Iteration #04 Report

Fraud Detection in Finance (IEEE-CIS Dataset)

Course: Essentials of Data Science

Team Members: Prajakta Avachat, Jay Bhuva

Submission Date: November 10, 2025

---

1 Dataset Description

Dataset Link:
IEEE-CIS Fraud Detection Dataset – Kaggle

Description:
The IEEE-CIS Fraud Detection dataset contains millions of anonymized online transaction records used for classifying fraudulent versus legitimate payments. It consists of two main tables, `transaction` and `identity`, joined on the key `TransactionID`. These files include variables such as transaction amount, product code, payment card type, device and browser information, and identity metadata.

Relevance and Suitability:
This dataset was selected because it aligns directly with our research objective: detecting fraudulent financial transactions using machine learning. Its large scale, diverse feature set, and real-world characteristics make it ideal for building predictive models. The combination of categorical and numerical data supports deep exploratory analysis, advanced feature engineering, and robust model training, which directly contributes to our system goal of improving fraud detection accuracy.

2 Tools and Methodologies

Tools and Libraries:

- **Programming Language:** Python 3.x
- **Libraries:** pandas, numpy, matplotlib, seaborn, scikit-learn, xgboost
- **Development Environment:** Jupyter Notebook / Google Colab
- **Documentation and Version Control:** Overleaf for reporting, GitHub for source code management

Methodology:
Our workflow follows the data science lifecycle — exploratory data analysis (EDA), preprocessing, feature selection, model training, and evaluation. We plan to train and compare multiple algorithms including Logistic Regression, Random Forest, and XGBoost, followed by advanced gradient-boosting models such as LightGBM and CatBoost. Imbalanced data handling techniques such as SMOTE and undersampling will be implemented. Evaluation will focus on metrics including Accuracy, Precision, Recall, F1-score, and ROC-AUC.

Justification:
Python offers a mature ecosystem for data preprocessing and machine learning. Libraries like `scikit-learn` and `xgboost` provide efficient, scalable implementations that suit the dataset's size and complexity. GitHub ensures reproducibility and collaborative workflow, while Overleaf enables professional documentation of progress.

3 Preliminary Timeline

Week	Task / Milestone	Deliverables
Week 1	Dataset exploration and EDA	Missing value report, fraud rate summary
Week 2	Data preprocessing and feature engineering	Clean dataset, feature importance charts
Week 3	Model training (Logistic Regression, Random Forest)	Baseline metrics (Accuracy, F1, ROC-AUC)
Week 4	Advanced modeling (XGBoost, LightGBM, CatBoost)	Optimized model performance report
Week 5	Model evaluation, visualization, and comparison	Evaluation plots and final summary table
Week 6	Documentation and final submission	Complete Overleaf report and updated GitHub repository

4 Team Member Contributions

Team Member	Role	Contributions
Prajakta Avachat	Data Preprocessing & Visualization	Led data cleaning, handled missing values, created EDA visualizations, and summarized key trends and distributions.
Jay Bhuva	Modeling & Evaluation	Developed ML pipeline, trained baseline and ensemble models, performed evaluation and documentation.

Collaboration:

Teamwork has been managed through GitHub with structured branches (`main` and `dev`). Regular commits, code reviews, and synchronization meetings ensured consistency. Prajakta focused primarily on data analysis and visualization, while Jay implemented machine learning models and evaluation metrics. Going forward, both members will collaborate on tuning, interpretability analysis, and report preparation.

5 Progress and Next Steps

Progress So Far:

- Successfully merged `train_transaction.csv` and `train_identity.csv` using `TransactionID`.
- Conducted full exploratory data analysis (EDA), including missing value analysis, feature distributions, and fraud rate visualization.
- Identified top correlated numerical variables with the target variable `isFraud`.
- Created preprocessing scripts for feature cleaning and one-hot encoding of categorical attributes.
- Generated preliminary evaluation metrics using Logistic Regression, Random Forest, and XGBoost baseline models.

Current Findings:
EDA revealed that the dataset contains substantial imbalance (fraud rate around 3.5%), necessitating resampling methods. Early model tests showed that ensemble-based approaches like Random Forest and XGBoost outperform simpler linear models in recall and ROC-AUC.

Next Steps:

- Implement advanced ML techniques such as LightGBM, CatBoost, and hyperparameter optimization.
- Explore feature selection using mutual information and SHAP importance.
- Improve handling of missing and imbalanced data using SMOTE and domain-specific encoding.
- Build detailed visualization dashboards to compare model performance metrics.
- Finalize evaluation reports and documentation for submission.

Planned Adjustments:
Based on EDA results, we will refine our feature selection and consider ensemble stacking for better generalization. The upcoming iterations will emphasize advanced model tuning, interpretability, and visualization integration.

6 Submission Details

Overleaf Project:
https://www.overleaf.com/read/rffbxngtfjqh#1d9f36

GitHub Repository:
https://github.com/jaybhuvaa/FraudBusters

All notebooks, scripts, visualizations, and updated project materials are maintained in the GitHub repository.

End of Report