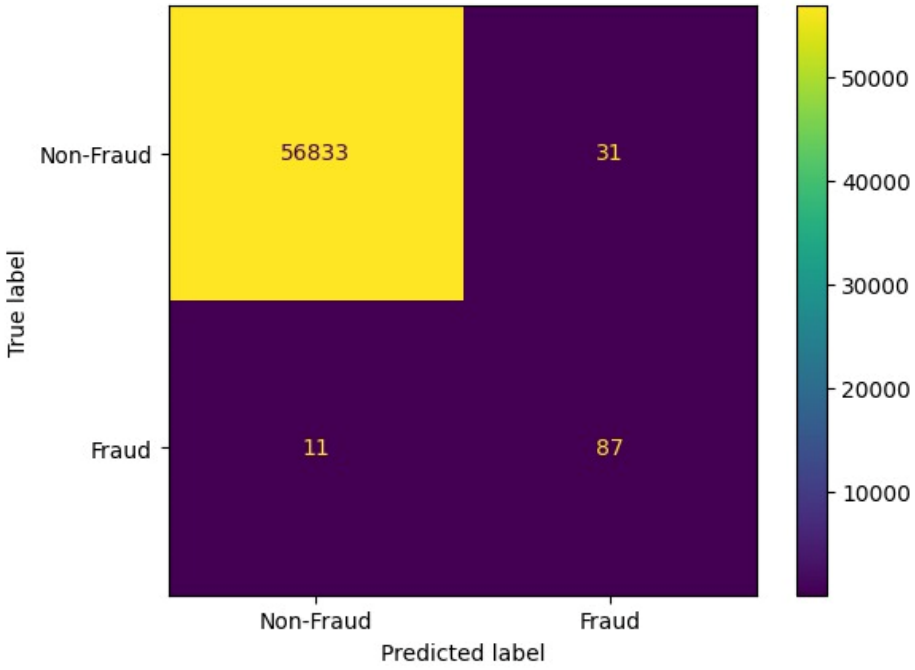
1-Confusion Matrix after Random Search



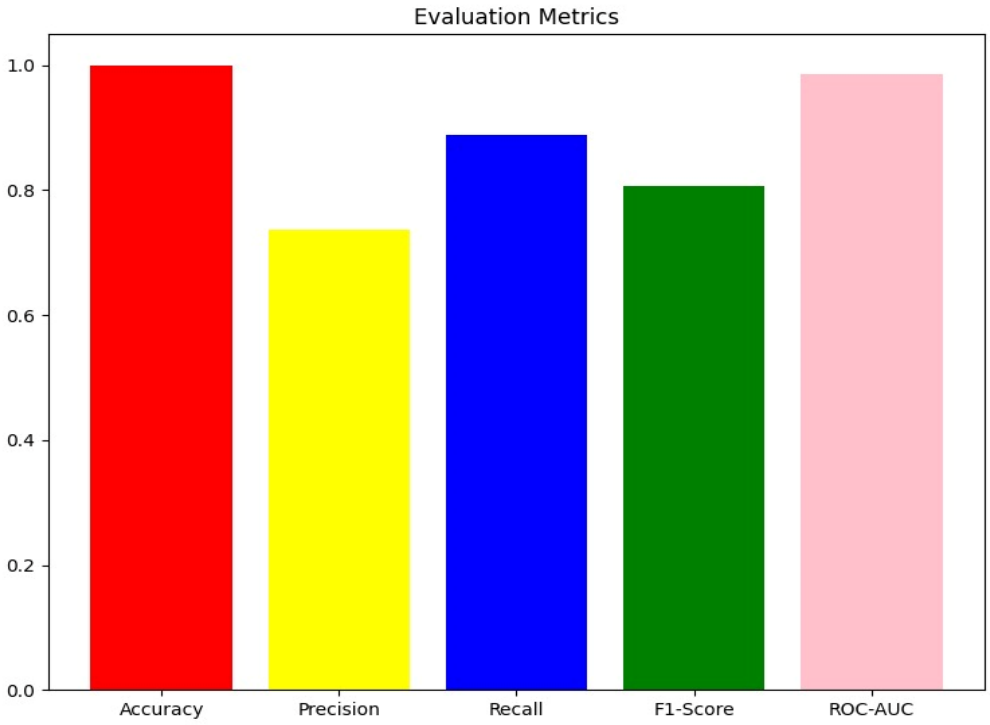
2- Bar Chart after Random Search



3- Confusion Matrix After Grid Search

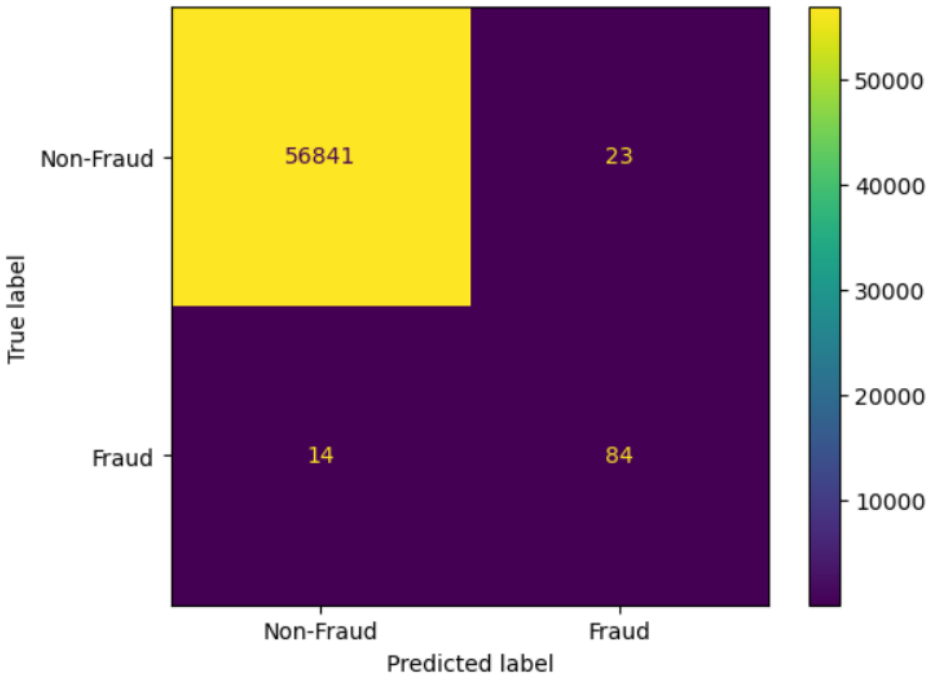


4- Bar Chat After Grid Search

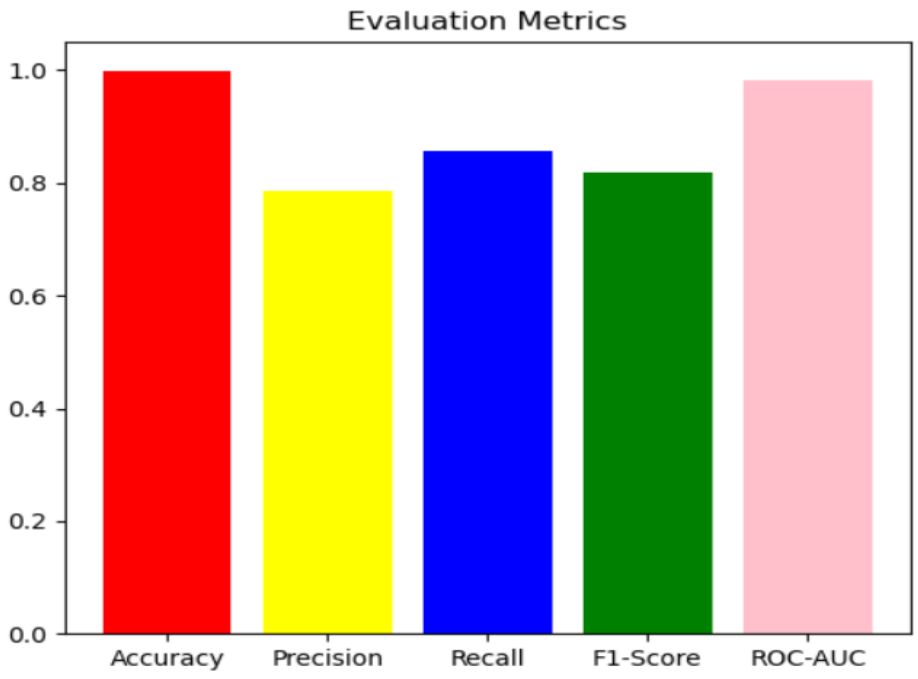


LIGHTGBM

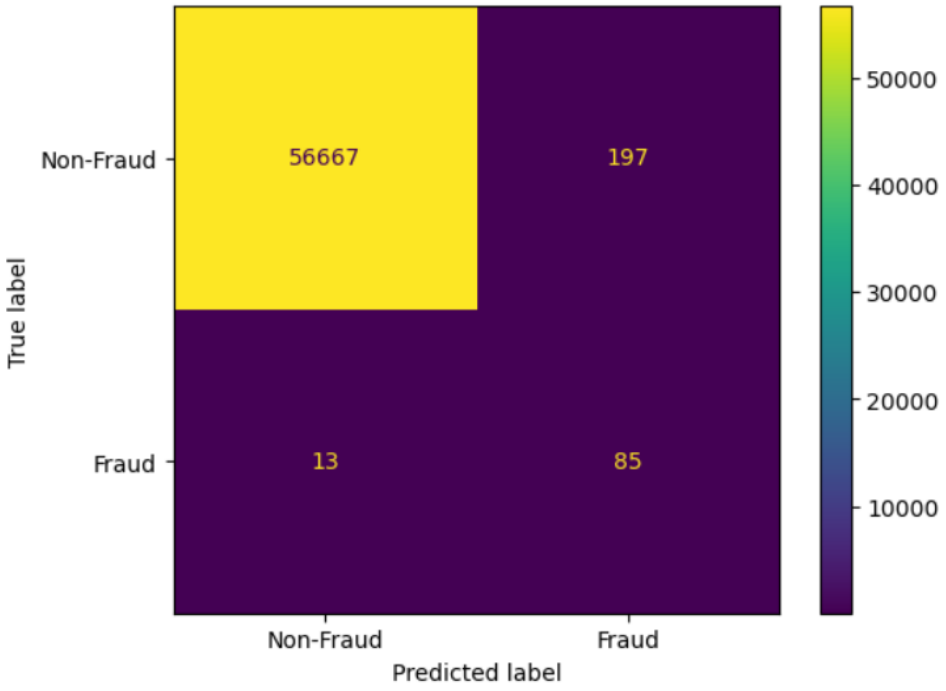
1. **Confusion Matrix After Random Search**



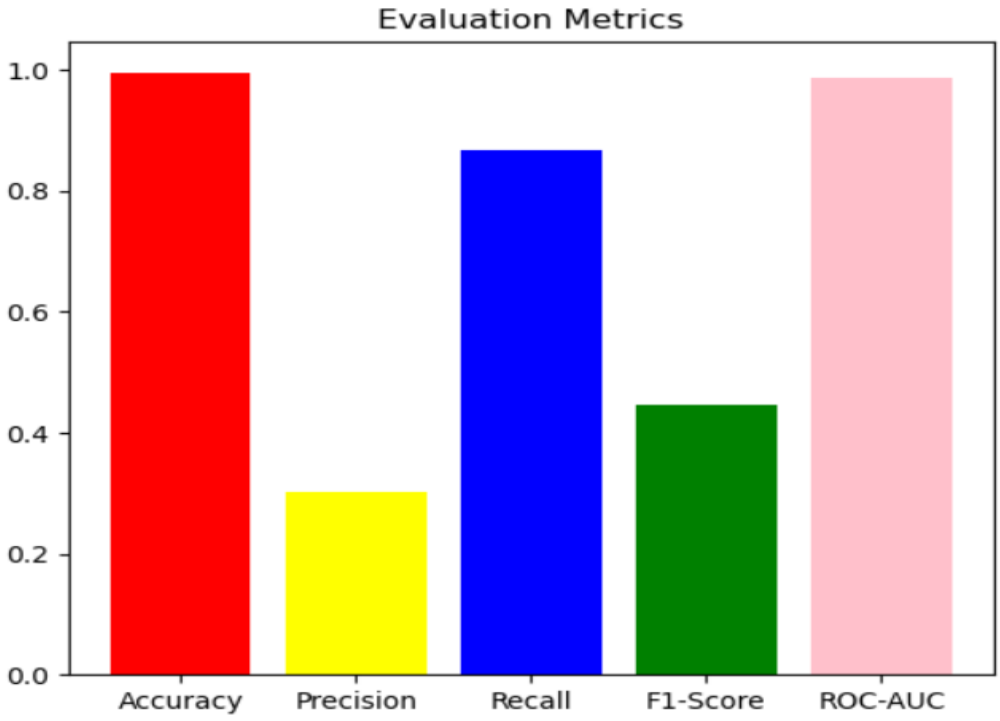
2. **Bar Chart After Random Search**



3. Confusion Matrix After Grid Search

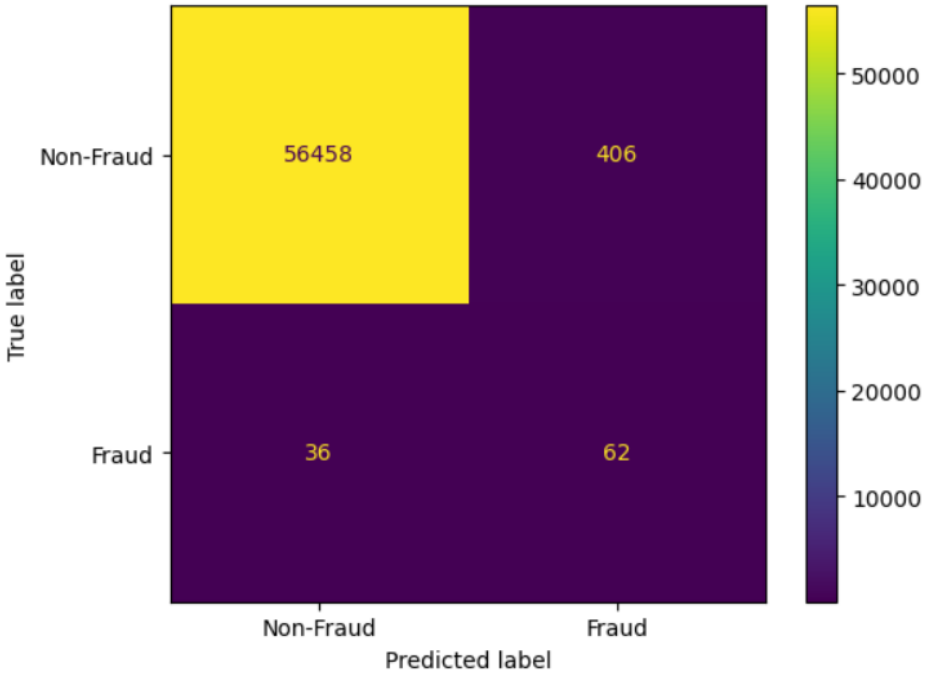


4. Bar Chart After Grid Search

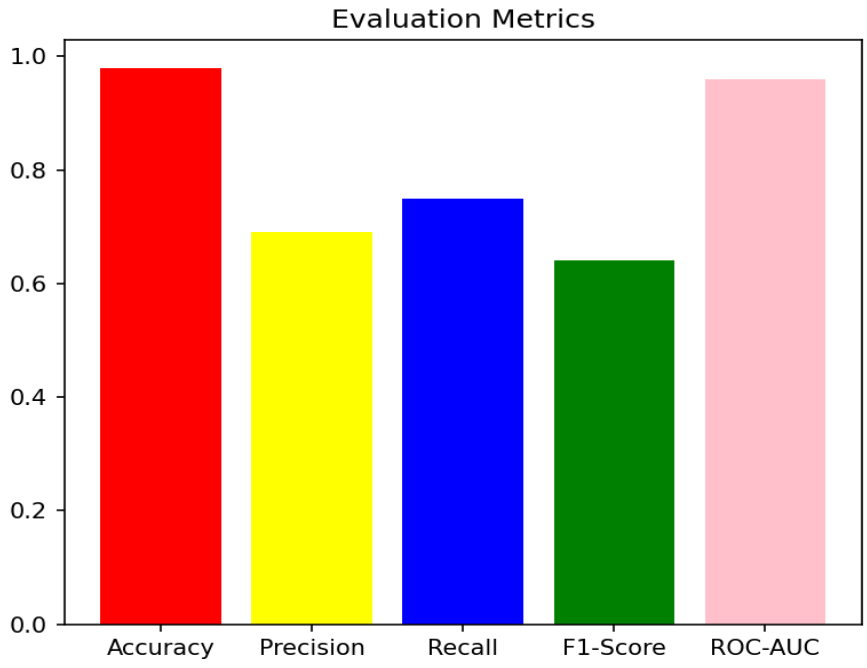


SUPPORT VECTOR MACHINE

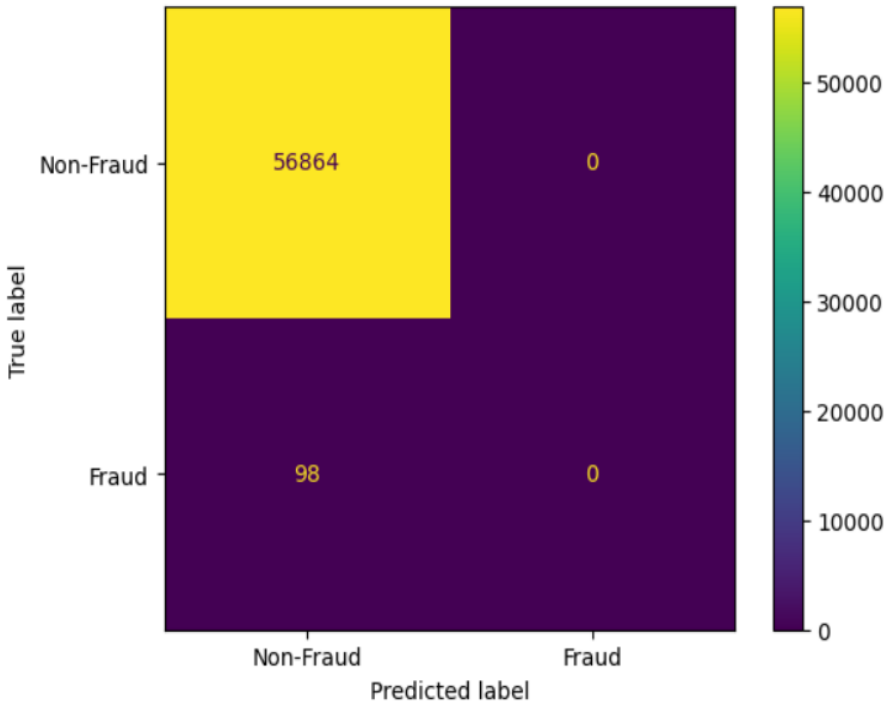
1. Confusion Matrix After Random Search



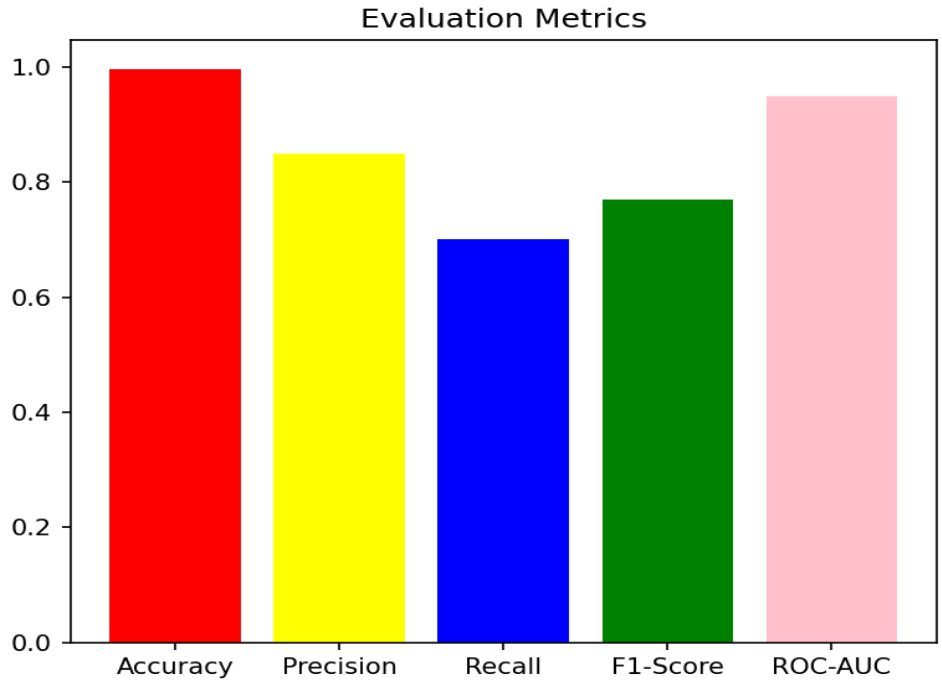
2. Bar Chart After Random Search



3. Confusion Matrix After Grid Search



4. Bar Chart After Grid Search



4- Analysis

Algorithm	Accuracy	Precision	Recall	F1	ROC-AUC	Best Hyper parameters	Execution Time	Remarks
Lightgbm (RandomSearch)	0.999729	0.792854	0.861987	0.824715	0.982132	feature_fraction= 0.8 learning_rate = 0.1 max_depth= 10 min_data_in_leaf= 30 n_estimators= 500 num_leaves= 100	10m24.8s	Good overall performance, potential imbalance
Lightgbm (GridSearch)	0.997933	0.30	0.873732	0.452588	0.96190	feature_fraction= 0.8 learning_rate = 0.05 max_depth= 10 min_data_in_leaf= 10 n_estimators= 200 num_leaves= 50	27m15.5s	Lower precision, longer execution time
SVM (Random Search)	0.997	0.85	0.70	0.77	0.90	C=1 Kernel = rbf Gamma = scale	225m12.5s	Lower recall, Longer Execution time
SVM (Grid Search)	0.98	0.69	0.75	0.64	0.96	C=10 Kernel = linear Gamma = scale	278m45.2s	Lower f1, Very slow execution
Xgboost (Random Search)	0.999455	0.838383	0.846938	0.842639		n_estimators=300, max_depth=4, learning_rate=0.5 subsample=1.0 scale_pos_weight=1 colsample_bytree=0.4	20m49s	Strong performance, fast execution
Xgboost (Grid Search)	0.999350	0.785046	0.857142	0.819512	0.980979	n_estimators=300, max_depth=6, learning_rate=0.3 subsample=0.6 scale_pos_weight=20 colsample_bytree=1	33m4s	Lower precision and recall, longer execution
Random Forest (Random Search)	0.999157	0.704918	0.877551	0.781818		n_estimators=30, max_depth=12, min_samples_split=6	1hr8m	Strong performance, longer execution

Random Forest (Grid Search)	0.999490	0.863157	0.836734	0.849740	0.986431	n_estimators=40, max_depth=12, min_samples_split=8	54m8s	Strong performance, faster execution