The Capstone Project of Data Science Module

"Fraud Detection Project"

1. Exploratory Data Analysis & Data Cleaning --> Gulcan
2. Data Preprocessing --> Gulcan

   Model Building --> Gulcan

   Logistic Regression --> Gulcan

   Random Forest Classifier --> Allen

   XGBoost Classifier --> Allen

3. Neural Network --> Sahinde
4. Model Deployement --> Sue & Sahinde

“creditcard.csv”

The datasets contains transactions made by credit cards in September 2013 by european cardholders.

Shape:(284807, 31)
There is 284807 observation and 31 columns in the dataset.

(i) The features (V1, V2, V3, up to V28) are the principal components (numerical input components) obtained using PCA.

(ii) The feature 'Time' contains the seconds elapsed between the first transaction in the data set and the subsequent transactions.

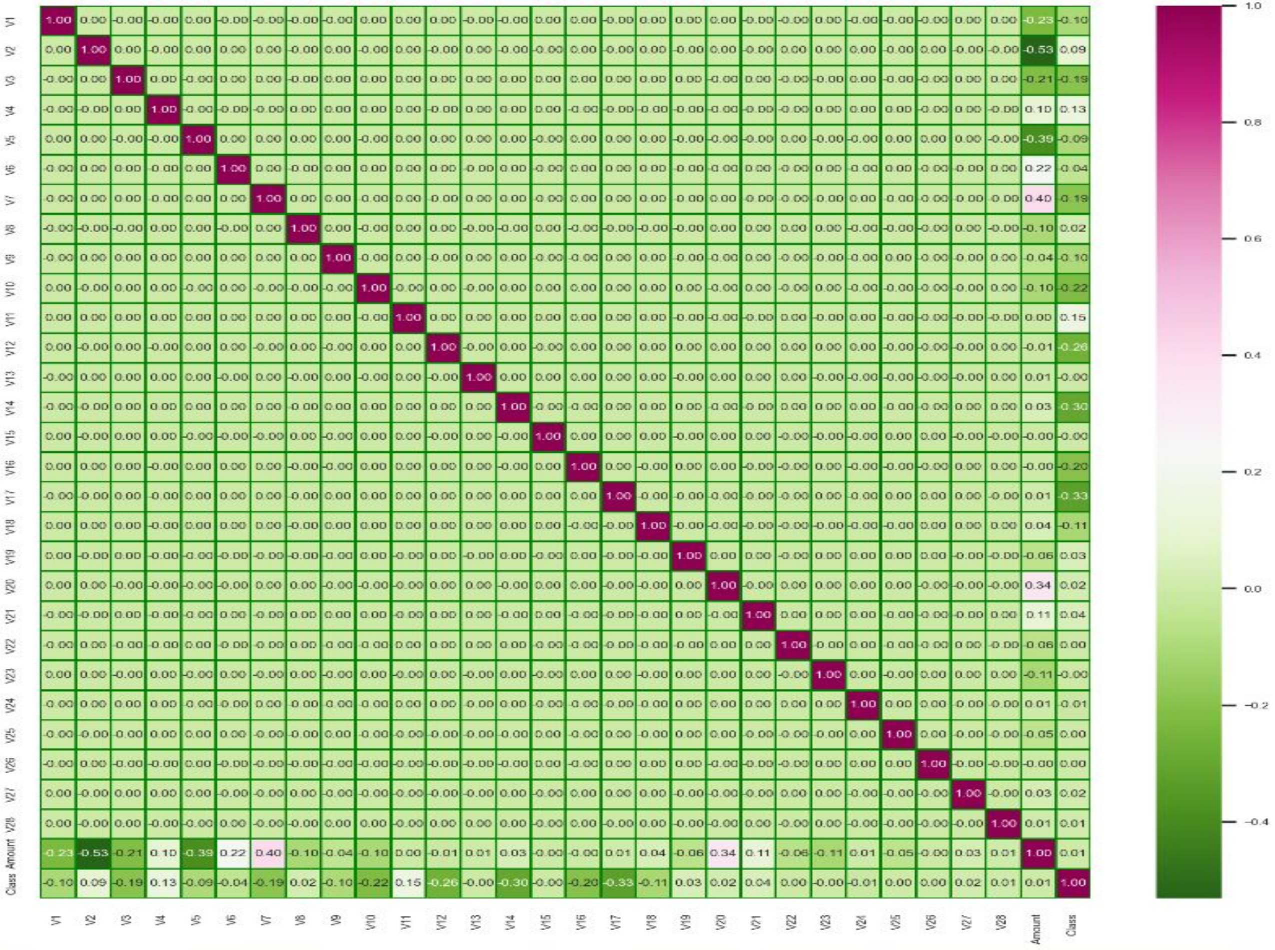
(iii) The feature 'Amount' is the transaction amount.

(iv) The feature 'Class' represents class labelling. (0: Normal Transactions, 1: Fraud Transactions)

Multi-Collinearity

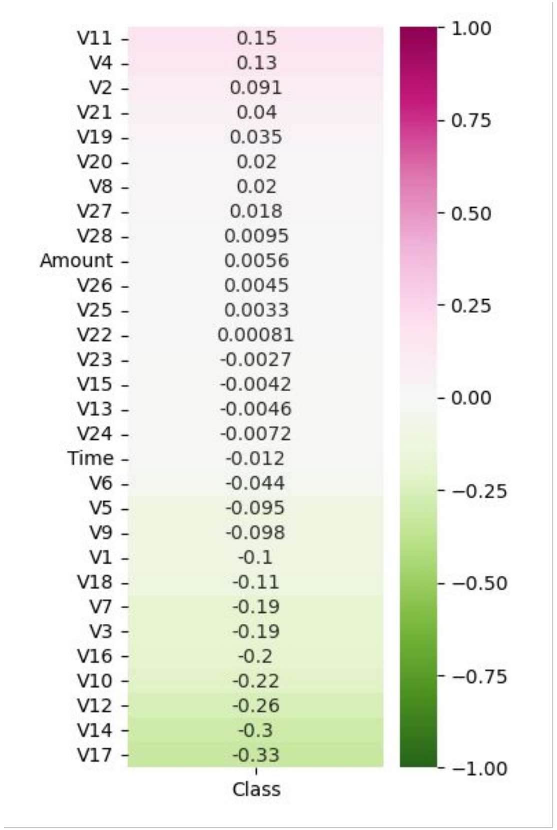
(i) (V1, V2 upto V28) clearly don't show any sign of multi-collinearity with each other.

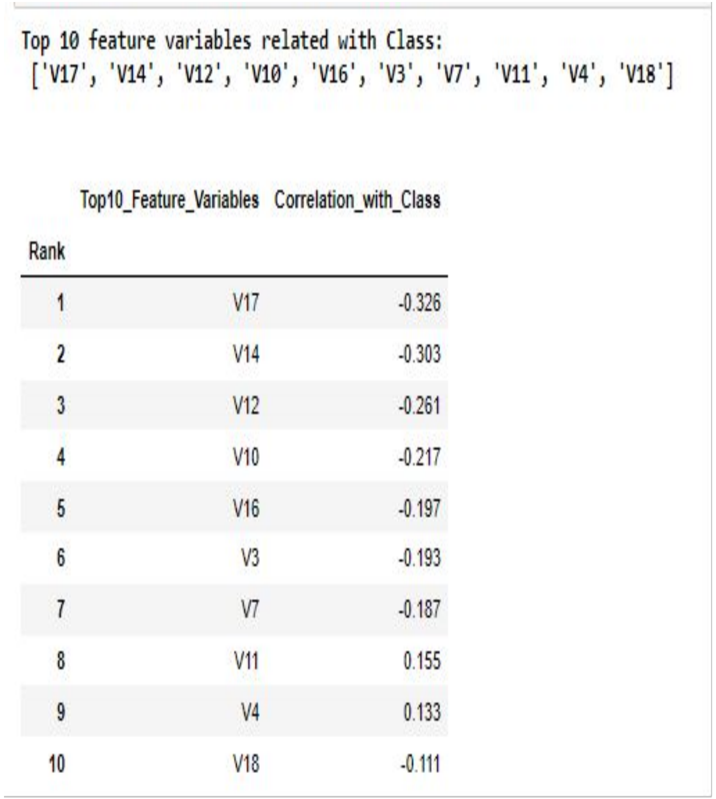
(ii) Some degree of correlation can be seen between (V1, V2 upto V28) and the 'Amount' feature.



	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	Class
V1	1.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.23	-0.10
V2	0.00	1.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.53	0.09
V3	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	-0.21	-0.19
V4	-0.00	-0.00	0.00	1.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.10	0.13
V5	0.00	0.00	-0.00	-0.00	1.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.39	-0.09
V6	-0.00	0.00	0.00	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	0.22	-0.04
V7	-0.00	0.00	0.00	-0.00	0.00	0.00	1.00	0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.40	-0.19
V8	-0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.10	0.02
V9	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.04	-0.10
V10	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	1.00	-0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.10	-0.22
V11	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.15
V12	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	1.00	-0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.01	-0.26
V13	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	1.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.01	-0.00
V14	-0.00	-0.00	0.00	0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	1.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	0.03	-0.30
V15	0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	1.00	0.00	0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00
V16	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	0.00	1.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.20
V17	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.01	-0.33
V18	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	1.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.04	-0.11
V19	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	1.00	0.00	0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.06	0.03
V20	0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	0.00	1.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.34	0.02
V21	-0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	-0.00	1.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.11	0.04
V22	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	1.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.06	0.00
V23	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	1.00	0.00	-0.00	0.00	0.00	0.00	-0.11	-0.00
V24	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	-0.00	-0.00	0.01	-0.01
V25	-0.00	-0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	1.00	0.00	-0.00	-0.00	-0.05	0.00
V26	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	0.00	1.00	-0.00	-0.00	-0.00	0.00
V27	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	-0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	-0.00	-0.00	0.00	0.00	-0.00	-0.00	-0.00	1.00	-0.00	0.03	0.02
V28	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	-0.00	0.00	0.00	-0.00	0.00	0.00	0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	0.00	-0.00	0.00	-0.00	-0.00	-0.00	-0.00	1.00	0.01	0.01
Amount	-0.23	-0.53	-0.21	0.10	-0.39	0.22	0.40	-0.10	-0.04	-0.10	0.00	-0.01	0.01	0.03	-0.00	-0.00	0.01	0.04	-0.06	0.34	0.11	-0.06	-0.11	0.01	-0.05	-0.00	0.03	0.01	1.00	0.01
Class	-0.10	0.09	-0.19	0.13	-0.09	-0.04	-0.19	0.02	-0.10	-0.22	0.15	-0.26	-0.00	-0.30	-0.00	-0.20	-0.33	-0.11	0.03	0.02	0.04	0.00	-0.00	-0.01	0.00	0.00	0.02	0.01	0.01	1.00

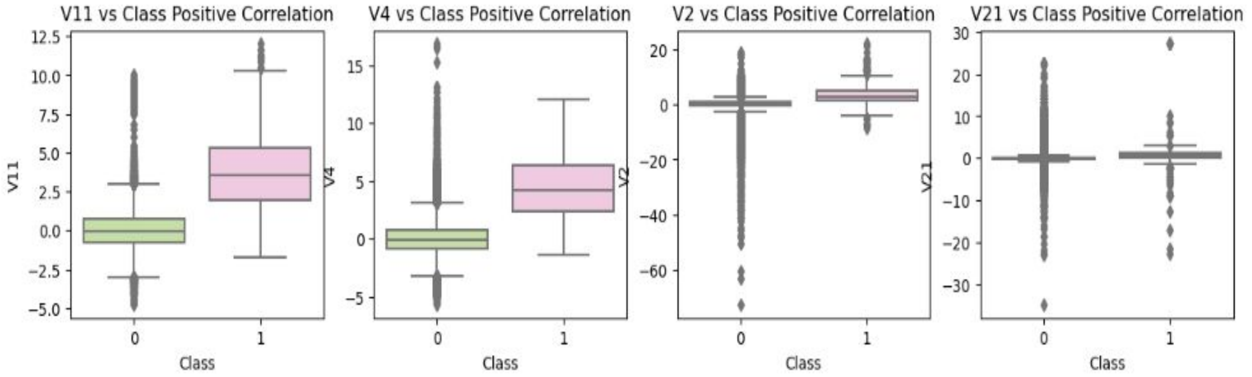
Our Highly Correlated Columns





```
Top 10 feature variables related with Class:
 ['V17', 'V14', 'V12', 'V10', 'V16', 'V3', 'V7', 'V11', 'V4', 'V18']
```

	Top10_Feature_Variables	Correlation_with_Class
Rank		
1	V17	-0.326
2	V14	-0.303
3	V12	-0.261
4	V10	-0.217
5	V16	-0.197
6	V3	-0.193
7	V7	-0.187
8	V11	0.155
9	V4	0.133
10	V18	-0.111



V11 vs Class Positive Correlation

V4 vs Class Positive Correlation

V2 vs Class Positive Correlation

V21 vs Class Positive Correlation

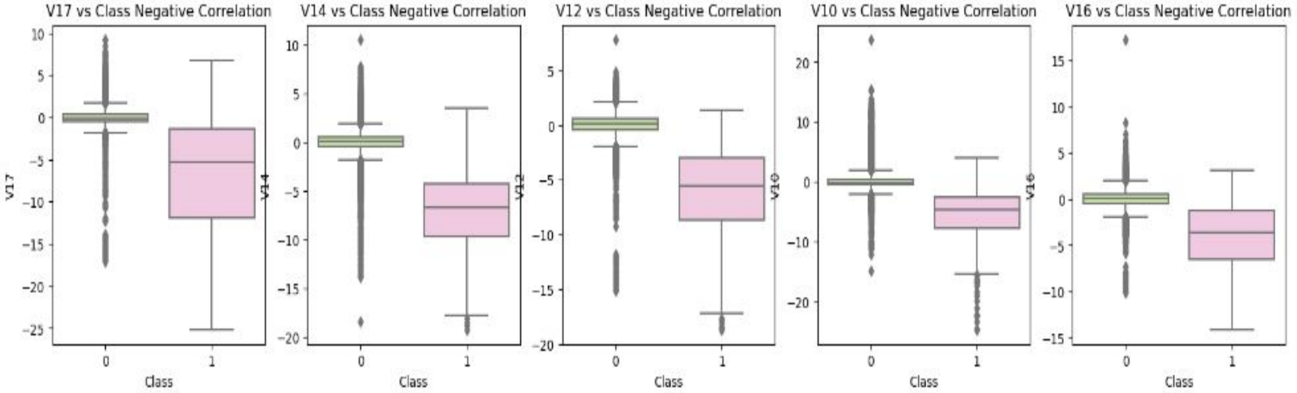
Positive Correlations: V11, V4 are positively correlated.

Notice how the higher these values are, the more likely the end result will be a fraud transaction.

Positive correlations (The higher the feature the probability increases that it will be a fraud transaction)

Negative Correlations: V17, V14, V12 , V10 and V16 are negatively correlated. Notice how the lower these values are, the more likely the end result will be a fraud transaction.

Negative Correlations with our Class (The lower our feature value the more likely it will be a fraud transaction)



V17 vs Class Negative Correlation

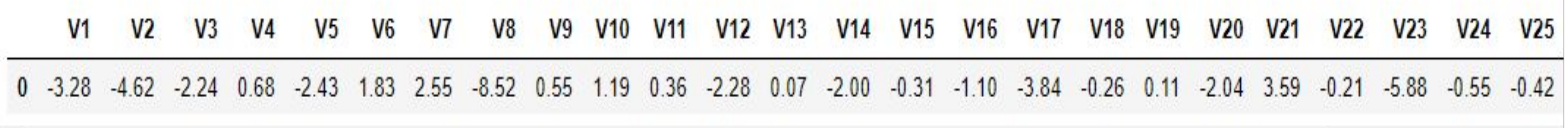
V14 vs Class Negative Correlation

V12 vs Class Negative Correlation

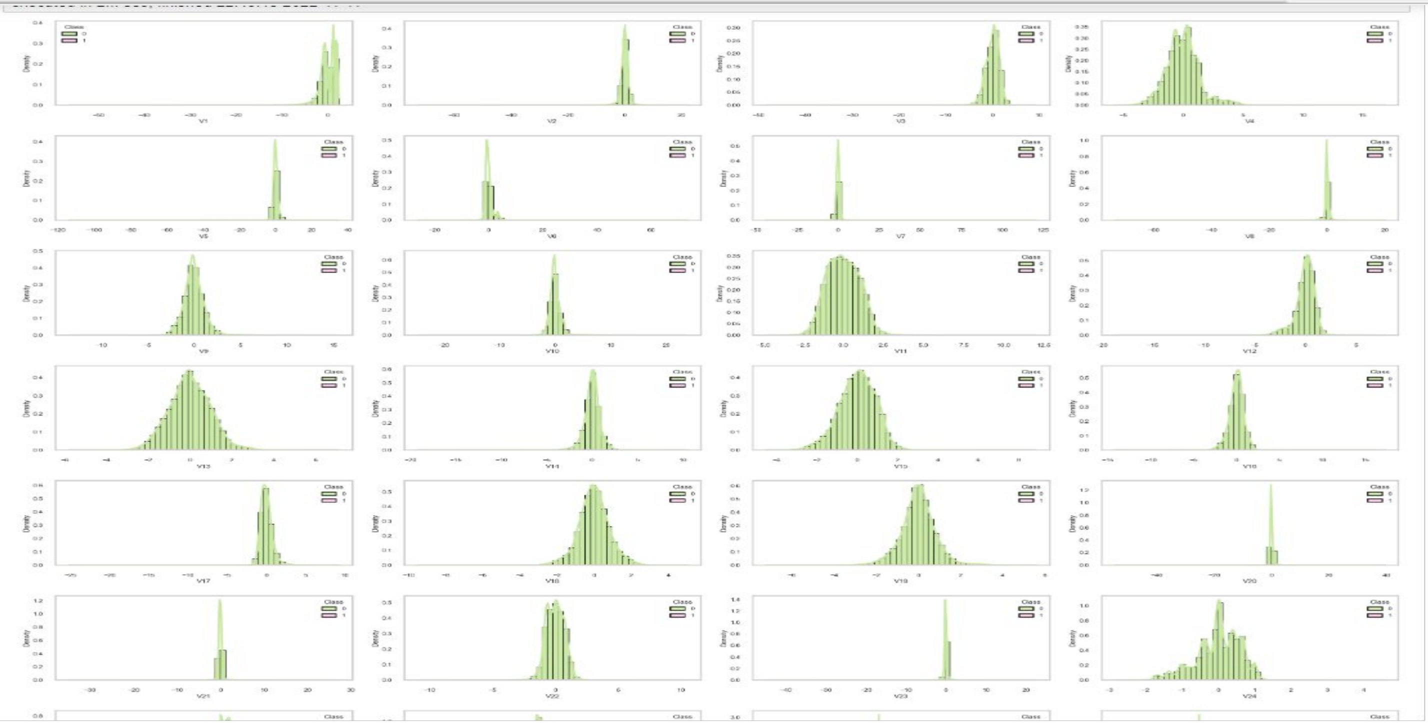
V10 vs Class Negative Correlation

V16 vs Class Negative Correlation

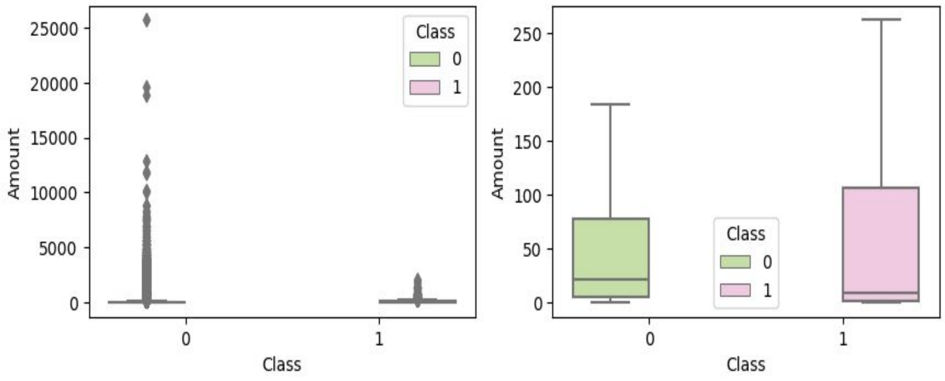
SKEWNESS



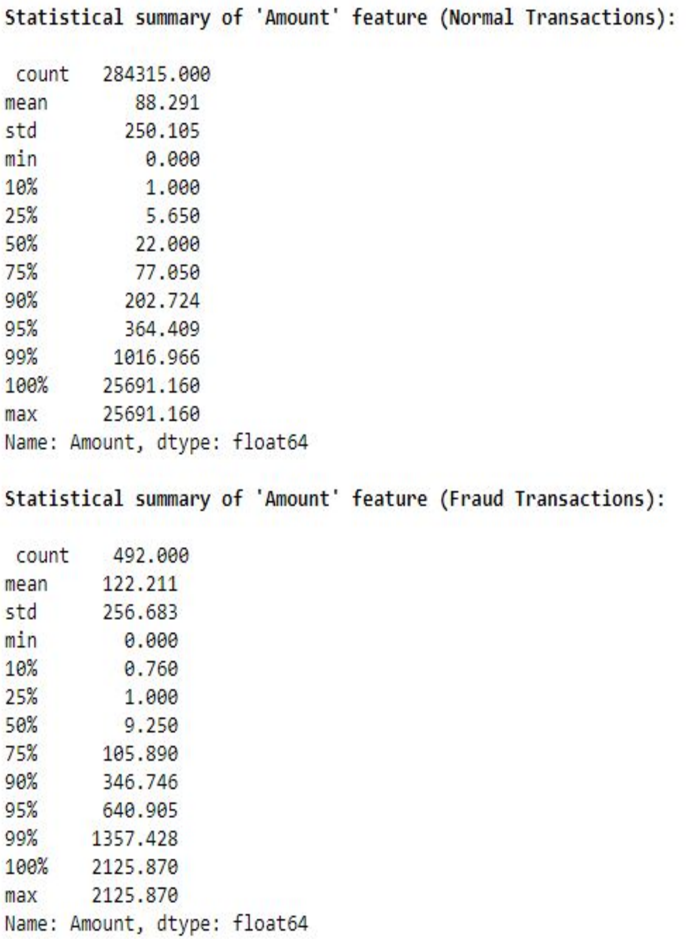
	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25
0	-3.28	-4.62	-2.24	0.68	-2.43	1.83	2.55	-8.52	0.55	1.19	0.36	-2.28	0.07	-2.00	-0.31	-1.10	-3.84	-0.26	0.11	-2.04	3.59	-0.21	-5.88	-0.55	-0.42



Amount

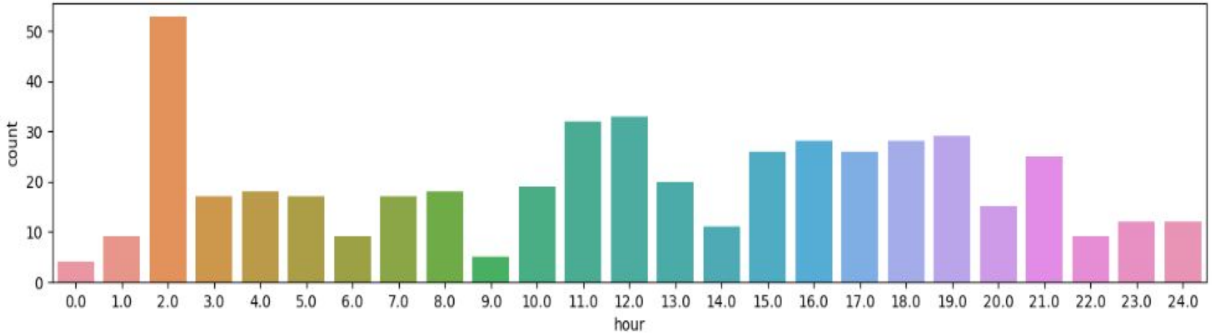


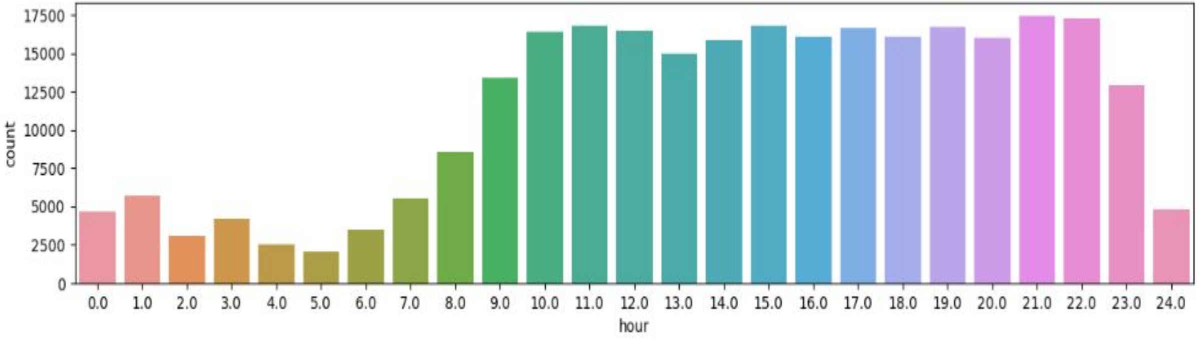
(i) From the plot of distribution of classes with respect to transaction amount, it is quite evident that the transaction amounts for fraud cases were lesser than approximately €2200.
(ii) Precisely, average fraud transaction amount is approximately €122 (from statistical summary) and maximum fraud transaction amount is €2125.87.
(iii) Normal transactions have a wide-range of amounts with certain outliers (as seen from the plot of Distribution of Class with Amount).



```
Statistical summary of 'Amount' feature (Normal Transactions):

 count    284315.000
mean         88.291
std         250.105
min           0.000
10%           1.000
25%           5.650
50%          22.000
75%          77.050
90%         202.724
95%         364.409
99%        1016.966
100%      25691.160
max       25691.160
Name: Amount, dtype: float64

Statistical summary of 'Amount' feature (Fraud Transactions):

 count     492.000
mean      122.211
std       256.683
min         0.000
10%         0.760
25%         1.000
50%         9.250
75%       105.890
90%       346.746
95%       640.905
99%      1357.428
100%     2125.870
max      2125.870
Name: Amount, dtype: float64
```

TIME- HOUR





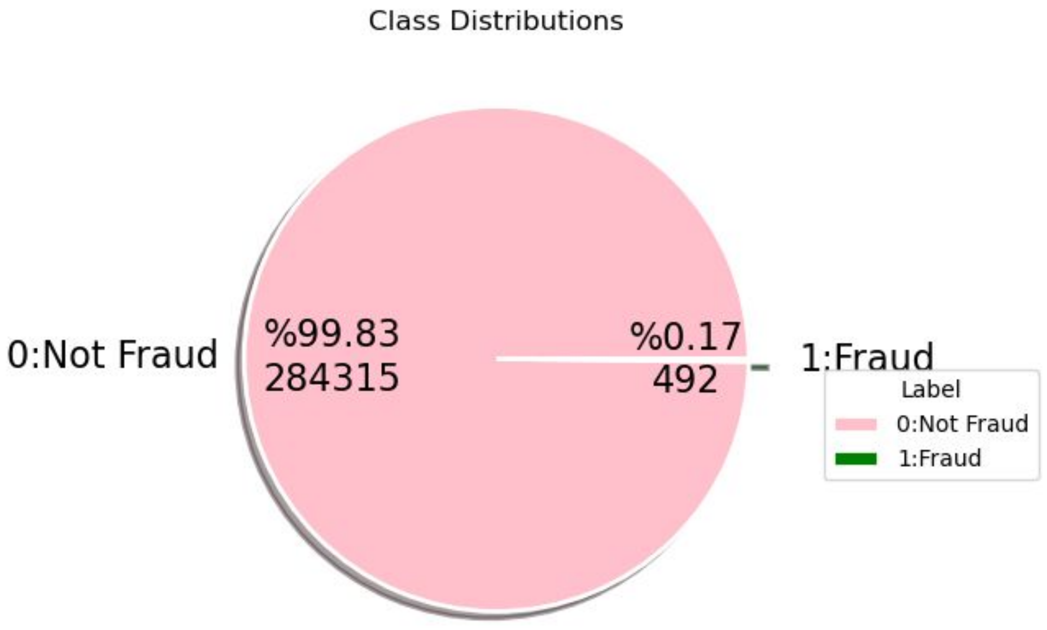
(i) No distinct fixed pattern (w.r.t. fraud transactions) can be seen from the hours.

(ii) From the distribution plot of normal transactions we can infer that most number of normal transactions took place from 08.00 hours till midnight (00.00 hours).

(iii) Normal transactions are less in number during night-time (from midnight (i.e. 00 hours) till 08.00 hours).

(iv) Most number of Fraud Transactions took place at 02.00 (no. of txns= 53), 12.00 (no. of txns= 33) and 11.00 (no. of txns= 32) hours respectively.

Class



Out of a total of 284315 transactions, 492 were fraudulent. This data set is highly unbalanced, with the positive class (frauds) accounting for approximately 0.172% of the total transactions.

As you see that value count of 1 is very less according to 0. It means that our dataset is imbalanced.

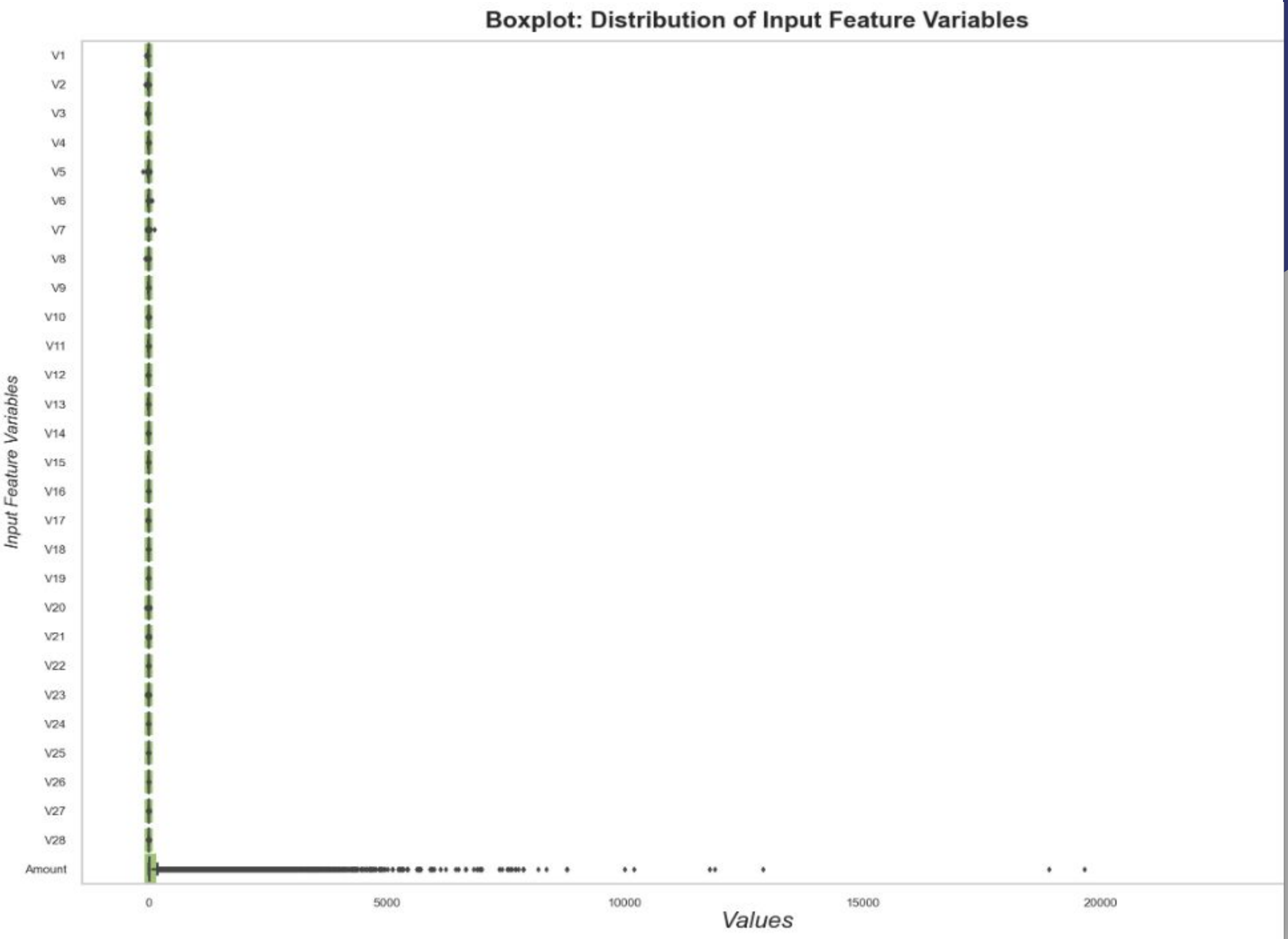
1 represent Fraud and 0 is No Fraud. This situation is quite natural for Fraud Detection studies.

Scaling

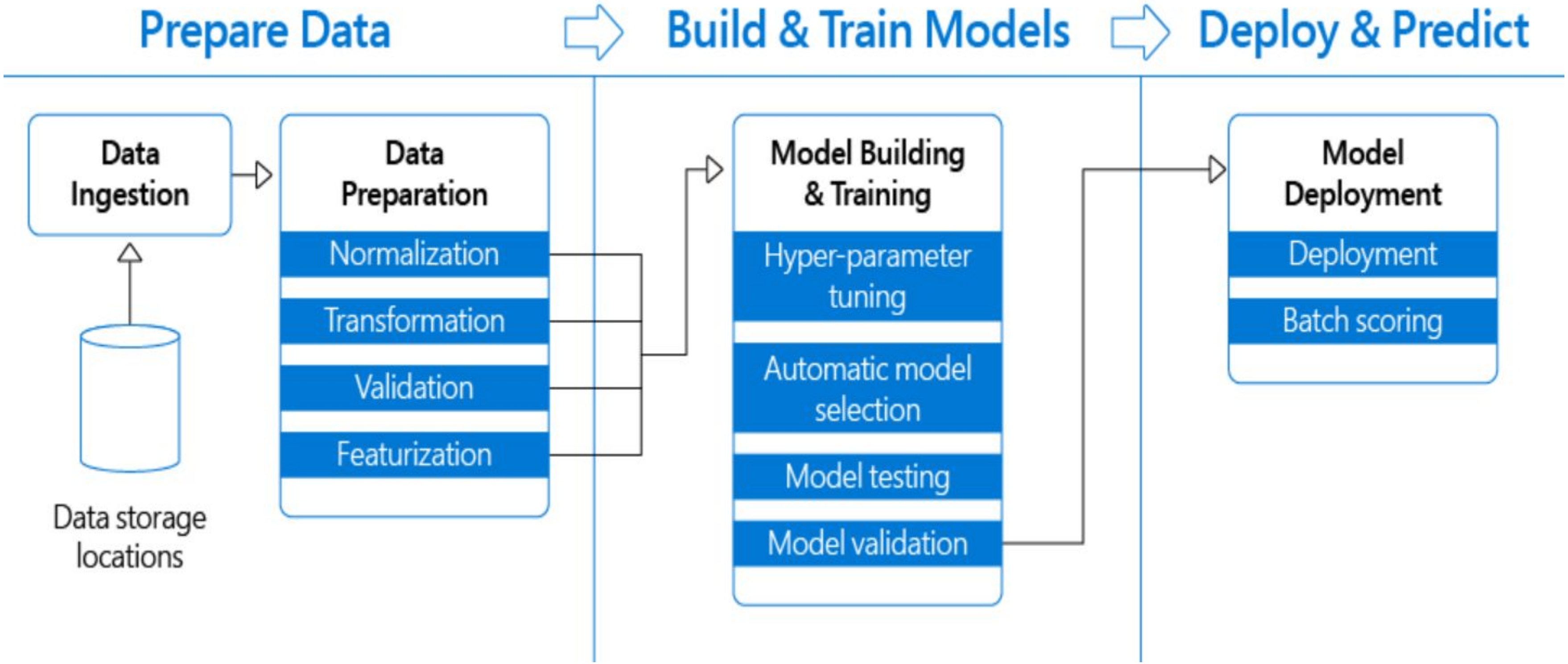
(i) Among the input numerical variables, the PCA features variables are centered across the mean 0. However, skewness in the distribution has to be checked in-depth.

(ii) 'Amount' feature is not centered across mean 0.

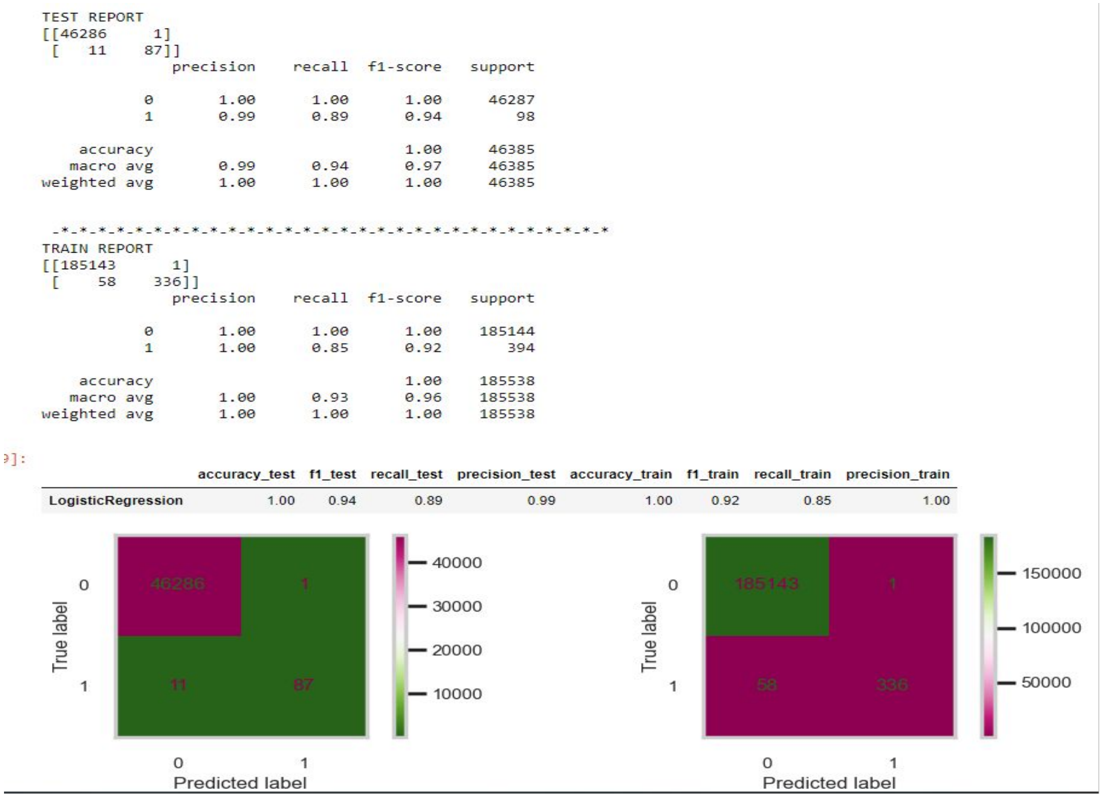
(iii) RobustScaler(), "Amount"---> RobustScaler is less prone to outliers.



MODEL BUILDING



Logistic Regression

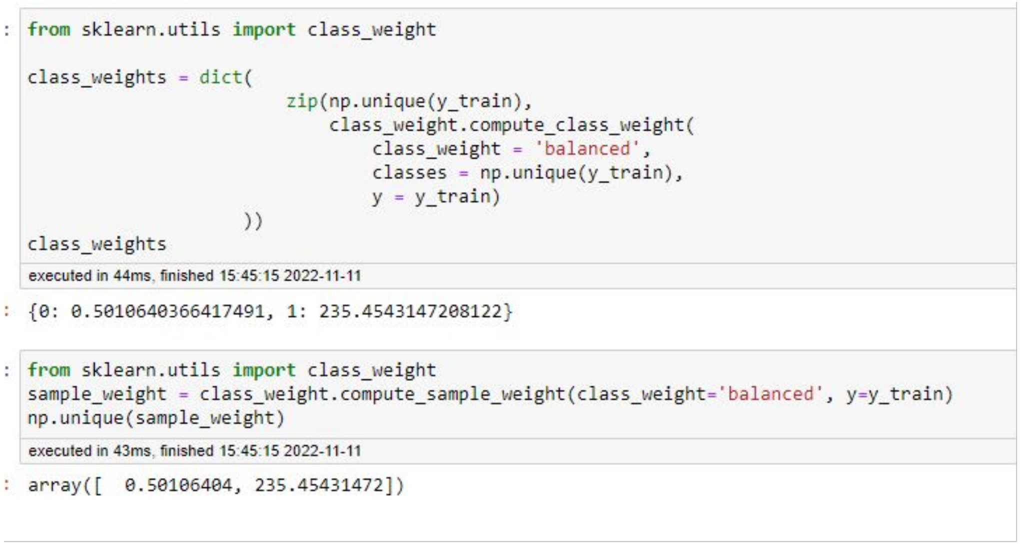


```
TEST REPORT
[[46286     1]
 [   11    87]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       0.99      0.89      0.94        98

    accuracy                           1.00     46385
   macro avg       0.99      0.94      0.97     46385
weighted avg       1.00      1.00      1.00     46385

 -*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
TRAIN REPORT
[[185143      1]
 [    58    336]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.85      0.92       394

    accuracy                           1.00    185538
   macro avg       1.00      0.93      0.96    185538
weighted avg       1.00      1.00      1.00    185538
```

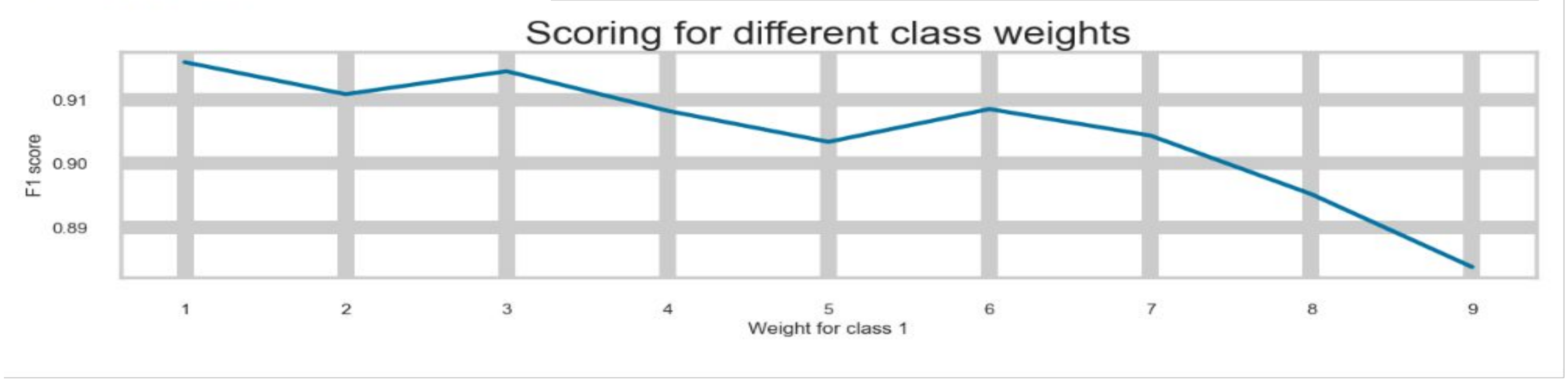
]:

	accuracy_test	f1_test	recall_test	precision_test	accuracy_train	f1_train	recall_train	precision_train
LogisticRegression	1.00	0.94	0.89	0.99	1.00	0.92	0.85	1.00

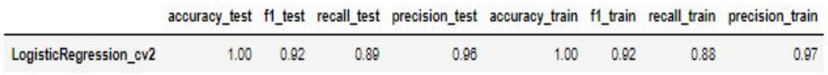
* With Best Parameter: class_weight (GridsearchCV)*



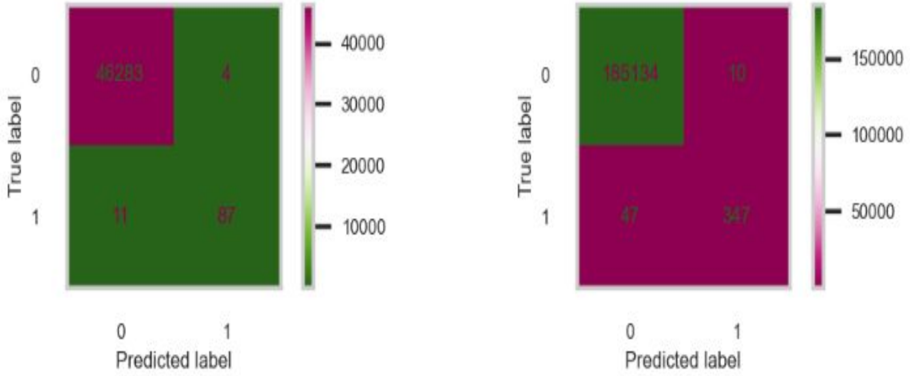
```
: from sklearn.utils import class_weight

  class_weights = dict(
                          zip(np.unique(y_train),
                              class_weight.compute_class_weight(
                                  class_weight = 'balanced',
                                  classes = np.unique(y_train),
                                  y = y_train)
                     ))
  class_weights
  executed in 44ms, finished 15:45:15 2022-11-11

: {0: 0.5010640366417491, 1: 235.4543147208122}

: from sklearn.utils import class_weight
  sample_weight = class_weight.compute_sample_weight(class_weight='balanced', y=y_train)
  np.unique(sample_weight)
  executed in 43ms, finished 15:45:15 2022-11-11

: array([  0.50106404, 235.45431472])
```

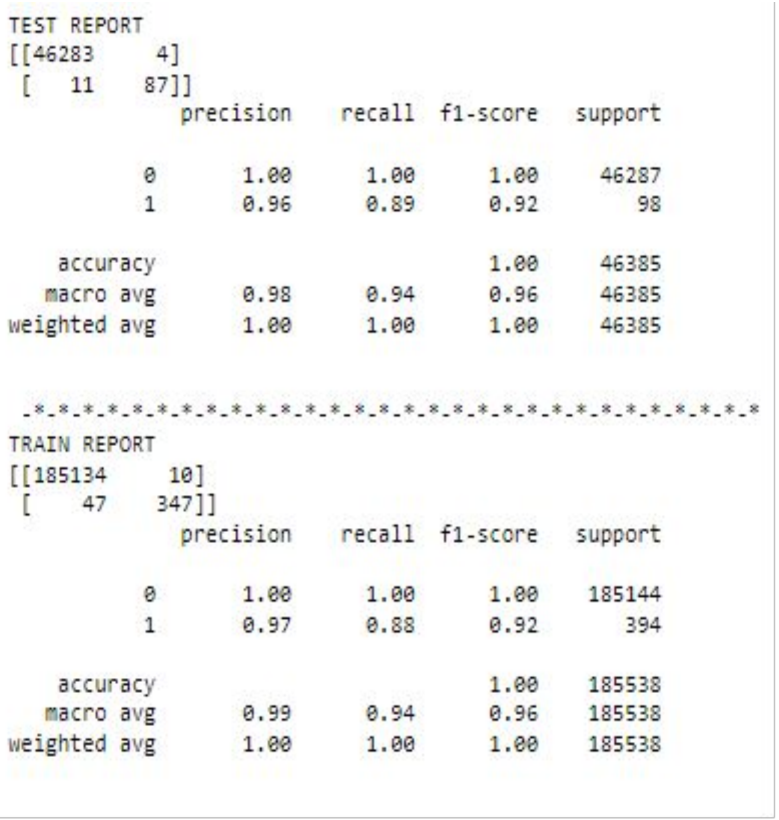


Logistic Regression With Best Parameters(GridsearchCV)



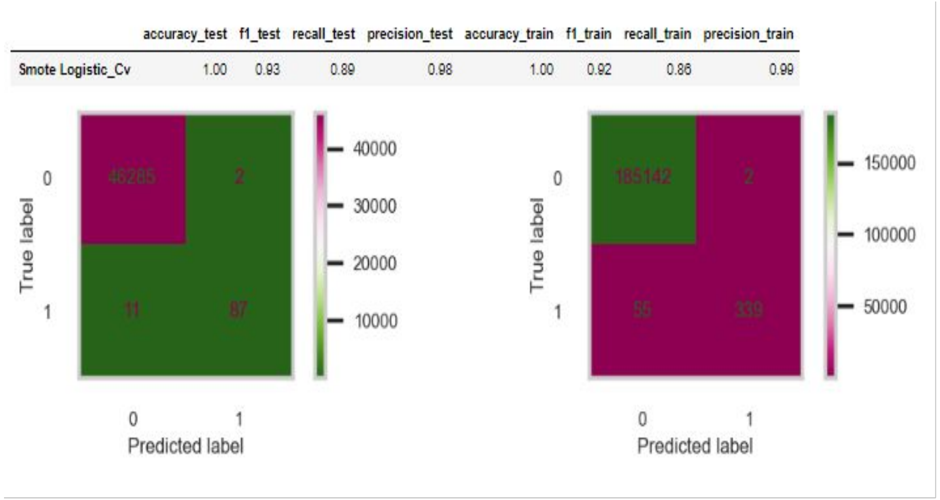
	accuracy_test	f1_test	recall_test	precision_test	accuracy_train	f1_train	recall_train	precision_train
LogisticRegression_cv2	1.00	0.92	0.89	0.96	1.00	0.92	0.88	0.97



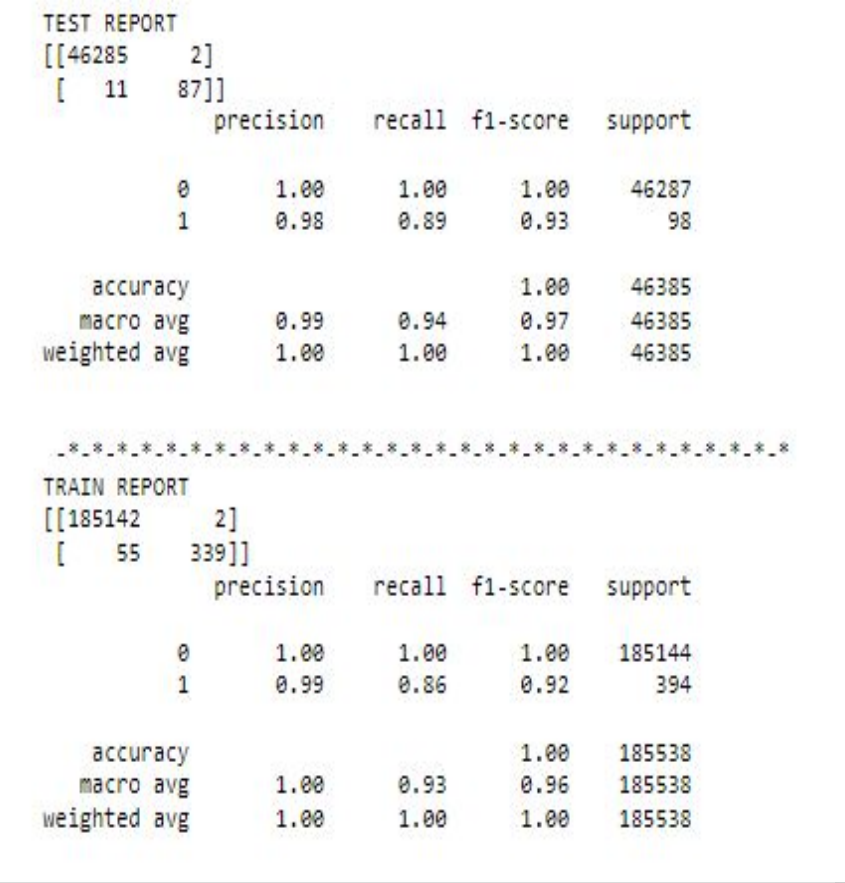


```
TEST REPORT
[[46283     4]
 [   11    87]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       0.96      0.89      0.92        98

    accuracy                           1.00     46385
   macro avg       0.98      0.94      0.96     46385
weighted avg       1.00      1.00      1.00     46385

 -*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
TRAIN REPORT
[[185134     10]
 [    47    347]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       0.97      0.88      0.92       394

    accuracy                           1.00    185538
   macro avg       0.99      0.94      0.96    185538
weighted avg       1.00      1.00      1.00    185538
```

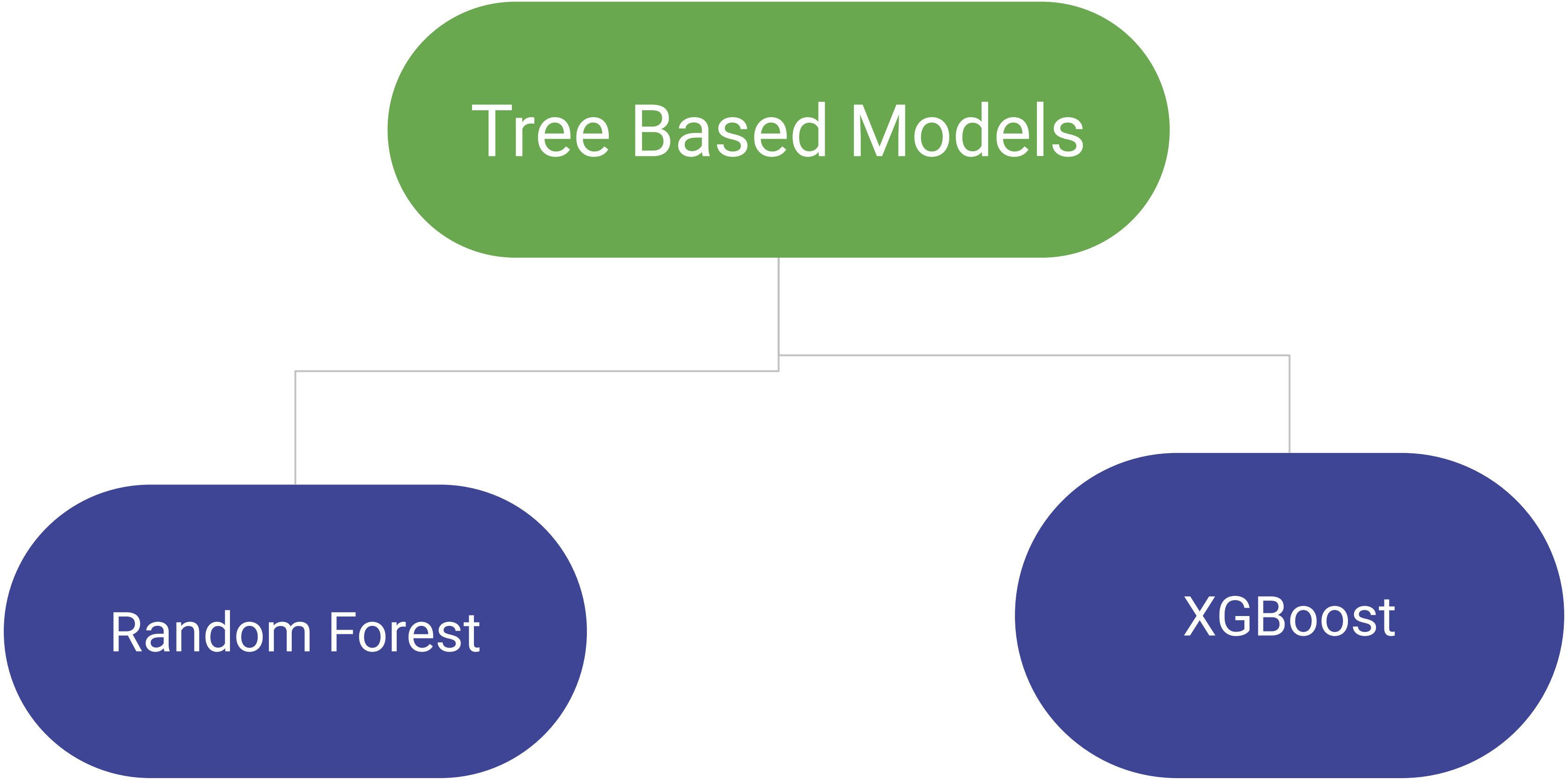
Logistic Regression Smote GridSearchCV



	accuracy_test	f1_test	recall_test	precision_test	accuracy_train	f1_train	recall_train	precision_train
Smote Logistic_Cv	1.00	0.93	0.89	0.98	1.00	0.92	0.86	0.99



```
TEST REPORT
[[46285     2]
 [   11    87]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       0.98      0.89      0.93        98

    accuracy                           1.00     46385
   macro avg       0.99      0.94      0.97     46385
weighted avg       1.00      1.00      1.00     46385

 -*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
TRAIN REPORT
[[185142      2]
 [    55    339]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       0.99      0.86      0.92       394

    accuracy                           1.00    185538
   macro avg       1.00      0.93      0.96    185538
weighted avg       1.00      1.00      1.00    185538
```



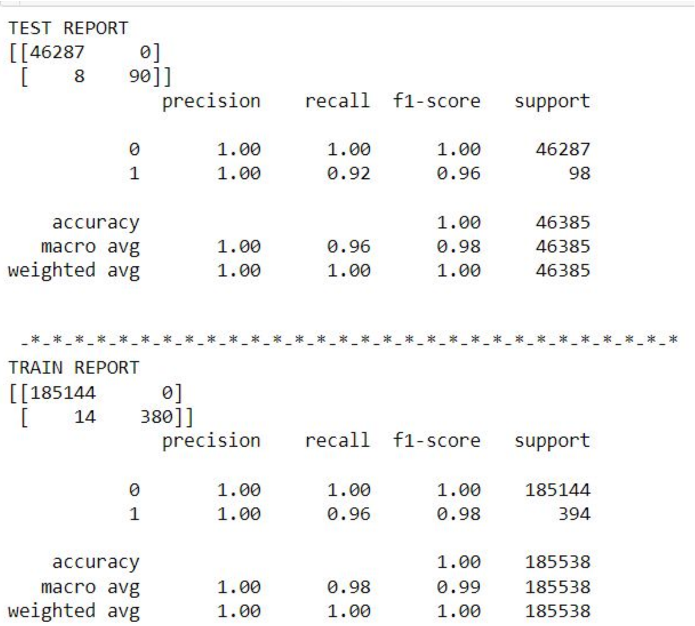
Tree Based Models

Random Forest

XGBoost

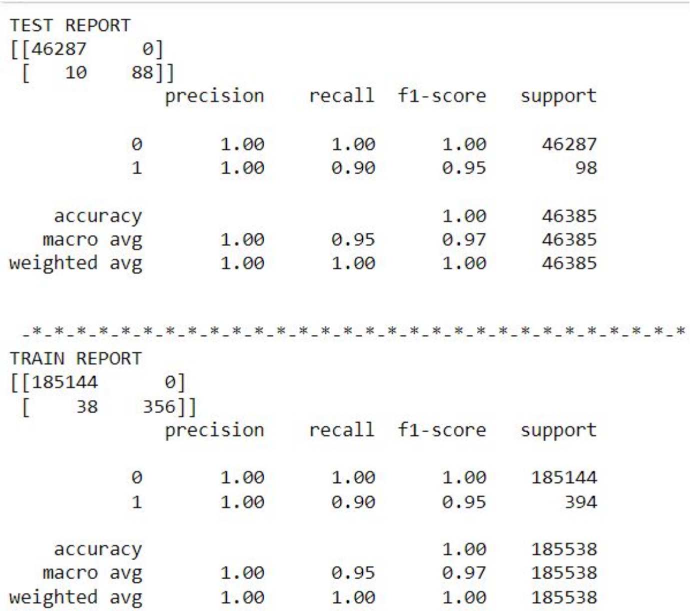
Random Forest Classifier with Unbalanced Data Techniques- class_weight

Before



```
TEST REPORT
[[46287     0]
 [    8    90]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.92      0.96        98

    accuracy                           1.00     46385
   macro avg       1.00      0.96      0.98     46385
weighted avg       1.00      1.00      1.00     46385

 _*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*
TRAIN REPORT
[[185144      0]
 [    14    380]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.96      0.98       394

    accuracy                           1.00    185538
   macro avg       1.00      0.98      0.99    185538
weighted avg       1.00      1.00      1.00    185538
```

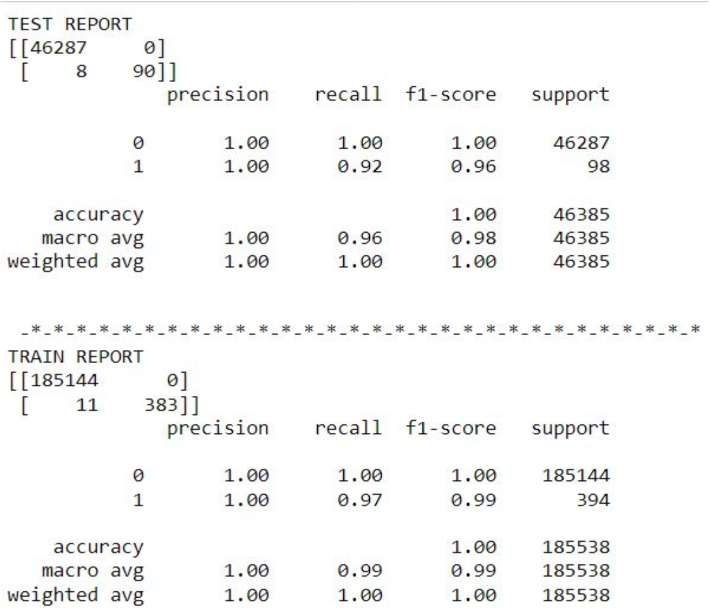
After



```
TEST REPORT
[[46287     0]
 [   10    88]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.90      0.95        98

    accuracy                           1.00     46385
   macro avg       1.00      0.95      0.97     46385
weighted avg       1.00      1.00      1.00     46385

 _*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*
TRAIN REPORT
[[185144      0]
 [    38    356]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.90      0.95       394

    accuracy                           1.00    185538
   macro avg       1.00      0.95      0.97    185538
weighted avg       1.00      1.00      1.00    185538
```

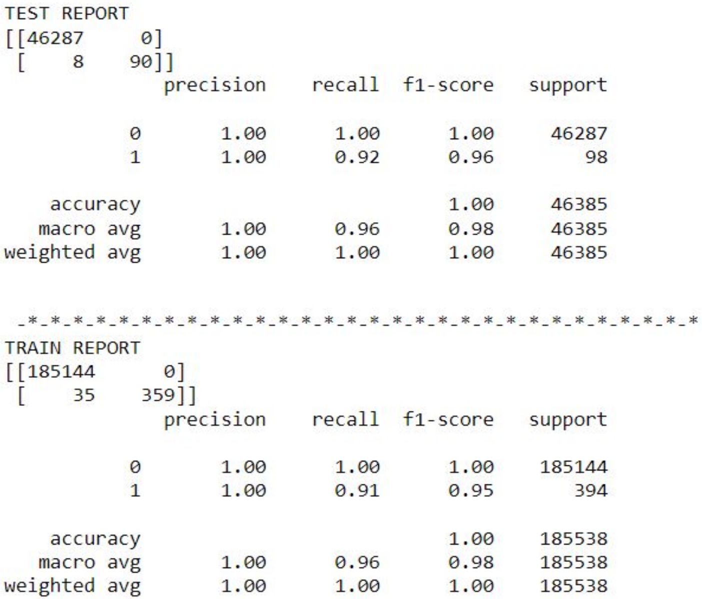
Random Forest Classifier with Unbalanced Data Techniques-SMOTE

Before



```
TEST REPORT
[[46287     0]
 [    8    90]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.92      0.96        98

    accuracy                           1.00     46385
   macro avg       1.00      0.96      0.98     46385
weighted avg       1.00      1.00      1.00     46385


 _*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*
TRAIN REPORT
[[185144      0]
 [    11    383]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.97      0.99       394

    accuracy                           1.00    185538
   macro avg       1.00      0.99      0.99    185538
weighted avg       1.00      1.00      1.00    185538
```

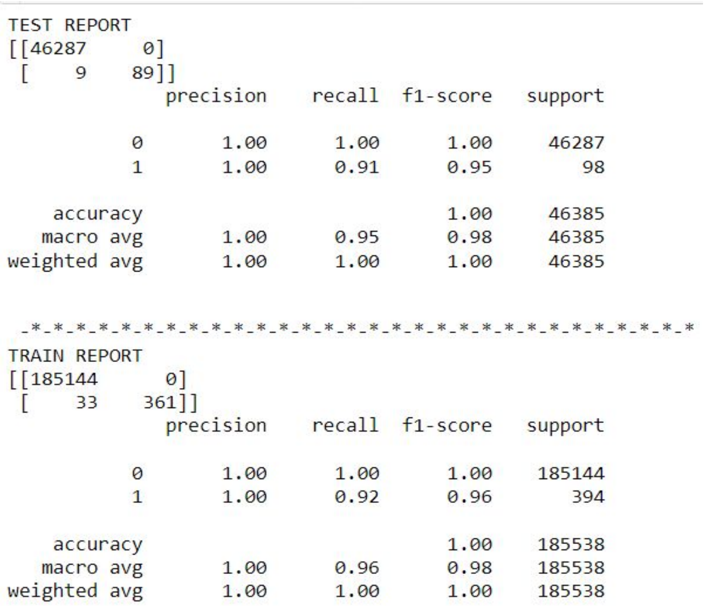
After



```
TEST REPORT
[[46287     0]
 [    8    90]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.92      0.96        98

    accuracy                           1.00     46385
   macro avg       1.00      0.96      0.98     46385
weighted avg       1.00      1.00      1.00     46385


 _*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*
TRAIN REPORT
[[185144      0]
 [    35    359]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.91      0.95       394

    accuracy                           1.00    185538
   macro avg       1.00      0.96      0.98    185538
weighted avg       1.00      1.00      1.00    185538
```

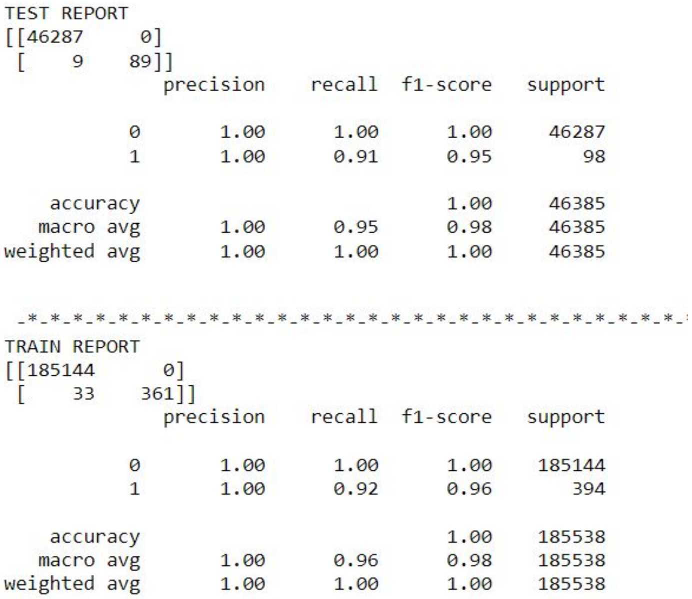
XGBoost Classifier with Unbalanced Data Techniques- class_weight

Before



```
TEST REPORT
[[46287     0]
 [    9    89]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.91      0.95        98

    accuracy                           1.00     46385
   macro avg       1.00      0.95      0.98     46385
weighted avg       1.00      1.00      1.00     46385


 _*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*
TRAIN REPORT
[[185144      0]
 [    33    361]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.92      0.96       394

    accuracy                           1.00    185538
   macro avg       1.00      0.96      0.98    185538
weighted avg       1.00      1.00      1.00    185538
```

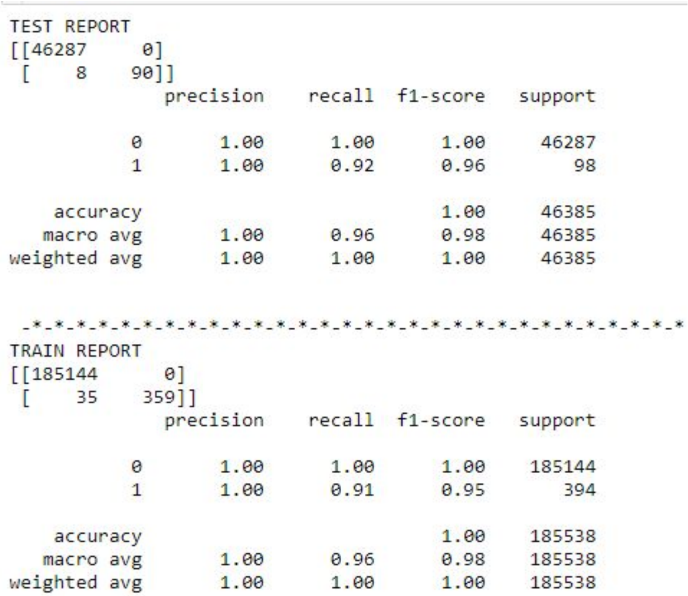
After



```
TEST REPORT
[[46287     0]
 [    9    89]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.91      0.95        98

    accuracy                           1.00     46385
   macro avg       1.00      0.95      0.98     46385
weighted avg       1.00      1.00      1.00     46385


 _*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_*_
TRAIN REPORT
[[185144      0]
 [    33    361]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.92      0.96       394

    accuracy                           1.00    185538
   macro avg       1.00      0.96      0.98    185538
weighted avg       1.00      1.00      1.00    185538
```

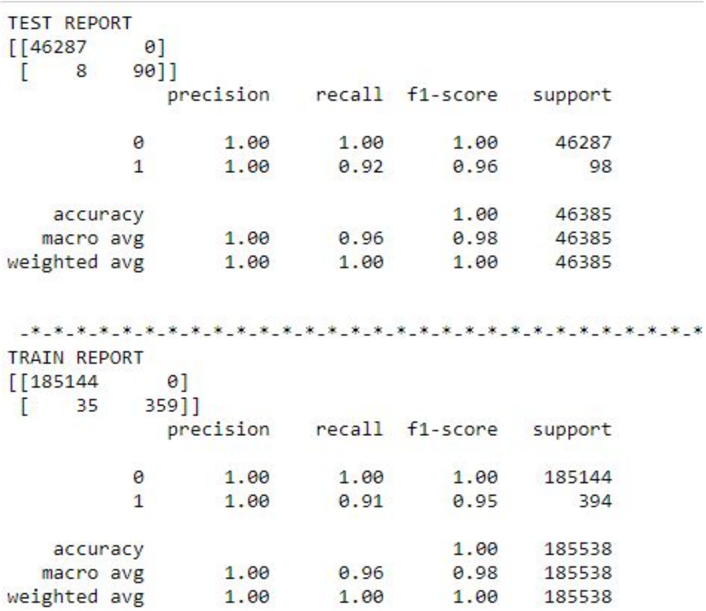
XGBoost Classifier with Unbalanced Data Techniques-SMOTE

Before



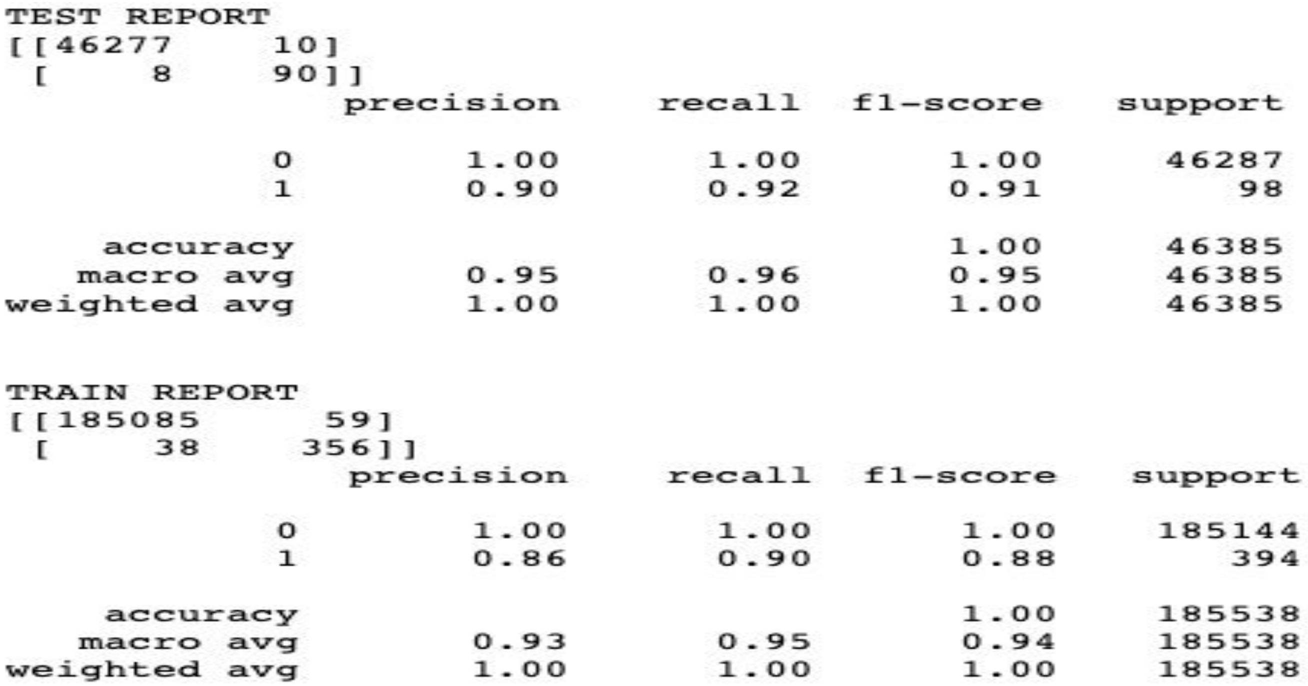
```
TEST REPORT
[[46287     0]
 [    8    90]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.92      0.96        98

    accuracy                           1.00     46385
   macro avg       1.00      0.96      0.98     46385
weighted avg       1.00      1.00      1.00     46385


 -*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
TRAIN REPORT
[[185144      0]
 [    35    359]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.91      0.95       394

    accuracy                           1.00    185538
   macro avg       1.00      0.96      0.98    185538
weighted avg       1.00      1.00      1.00    185538
```

After

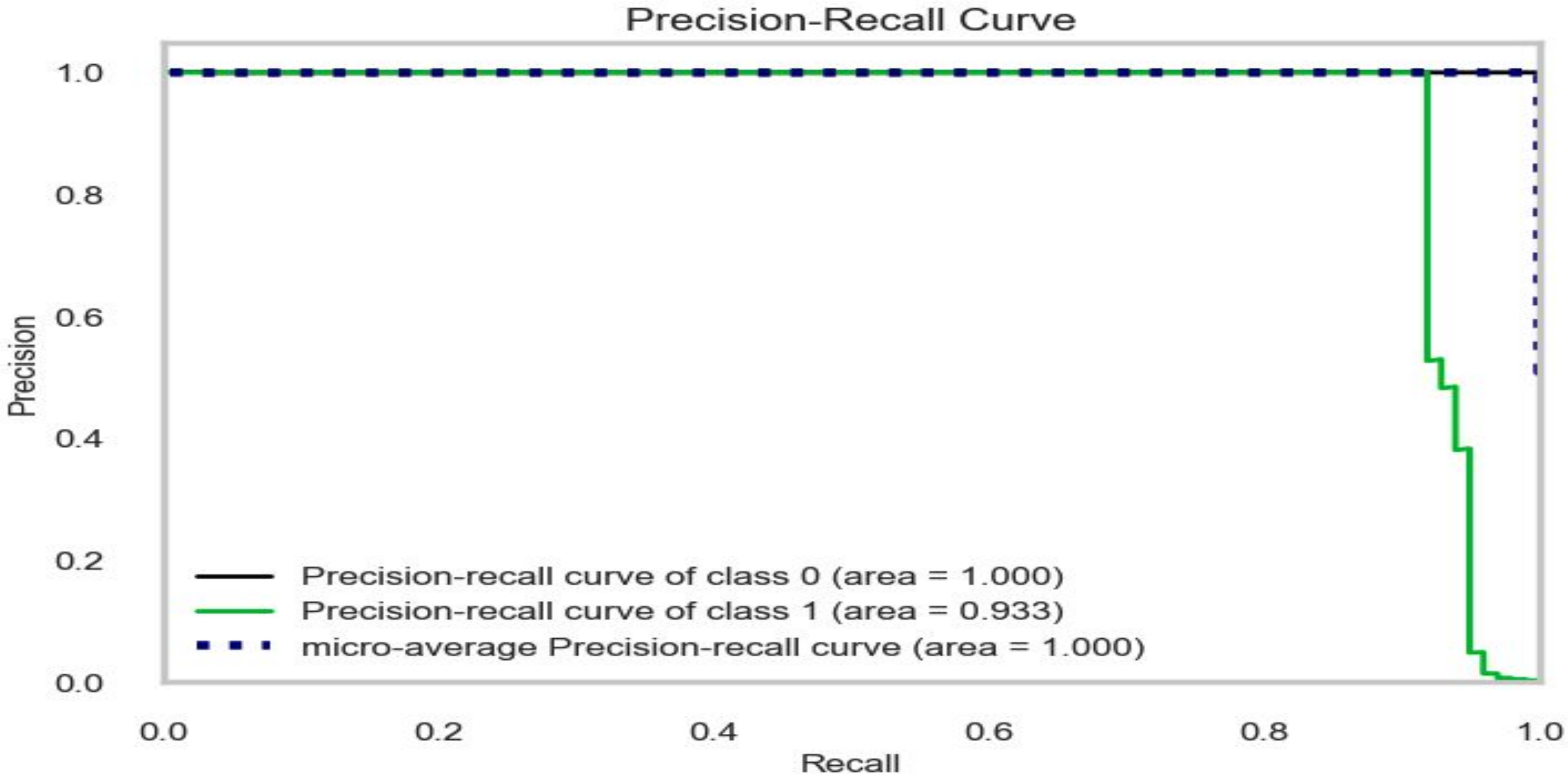


```
TEST REPORT
[[46287     0]
 [    8    90]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       1.00      0.92      0.96        98

    accuracy                           1.00     46385
   macro avg       1.00      0.96      0.98     46385
weighted avg       1.00      1.00      1.00     46385


 -*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
TRAIN REPORT
[[185144      0]
 [    35    359]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       1.00      0.91      0.95       394

    accuracy                           1.00    185538
   macro avg       1.00      0.96      0.98    185538
weighted avg       1.00      1.00      1.00    185538
```

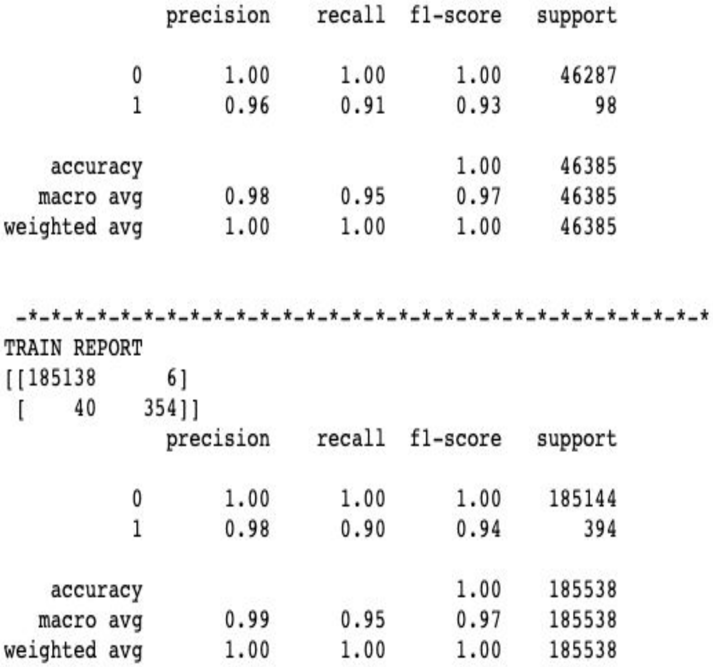
NEURAL NETWORK



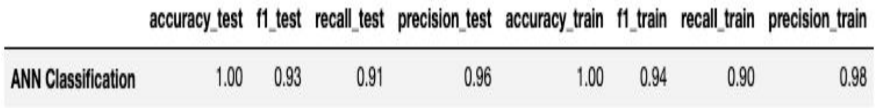
```
TEST REPORT
[[46277    10]
 [    8    90]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       0.90      0.92      0.91        98

    accuracy                           1.00     46385
   macro avg       0.95      0.96      0.95     46385
weighted avg       1.00      1.00      1.00     46385


TRAIN REPORT
[[185085     59]
 [    38    356]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       0.86      0.90      0.88       394

    accuracy                           1.00    185538
   macro avg       0.93      0.95      0.94    185538
weighted avg       1.00      1.00      1.00    185538
```


Precision-Recall Curve
1.0
0.8
0.6
0.4
0.2
0.0
Precision
0.0
0.2
0.4
0.6
0.8
1.0
Recall
Precision-recall curve of class 0 (area = 1.000)
Precision-recall curve of class 1 (area = 0.933)
micro-average Precision-recall curve (area = 1.000)

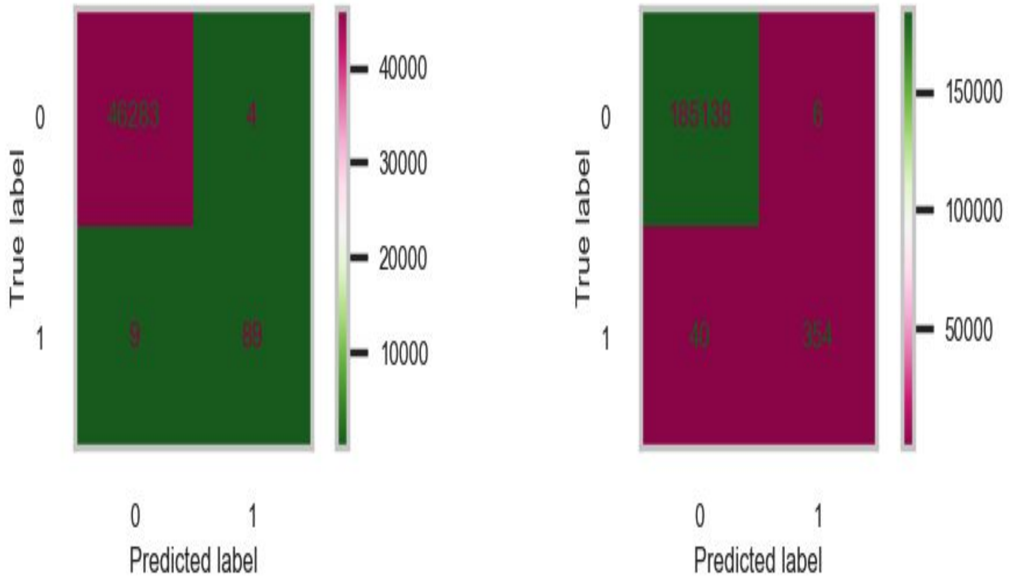
NEURAL NETWORK WITH GRID SEARCH CV

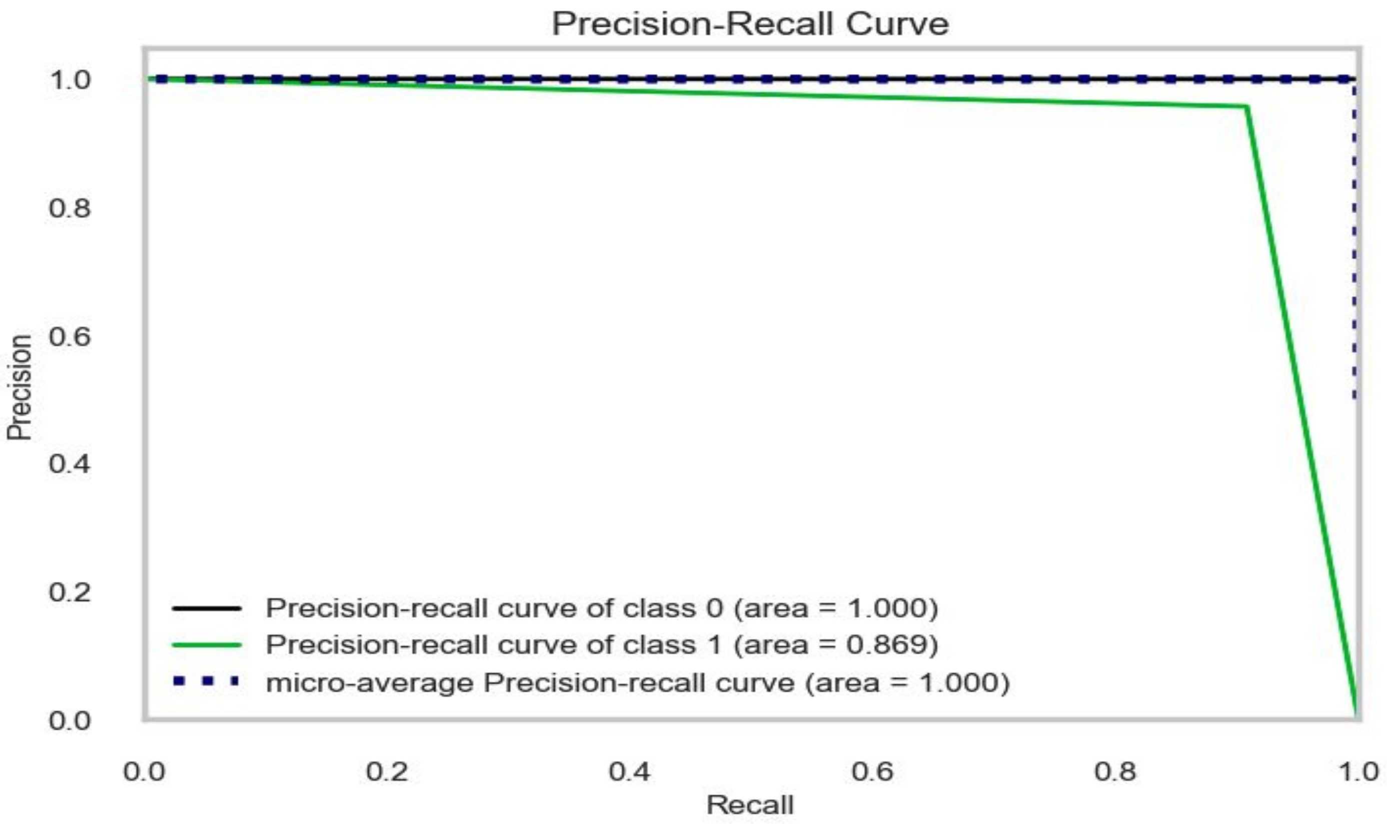


```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       0.96      0.91      0.93        98

    accuracy                           1.00     46385
   macro avg       0.98      0.95      0.97     46385
weighted avg       1.00      1.00      1.00     46385

 -*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
TRAIN REPORT
[[185138      6]
 [    40    354]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       0.98      0.90      0.94       394

    accuracy                           1.00    185538
   macro avg       0.99      0.95      0.97    185538
weighted avg       1.00      1.00      1.00    185538
```

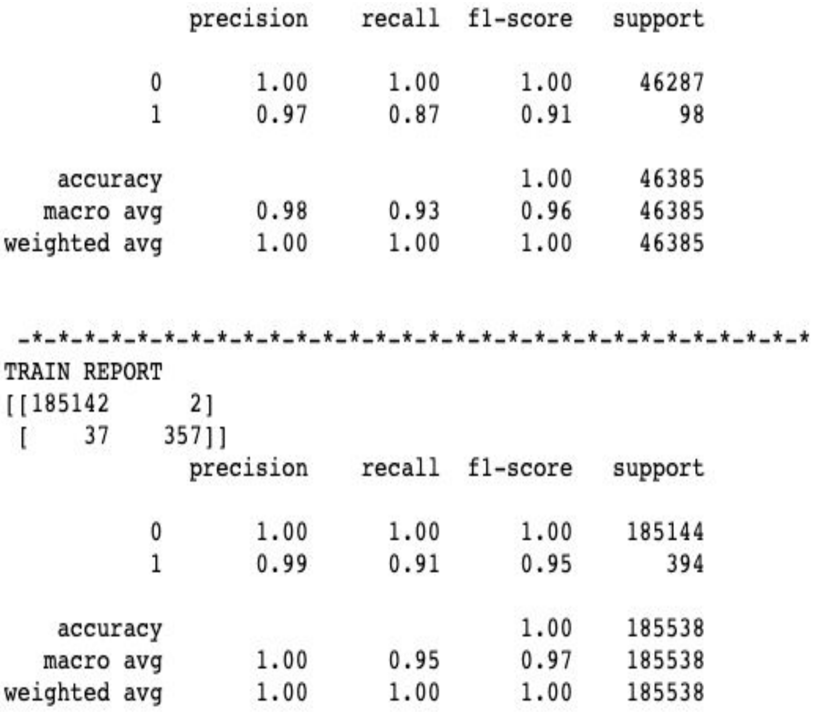


	accuracy_test	f1_test	recall_test	precision_test	accuracy_train	f1_train	recall_train	precision_train
ANN Classification	1.00	0.93	0.91	0.96	1.00	0.94	0.90	0.98

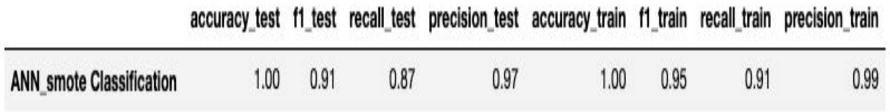



Precision-Recall Curve
1.0
0.8
0.6
0.4
0.2
0.0
Precision
0.0
0.2
0.4
0.6
0.8
1.0
Recall
Precision-recall curve of class 0 (area = 1.000)
Precision-recall curve of class 1 (area = 0.869)
micro-average Precision-recall curve (area = 1.000)

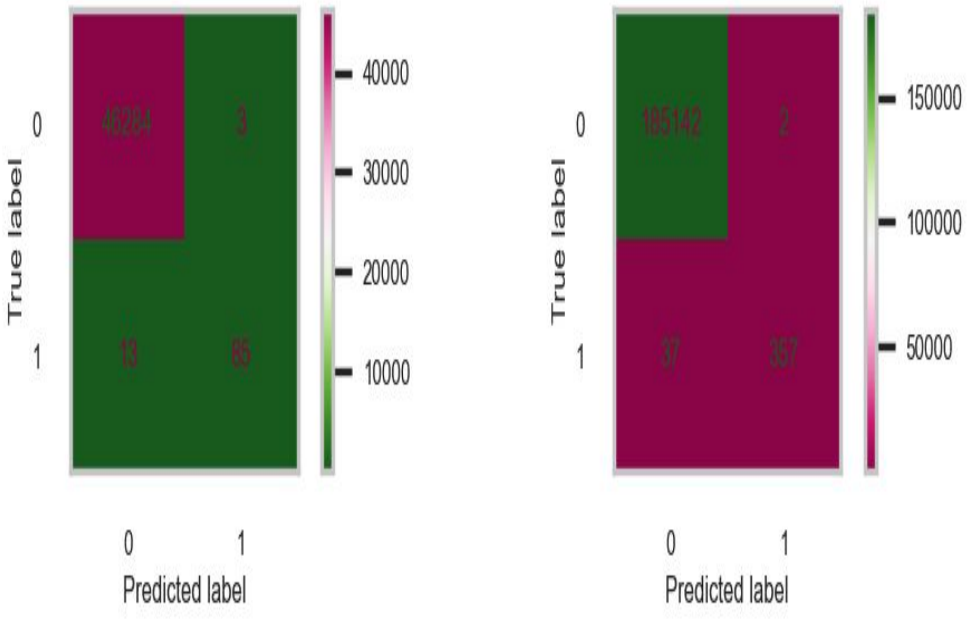
NEURAL NETWORK WITH SMOTE



```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     46287
           1       0.97      0.87      0.91        98

    accuracy                           1.00     46385
   macro avg       0.98      0.93      0.96     46385
weighted avg       1.00      1.00      1.00     46385


 -*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
TRAIN REPORT
[[185142      2]
 [    37    357]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    185144
           1       0.99      0.91      0.95       394

    accuracy                           1.00    185538
   macro avg       1.00      0.95      0.97    185538
weighted avg       1.00      1.00      1.00    185538
```

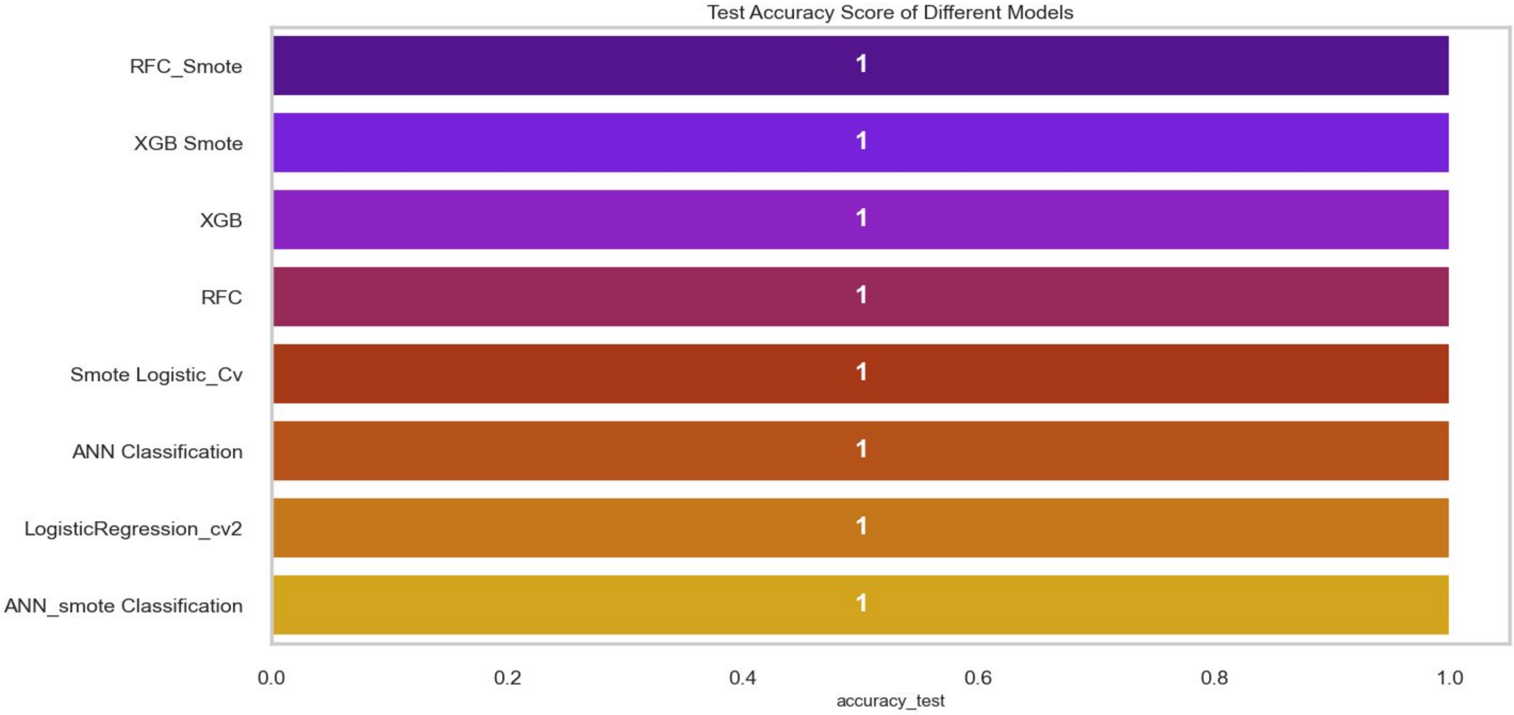


	accuracy_test	f1_test	recall_test	precision_test	accuracy_train	f1_train	recall_train	precision_train
ANN_smote Classification	1.00	0.91	0.87	0.97	1.00	0.95	0.91	0.99



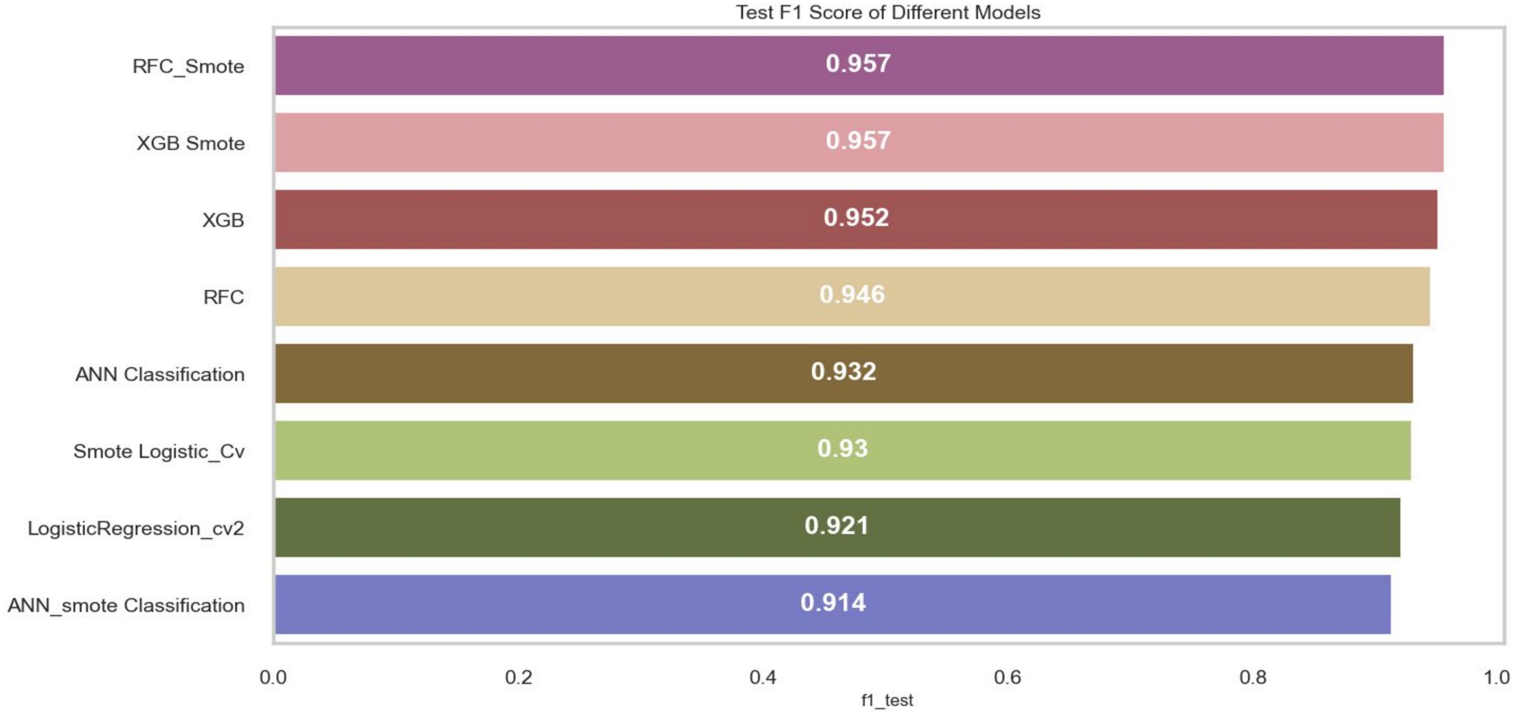
Compare Models

Accuracy Comparison



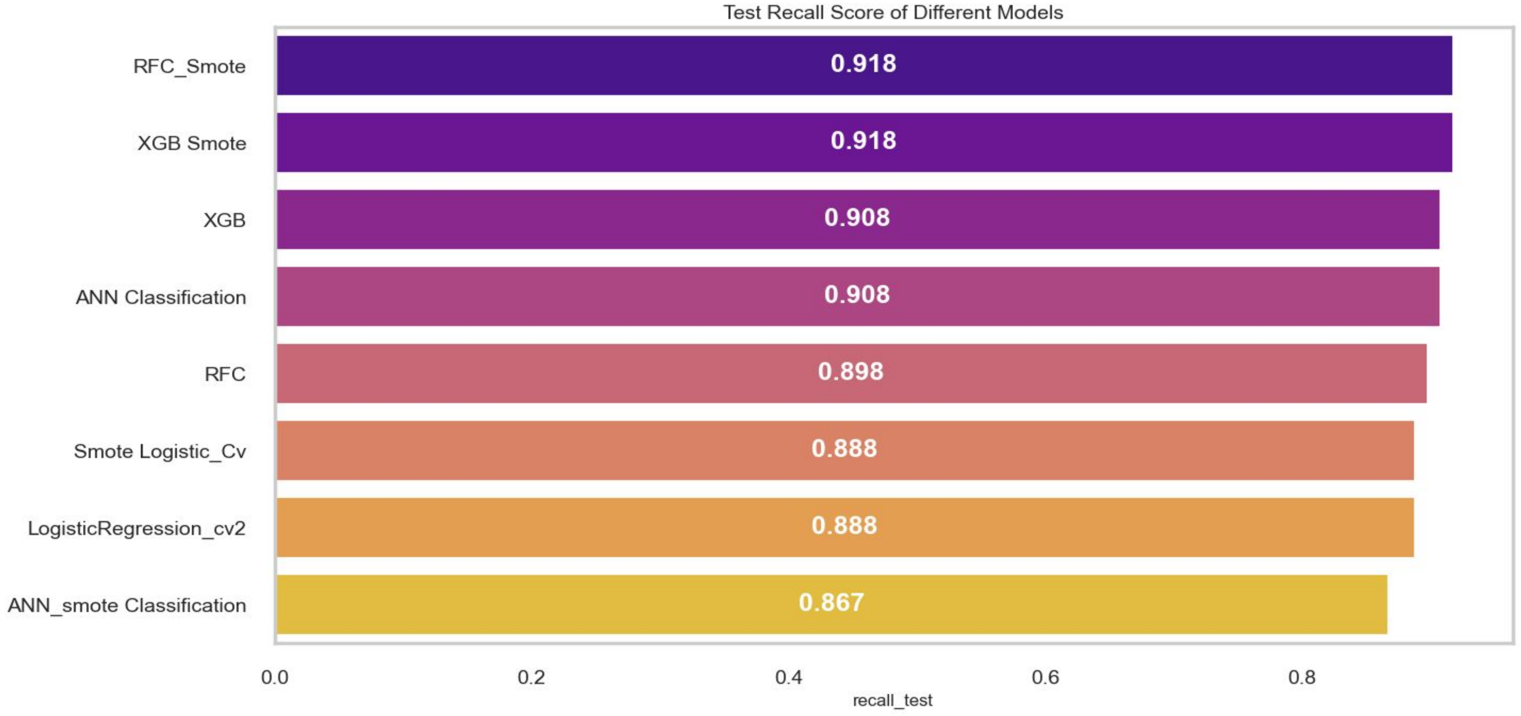
Compare Models

F1 Score Comparison



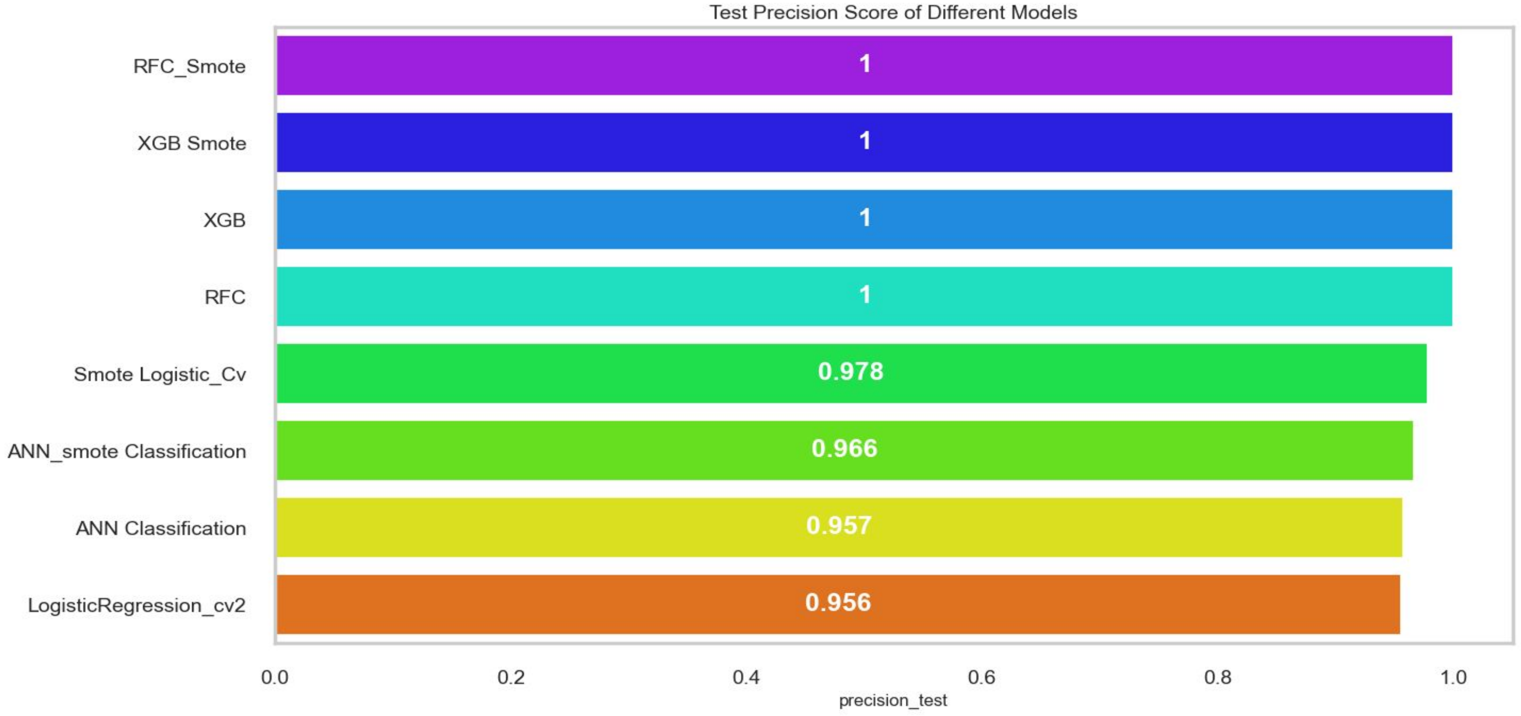
Compare Models

Recall Score Comparison

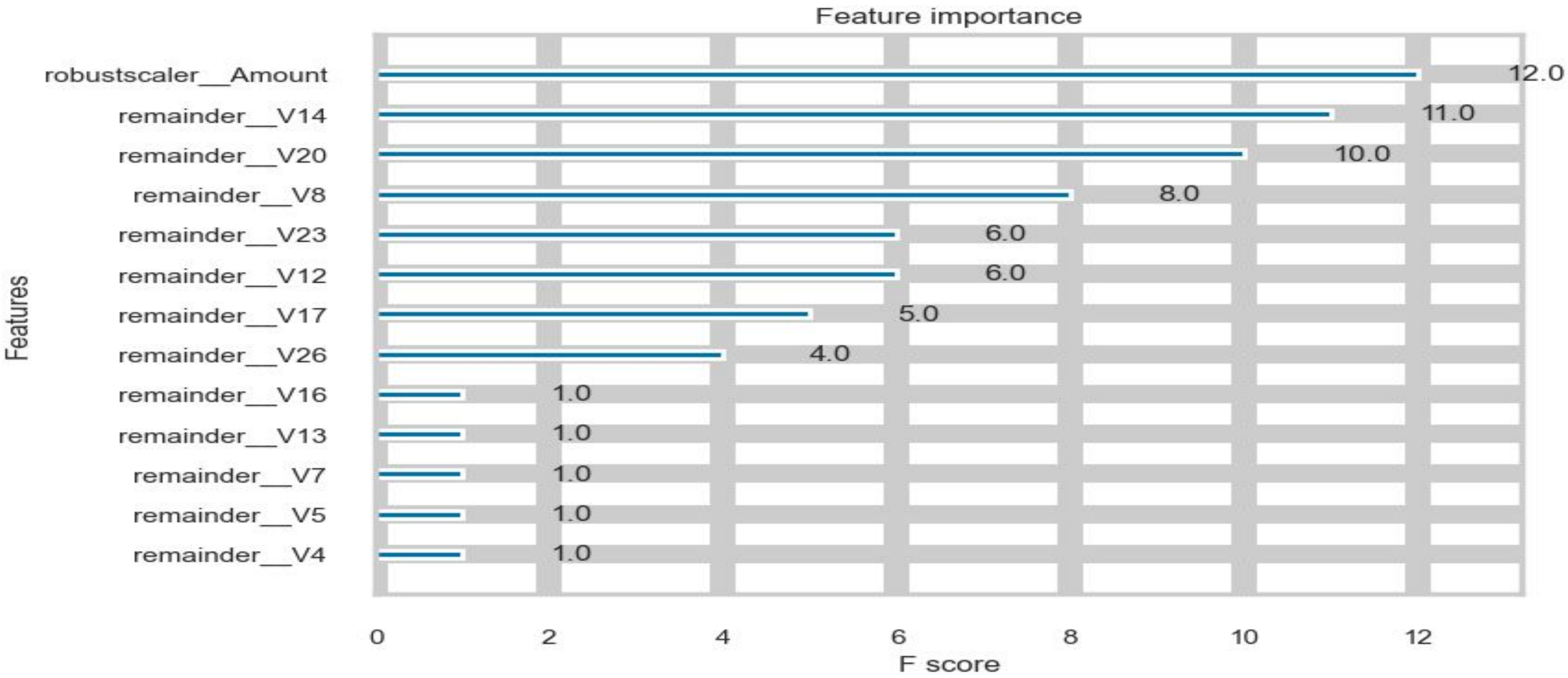


Compare Models

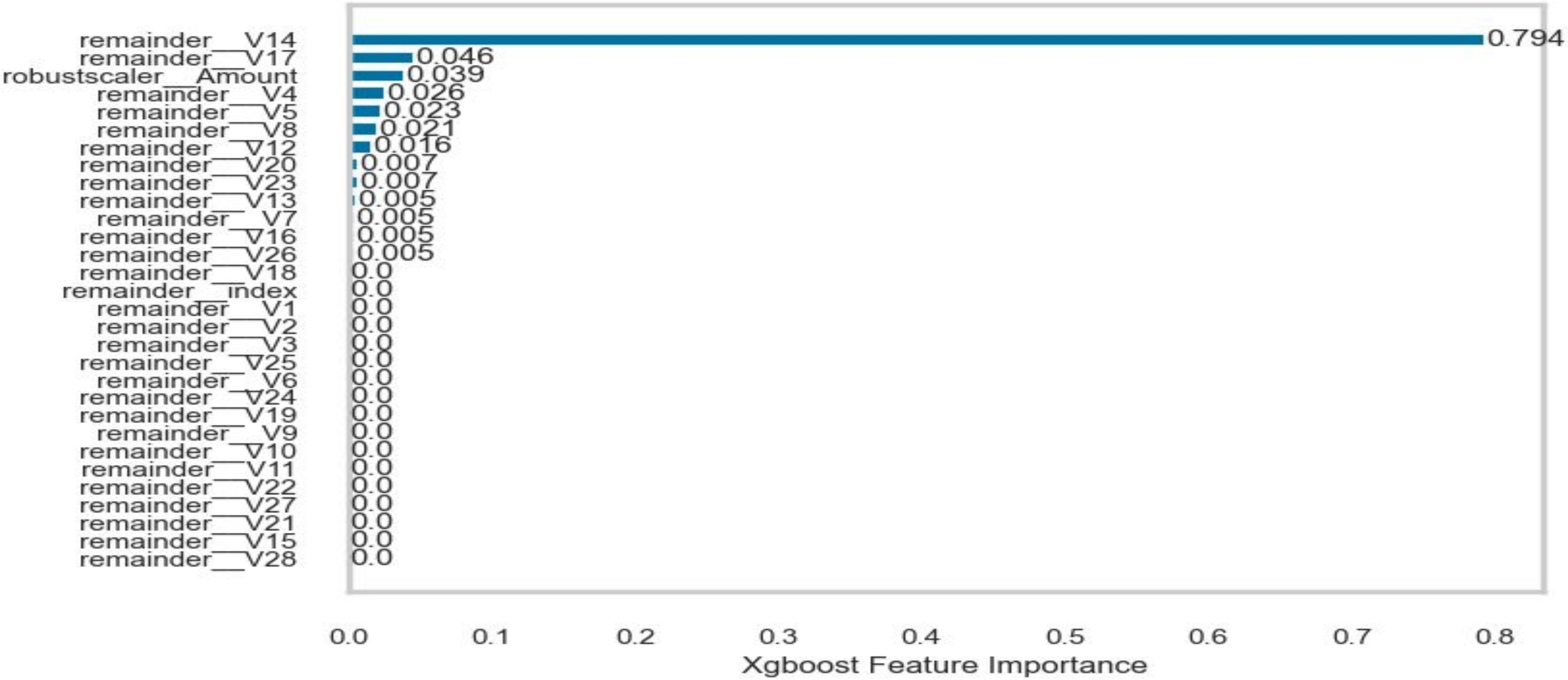
Precision Score Comparison



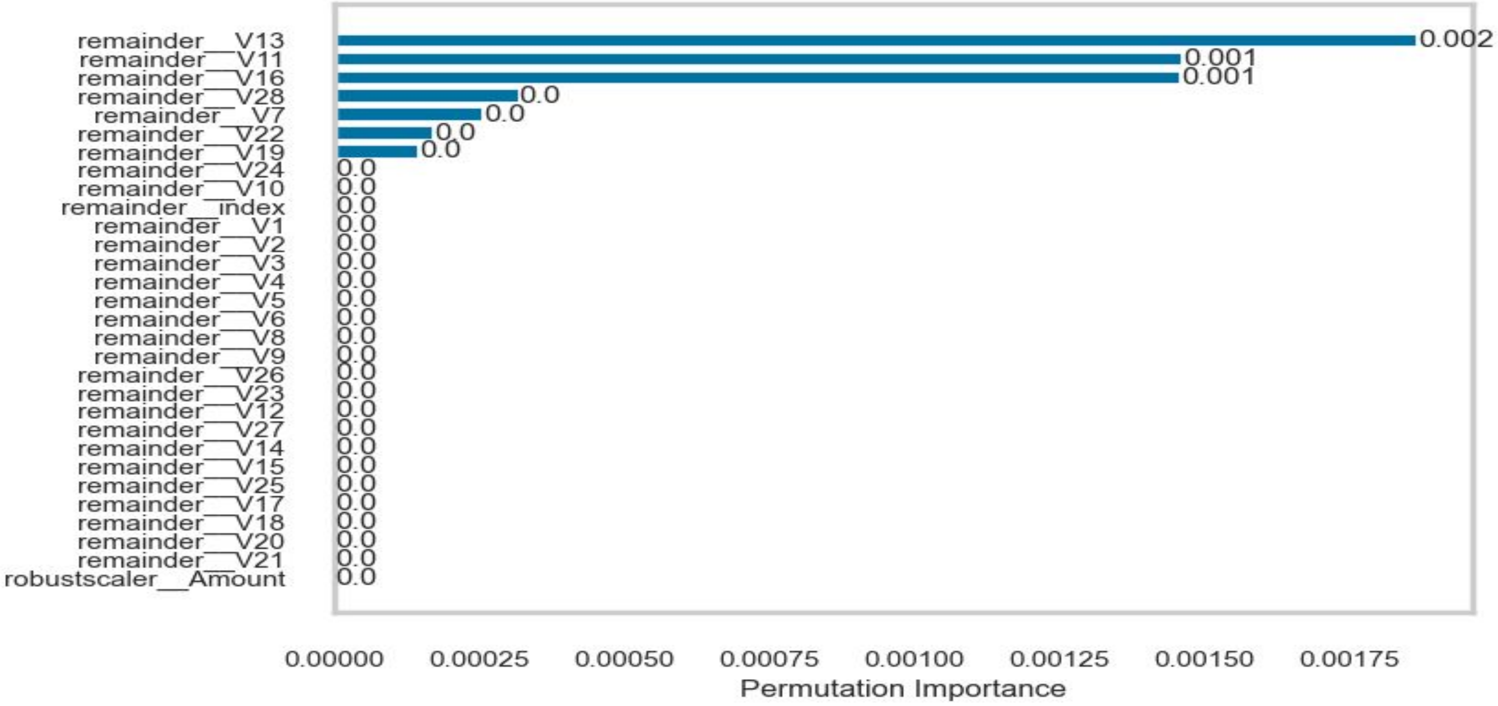
XGBoost Feature Importance Computed in 3 Ways with Python



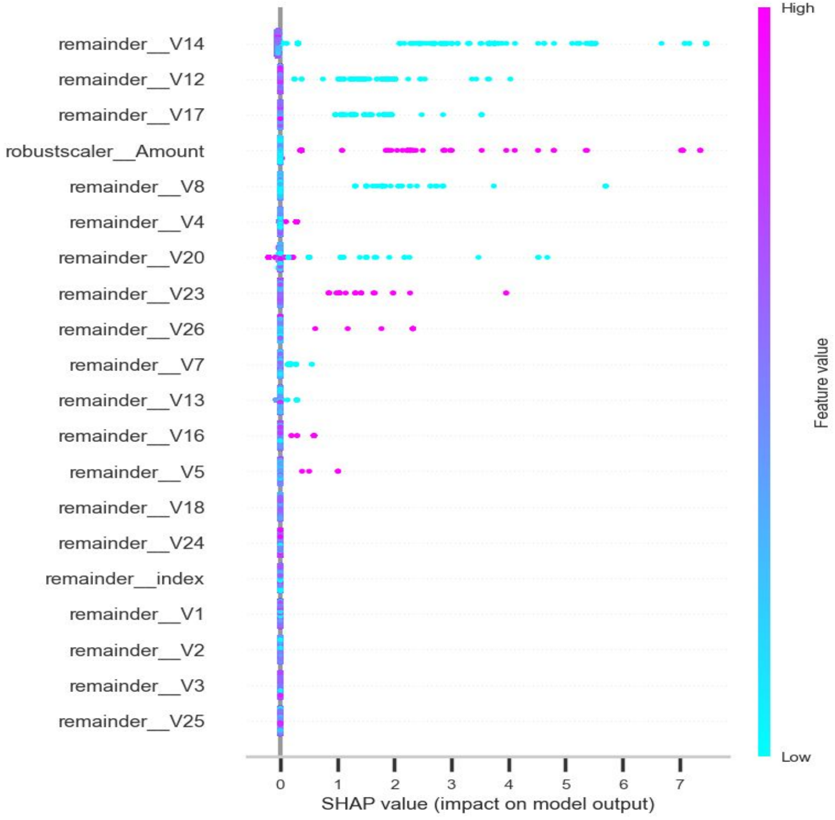
XGBoost Feature Importance Computed in 3 Ways with Python

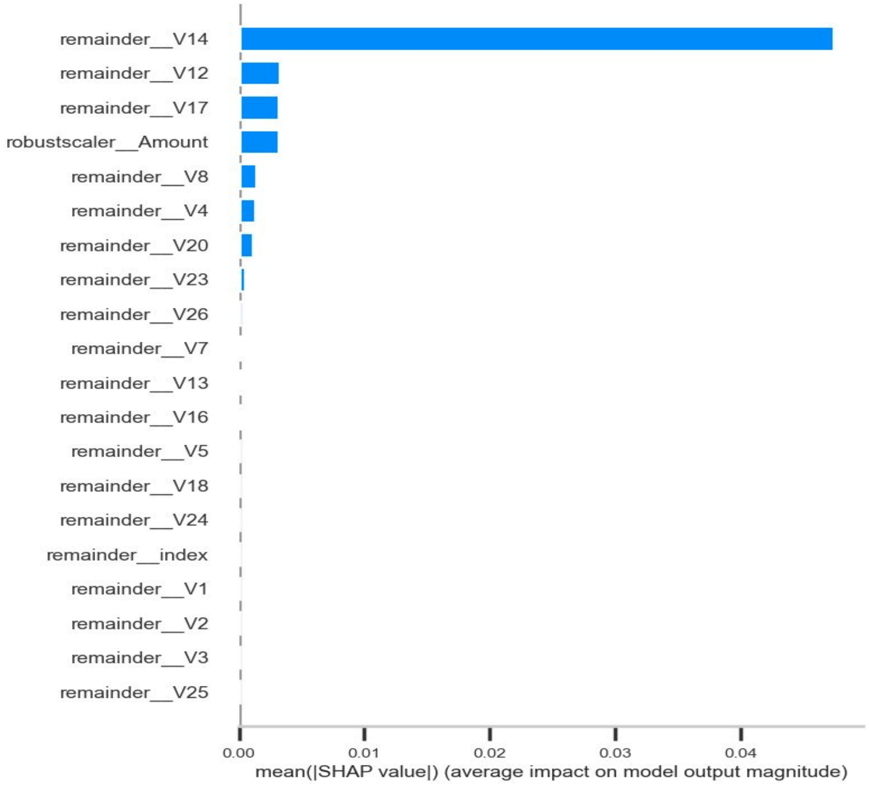


Permutation Based Feature Importance (with scikit-learn)



Importance computed with SHAP values

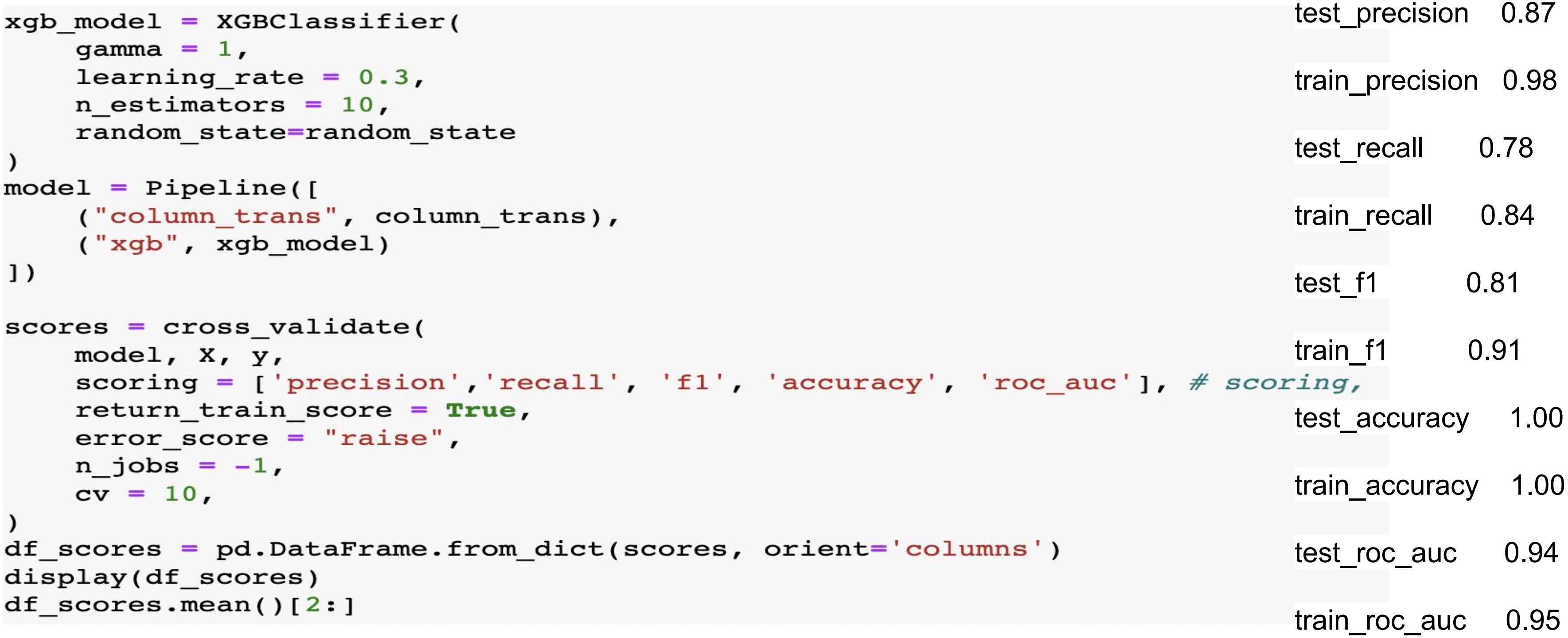




Final Model

Features that we build a model with them;

"Amount", "V14", "V20", "V8", "V23", "V12", "V17","V4","V7","V13","V5"



```
xgb_model = XGBClassifier(
    gamma = 1,
    learning_rate = 0.3,
    n_estimators = 10,
    random_state=random_state
)
model = Pipeline([
    ("column_trans", column_trans),
    ("xgb", xgb_model)
])

scores = cross_validate(
    model, X, y,
    scoring = ['precision','recall', 'f1', 'accuracy', 'roc_auc'], # scoring,
    return_train_score = True,
    error_score = "raise",
    n_jobs = -1,
    cv = 10,
)
df_scores = pd.DataFrame.from_dict(scores, orient='columns')
display(df_scores)
df_scores.mean()[2:]
```

test_precision	0.87
train_precision	0.98
test_recall	0.78
train_recall	0.84
test_f1	0.81
train_f1	0.91
test_accuracy	1.00
train_accuracy	1.00
test_roc_auc	0.94
train_roc_auc	0.95

Streamlit Link:

https://sue-yavuz-streamlit-fraud-detection-project03-streamlit-zqjjqa.streamlit.app/

THANKS FOR LISTENING