

Detecting Fraudulent Transactions in Online Payments using Supervised Learning

Valentina Bertei
Francesco Tarchi

Data Mining and Machine Learning Project
A.Y. 2024/2025

Problem Description



Problem Overview

Online transactions are increasingly exposed to fraudulent activities, posing risks to both consumers and businesses.

Credit card fraud detection is critical in preventing unauthorized access and minimizing financial losses.



Relevance of DMML Techniques

The challenge is inherently a **binary classification task** (*fraudulent* vs. *legitimate* transactions). This calls for robust **supervised** machine learning methods, advanced feature engineering, and imbalance handling strategies.



Proposed Approach

- Compare multiple classification algorithms to identify the most effective model for fraud detection.
- Candidate models:
 - *KNN*
 - *Gaussian NaiveBayes*
 - *DecisionTree*
 - *RandomForest*
 - *AdaBoost*
 - *XGBoost*
- A set of evaluation metrics will be used to compare the classifiers and determine the **best-performing** model.

Dataset Description



Dataset Source

- Publicly available on Kaggle: [IEEE-CIS Fraud Detection](#)
- Originally provided by Vesta Corporation, a real-world e-commerce platform



Collection Details

The dataset contains historical online transaction data enriched by behavioral signals and device information.

Dataset Properties:

- **Size:** 590,540 transactions (1.08 GB)
- **Fraudulent samples:** 20,663 (approx. 3.5%)
- **Columns:** 432 total columns including transaction details, card info, identity data, and 339 engineered Vesta features
- **Label:** isFraud (1 = fraudulent, 0 = normal)
- **Input/Output Format:**
 - Input: Multivariate features including TransactionAmt, card1–card6, addr1, D1, C1, M1, V1–V339, etc.
 - Output: Binary class label isFraud ∈ {0, 1}

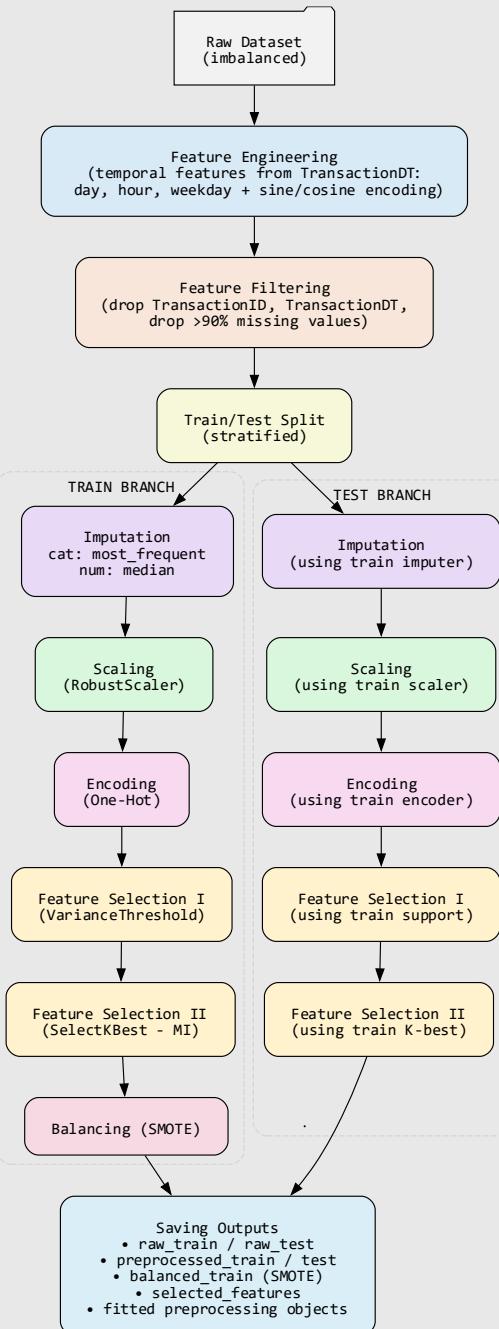


Preprocessing

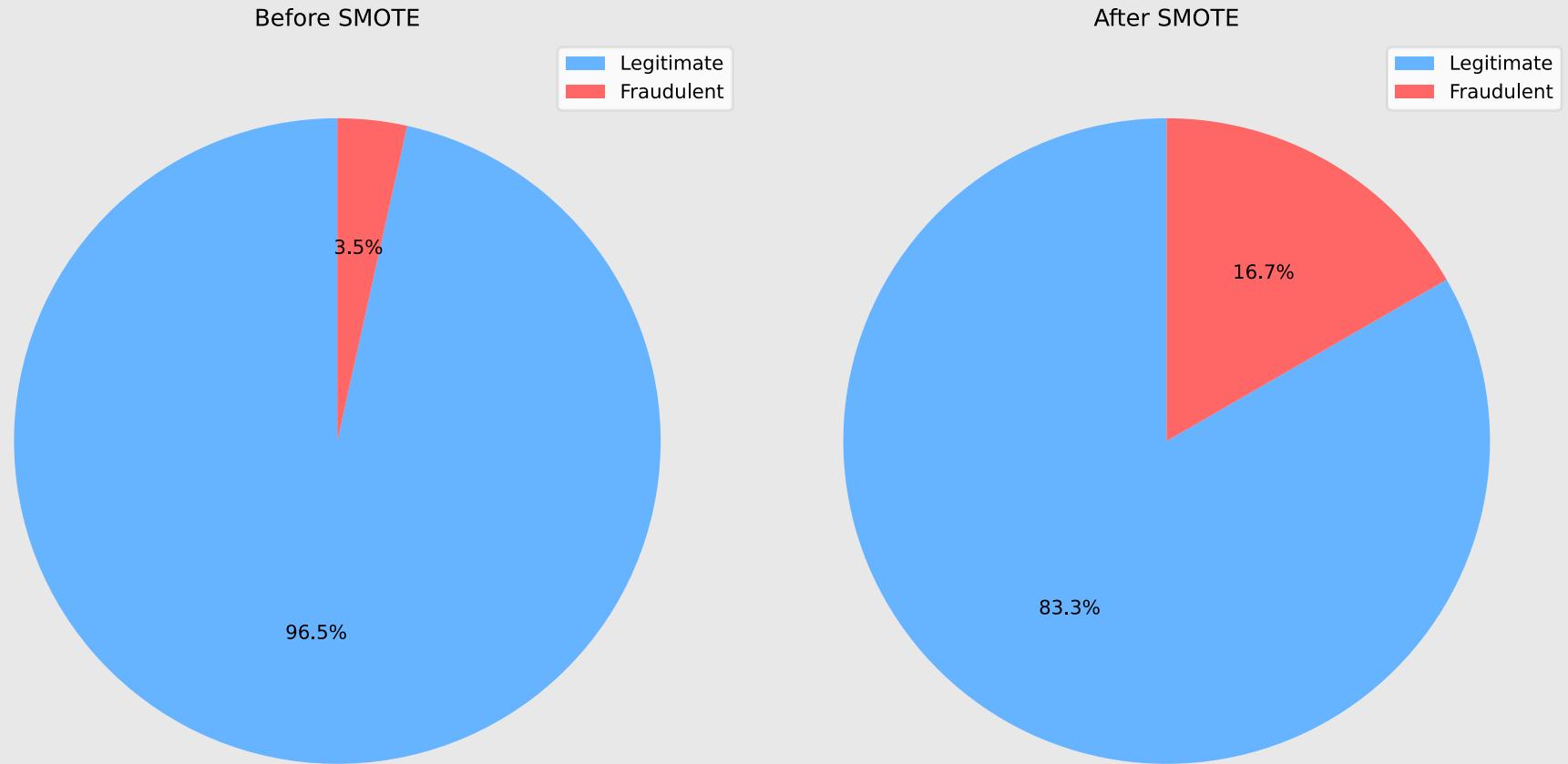


Raw dataset cleaning

- **Feature engineering:** temporal features from `TransactionDT` → day, hour, weekday + cyclic encoding (sine/cosine).
- **Feature filtering:** removing *useless* or *non-informational* features.
- **Train/test split:** *stratified* sampling to preserve label distribution.
- **Imputation:** missing values replaced with *median* or *mode*.
- **Scaling:** `RobustScaler` to reduce outlier impact.
- **Encoding:** `OneHotEncoding` on categorical features.
- **Feature selection:**
 - ✓ Variance Threshold → remove low-variability features;
 - ✓ `SelectKBest` with Mutual Information → retain most informative features.
- Class imbalance handling: **SMOTE** applied only to training set only (*sampling strategy* set to 0.2 → fraud rate from 3.5% to 16.7%).

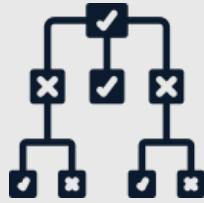


Preprocessing



Comparison of the class distribution in the training set before (left) and after (right) the application of **SMOTE**.

Training and Testing



Classifiers evaluated

- KNN (*K-Nearest Neighbors*)
- NB (*Gaussian NaiveBayes*)
- DT (*DecisionTree*)
- RF (*RandomForest*)
- ADA (*AdaBoost*)
- XGB (*XGBoost*)

Hyperparameter tuning

- **Grid Search** (`GridSearchCV`) with 5-fold CV
- Scoring metric: *f1-score* → balances *precision* & *recall* on imbalanced data
- Goal: find best hyperparameters for each classifier

Training & Testing

- Train on *SMOTE*-rebalanced train set
- 10-fold CV during training → robust validation metrics
- Test on imbalanced test set → evaluate predictions

Evaluation Metrics



Metrics used

- *Confusion Matrix* → base for other metrics (TN, FP, FN, TP)
- *Precision & Recall* → focus
- *f1-score* → balances precision & recall (used in Grid Search)
- *Accuracy & Balanced Accuracy* → quick overview (the 2nd one robust to imbalanced datasets)
- Weighted versions → balancing the metrics between the 2 classes
- *ROC AUC & PR AUC* → compare overall classifier performance

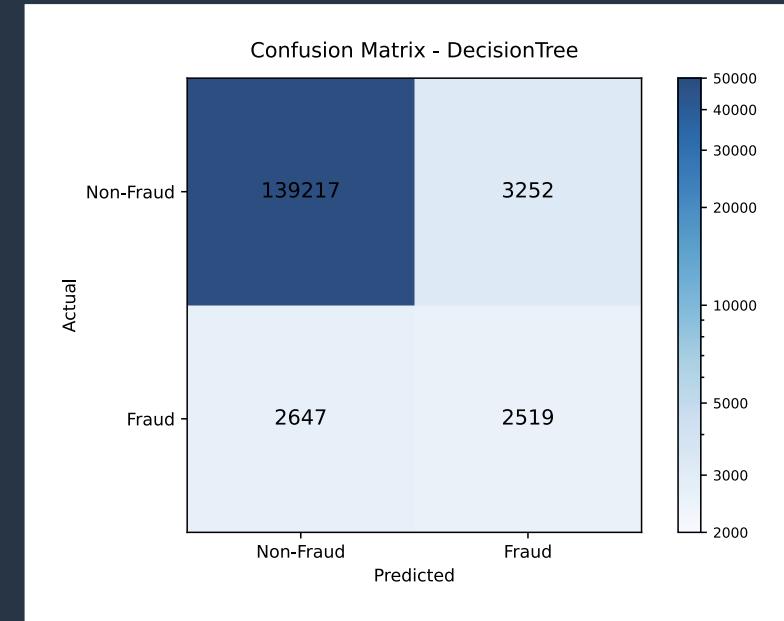
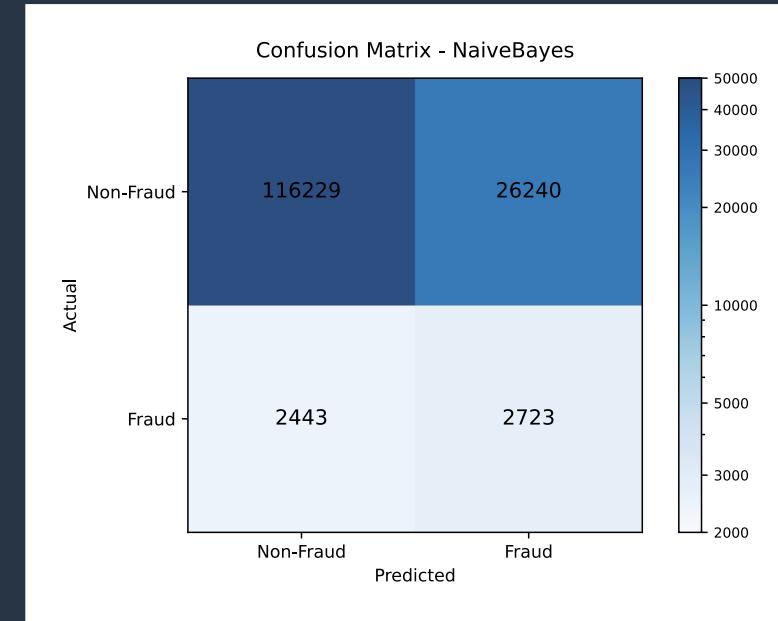
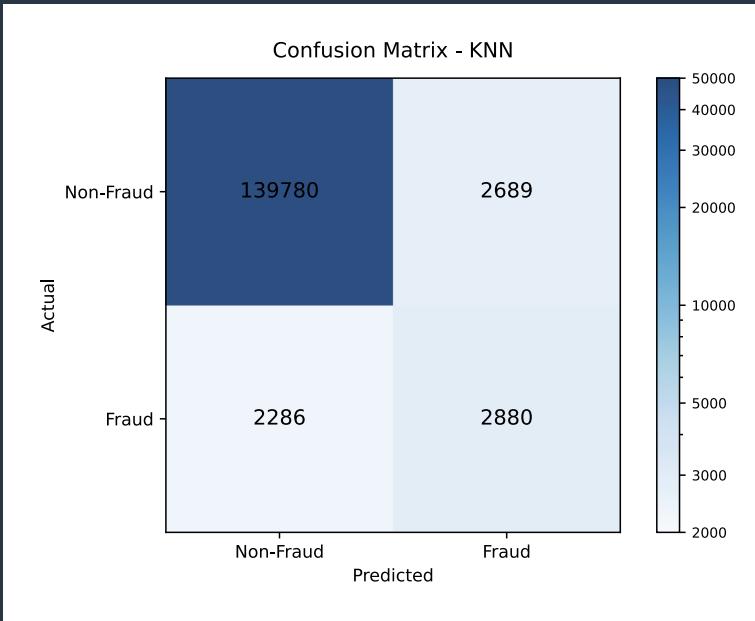


Acceptable Level of Performance (ALP)

- Defined as $\text{TPR} \geq 0.8$
→ correctly identify $\geq 80\%$ of frauds
- **ALP_threshold** → decision threshold where ALP is reached
- **ALP_FPR** → FPR when ALP is reached
- Analysis via ROC curves
→ identify best trade-off between TPR and FPR

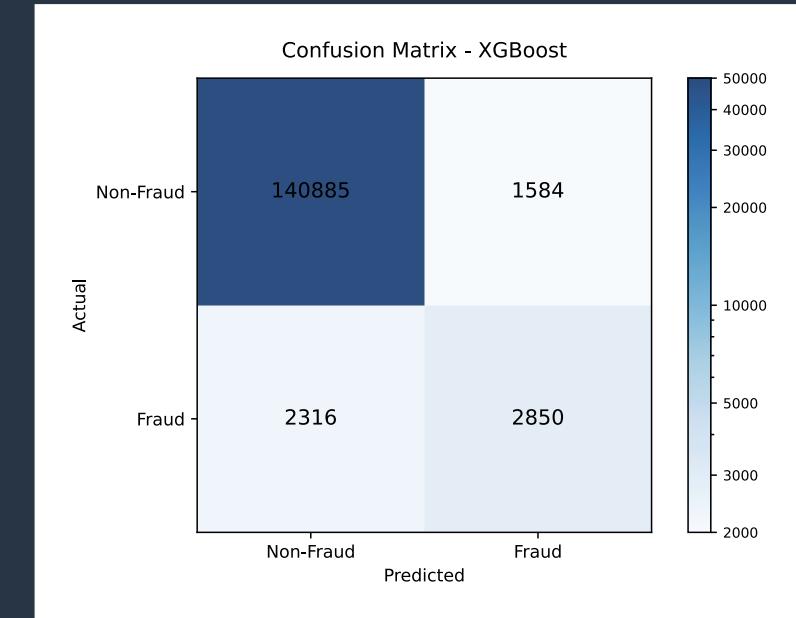
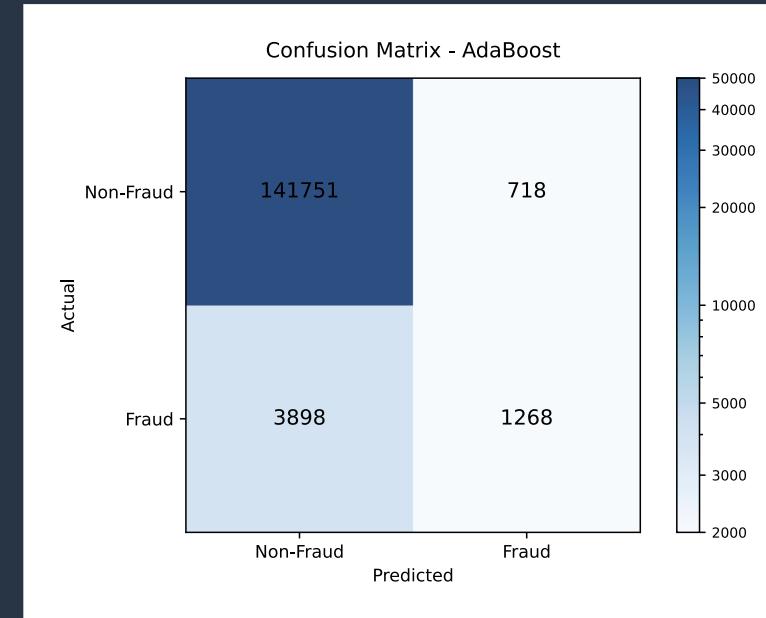
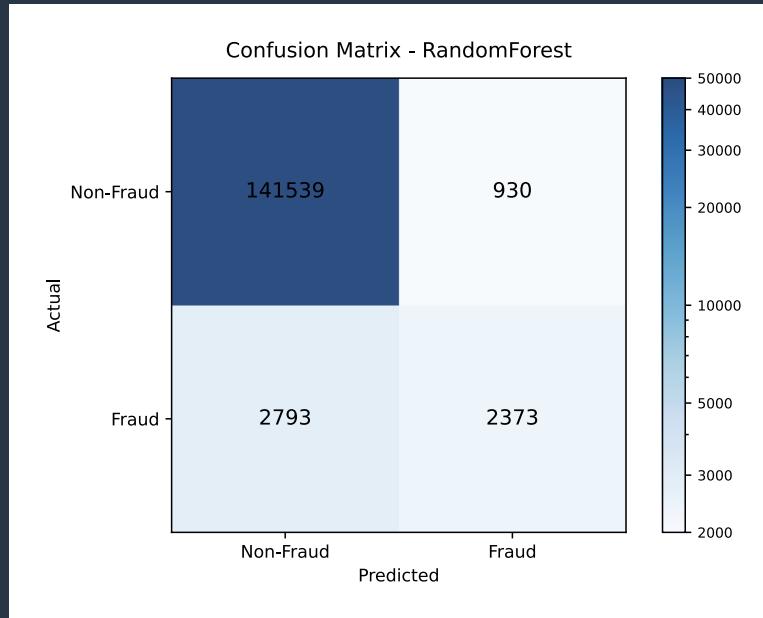
Individual results

Confusion matrices



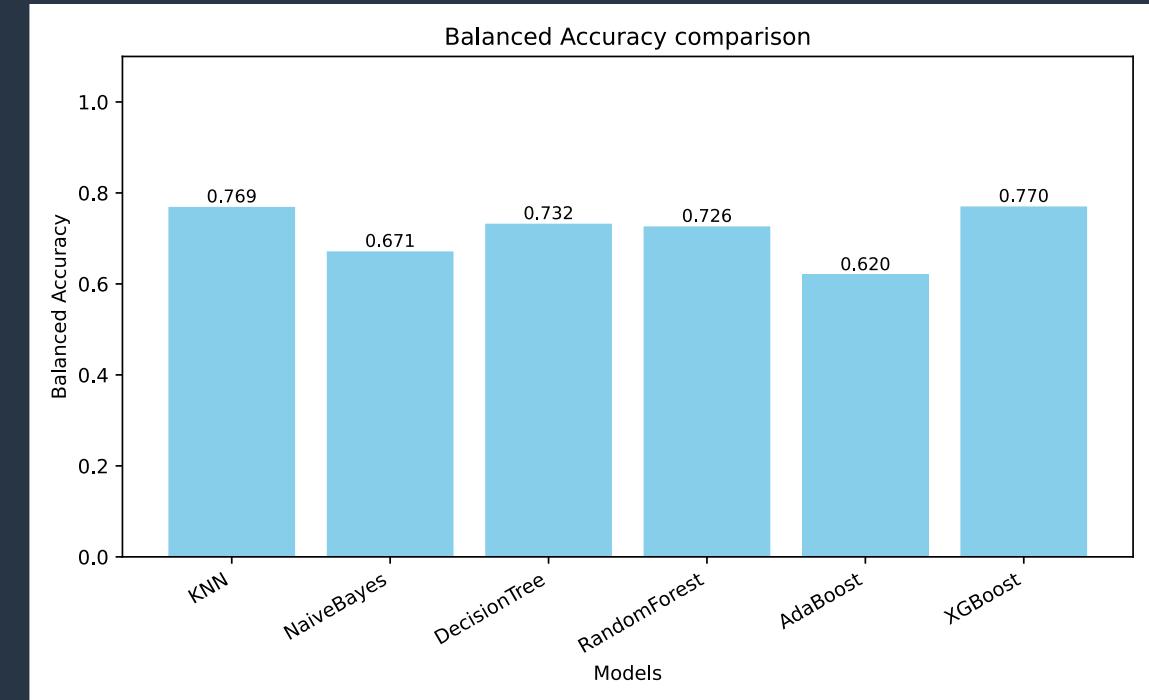
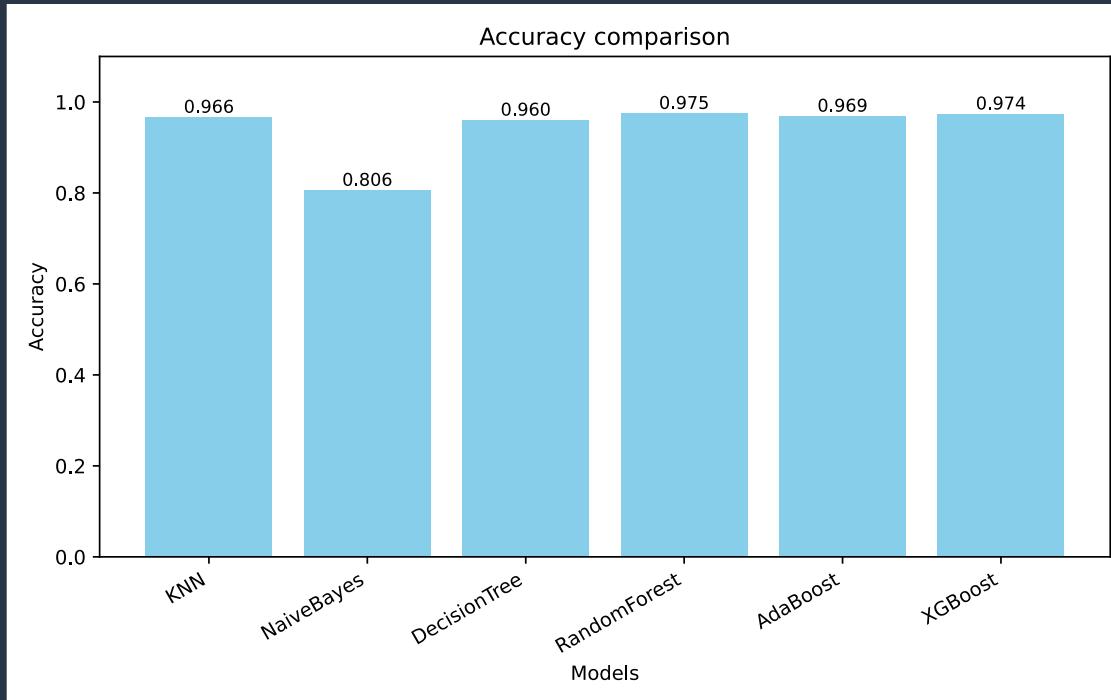
Individual results

Confusion matrices



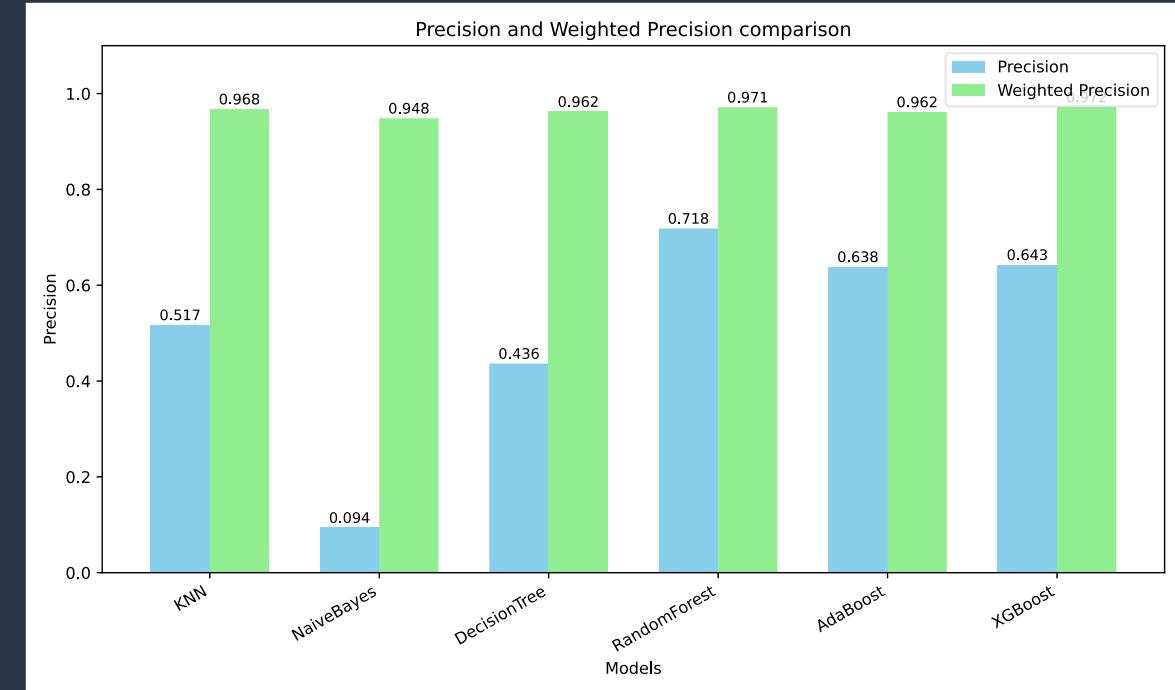
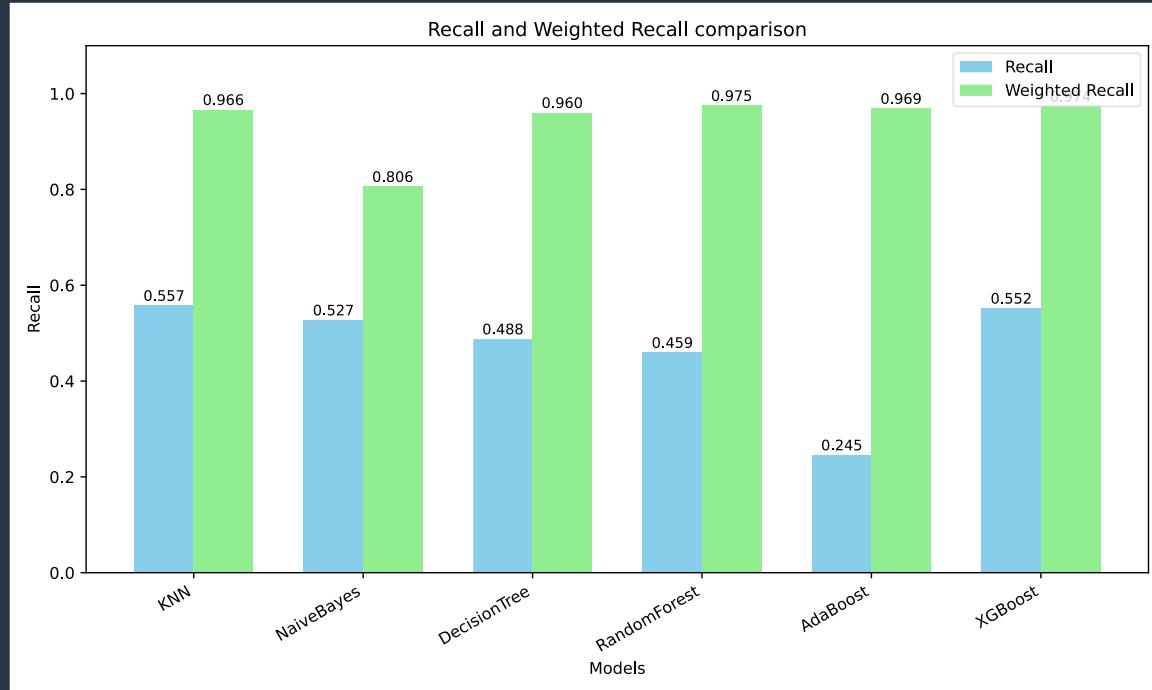
Model comparison

Accuracy & Balanced accuracy



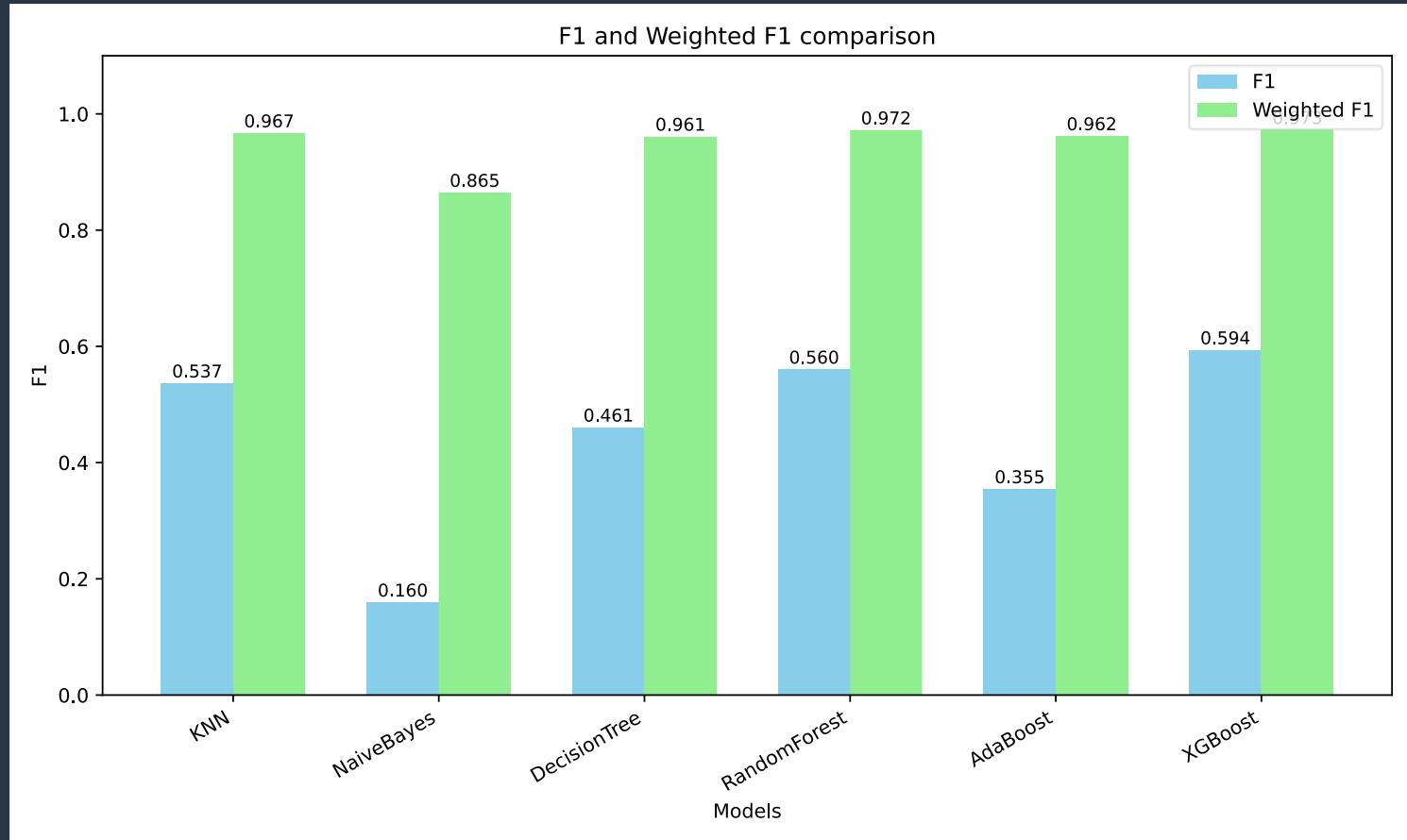
Model comparison

Recall & Precision



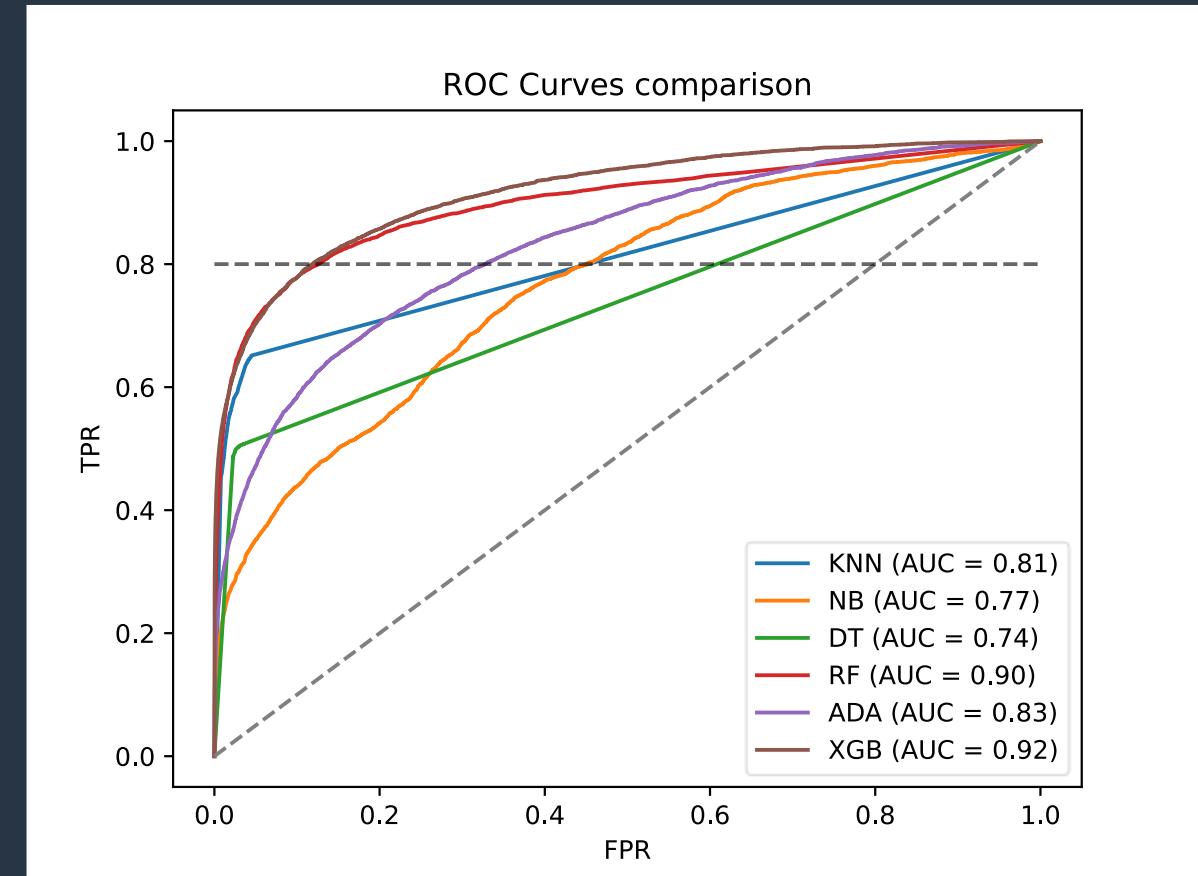
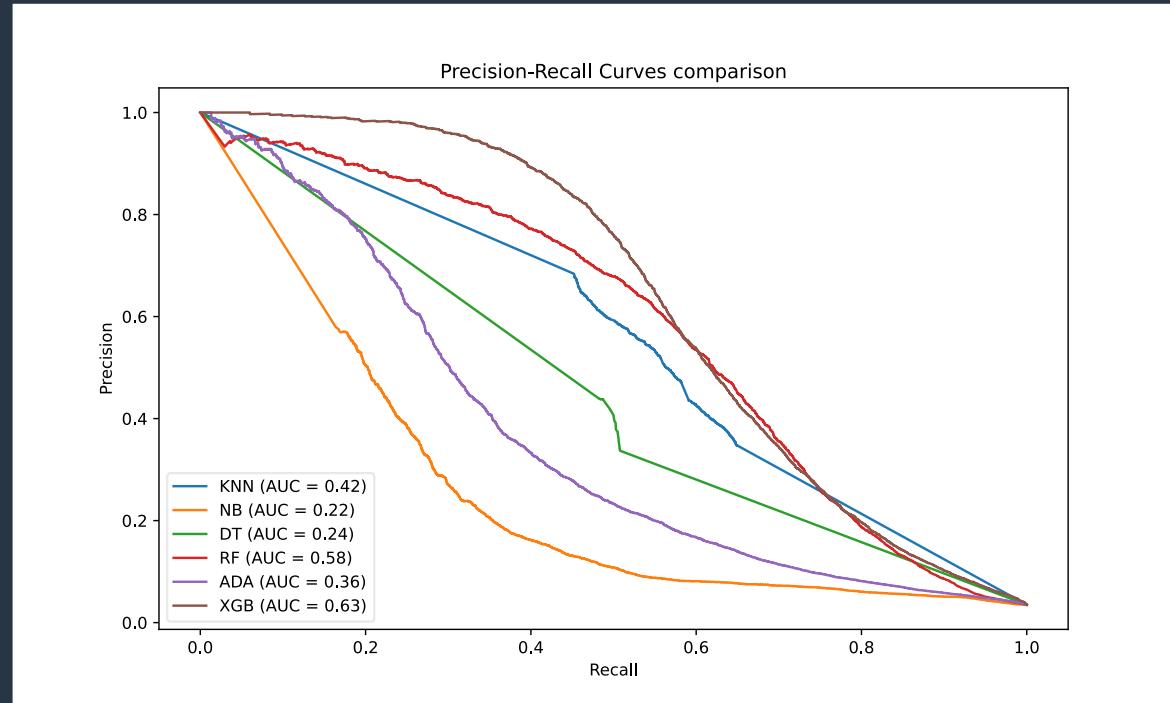
Model comparison

F1-score



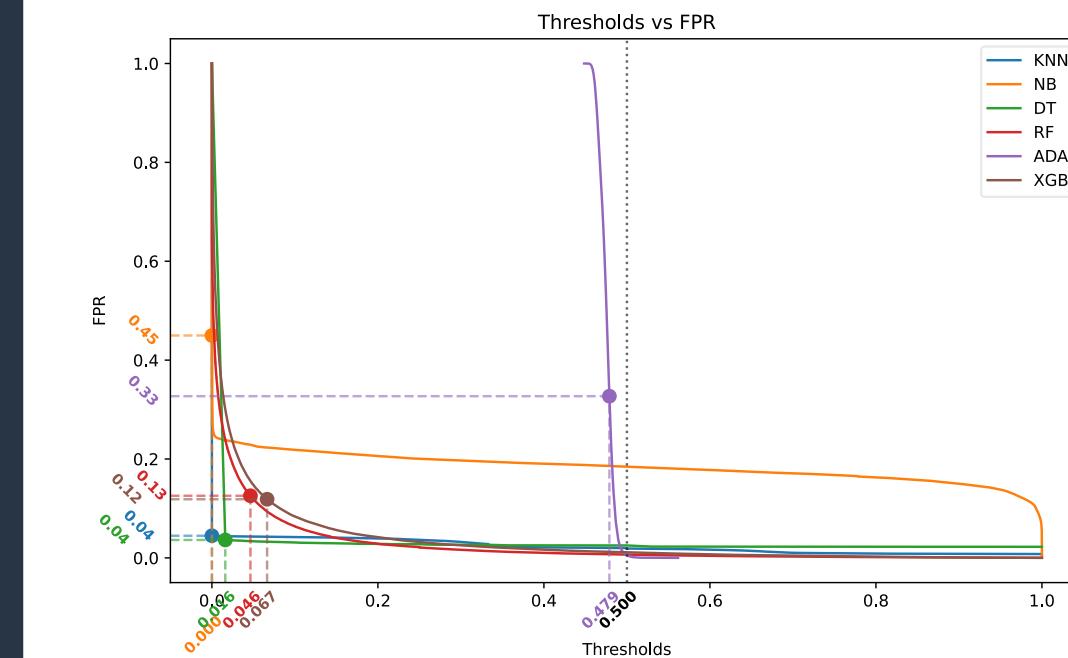
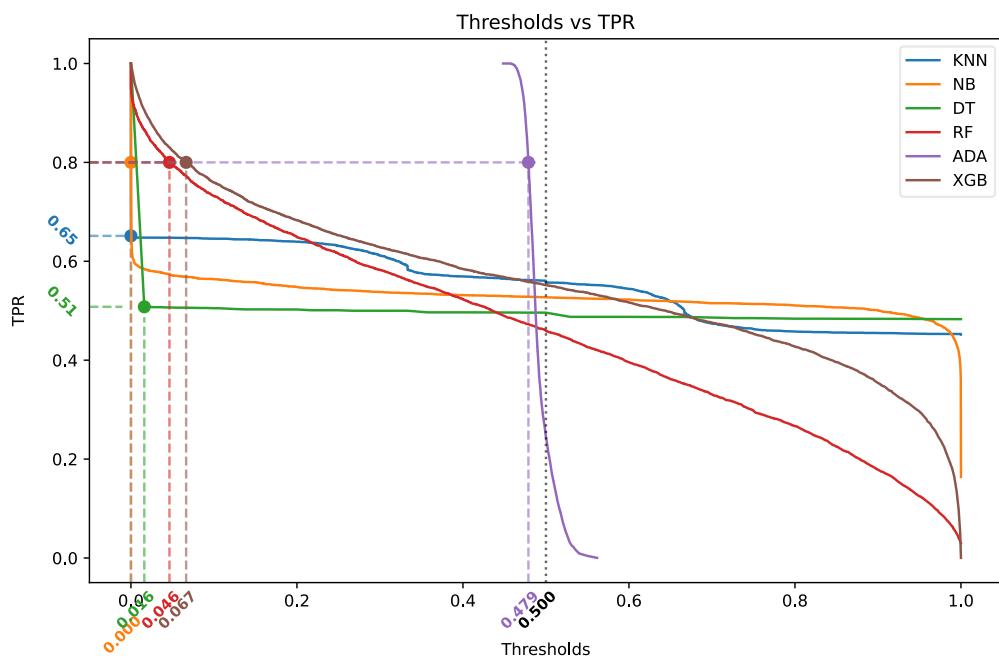
Model comparison

Precision-Recall & ROC



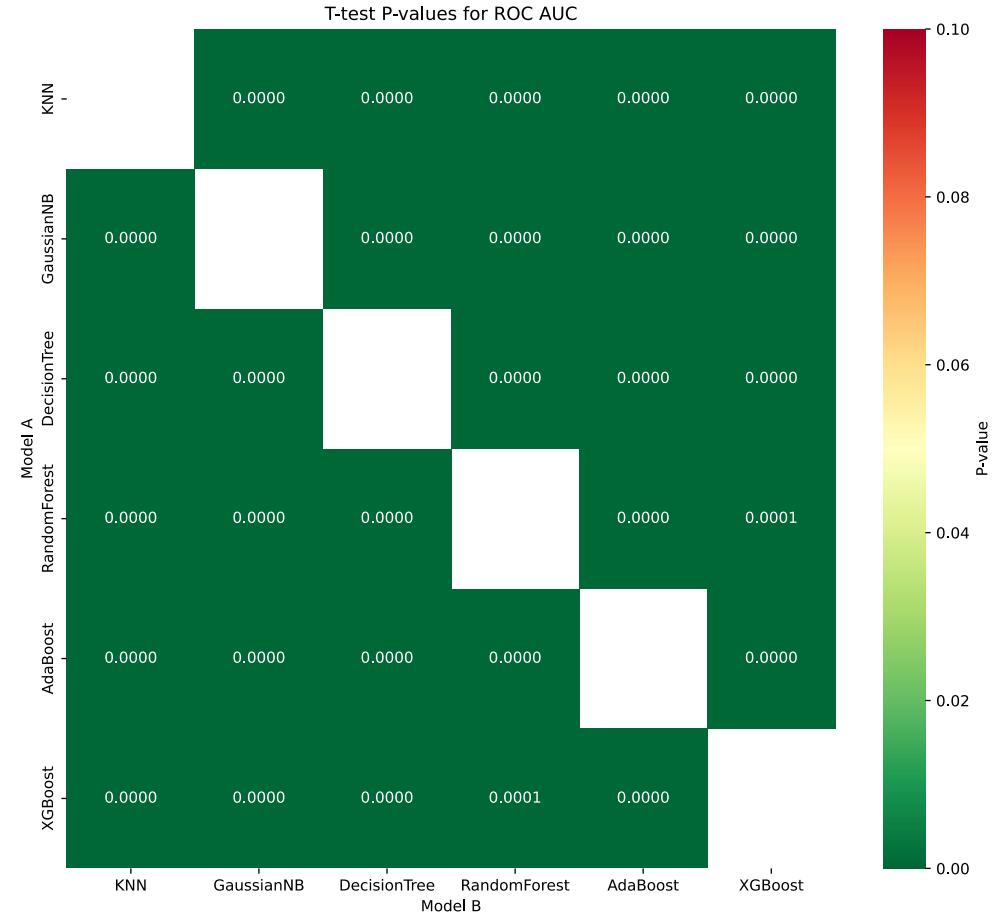
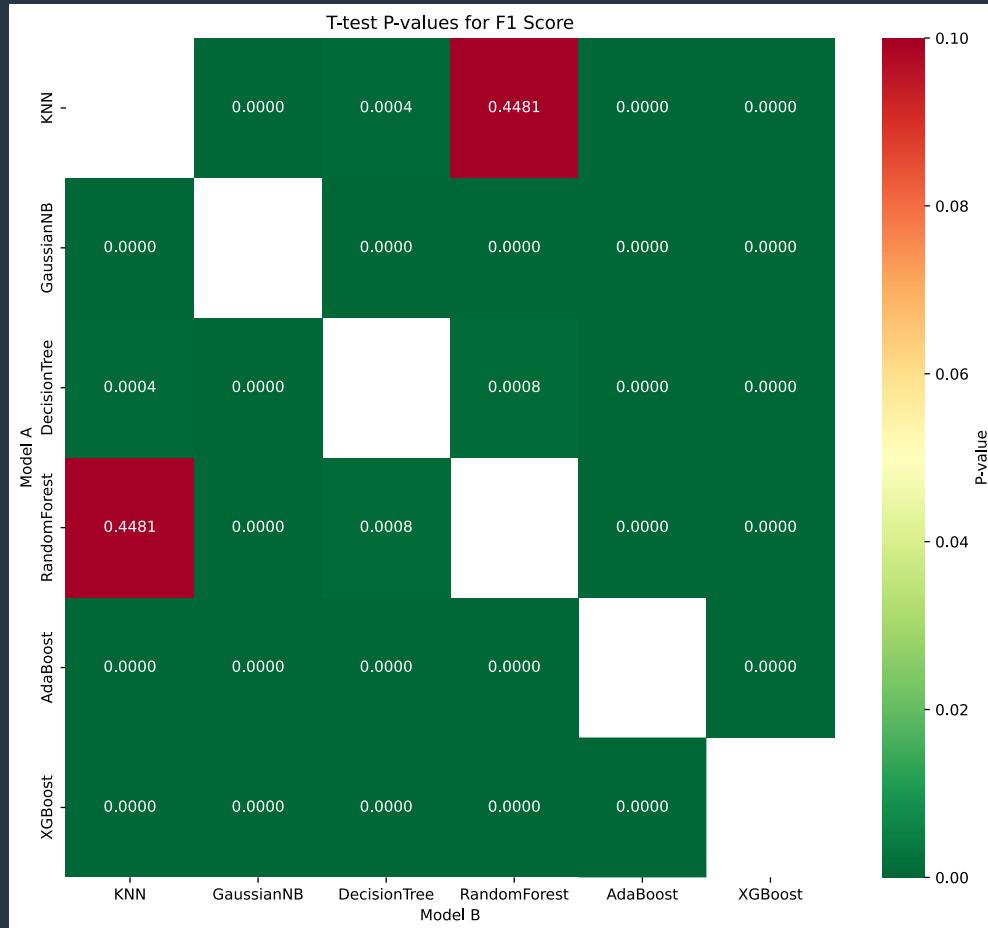
Model comparison

Threshold analysis



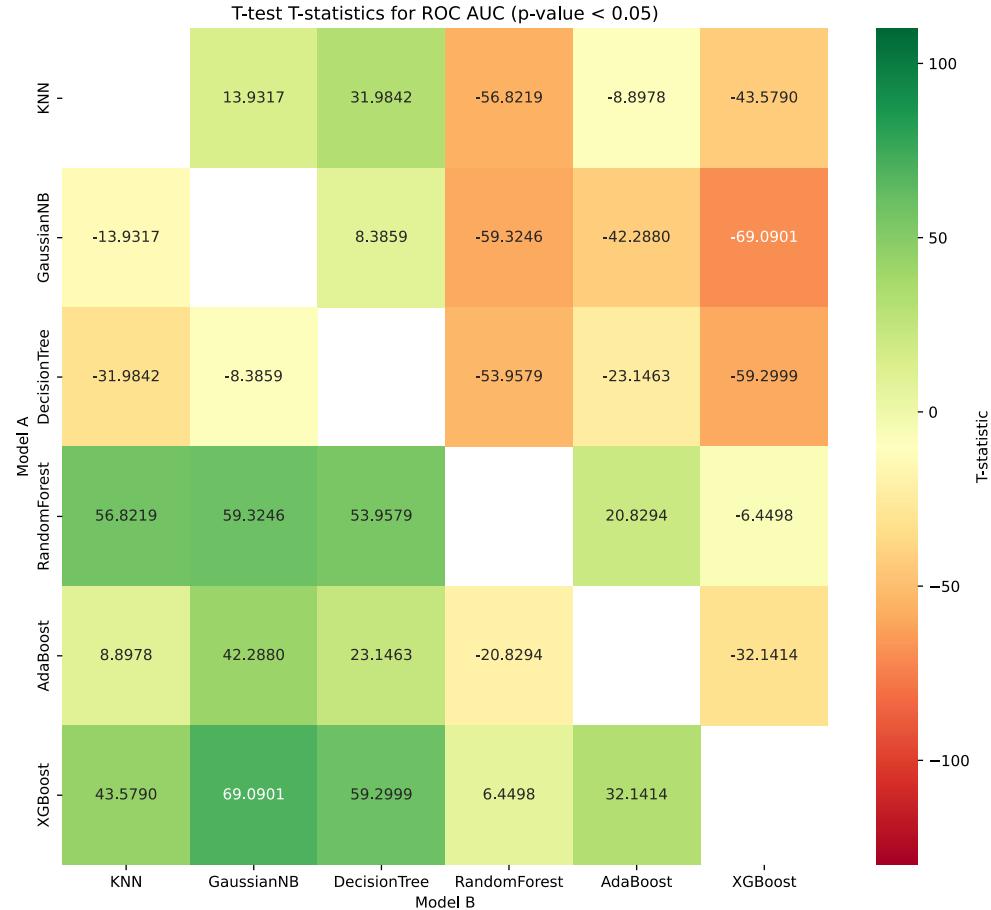
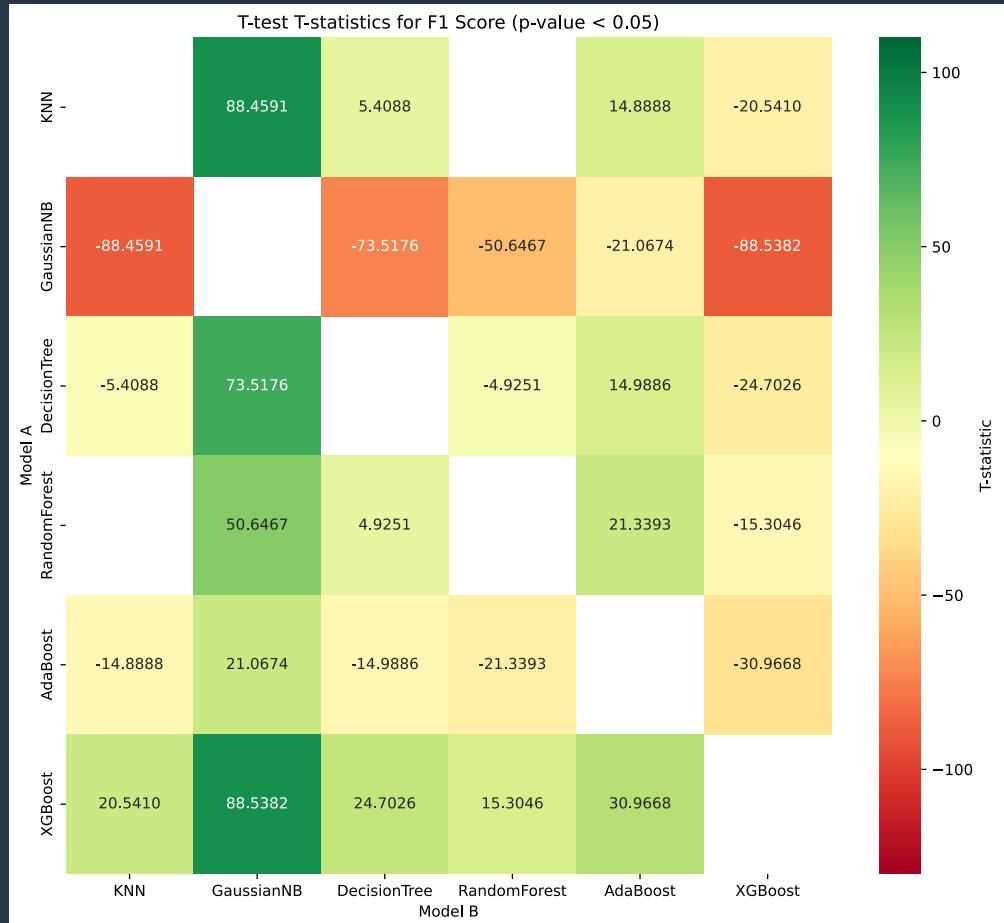
Statistical tests (*paired t-test*)

p-values



Statistical tests (*paired t-test*)

t-statistics



Model explainability



Goal

Understanding predictions & identifying influential features.



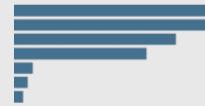
Feature Importances

- For DT, RF, ADA, XGB.
- Horizontal bar plots → top contributing features.



SHAP Values

- Quantify feature contribution for individual predictions.
- Summary plots → global feature impact.



Permutation Feature Importance

- Model-agnostic method.
- Measures decrease in performance when a feature is shuffled.



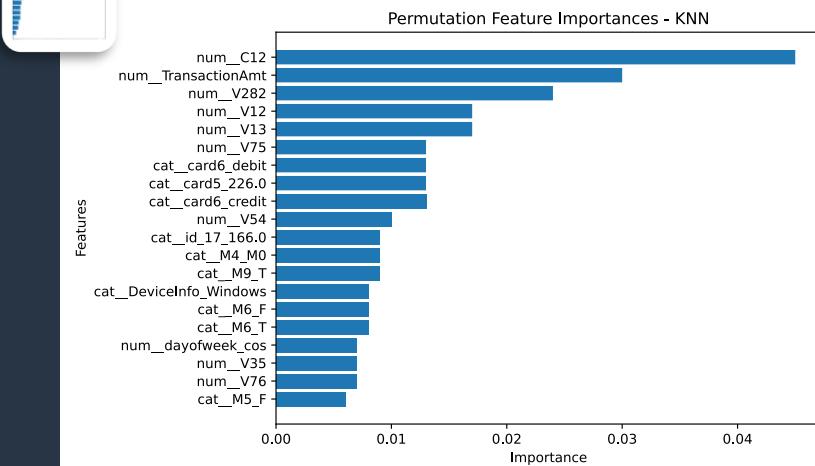
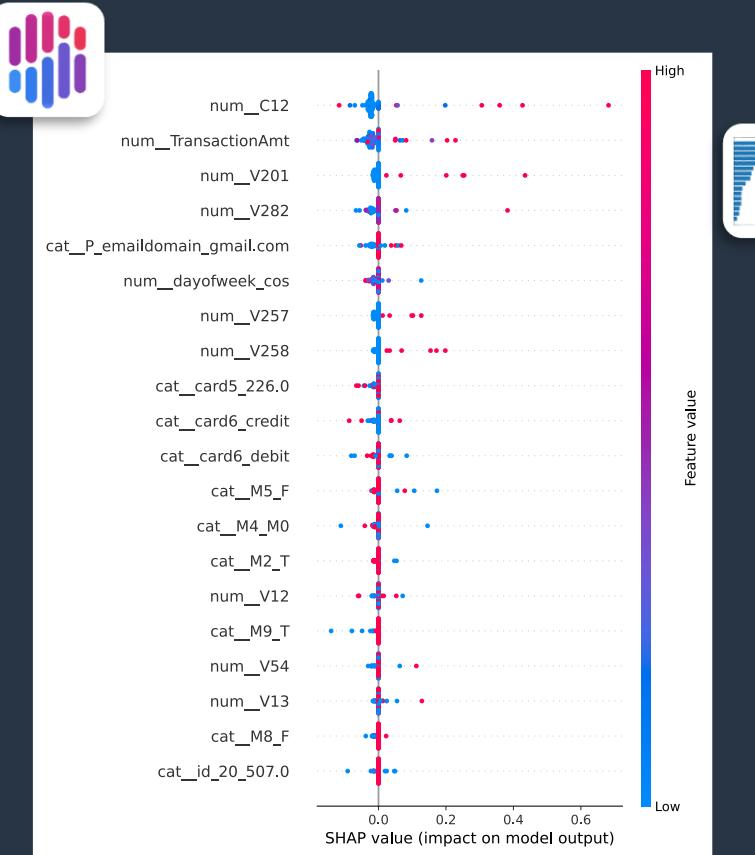
Rule Extraction via Surrogate Models

- Surrogate decision trees (depth=3).
- Extracts explicit, human-readable IF-THEN rules.

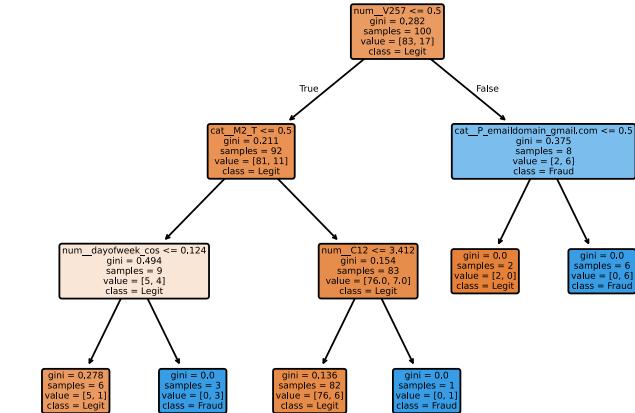


Explanations of the models

K-Nearest Neighbours

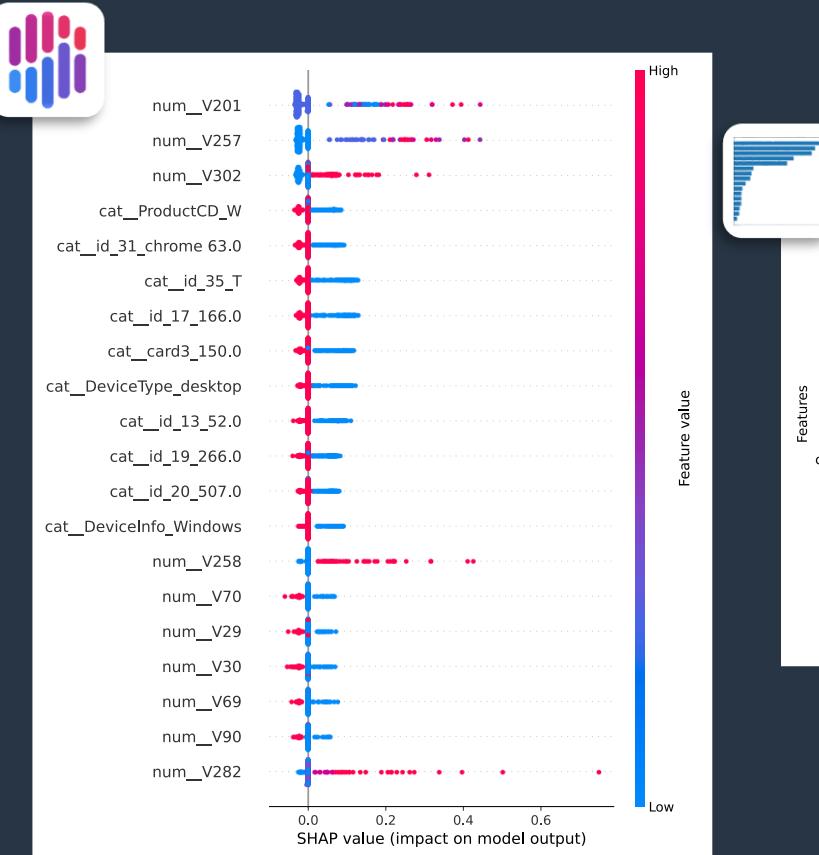


Surrogate Decision Tree - KNN

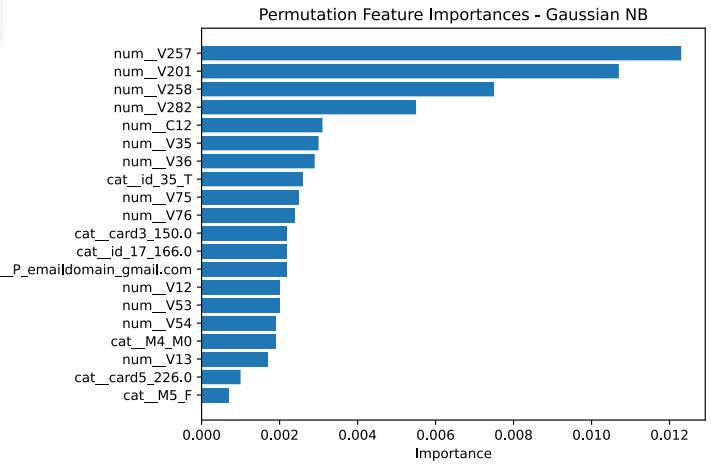


Explanations of the models

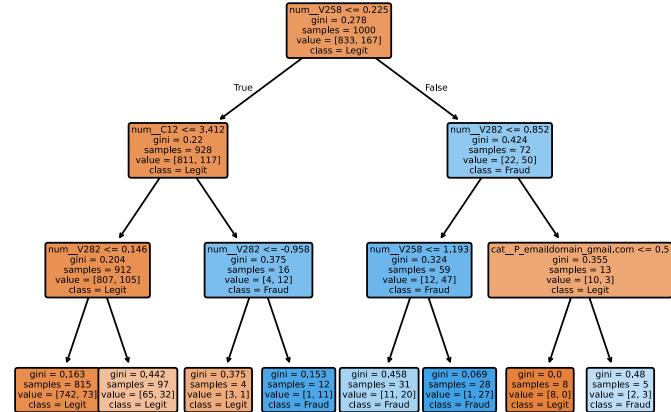
Gaussian NaiveBayes



Features

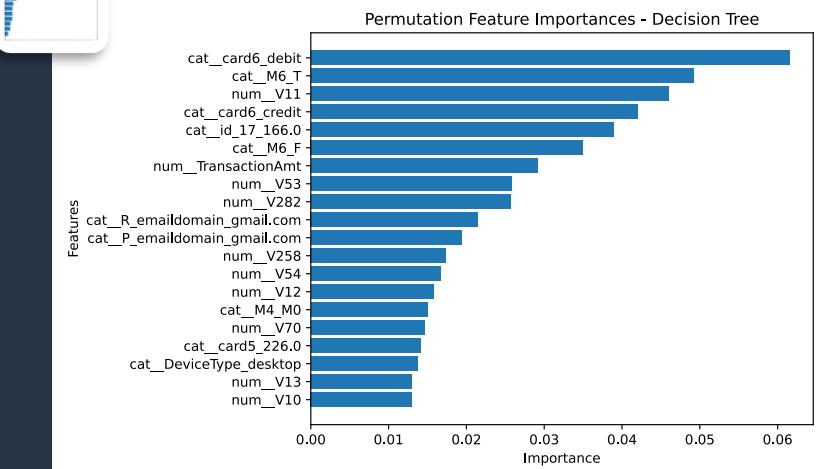
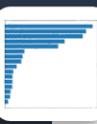
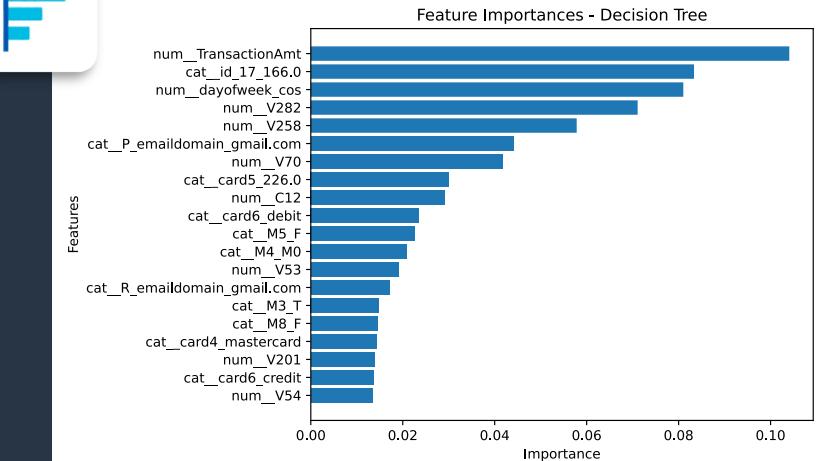
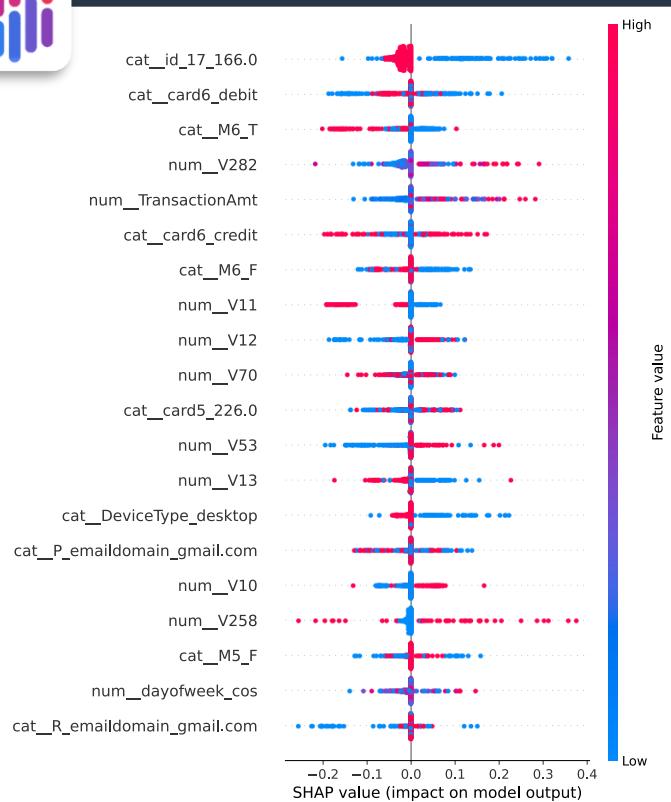


Surrogate Decision Tree - Gaussian NB

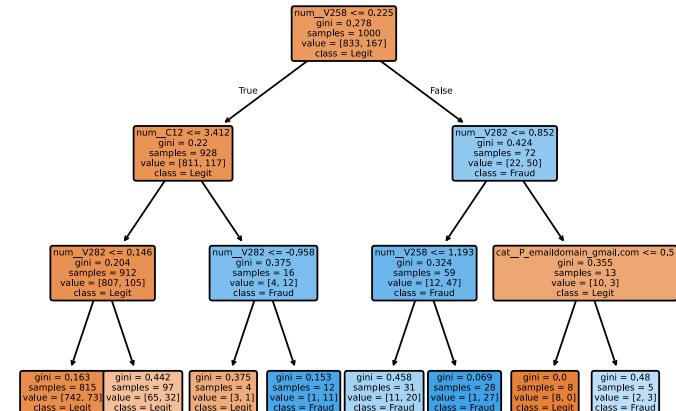


Explanations of the models

DecisionTree

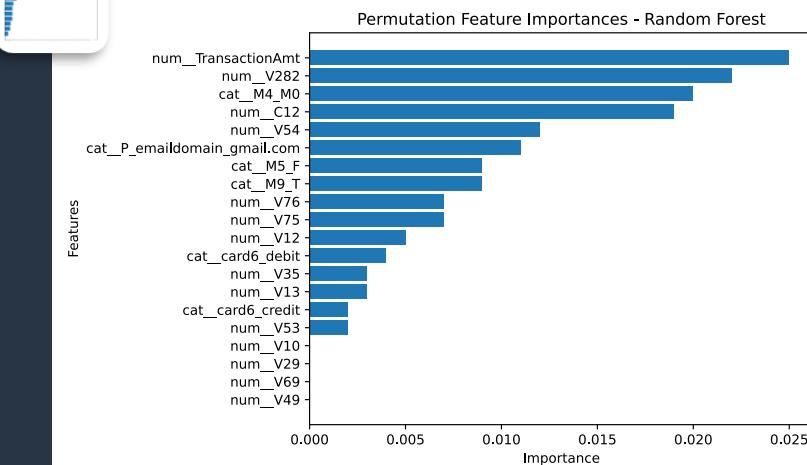
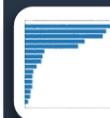
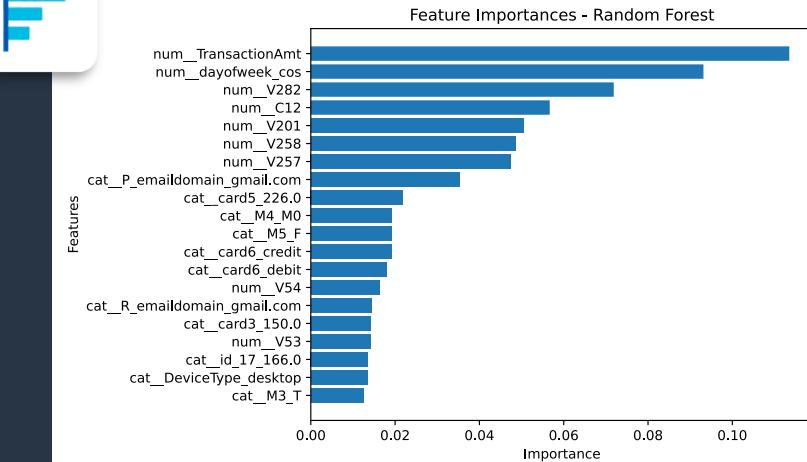
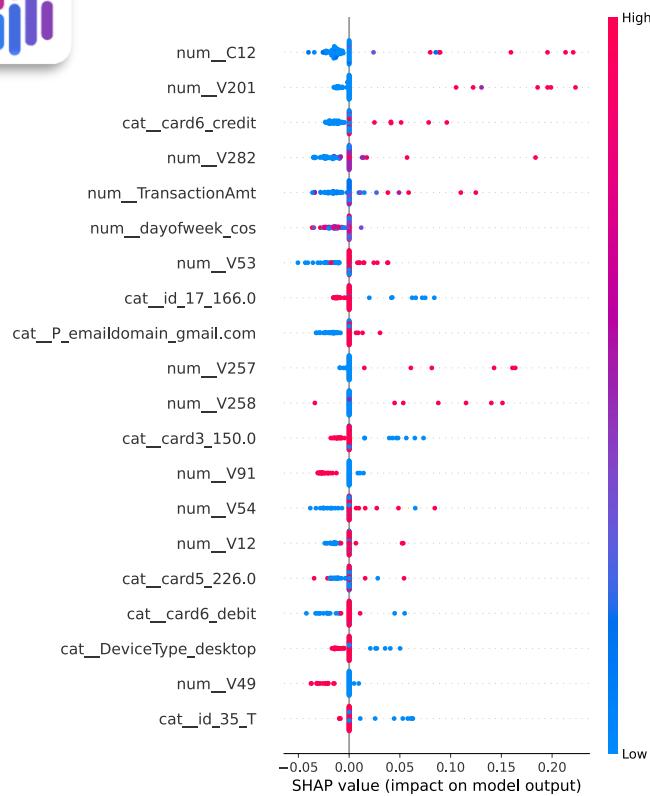


Surrogate Decision Tree - Decision Tree

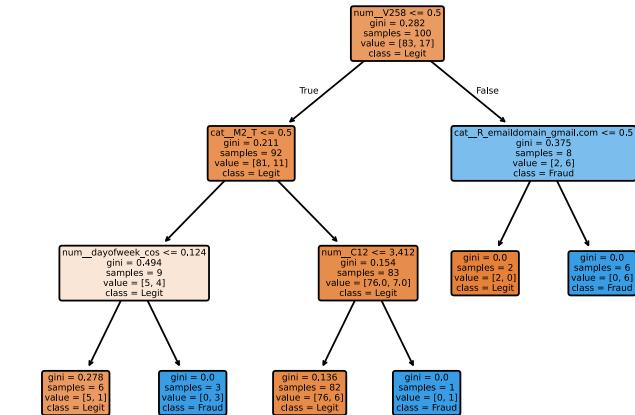


Explanations of the models

RandomForest

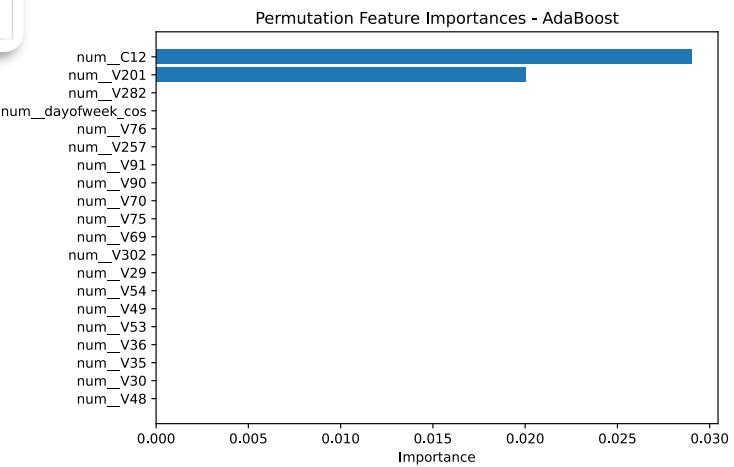
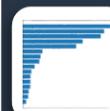
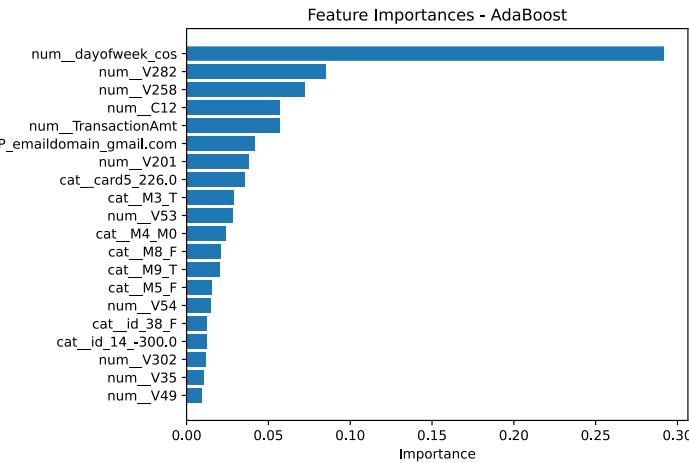
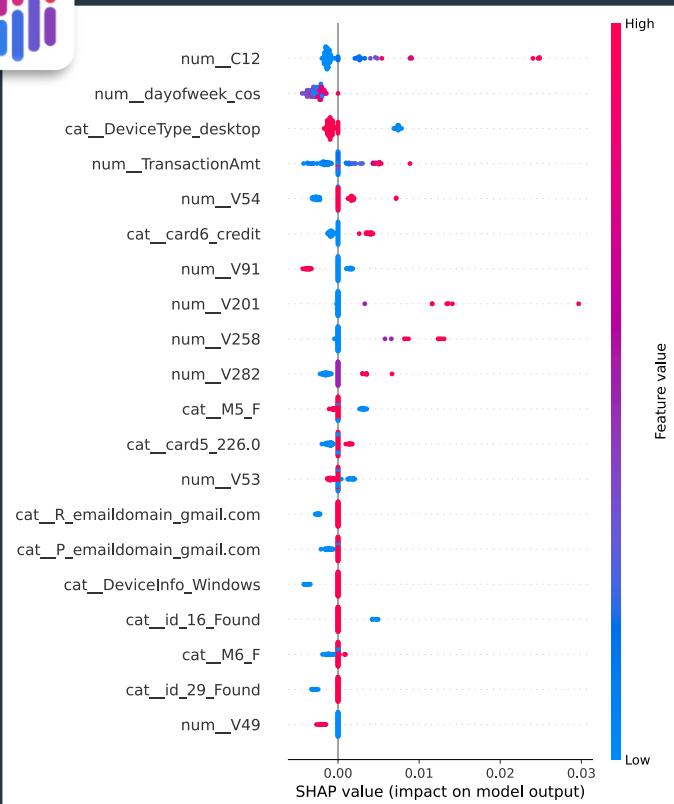


Surrogate Decision Tree - Random Forest

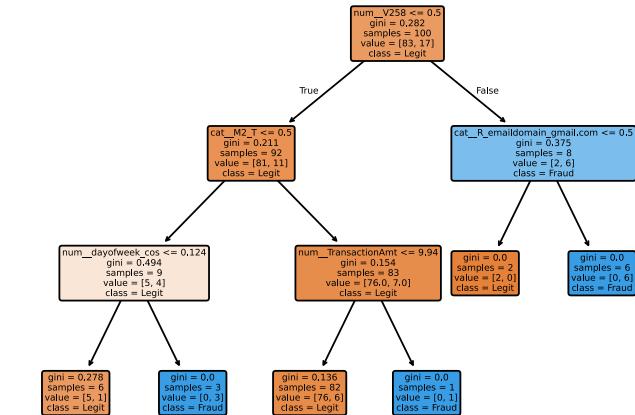


Explanations of the models

AdaBoost

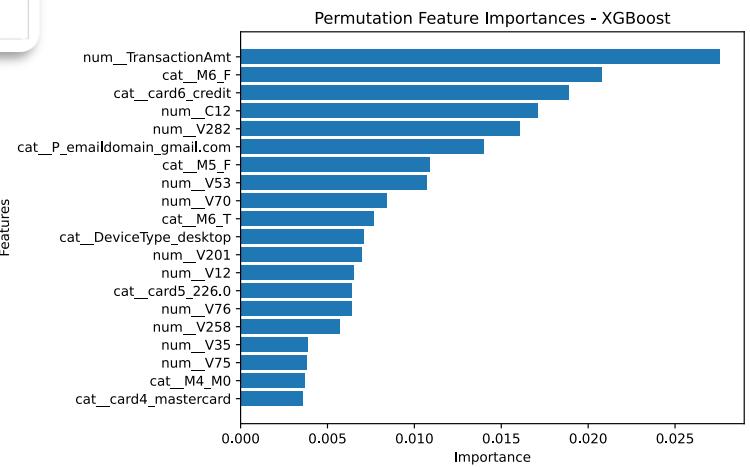
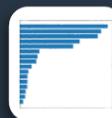
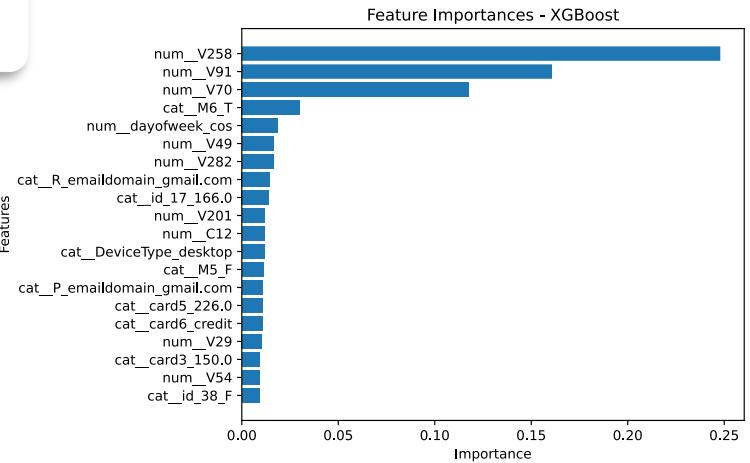
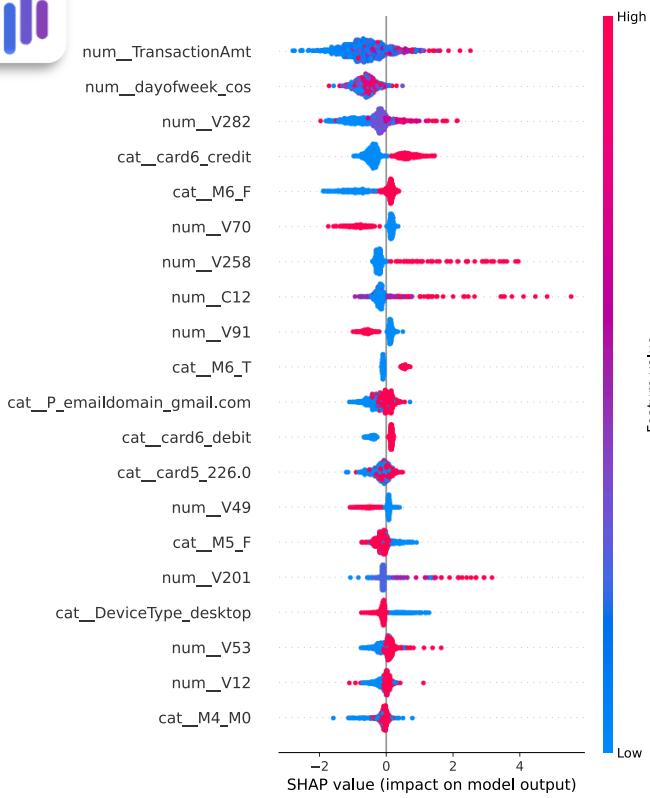


Surrogate Decision Tree - AdaBoost

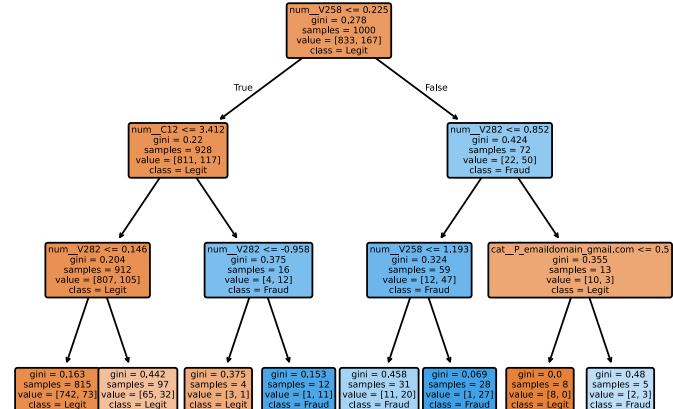


Explanations of the models

XGBoost



Surrogate Decision Tree - XGBoost



Real-world Application



Scenario

- Online payment systems → high risk of fraud
- Goal: real-time detection for banks, e-commerce, payment processors



Prototype

- Web interface built with *Streamlit*
- Users manually input transaction details



Pipeline

1. Pre-processing
 - Same as training: missing values, scaling, encoding, temporal features, feature selection
2. Classification
 - 6 classifiers: Decision Tree, Random Forest, Naive Bayes, KNN, AdaBoost, XGBoost
 - Output: Legitimate / Fraudulent
3. Ensemble decision
 - Majority voting for final prediction
4. Explainability (XAI)
 - SHAP waterfall plots to visualize how features contribute to increasing or decreasing the probability of fraud

Real-world Application

Credit Card Fraud Detection

Insert a transaction to classify:

TransactionID_x	TransactionDT	TransactionAmt	ProductCD	card1	card2
3457624	12153579	724	4	7826	481

Predict

Model Predictions:

Model	Prediction
Random Forest	Legitimate
XGBoost	Legitimate
Decision Tree	Legitimate
Gaussian NB	Legitimate
KNN	Legitimate
AdaBoost	Legitimate

Final Model Vote:

Transaction classified as Legitimate by 6/6 models!

Explanations (per model):

Explanation for Gaussian NB

Explanation for XGBoost

Explanation for RandomForest

Explanation for Decision Tree

Explanation for KNN

Explanation for AdaBoost

Conclusions

Key Findings



Best performance

Ensemble models: XGBoost
better even than
RandomForest and *AdaBoost*
(both $f1$ -score, ROC AUC).



Acceptable Level of Performance (ALP)

Some models can correctly identify $\geq 80\%$ frauds with limited false positives.



Threshold analysis

KNN & DecisionTree cannot match top performers.

AdaBoost → “conservative” (small threshold deviation), but high false positive rate.

Worst model: *Gaussian NaiveBayes*.

References



Related Work

- Cho Do Xuan, Dang Ngoc Phong, Nguyen Duy Phuong. *A new approach for detecting credit card fraud transaction*, International Journal of Nonlinear Analysis and Applications, Vol. 14 (2023), pp. 133–146.
Available at: https://ijnaa.semnan.ac.ir/article_7623_b95b41b8707a1ba645b2ad938f3cd76f.pdf



Bibliography

- *Kaggle Dataset: IEEE-CIS Fraud Detection*
Available at: <https://www.kaggle.com/datasets/phambacong/ieee-cis-fraud-detection>
- T. Chen and C. Guestrin, *XGBoost: A scalable tree boosting system* (2016).
Available at: <https://arxiv.org/abs/1603.02754>
- C. Nadeau, Y. Bengio, *Inference for the Generalization Error* (2003).
Available at: <https://doi.org/10.1023/A:1024068626366>
- Z. Hanusz, J. Tarasinska, and W.Zieliński, *Shapiro-wilk test with known mean* (2016).
Available at: https://www.researchgate.net/publication/298706800_Shapiro-Wilk_test_with_known_mean



Thanks for your attention!