

MACHINE LEARNING

Leakage-Resistant and Explainable Machine Learning for Bitcoin Transaction Analysis

Aastha Dua, Arshdeep Kaur, Amanjot Kaur

Department of Computer Science and Engineering, Malout Institute of Management and Information Technology, Malout, Punjab, India

Published October 2026

Abstract

Bitcoin provides a transparent public ledger, but identifying illicit transactions remains challenging because of severe class imbalance, evolving transaction patterns, and the graph structure of the transaction network. Conventional random train-test splitting can produce overly optimistic performance when structurally related transactions appear across different data partitions. This paper presents a leakage-resistant and explainable machine learning framework for Bitcoin transaction analysis. Experiments use the Elliptic Bitcoin Transaction Dataset containing 203,769 transactions, 234,355 transaction edges, and 46,564 labeled transactions. Logistic Regression, Random Forest, and histogram-based Gradient Boosting are evaluated chronologically, with time steps 1–34 used for training, 35–41 for validation, and 42–49 for testing. Accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC are reported. A separate random-split experiment investigates structural overlap. Temporal Random Forest achieves an F1-score of 0.617 and PR-AUC of 0.547, while random-split evaluation produces an F1-score of 0.937 and PR-AUC of 0.982. Edge analysis identifies substantial cross-partition connections in the random split, whereas no direct cross-partition edges are observed in the temporal split. SHAP is used to improve model interpretability. The results demonstrate the importance of leakage-resistant evaluation for reliable Bitcoin transaction classification.

Keywords: Bitcoin; data leakage; explainable artificial intelligence; illicit transaction detection; machine learning; SHAP; temporal evaluation; transaction analysis

1 Introduction

Bitcoin is a decentralized digital currency based on blockchain technology. Transactions are recorded on a public distributed ledger, making transaction information available for analysis. Although this transparency provides opportunities for financial analysis and forensic investigation, Bitcoin can also be associated with illicit activities such as fraud, scams, ransomware, and money laundering.

Machine learning has become an important approach for identifying suspicious Bitcoin transactions. However, transaction classification presents several challenges. The dataset contains a highly imbalanced distribution between licit and illicit transactions. In addition, Bitcoin transactions form a graph in which transactions can be connected through relationships represented by edges. These relationships create an important consideration during model evaluation.

A conventional random train-test split may distribute structurally related transactions across training and testing sets. Consequently, a model may obtain information indirectly related to test samples during training, resulting in an overly optimistic estimate of generalization performance. Chronological evaluation is more representative of deployment because future transactions are evaluated using information from earlier periods.

The contributions of this work are: (1) a chronological evaluation strategy; (2) a comparison of temporal and random splits using transaction-edge overlap; (3) evaluation of three machine learning models; (4) reporting of multiple metrics; (5) SHAP-based feature explanations; and (6) a sensitivity experiment that removes the explicit time-step feature.

2 Related work

Previous research has demonstrated the usefulness of machine learning and graph-based approaches for identifying illicit activity in cryptocurrency transaction networks. The Elliptic dataset has been widely used for Bitcoin transaction classification and financial-forensics research.

Weber et al. [1] investigated illicit Bitcoin transaction detection using graph convolutional networks and demonstrated the importance of graph structure for cryptocurrency analysis. Other studies have explored machine learning-based approaches for cryptocurrency anti-money-laundering applications.

High predictive performance does not necessarily guarantee reliable real-world performance. In transaction networks, inappropriate data splitting can introduce structural relationships between training and testing samples. This study therefore investigates temporal versus random splitting while incorporating explainability.

3 Dataset and preprocessing

3.1 Dataset description

The experiments use the Elliptic Bitcoin Transaction Dataset [2], which contains transaction-level features, transaction classes, and an edge list describing relationships between transactions. Class 1 is treated as illicit and class 2 as licit. Unknown transactions are excluded from supervised classification because their ground-truth labels are unavailable.

The feature representation contains anonymized variables. Feature names are retained without assigning unsupported real-world meanings to them.

Table 1. Dataset statistics.

Property	Value
Total transactions	203,769
Total edges	234,355
Total time steps	49
Licit transactions	42,019
Illicit transactions	4,545
Unknown transactions	157,205
Labeled transactions	46,564
Features used	166

3.2 Data preparation

Transaction features and class labels are merged using the transaction identifier (txId). The identifier and class columns are removed from the model input. The target is encoded as 0 for licit and 1 for illicit. The dataset is highly imbalanced, with illicit transactions representing a substantially smaller proportion of labeled transactions. Class weighting is used for Logistic Regression and Random Forest.

For Logistic Regression, a StandardScaler is fitted only on training data and then applied to validation and test data.

4 Leakage-resistant evaluation methodology

4.1 Temporal split

To simulate prediction of future transactions, labeled data are divided chronologically. Time steps 1–34 are used for training, 35–41 for validation, and 42–49 for testing. The temporal test set contains 8,433 licit and 408 illicit transactions.

All preprocessing operations requiring fitting are performed using the training data before being applied to validation and test data.

Table 2. Temporal data partitions.

Partition	Time Steps	Samples
Training	1–34	29,894
Validation	35–41	7,829
Testing	42–49	8,841

4.2 Random split comparison

A stratified random split is also performed to demonstrate the difference between conventional evaluation and chronological evaluation.

Table 3. Cross-partition edge comparison.

Split Type	Train–Val	Train–Test	Val–Test
Temporal	0	0	0
Random	7,645	7,858	1,713

The temporal split shows no direct edge overlap between the labeled partitions examined. In contrast, the random split contains thousands of cross-partition edges. This analysis tests direct edge overlap only; it does not prove that every possible form of graph or temporal leakage has been eliminated.

5 Machine learning models

5.1 Logistic Regression

Logistic Regression is used as a linear baseline. Features are standardized using a StandardScaler fitted only on the training set. Class weighting is set to balanced to address class imbalance.

5.2 Random Forest

Random Forest is an ensemble learning method consisting of multiple decision trees. The model uses 200 trees, balanced class weights, and a fixed random seed of 42.

5.3 Gradient Boosting

Histogram-based Gradient Boosting is used as the third model. The configuration contains 200 iterations, a learning rate of 0.05, maximum 31 leaf nodes, L2 regularization of 1.0, and random seed 42.

6 Performance evaluation

The models are evaluated using accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC. Accuracy = $(TP + TN) \div (TP + TN + FP + FN)$. Precision = $TP \div (TP + FP)$. Recall = $TP \div (TP + FN)$. F1 = $2 \times (Precision \times Recall) \div (Precision + Recall)$. ROC-AUC and PR-AUC are additionally reported because accuracy alone can be misleading for highly imbalanced classification problems.

7 Experimental results

Table 4. Temporal test-set performance.

Model	Accuracy	Precision	Recall	F1	ROC-AUC	PR-AUC
Logistic Regression	0.725	0.122	0.799	0.212	0.855	0.200
Random Forest	0.974	0.964	0.453	0.617	0.857	0.547
Gradient Boosting	0.971	0.814	0.471	0.596	0.869	0.558

The results show different trade-offs. Logistic Regression achieves the highest recall but produces many false positives. Random Forest provides the highest precision and F1-score. Gradient Boosting obtains the highest ROC-AUC and PR-AUC among the three models.

The Random Forest confusion matrix contains 8,426 true negatives, 7 false positives, 223 false negatives, and 185 true positives. The model produces few false positives but misses many illicit transactions, explaining its high precision and comparatively lower recall.

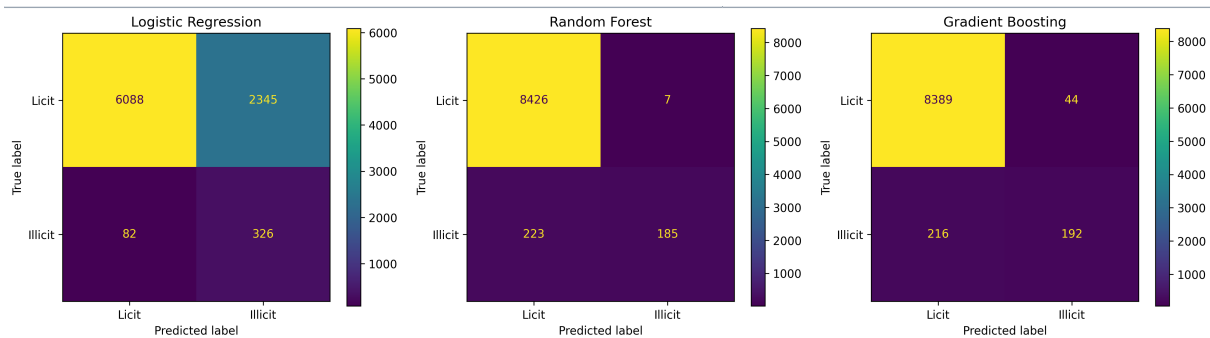


Figure 1. Confusion matrices for Logistic Regression, Random Forest, and Gradient Boosting on the temporal test set.

8 Random versus temporal evaluation

Table 5. Random versus temporal Random Forest performance.

Evaluation	Accuracy	Precision	Recall	F1	ROC-AUC	PR-AUC
Temporal test	0.974	0.964	0.453	0.617	0.857	0.547
Random test	0.988	0.995	0.886	0.937	0.997	0.982

Under the random split, F1-score increases from 0.617 to 0.937, while PR-AUC increases from 0.547 to 0.982. The random evaluation also contains thousands of cross-partition edges. Together, these observations indicate that random splitting may provide an overly optimistic estimate of generalization for this transaction graph. Cross-partition edges do not by themselves prove that each connection causes leakage, but they motivate careful split design.

9 Explainable machine learning using SHAP

To examine why transactions are classified as illicit, SHAP (SHapley Additive exPlanations) [3] is applied to the Random Forest model. A sample of 500 temporal test transactions is used. Mean absolute SHAP values identify features with the largest contribution to model predictions.

Table 6. Selected top SHAP features.

Rank	Feature	Mean SHAP
1	feature_53	0.029524
2	feature_55	0.026264
3	feature_47	0.016854
4	feature_52	0.014783
5	feature_43	0.013727
6	feature_41	0.013714
7	feature_18	0.012650
8	feature_49	0.012076
9	feature_29	0.011613
10	Feature 90	0.011590

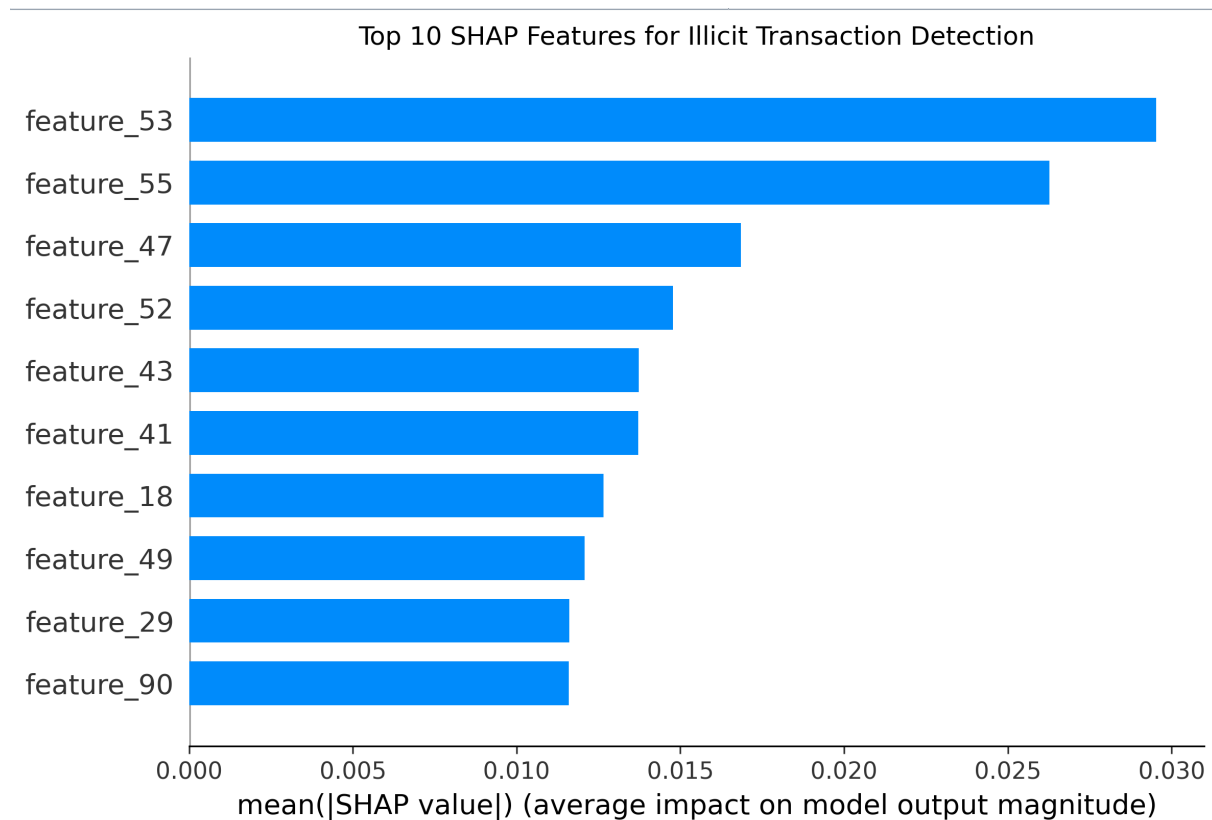


Figure 2. Top SHAP features contributing to illicit transaction detection.

Because the original feature representation is anonymized, these variables are interpreted only as features contributing to model predictions; unsupported semantic meanings are not assigned. SHAP values explain contributions to predictions and are not proof of causal relationships.

10 Time-step sensitivity analysis

Table 7. Time-step sensitivity analysis.

Metric	With Time Step	Without Time Step
Accuracy	0.9740	0.9742
Precision	0.9635	0.9688
Recall	0.4534	0.4559
F1-score	0.6167	0.6200
ROC-AUC	0.8568	0.8457
PR-AUC	0.5467	0.5427

Removing time_step produces only small changes in the classification metrics. F1-score increases slightly, whereas ROC-AUC and PR-AUC decrease modestly. Overall, the experiment indicates that classification performance is largely maintained without explicitly providing the time-step variable to the model.

11 ROC and precision–recall analysis

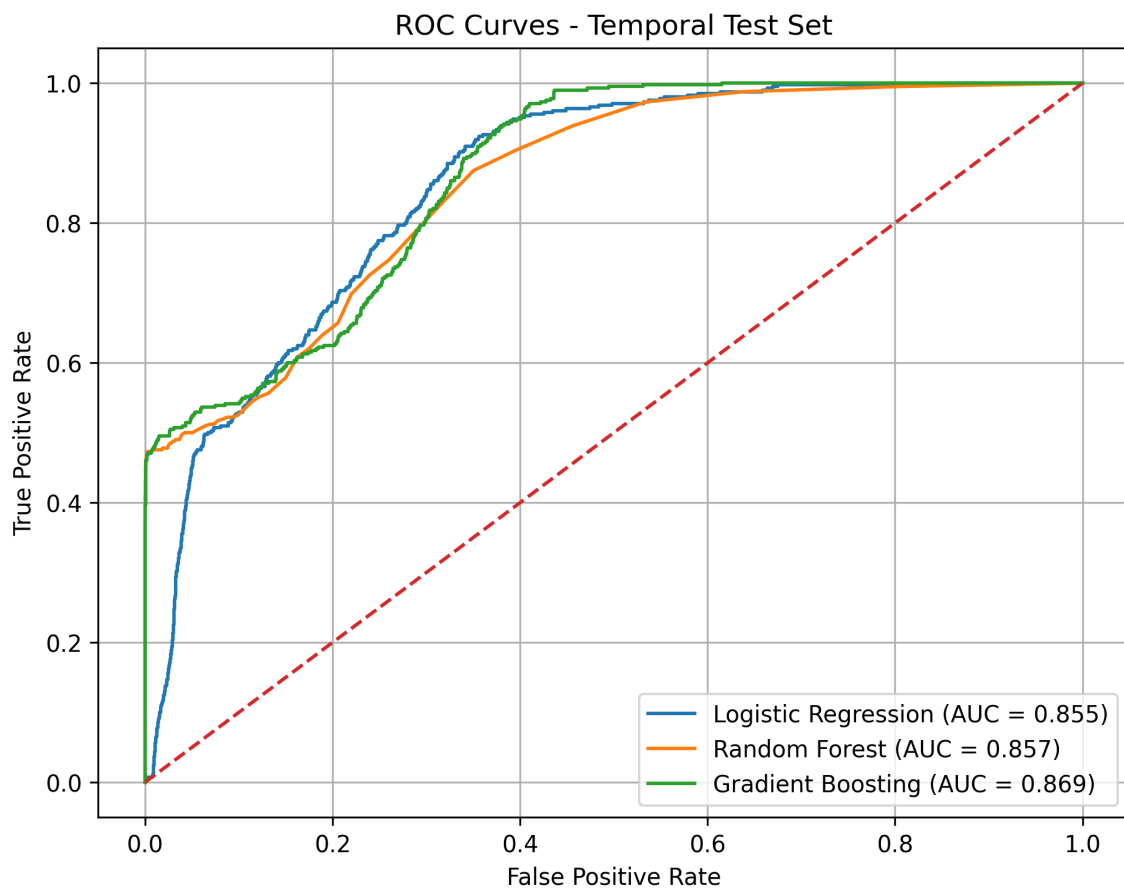


Figure 3. ROC curves for Logistic Regression, Random Forest, and Gradient Boosting on the temporal test set.

The ROC-AUC values are 0.855 for Logistic Regression, 0.857 for Random Forest, and 0.869 for Gradient Boosting. Gradient Boosting provides the strongest ranking performance according to ROC-AUC.

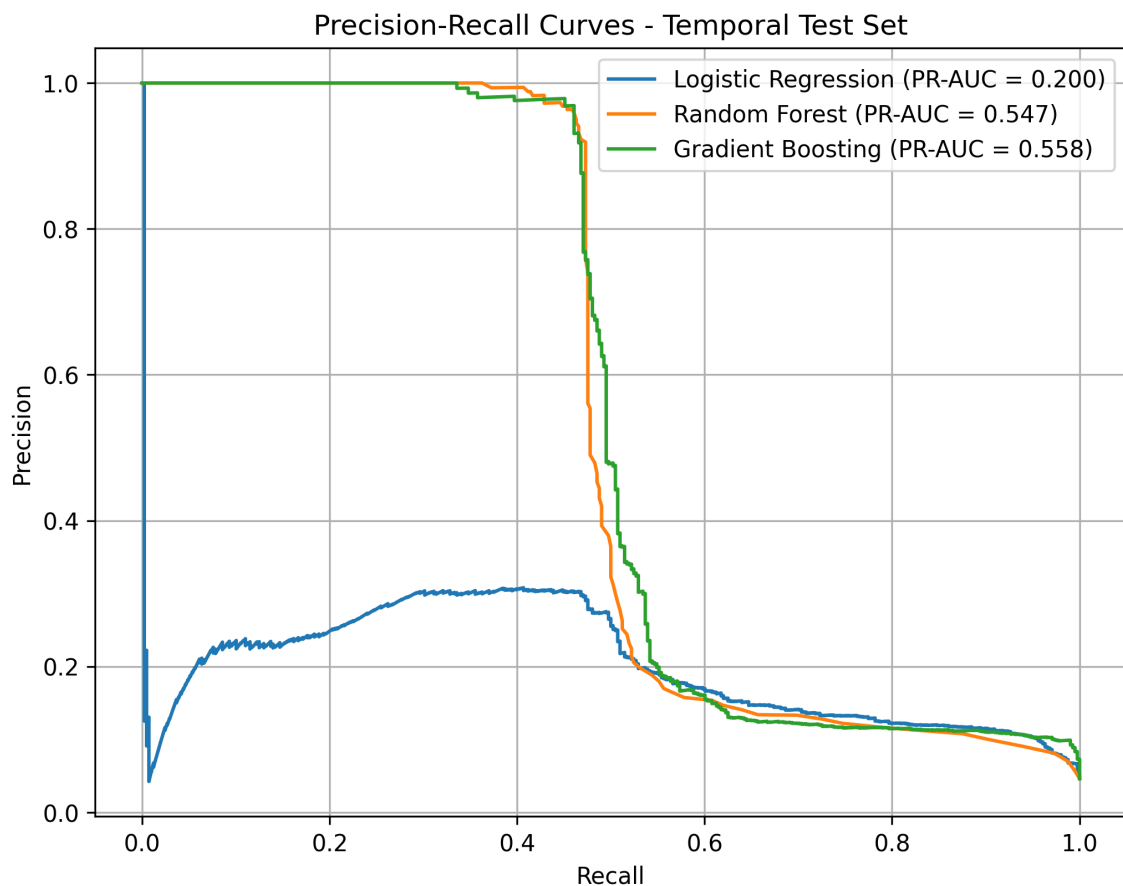


Figure 4. Precision–recall curves for Logistic Regression, Random Forest, and Gradient Boosting on the temporal test set.

The PR-AUC results are 0.200, 0.547, and 0.558 for Logistic Regression, Random Forest, and Gradient Boosting, respectively. The results highlight the difficulty of detecting the minority illicit class under chronological evaluation.

12 Discussion

The experiments demonstrate that evaluation methodology has a major effect on reported performance. Random Forest obtains an F1-score of 0.937 under random evaluation but only 0.617 under chronological evaluation. PR-AUC decreases from 0.982 to 0.547. Edge analysis finds substantial cross-partition connections in the random split and no direct cross-partition edges in the temporal split.

These observations suggest that random splitting can expose the learning process to structural relationships spanning training and evaluation sets. Among the temporal models, Random Forest offers the strongest F1-score and precision, making it attractive when false-positive control is important. Gradient Boosting provides the strongest ROC-AUC and PR-AUC, while Logistic Regression provides the highest recall. Model selection should therefore depend on the operational objective.

13 Conclusion

This paper investigated leakage-resistant and explainable machine learning for Bitcoin transaction analysis. Using the Elliptic dataset, Logistic Regression, Random Forest, and Gradient Boosting were

evaluated under a chronological split designed to represent prediction of future transactions. Random Forest achieves an F1-score of 0.617 and PR-AUC of 0.547 under temporal evaluation, compared with 0.937 and 0.982 under random evaluation. The random split also contains substantial cross-partition transaction connections, whereas the temporal split shows no direct cross-partition edges among the evaluated labeled transactions. These findings emphasize that evaluation methodology is critical for graph-linked financial data. SHAP provides a mechanism for examining which anonymized features contribute most strongly to predictions. Overall, the study supports chronological evaluation and explainability as important components of reliable Bitcoin transaction classification.

References

- [1] Weber M, Domeniconi G, Chen J, Weidele DK, Bellei C, Robinson T, et al. Anti-money laundering in Bitcoin: experimenting with graph convolutional networks for financial forensics. In: KDD '19 Workshop on Anomaly Detection in Finance (25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining); 2019. Available from: <https://arxiv.org/abs/1908.02591>
- [2] Elliptic. *Elliptic data set* [Internet]. Kaggle. Available from: <https://www.kaggle.com/ellipticco/elliptic-data-set>
- [3] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems 30 (NIPS 2017); 2017. Available from: <https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html>
- [4] Pedregosa F, et al. Scikit-learn: machine learning in Python. *Journal of Machine Learning Research*. 2011;12:2825–2830. Available from: <https://www.jmlr.org/papers/v12/pedregosa11a.html>
- [5] Scikit-learn. *Classification metrics and model evaluation* [Internet]. scikit-learn documentation. Available from: https://scikit-learn.org/stable/modules/model_evaluation.html
- [6] Pocher A, et al. Detecting anomalous cryptocurrency transactions: an AML/CFT application of machine learning-based forensics. *Electronic Markets*. 2023;33. <https://doi.org/10.1007/s12525-023-00654-3>