ComputerEngineering.Msc

2024.Budapest

ML Techniques for Detecting { Credit Card Fraud In Imbalanced Datasets }

< Alejandra Cuadros Rivas >

Table of Contents {

}

01. Introduction; {

Challenges of fraud detection

- Fraudulent transactions make up a very small percentage of all transactions
- High class imbalance affects model performance

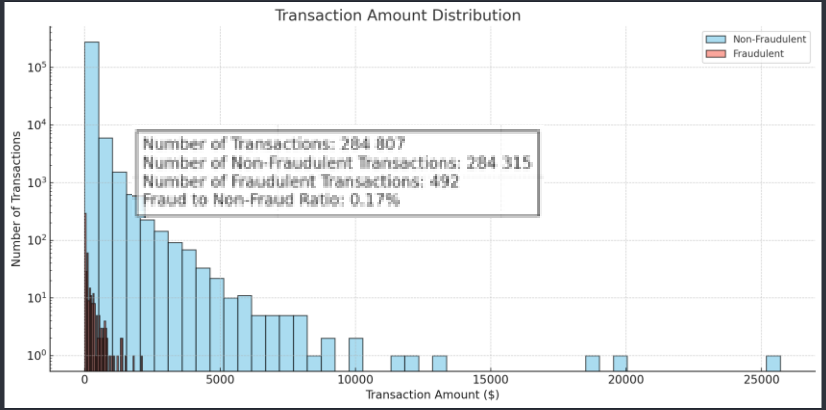
Importance of reducing false negatives

- Protect financial assets (reduce financial losses)
- Maintain system trust

}

02. Dataset Overview; {

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
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	0.128539	-0.189115	0.133558	-0.021053	149.62	0
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	0.167170	0.125895	-0.008983	0.014724	2.69	0
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	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0
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	0.647376	-0.221929	0.062723	0.061458	123.50	0
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	-0.206010	0.502292	0.219422	0.215153	69.99	0



- Features: Time, Amount, anonymized features (V1–V28).
- Preprocessing:
  - Scaling: Standardized Amount and Time.
  - Balancing

}

03. Methods Overview; {

Data balancing

< Under-sampling, SMOTE, combination>

Models

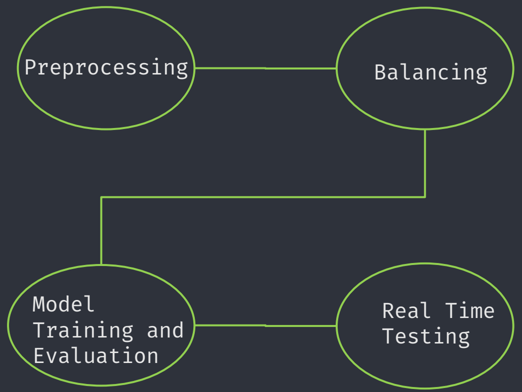
< Logistic Regression,
Random Forest, Gradient Boosting>

Advanced methods

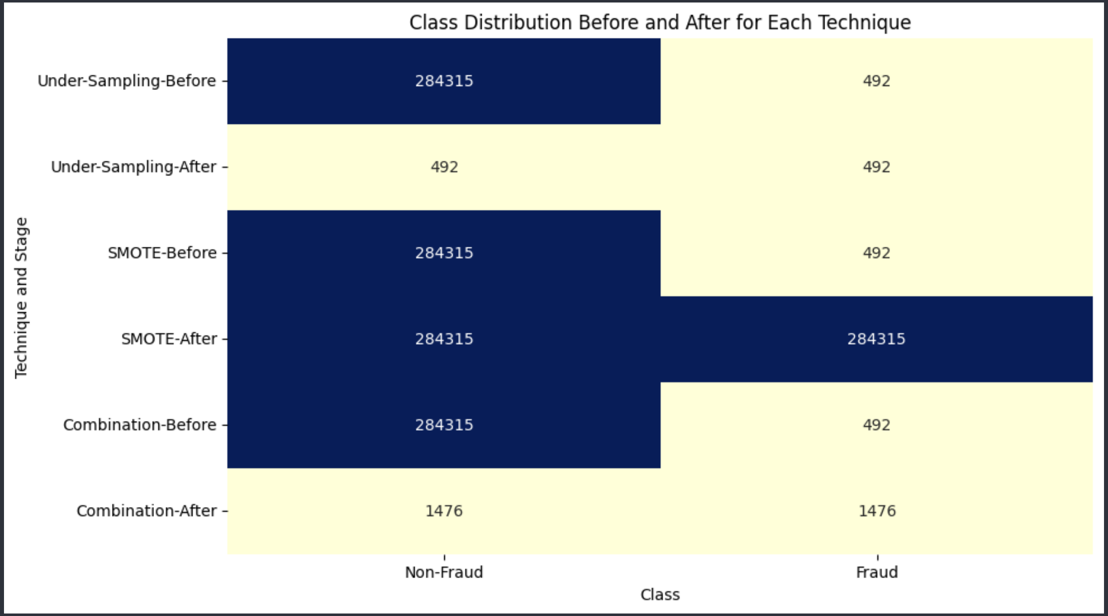
< SHAP, LIME, real-time simulation.>

}

Pipeline



04. Data Balancing Techniques; {



< Under-sampling: Low precision and F1-score.

SMOTE: Best recall.

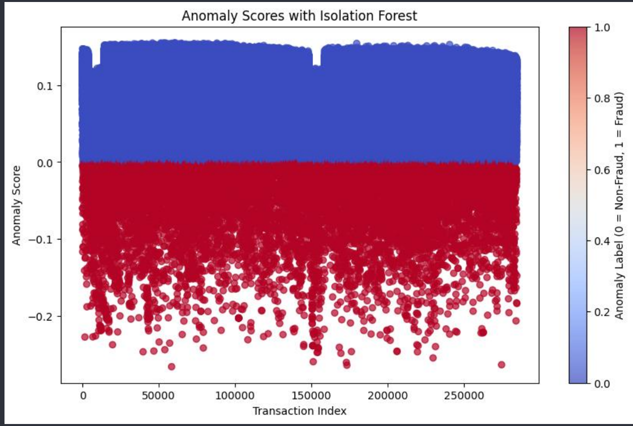
Combination: Balanced performance>

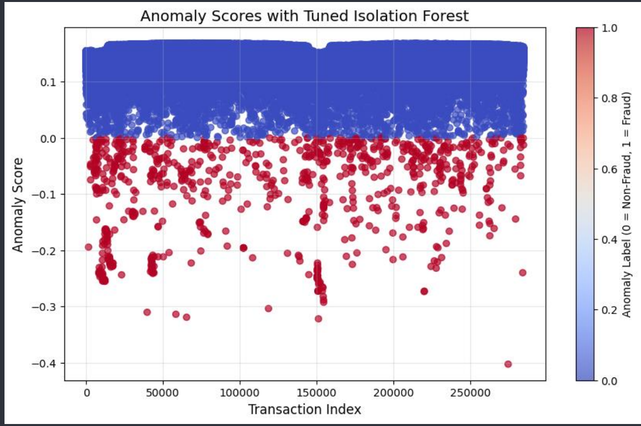
}

05. Unsupervised Learning; {

Isolation Forest

< Anomaly detection, useful for exploratory analysis.>





- Overlap between fraud and non-fraud transactions
- Challenge in separating the two classes and potentially contributing to false positives

- Reasonably separating transactions.
- Model's challenges in distinguishing.

```
n_estimators=200,  # Increase number of trees for better partitioning
max_samples=0.8,  # Use 80% of the data to fit each tree
contamination=0.002,  # Approximate proportion of fraud transactions
```

}

05. Unsupervised Learning; {

Isolation Forest

Comparison of Untuned vs Tuned Isolation Forest Metrics

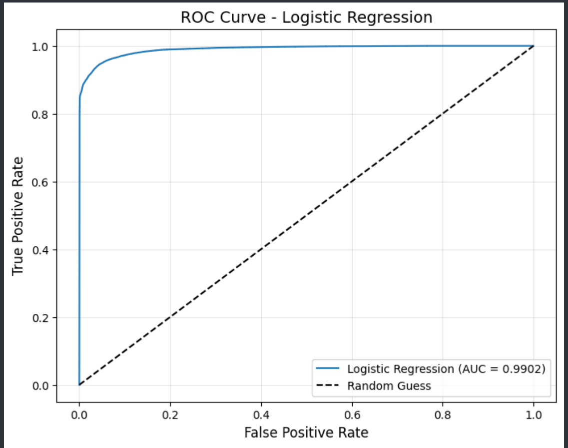
Metric	Untuned Model	Tuned Model
True Negatives	273961.0	283746.0
False Positives	10354.0	569.0
False Negatives	80.0	215.0
True Positives	412.0	277.0
Precision (Fraud)	0.04	0.33
Recall (Fraud)	0.84	0.56
F1-Score (Fraud)	0.07	0.41
ROC-AUC	0.047	0.046

< The tuned model strikes a better balance, improving overall fraud detection reliability.>

}

06. Supervised Learning; {

6.1 Logistic Regression



< Good performance on SMOTE balanced dataset>

< High ROC-AUC (Area under receiver operating characteristic curve) effective separation of classes>

<4541 fraud transactions missed>

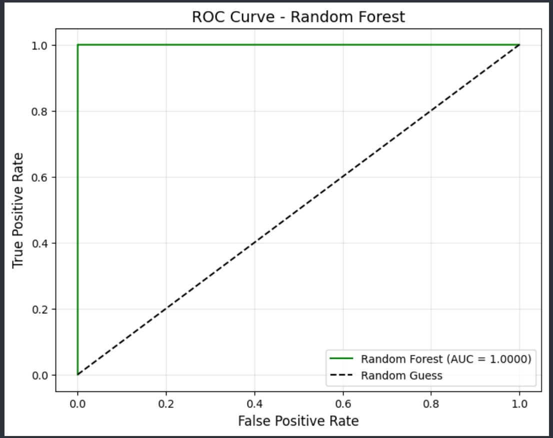
}

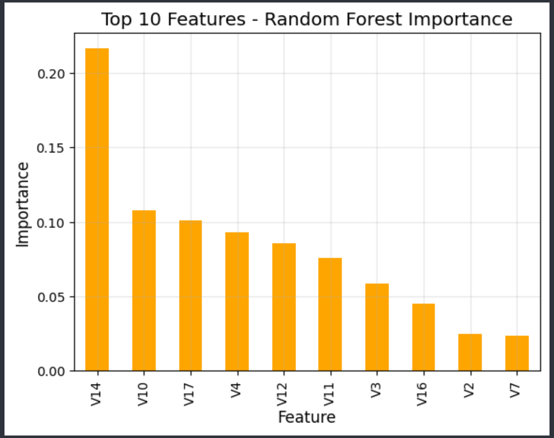
```
Confusion Matrix:
[[55360  1503]
 [ 4541 52322]]
```

TN – **FP**
FN – **TP**

06. Supervised Learning; {

6.2. Random Forest





< Outstanding performance on SMOTE balanced dataset>

< Overfitting or Perfect accuracy>

< No Overfitting Detected, training and test identical metrics>

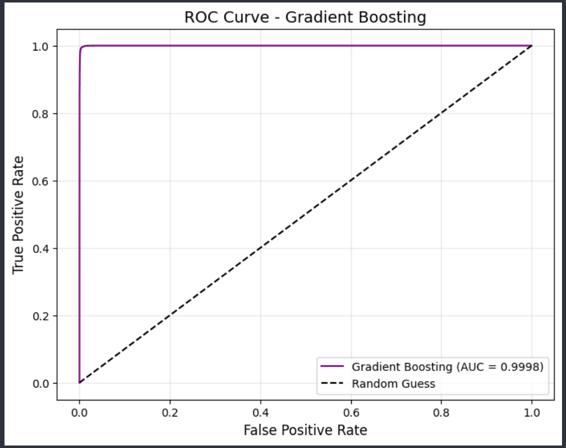
```
Confusion Matrix:
[[56847    16]
 [    0 56863]]
```

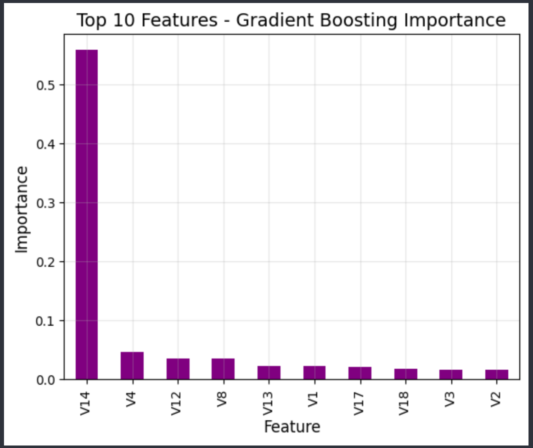
TN – FP
FN – TP

}

06. Supervised Learning; {

6.3 Gradient Boosting





< Excellent performance on SMOTE balanced dataset>

< More false positives and false negatives >

< ROC curve indicates that the model excels at distinguishing between fraudulent and non-fraudulent transactions>

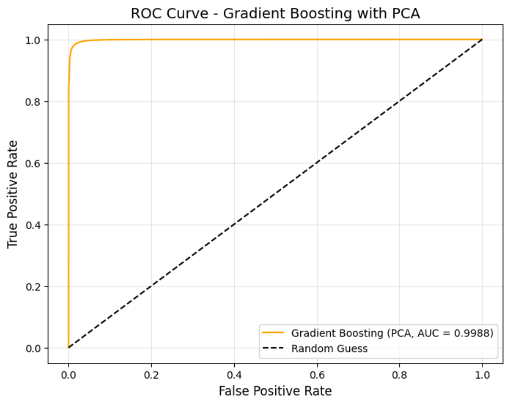
```
Confusion Matrix:
[[56521   342]
 [  256 56607]]
```

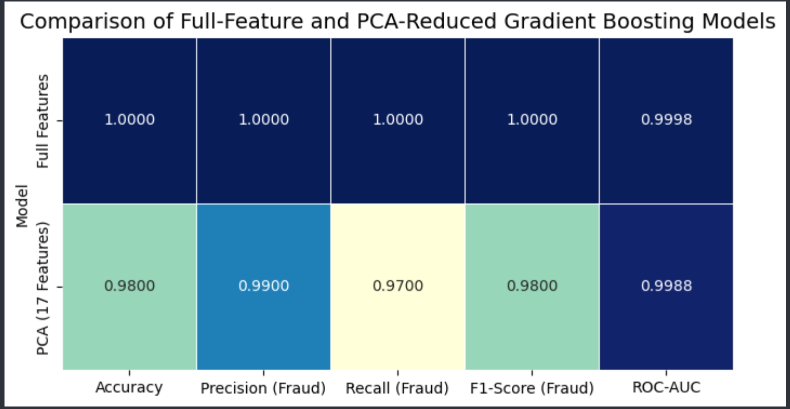
TN – FP
FN – TP

}

06. Supervised Learning; {

6.4 Gradient Boosting and PCA






```
Original Feature Count: 30
Reduced Feature Count: 17
Confusion Matrix (PCA):
[[56357   506]
 [ 1492 55371]]
```

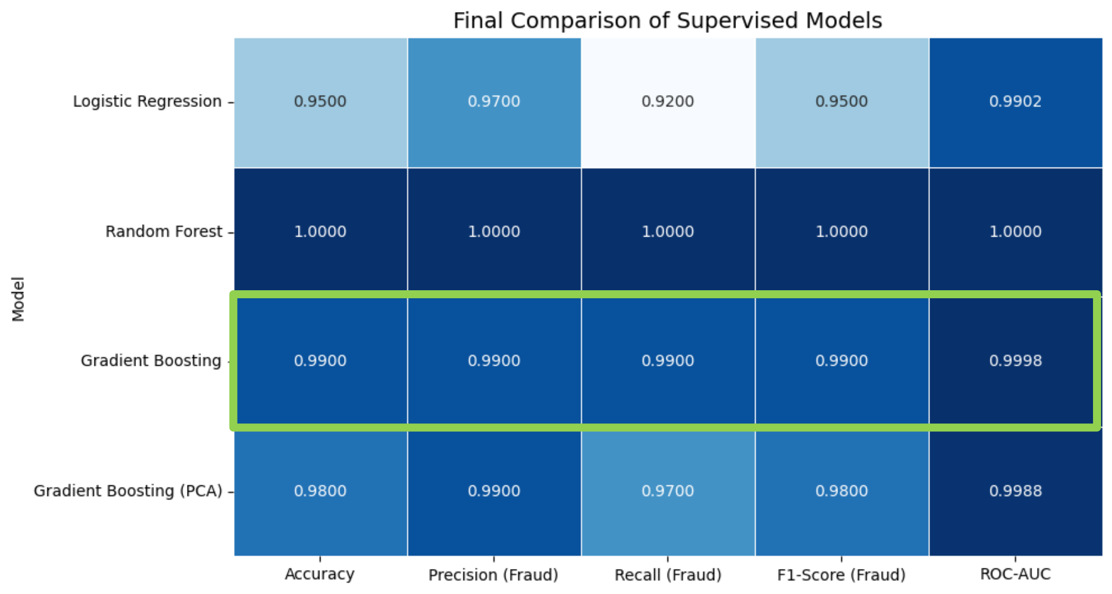
TN – FP
FN – TP

< PCA improve computation time and reduce overfitting risk, but misclassifies more >
< PCA demonstrates dimensionality reduction by achieving comparable results with fewer features>

}

06. Supervised Learning; {

6.5 Final Comparison



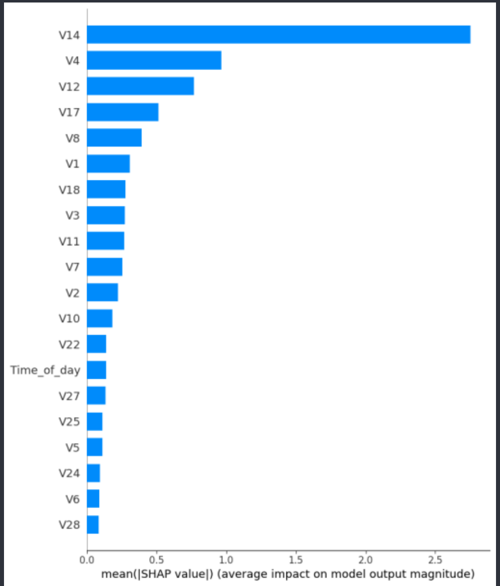
< Gradient Boosting Strikes a balance between accuracy and robustness >

< High metrics and slightly lower susceptibility to overfitting than Random Forest >

}

07. Feature Importance and Explainability; {

7.1 SHAP - (SHapley Additive Explanations)



< Mean absolute SHAP value for each feature >

< The impact of a single feature on a specific prediction >

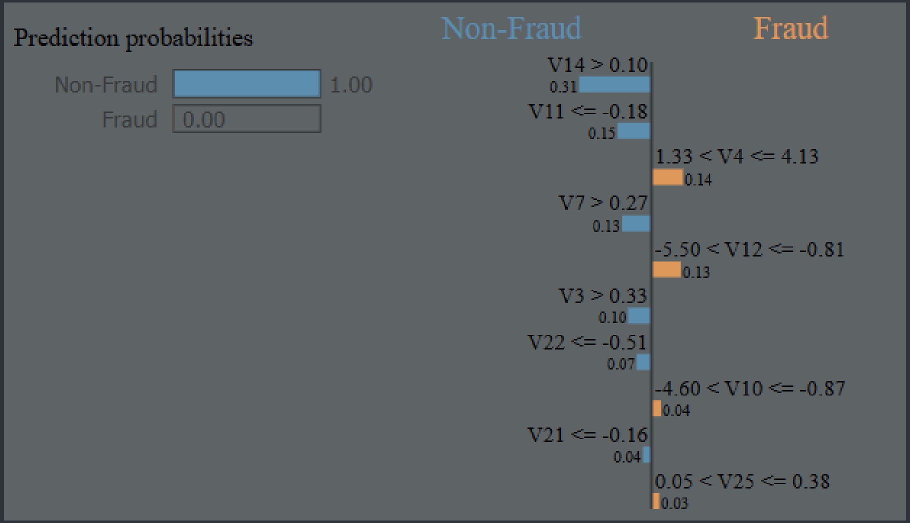
< Feature V14 is the most significant, followed by V4 and V12 >

< These variables play a key role in predicting fraudulent transactions>

}

07. Feature Importance and Explainability; {

7.2 LIME – (Local Interpretable Model-Agnostic Explanations)



Feature	Value
V14	0.52
V11	-1.51
V4	1.66
V7	1.28
V12	-2.95
V3	1.64
V22	-0.98
V10	-0.90
V21	-0.47
V25	0.21

< Random Sample >

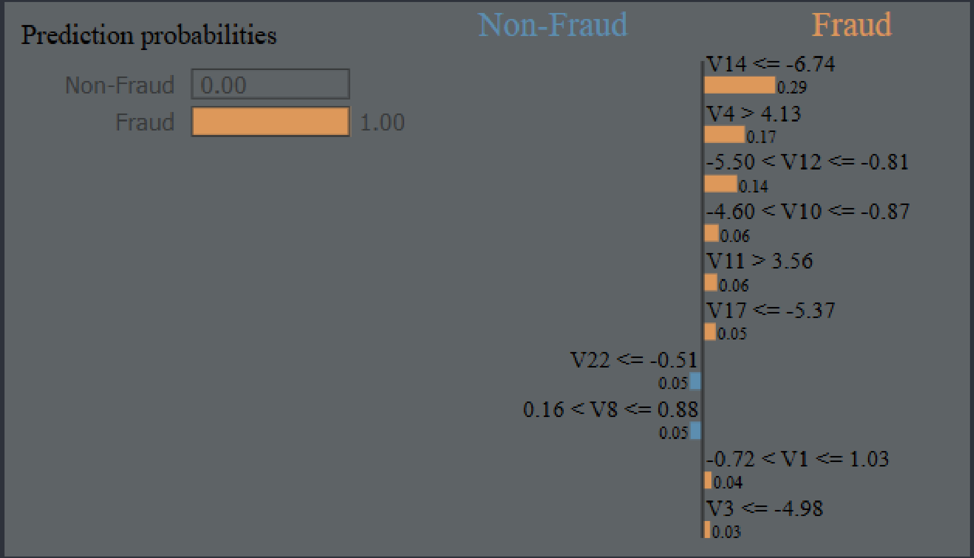
< Weight of Features>

< Feature value contribution>

}

07. Feature Importance and Explainability; {

7.2 LIME - (Local Interpretable Model-Agnostic Explanations)



Feature	Value
V14	-9.39
V4	5.90
V12	-5.33
V10	-4.29
V11	3.75
V17	-5.94
V22	-0.90
V8	0.60
V1	0.69
V3	-5.24

< Random Sample >

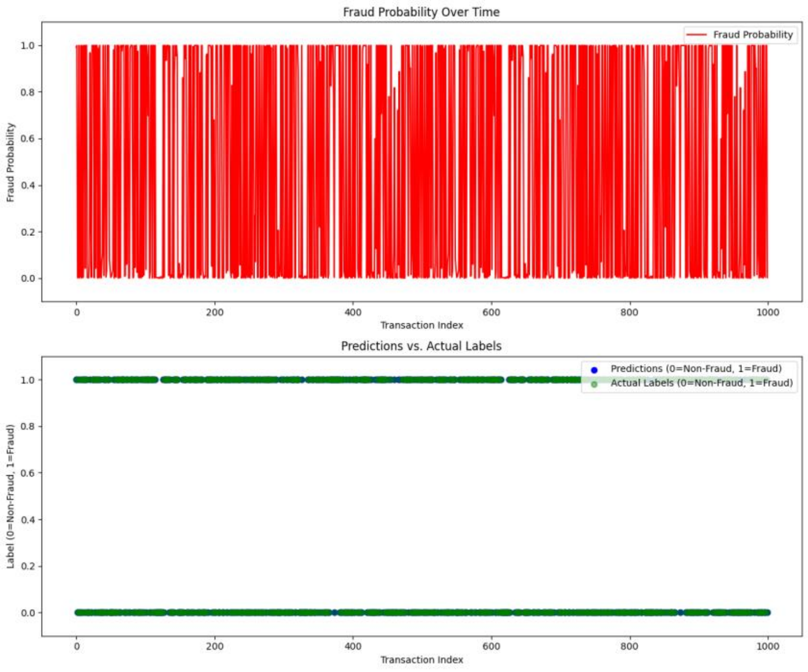
< Weight of Features>

< Feature value contribution>

}

08. Real Time Simulation; {

1000 Transactions




```
Confusion Matrix:
[[520   4]
 [  0 476]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.99      1.00       524
           1       0.99      1.00      1.00       476

    accuracy                           1.00      1000
   macro avg       1.00      1.00      1.00      1000
weighted avg       1.00      1.00      1.00      1000
```

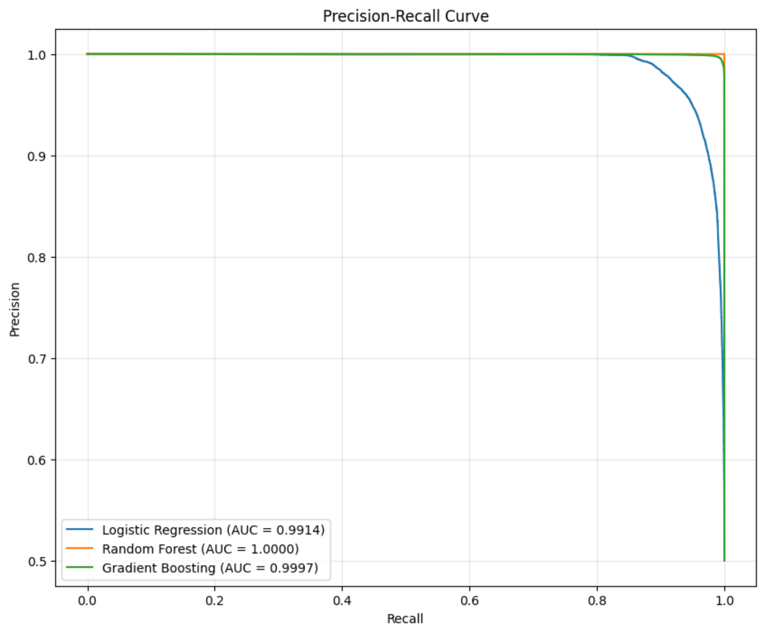
< TN (520): Non-fraudulent transactions correctly classified.
FP (4): Non-fraudulent transactions incorrectly classified as fraud.
TP (476): Fraudulent transactions correctly classified.
FN(0): No fraudulent transactions were missed.>

}

This model is highly reliable and suitable for real-world fraud detection applications.

09. Results and Conclusions; {

9.1 Precision-Recall Curve (AUPRC)



< Highlights the trade-off between precision (accuracy of positive predictions) and recall (proportion of true positives identified) >

< Logistic Regression model – limitations with complex decisions >

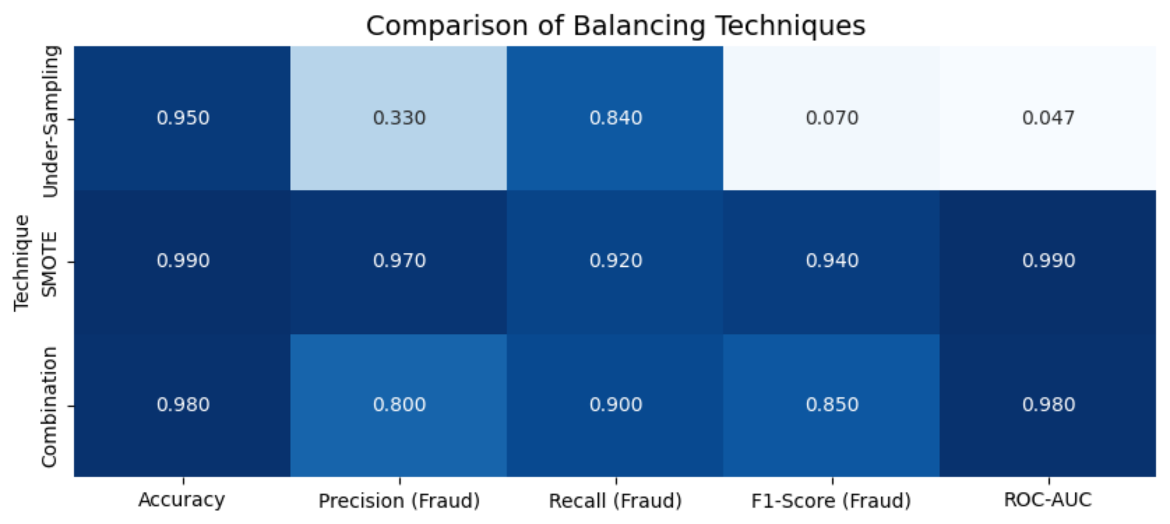
< Random Forest – Perfect results>

< Gradient Boosting – maintains robust performance >

}

09. Results and Conclusions; {

9.2 Balancing Techniques



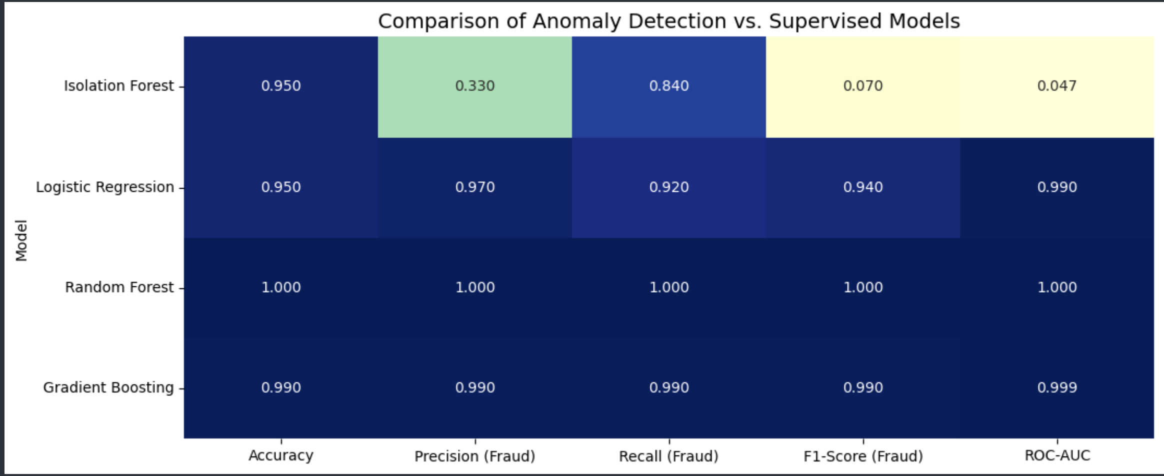
< **SMOTE**
Precision (97%) and recall (92%) demonstrate excellent fraud detection capabilities >

< **SMOTE** provides the best balance for fraud detection, while under-sampling struggles due to information loss from reducing the majority class>

}

09. Results and Conclusions; {

9.3 Unsupervised Models vs. Supervised Models



< Supervised models **outperform** anomaly detection methods >

< Random Forest and Gradient Boosting show superior results, careful evaluation to avoid overfitting>

}

< **SMOTE** for **balancing** and **Gradient Boosting** for **modeling** may provide the best trade-off between performance and reliability>

10. Future work; {

1. Adding cost sensitive learning and testing with larger and more diverse datasets.
    1. Reduces false negatives by penalizing false misclassification.
    2. Complex and Time Consuming.

}

CreditCardFraud.html BME.py

{

Thanks;

}