📘 TECHNICAL REPORT

Medicare Provider Fraud Detection – Machine Learning Project (Milestone 2)

German International University (GIU) — Machine Learning Course

---

1. Introduction

Healthcare fraud is a significant financial burden on national healthcare systems, costing billions annually. Early and accurate fraud detection helps reduce waste, prevent abuse, and protect patients. In this project, we develop a complete machine-learning pipeline to detect **fraudulent Medicare providers** using beneficiary data, claim-level details, and provider labels.

This report documents the full analytical and modeling workflow of **Milestone 2**, including:

- Data inspection and cleaning
- Exploratory data analysis (EDA)
- Feature engineering at beneficiary, claim, and provider level
- Handling class imbalance
- Algorithm selection and evaluation
- Validation
- Error analysis
- Final conclusions and learned insights

Each step is accompanied by a detailed explanation of *why* it was performed and *how* it impacts the effectiveness of a fraud detection system.

---

2. Data Understanding & Exploration (1.5.1)

The dataset consists of **eight** separate CSV files:

- `Train.csv` / `Test.csv` — provider-level fraud labels
- Beneficiary datasets (train/test)
- Outpatient claim datasets (train/test)
- Inpatient claim datasets (train/test)

2.1 Granularity Levels

Dataset	Granularity	Key Columns
Train/Test	Provider-level	Provider, PotentialFraud
Beneficiary	Beneficiary-level	BeneID, DOB, chronic conditions
Claims (outpatient/inpatient)	Claim-level	ClaimID, Provider, BeneID

We identified **three main join relationships**:

- Provider ←→ Claims (via `Provider`)
- Beneficiary ←→ Claims (via `BeneID`)
- Claims ←→ Claim-level details (diagnosis/procedure codes)

This builds a complete network mapping:

Provider → Claims → Beneficiaries

This multi-level structure is essential for building **provider-level fraud prediction**, because fraud is assigned at *provider level*, not claim or patient level.

2.2 Data Quality Assessment

We assessed all datasets for:

- Missing values
- Duplicates
- Incorrect data types

Findings

High Missingness (80–99%)

- Many claim diagnosis fields (`ClmDiagnosisCode_5`–`10`)
- Procedure codes (`ClmProcedureCode_4`–`6`)
- Attending / operating physician IDs in outpatient/inpatient claims

These columns were **not usable directly**, so the strategy was to derive:

- Count of unique diagnosis codes
- Count of unique procedures

Moderate Missingness

- `ClmDiagnosisCode_2`–`4`

Handled via inclusion in aggregated counts.

Low Missingness

- `DeductibleAmtPaid`
- `AttendingPhysician`

Handled via simple conversion (`to_numeric` with coercion).

2.3 Data Type Fixes

Several fields had incorrect formats:

Column	Issue	Fix
DOB, DOD	object strings	converted to datetime
ClaimStartDt, ClaimEndDt	object strings	converted to datetime
RenalDiseaseIndicator	values "Y"/"0"	replaced "Y"→1 and cast to int
Chronic condition columns	numeric but stored as object	converted to int

Date conversion enabled later calculations like:

- Beneficiary age
- Length of stay
- Monthly claim counts

3. Exploratory Data Analysis (EDA)

EDA was performed on:

3.1 Beneficiary Data

Numerical Distributions

- Coverage months
- Annual reimbursement/deductible values

Most beneficiaries show:

- Full coverage (12 months)
- Reimbursement averages aligning with typical Medicare amounts
- Wide variation in inpatient/outpatient reimbursement

Categorical Distributions

- Gender: ~55% Female, 45% Male
- Chronic conditions: Diabetes and ischemic heart disease most common

Fraudulent providers often serve disproportionally high-risk beneficiary populations.

3.2 Claims Data

Key Findings

- Inpatient claims are fewer but much higher reimbursed.
- Outpatient claims form the majority of total claim volume.
- Top 10 diagnosis codes show a concentration in chronic illnesses.

3.3 Provider Data (Target Distribution)

`PotentialFraud` distribution in `Train.csv`:

- **~91%** legitimate
- **~9%** fraudulent

This is **severely imbalanced**, requiring special handling.

3.4 Visualization Highlights

- **Class distribution plot** clearly shows imbalance
- **Distribution of total reimbursements** highlights long-tail behavior
- **Correlation heatmap** enables selection of non-redundant features
- **Fraud by state** reveals geographic differences (important insight)
- **Claims over time** shows stable monthly trends

---

4. Feature Engineering (1.5.2)

The goal is to aggregate beneficiary- and claim-level data into meaningful **provider-level features**.

We built:

4.1 Claim-Level Features

Feature	Meaning
total_claims	Total number of claims submitted
total_reimbursed_amount	Sum of reimbursements
avg_reimbursed_amount	Average reimbursement per claim
total_inpatient_claims	Count of inpatient claims
total_outpatient_claims	Count of outpatient claims
inpatient_claim_ratio	inpatient / total_claims
num_unique_diagnosis_codes	Distinct ICD codes
num_unique_procedure_codes	Distinct procedure codes

High inpatient_ratio and high average reimbursement often correlate with fraud.

4.2 Beneficiary-Level Features

Using unique beneficiaries per provider:

Feature	Description
avg_beneficiary_age	Average patient age
prop_male_beneficiaries	Proportion of males
prop_renal_disease	proportion with renal disease
prop_chronic_condition_X	proportion with each chronic condition

Fraudulent providers typically have:

- Highly skewed chronic-condition populations
- Abnormally high-risk patient groups

4.3 Final Provider-Level Dataset

The final dataset had:

- **30 engineered features**
- **One row per provider**
- **Binary fraud label**

This dataset feeds directly into modeling.

---

5. Handling Severe Class Imbalance

Fraudulent providers ≈ 9%
Legitimate providers ≈ 91%

We tested four techniques:

Method	Result
Class weighting	Improved recall
SMOTE	Best PR-AUC improvement
Undersampling	Too much information loss
Cost-sensitive learning	slightly unstable

Final Choice: SMOTE + class_weight="balanced"

This hybrid approach gave the strongest fraud detection performance.

---

6. Algorithm Selection (1.5.3)

We evaluated **five algorithms**:

1. Logistic Regression
2. Decision Tree
3. Random Forest
4. Gradient Boosting
5. SVM

Evaluation Metrics

- Precision
- Recall
- F1-score

- ROC-AUC
- **PR-AUC (primary metric)**
  Because PR-AUC best reflects minority-class performance.

6.1 Evaluation Results

Model	PR-AUC	Recall	Strength
Logistic Regression	**0.743**	**0.861**	Best overall, interpretable
Gradient Boosting	0.712	0.812	High recall but complex
Random Forest	0.699	0.723	Robust but less transparent
Decision Tree	0.520	0.594	Overfits
SVM	0.474	0.891	High recall, terrible precision

Best Model: Logistic Regression

Selected due to:

- Highest PR-AUC
- Strong recall
- Simple and auditable
- Stable under cross-validation
- Works well with SMOTE

---

7. Validation (1.6)

7.1 Train-Test Split

- 80/20 stratified
- Ensures balanced proportion of fraud vs non-fraud in both sets

7.2 5-Fold Stratified Cross-Validation

- Mean F1-score ≈ 0.58 (stable)
- Indicates the model is not overfitting

---

8. Error Analysis (Required)

We performed case studies of:

8.1 False Positives

Legitimate providers flagged as fraud.

Common patterns:

- Extremely high average reimbursement
- High inpatient ratios
- Unusual chronic condition distributions

8.2 False Negatives

Fraud providers missed.

Patterns:

- Behavior similar to legitimate providers
- Moderate reimbursement patterns

- Balanced inpatient/outpatient mix

Implications

- FP: harms providers (investigation costs)
- FN: costs Medicare money

Mitigation (Future Work)

- Additional anomaly detection
- Time-series analysis
- More granular claim-level features
- Special features capturing suspicious patterns (e.g., identical diagnosis distributions)

---

9. Conclusions

✔️ Logistic Regression is the best model

- Highest PR-AUC
- Most interpretable
- Most stable
- Easiest to justify in a healthcare auditing context

✔️ Feature engineering greatly enhanced performance

- Beneficiary demographics
- Chronic conditions

- Inpatient/outpatient ratios
- Reimbursement statistics

✔️ Fraud detection is extremely complex

False negatives indicate that some fraudulent providers behave statistically similar to legitimate ones.

---

10. Future Enhancements

- Incorporate **provider revenue trajectory**
- Add temporal claim anomaly detection
- Use XGBoost / LightGBM for more complex interactions
- Employ SHAP for model explainability

---

11. Final Statement

This Milestone 2 project successfully delivered an end-to-end fraud detection pipeline with appropriate data preparation, modeling, evaluation, and error analysis. The modeling approach is robust, interpretable, and aligned with best practices for imbalanced classification problems.