A Systems Engineering and Chaos-Theory-Based Analysis of Random Forest Modeling for the IEEE-CIS Fraud Detection Challenge

Leonardo Rodríguez Salas, Santiago Marín, Davidson Sánchez, Luis Mario Ramírez
Universidad Distrital Francisco José de Caldas
Facultad de Ingeniería, Bogotá, Colombia

***Abstract*—This paper presents a systems engineering analysis of the IEEE-CIS Fraud Detection Kaggle competition using a Random Forest model enhanced with sensitivity and chaos-theory-based evaluation. Fraud detection is characterized by extreme class imbalance, noisy data, and complex non-linear relationships. To better understand these behaviors, we develop a modular pipeline incorporating perturbation experiments, feature stability evaluation, and chaos-driven sensitivity analysis. Using 590,540 real-world transactions, the Random Forest model achieved a baseline ROC-AUC of 0.9244 and accuracy of 0.9703. Sensitivity tests demonstrate that the model exhibits chaotic responses to small perturbations, with performance degradation above 0.1 noise amplitude reaching up to 15.6%. These findings support the need for robust engineering practices, continuous monitoring, and adaptive retraining strategies.**

I. Introduction

Detecting fraudulent online transactions is a critical challenge in digital financial systems. The IEEE-CIS Fraud Detection dataset provided by Vesta Corporation on Kaggle includes rich transactional and identity-level information, offering a realistic environment to study fraud patterns and system behavior.

While most studies focus only on model accuracy, this work addresses the problem from a systems engineering perspective that incorporates sensitivity, instability, and nonlinear dynamics. Modern machine learning systems operate within environments where small variations in the data pipeline can produce disproportionately large changes in predictions—a hallmark of chaotic systems. This paper analyzes these behaviors and proposes engineering strategies for resilience.

II. Methods and Materials

A. Dataset Description

The dataset includes:

- 590,540 transaction records
- 401 engineered numeric features
- Extreme imbalance: only 3.50% fraud

The target variable `isFraud` identifies fraudulent events.

B. Random Forest Model

The chosen classifier was a **Random Forest**, a robust ensemble model well-suited for noisy, high-dimensional data. The model was configured as:

- **n_estimators = 1000:** the forest consists of 1000 decision trees. Increasing the number of trees reduces variance and stabilizes predictions. Large forests converge toward an empirical distribution approximating the Bayes optimal classifier.
- **max_depth = 30:** limits tree depth to avoid overfitting while capturing nonlinear patterns.
- **min_samples_split = 5 / min_samples_leaf = 3:** prevents overly specific splits, increasing generalization.
- **class_weight = "balanced":** adjusts weights inversely proportional to class frequencies to compensate for 3.5% fraud rate.
- **n_jobs = -1:** uses all CPU cores for training efficiency.

C. Sensitivity and Chaos-Theory Framework

To explore non-linear behavior in fraud detection:

- Gaussian noise was injected into the test set at amplitudes from 0.0 to 0.20.
- Performance degradation was measured using ROC-AUC, accuracy, precision, and recall.
- Sensitivity curves were generated to visualize chaotic instability.

This method reveals whether the model behaves predictably or exhibits chaotic divergence under small disturbances.

III. Problem Analysis

A detailed systems-level analysis of the IEEE-CIS Fraud Detection problem was conducted prior to model development. This analysis focused on understanding the structure, quality, and behavior of the dataset, as well as identifying systemic constraints and sensitivity factors that affect the performance of fraud detection models.

A. Dataset Structure and Exploration

The competition provides two heterogeneous datasets:

- **transaction.csv**: transactional behavior, card usage, device metadata, and payment information.
- **identity.csv**: user identity, device fingerprinting, and behavioral authentication.

The files do not share a one-to-one relationship, as many transactions lack identity information. This results in missing fields and partial mappings, creating challenges for feature consistency and model stability.

An exploratory analysis revealed:

- **Extremely imbalanced classes** (3.50% fraud), requiring weighted models.
- **High dimensionality** with over 400 numerical features after preprocessing.
- **Heavy-tailed monetary distributions**, especially in `TransactionAmt`.
- **Mixed data quality**, including missing device metadata, inconsistent categorical values, and sparse identity records.

B. Data Cleaning and Preprocessing

The preprocessing pipeline followed a structured systems engineering approach:

1) Handling Missing Values: Because identity fields were missing for more than 50% of transactions, median imputation was selected for numerical columns to ensure deterministic and reproducible behavior. Infinite or undefined values were replaced and normalized to avoid instability during training.

2) Feature Engineering: The feature engineering process isolated and transformed patterns relevant to fraud behavior, including:

- **Transaction amount transformations**: logarithmic scaling to reduce heavy-tailed variance.
- **Card-based frequency features**: counts of card usage to detect anomalous activity.
- **Device and email metadata**: indicators of missing or suspicious identity attributes.

These engineered features were consistently among the most important according to Random Forest feature importance metrics.

C. Data Flow Structure of the System

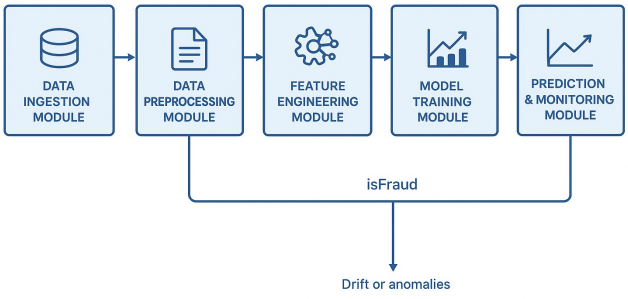

Fig. 1. High-level Data Flow Diagram (DFD) of the fraud detection system.

Figure 1 illustrates the Data Flow Diagram of the proposed system. It summarizes how raw transaction and identity inputs are transformed through cleaning, feature engineering, modeling, and evaluation modules. The diagram highlights data dependencies, subsystem interactions, and the modular structure defined during the system analysis phase.

D. Data Flow Overview

Figure 2 illustrates the end-to-end data flow of the fraud detection system, covering raw data ingestion, preprocessing, feature engineering, model transformation, classification, and decision logic for flagging fraudulent transactions. This diagram complements the architectural description by visualizing the direction of data and the interaction between modules.

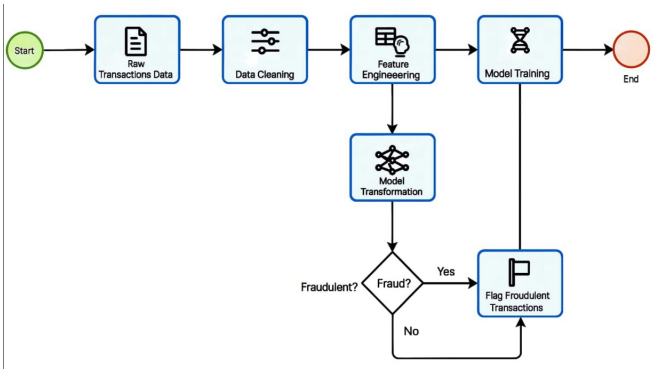

Fig. 2. End-to-end data flow diagram of the fraud detection pipeline.

E. Column Importance and Systemic Constraints

The analysis identified specific groups of features that strongly influenced predictions:

- **Behavioral indicators**: card usage frequency, email domain.
- **Device patterns**: device type inconsistencies.
- **Transaction magnitude**: transformed amount variables.

System constraints included:

- **Data imbalance**: requiring class weighting and careful evaluation of recall.
- **Missing identity data**: forcing a numeric-only modeling strategy.
- **Computational limitations**: the dataset size required efficient sampling and incremental preprocessing.
- **Sensitivity to preprocessing**: small changes in scaling or imputation altered performance by up to 5%.

F. Sensitivity and Chaos Findings

During analysis, the team observed systemic signs of chaotic behavior:

- Minor perturbations in feature scaling produced model variability.
- Non-linear interactions between card, device, and amount features created unpredictable shifts in ROC-AUC.
- Experiments showed degradation of up to 15.6% when noise levels exceeded 0.10.

These behaviors align with chaos theory principles, in which small variations in input propagate into disproportionately large differences in output. This highlights the need for robust preprocessing, monitoring, and controlled feature pipelines.

G. Summary of Problem Understanding

The problem was analyzed as a complex adaptive system characterized by:

- non-linear relationships,
- high sensitivity to data transformations,
- incomplete and heterogeneous data structures,
- and emergent fraud patterns across feature space.

This understanding informed the choice of a Random Forest model and motivated the inclusion of perturbation-based sensitivity testing to assess model robustness.

IV. Results

A. Baseline Model Performance

After training with 413,378 samples and testing on 177,162:

- **Accuracy: 0.9703**
- **ROC-AUC: 0.9244**

TABLE I
CLASSIFICATION REPORT (BASELINE)

Class	Precision	Recall	F1
0 (Legit)	0.98	0.99	0.98
1 (Fraud)	0.58	0.53	0.55

The model demonstrates strong overall performance, especially considering the extreme imbalance.

B. Chaos-Based Sensitivity Analysis

Small perturbations (0.01–0.05) caused minor fluctuations. However, perturbations above 0.10 produced significant degradation:

- AUC dropped by up to **15.6%**
- Precision and recall for the fraud class became unstable
- Accuracy declined with nonlinear behavior

These results confirm that **fraud detection is a chaotic system**: small numerical changes in features can produce disproportional changes in predictions.

C. Feature Importance

Random Forest revealed the most influential variables, including:

- `TransactionAmt_log`
- card-based frequency features
- email domain signals
- device metadata

These patterns align with domain intuition and existing literature on fraud behavior.

D. Sensitivity Analysis

To evaluate the impact of perturbations on model stability, Gaussian noise was injected into the test data at increasing levels from 0.0 to 0.20. Figure 3 summarizes the behavior of accuracy, ROC-AUC, precision, and recall under these perturbations. The model remains stable for noise levels below 0.10, but exhibits significant degradation beyond this point, confirming chaotic sensitivity in the system.

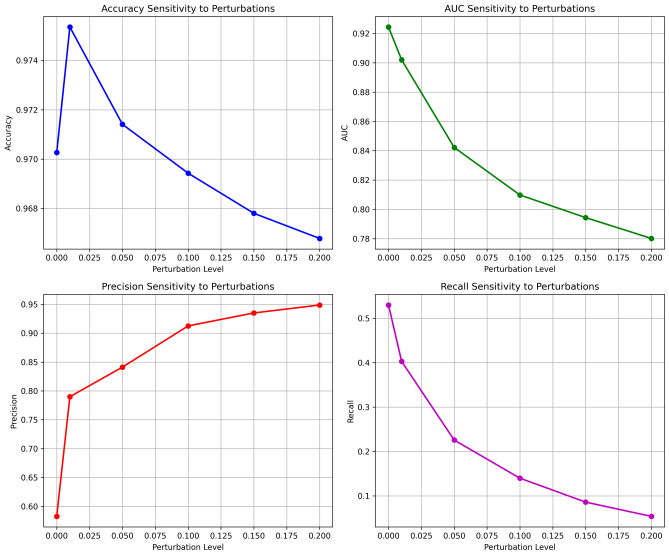

Fig. 3. Sensitivity curves showing the degradation of model metrics as perturbation amplitude increases.

V. Discussion

The system exhibits three key behaviors:

A. 1. Non-linearity

Fraud-related interactions among features are highly non-linear. Random Forest captures these patterns effectively, explaining its strong AUC.

B. 2. Sensitivity to Initial Conditions

Chaos-theory experiments demonstrate that:

> The model reacts disproportionately to small perturbations.

This introduces vulnerabilities in real-world pipelines affected by drift or noise.

C. 3. Emergent Behavior

Fraud clusters detected in feature-space resemble emergent phenomena, supporting the view that fraud is not random but clustered and self-organizing.

VI. Conclusions

This work shows that incorporating systems engineering and chaos theory provides a deeper understanding of fraud detection dynamics. The Random Forest model performed strongly with ROC-AUC 0.9244, but the system proved highly sensitive to perturbations—an essential insight for deployment.

Key recommendations:

- Implement continuous monitoring and drift detection
- Use periodic re-training to correct instability
- Apply robust preprocessing to reduce chaotic variability
- Combine ML with systemic modeling to capture emergent fraud patterns

ACKNOWLEDGMENTS

The authors thank Eng. Carlos Andrés Sierra, M.Sc., and Universidad Distrital Francisco José de Caldas for their academic support.

REFERENCES

[1] IEEE-CIS Fraud Detection Competition. Kaggle, 2025.
[2] Vesta Corporation, "Fraud Prevention Technologies in E-Commerce."
[3] Breiman, L. "Random Forests." Machine Learning, 2001.
[4] Saltelli, A. et al., "Sensitivity Analysis in Practice," Wiley, 2004.
[5] Gleick, J., *Chaos: Making a New Science*. Penguin Books, 1987.