German International University of Applied Sciences
Informatics and Computer Science

DataOrbit - Healthcare Provider Fraud Detection Project

Machine Learning (Winter 2025) - Project 2

Prepared by:
Ali Tharwat
Amr Khaled
Mostafa Ahmed
Lakshy Rupani

Supervised by:
Dr. Caroline Sabty
TA Nouran Khaled
TA Sandra Samuel
TA Sarah Hatem

Prepared for:
Centers for Medicare & Medicaid Services (CMS)

Executive Summary

Healthcare fraud imposes a significant burden on the U.S. healthcare system, with estimated annual losses exceeding $68 billion. The Centers for Medicare & Medicaid Services (CMS) contracted DataOrbit to develop a robust, data-driven solution to identify fraudulent healthcare providers. The objective was to move beyond traditional rule-based systems, which often fail to detect sophisticated fraud schemes such as upcoding, unbundling, and billing for services not rendered.

This report documents the end-to-end development of a machine learning pipeline designed to flag high-risk providers while minimizing false positives to preserve investigative resources.

Key Project Outcomes:

- **Data Integration:** Successfully merged beneficiary, inpatient, and outpatient data into a unified provider-level dataset comprising 5,410 providers and 558,211 claims.
- **Advanced Feature Engineering:** Developed 99 predictive features, including novel metrics for claim duration, diagnosis code diversity, and renal disease prevalence. Granular ICD-9 codes were mapped to 17 high-level clinical groups to reduce dimensionality.
- **Imbalance Mitigation:** Addressed the severe 9.35% fraud class imbalance by testing eight distinct strategies. While Random Under-Sampling achieved the highest recall (92.76%), it compromised precision. The final model utilized a balanced approach to optimize F1-score and Net Savings.
- **Model Selection:** After rigorous testing of Logistic Regression, Support Vector Machines (SVM), Random Forest, AdaBoost, and Gradient Boosting, the **Gradient Boosting Classifier** was selected as the champion model.
- **Performance:** The champion model achieved a Recall of **82.89%**, detecting the

vast majority of fraud cases, with an ROC-AUC of **0.9406**, indicating exceptional discriminatory power.

- **Financial Impact:** Based on a cost-benefit analysis (assuming $50,000 per missed fraud and $1,000 per investigation), the model is projected to generate **net savings of $4.76 million** on the test set alone.

The delivered solution provides a scalable, interpretable, and highly effective tool for prioritizing fraud investigations.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Project Background

The integrity of the Medicare system is constantly threatened by fraudulent activities. Traditional detection methods rely on static rules (e.g., checking for duplicate claims) which are reactive and easily circumvented by adaptive fraudsters. The DataOrbit team was tasked with creating a predictive model that uses historical claims data to identify behavioral patterns associated with fraud.

1.2 Problem Statement

The core challenge is a binary classification problem: predicting whether a healthcare provider is "Potential Fraud" or "Non-Fraud". This task is complicated by:

- **Class Imbalance:** Fraudulent providers constitute a small minority (approx. 1 in 10), making standard algorithms biased towards the majority class.
- **Data Complexity:** The raw data consists of transactional claims with high-cardinality categorical features (thousands of diagnosis codes).
- **Cost Sensitivity:** False Negatives (missing fraud) are extremely expensive due to lost funds, while False Positives (wrongly accusing a provider) incur investigation costs and reputational damage.

1.3 Project Objectives

1. **Data Unification:** Consolidate disparate datasets (Beneficiary, Inpatient, Outpatient) into a coherent analytical structure.
2. **Behavioral Profiling:** Engineer features that capture the "fingerprint" of fraud, such as abnormal claim durations or unusual diagnostic patterns.
3. **Predictive Modeling:** Train and validate machine learning models to classify providers.
4. **Business Value Analysis:** Quantify the financial impact of the model using realistic cost assumptions.

Chapter 2

Data Exploration and Cleaning

2.1 Dataset Overview

The analysis utilized the following datasets provided by CMS:

- **`Train_Beneficiarydata.csv` (138,556 records):** Contains patient demographics (DOB, Gender, Race), chronic conditions (Alzheimer's, Diabetes, etc.), and insurance coverage details.
- **`Train_Inpatientdata.csv` (40,474 records):** Claims for hospital admissions, including admission dates, discharge dates, and reimbursement amounts.
- **`Train_Outpatientdata.csv` (517,737 records):** Claims for clinic visits, consisting of procedure codes and physician IDs.
- **`Train_labels.csv` (5,410 records):** The target labels for providers.

2.2 Data Cleaning and Preprocessing

Extensive data cleaning was performed to ensure data quality before modeling.

2.2.1 Handling Missing Values

- **Procedure Codes:** Columns `ClmProcedureCode_6` contained 100% missing values in both inpatient and outpatient files and were dropped.

- **Deductible Amounts:** The `DeductibleAmtPaid` column had missing values. Domain knowledge dictates that Inpatient claims typically have a standard deductible, while Outpatient claims often do not. We imputed missing values based on the `IsInpatient` flag: $1,068 for Inpatient and $0 for Outpatient.
- **Physician IDs:** Missing values in `AttendingPhysician`, `OperatingPhysician`, and `OtherPhysician` were treated as a distinct category ("Unknown") rather than dropped, as the absence of a physician ID can be a signal in itself.

2.2.2 Date Logic Correction

A crucial data integrity issue was identified regarding `AdmissionDt` (Admission Date) and `ClaimStartDt` (Claim Start Date) in the inpatient data.

- **Findings:** In 32 cases, the `AdmissionDt` preceded the `ClaimStartDt`.
- **Implication:** Using `ClaimStartDt` would underestimate the length of stay.
- **Resolution:** We overwrote `ClaimStartDt` with the value of `AdmissionDt` to capture the true physical duration of the hospital stay. Following this, `AdmissionDt` was dropped to avoid multicollinearity.

2.2.3 Demographic Standardization

- **Gender:** Coded as 1 (Male) and 2 (Female). Transformed to a binary 0/1 variable (`IsMale`).
- **Race:** Codes (1, 2, 3, 5) were mapped to descriptive categories (White, Black, Asian/Pacific, Hispanic) and then one-hot encoded for analysis.
- **Renal Disease Indicator:** The column contained 'Y' for Yes and '0' for No. This was mapped to a standard binary 1/0 integer format.

2.3 Exploratory Data Analysis (EDA)

2.3.1 Class Distribution

The dataset is moderately imbalanced.

Table 2.1: Target Variable Distribution

Class	**Count**	**Percentage**
Non-Fraud (0)	3,433	90.65%
Fraud (1)	354	9.35%

2.3.2 Financial Patterns

Analysis of reimbursement amounts revealed that fraudulent providers tend to have higher average claim amounts and higher variability (standard deviation) in their claims. This suggests "upcoding" behavior where providers bill for more expensive services than were rendered.

2.3.3 Geographic Distribution

Using the State and County codes, we analyzed the geographical spread. While fraud was present across many regions, certain states exhibited higher concentrations of flagged providers.

Chapter 3

Feature Engineering

To transform raw transactional data into a format suitable for machine learning, we aggregated data from the claim level (558,211 rows) to the provider level (5,410 rows). This process generated 99 distinct features.

3.1 Feature Construction

3.1.1 Temporal Features

- **Claim Duration:** Calculated as `ClaimEndDt - ClaimStartDt`. We derived the Mean, Maximum, and Standard Deviation of claim duration for each provider. High variance in duration can indicate inconsistency typical of fraud.
- **IsPostDischargeBilling:** A critical binary flag was created for inpatient claims where the `ClaimEndDt` occurred *after* the `DischargeDt`. This is a logical impossibility for legitimate claims (billing for services after the patient has left). This feature proved highly predictive.
- **Age:** Patient age was calculated derived from `DOB` relative to a reference date of 2010-01-01.

3.1.2 Medical Code Grouping (Dimensionality Reduction)

The dataset contained thousands of unique ICD-9 diagnosis and procedure codes. Using these raw codes would result in a sparse matrix ("curse of dimensionality"). We mapped

these codes into 17 high-level clinical "Super Groups" to capture broader patterns.

- **Groups Created:** Circulatory System, Respiratory System, Digestive System, Injury & Poisoning, Musculoskeletal, Neoplasms, etc.
- **Aggregation:** For each provider, we calculated the count of claims belonging to each super group. This allows the model to detect if a provider specializes excessively in high-cost disease categories.

3.1.3 Operational Features

- **Unique Diagnosis Count:** The number of unique diagnosis codes submitted by a provider. Fraudulent providers often reuse the same set of codes (cloning claims).
- **Physician Network Size:** Count of unique `AttendingPhysician` and `OperatingPhysician` IDs associated with a provider.
- **Deductible Variability:** Standard deviation of `DeductibleAmtPaid`. Legitimate outpatient clinics usually have $0 deductibles; variations here can signal anomalies.

3.2 Aggregation Strategy

Since the target variable exists at the provider level, all features were aggregated using the following logic:

Table 3.1: Provider-Level Aggregation Logic

Feature Type	**Aggregation Functions**
Financial (Reimbursement)	Mean, Standard Deviation, Max
Temporal (Duration)	Mean, Max, Standard Deviation
Categorical (Codes, IDs)	Count Unique (Diversity)
Binary (Chronic Conditions)	Sum (Prevalence)

Chapter 4

Methodology and Modeling

4.1 Training Protocol

- **Data Split:** The data was split into Training (80%) and Testing (20%) sets using stratified sampling to maintain the 9.35% fraud ratio in both sets.
- **Cross-Validation:** We employed 5-Fold Stratified Cross-Validation during training to ensure model stability and prevent overfitting.
- **Scaling:** `StandardScaler` was applied to normalize numerical features, which is crucial for algorithms like SVM and Logistic Regression.

4.2 Handling Class Imbalance

Given the 1:10 imbalance, training on raw data would yield a model biased toward the majority class (Non-Fraud). We extensively tested eight different resampling strategies:

1. **No Resampling:** Using `class_weight='balanced'` to penalize misclassification of the minority class.
2. **SMOTE:** Synthetic Minority Over-sampling Technique.
3. **ADASYN:** Adaptive Synthetic sampling.
4. **Random Over-Sampling:** Duplicating minority samples.

5. **Random Under-Sampling:** Reducing majority samples to match the minority count.
6. **NearMiss:** An under-sampling technique based on distance.
7. **SMOTE + Tomek Links:** A hybrid approach.
8. **SMOTE + ENN:** Another hybrid approach.

4.3 Model Selection

We trained and tuned the following classifiers using GridSearchCV:

- **Logistic Regression:** A linear baseline.
- **Support Vector Machine (SVM):** For finding complex decision boundaries.
- **Random Forest:** An ensemble of decision trees to reduce variance.
- **AdaBoost:** A boosting algorithm focusing on hard-to-classify errors.
- **Gradient Boosting:** A powerful boosting method that optimizes a loss function.

Chapter 5

Results and Analysis

5.1 Imbalance Strategy Performance

The choice of imbalance strategy significantly impacted model behavior.

- **Random Under-Sampling** yielded the highest Recall (approx 92%) but suffered from very low Precision (below 33%), meaning it generated too many false alarms.
- **SMOTE** provided a middle ground but introduced some noise.
- **No Resampling (Class Weights)** provided the most balanced performance when combined with robust algorithms like Gradient Boosting.

5.2 Final Model Comparison

The table below summarizes the performance of the top three models on the unseen Test Set (1,623 providers).

Table 5.1: Test Set Performance Comparison

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC	PR-AUC
Gradient Boosting	0.9156	0.5316	**0.8289**	**0.6478**	**0.9406**	**0.6322**
AdaBoost	0.9199	0.5585	0.6908	0.6176	0.8509	0.5033
SVM	0.9162	0.5476	0.6053	0.5750	0.9107	0.6159

Analysis:

- **Gradient Boosting** is the clear winner. It achieved the highest Recall (82.89%), meaning it detected the most fraud cases. It also achieved the highest F1-Score and ROC-AUC.
- **AdaBoost** had slightly higher precision but significantly lower recall (missing roughly 30% of fraud cases).
- **SVM** performed well but lagged behind Gradient Boosting in detecting fraud (lower recall).

5.3 Confusion Matrix: Gradient Boosting

Table 5.2: Confusion Matrix (Gradient Boosting on Test Set)

	Predicted Non-Fraud	**Predicted Fraud**
Actual Non-Fraud	1360 (TN)	111 (FP)
Actual Fraud	26 (FN)	126 (TP)

Interpretation:

- **TP (126):** We successfully caught 126 fraudulent providers.
- **FN (26):** We missed 26 cases. This is an acceptable trade-off to avoid flagging too many innocents.
- **FP (111):** We flagged 111 legitimate providers. This represents a False Positive Rate of 7.55%, which is a manageable workload for investigators.

5.4 Feature Importance

The Gradient Boosting model identified the following as the top predictive features:

1. `ClaimDurationDays_max`: The maximum length of a claim.
2. `DiagnosisGroupCode_nunique`: Diversity of diagnosis codes.
3. `IsInpatient_std`: Variability in inpatient vs outpatient claims.

4. **`DeductibleAmtPaid_std:`** Variance in deductibles.

5. **`InscClaimAmtReimbursed_p99_5:`** The 99.5th percentile of reimbursement amounts (detecting high-value outliers).

Chapter 6

Financial Evaluation

To demonstrate business value, we conducted a cost-benefit analysis using the following assumptions:

- **Cost of Missed Fraud (FN):** $50,000 (Average loss per undetected fraudster).
- **Cost of Investigation (FP):** $1,000 (Administrative cost to audit a flagged provider).
- **Cost of Verification (TP):** $500 (Cost to confirm a true fraud).

6.1 Comparative Cost Analysis

We calculated the total estimated cost for each model on the test set.

Table 6.1: Financial Impact Analysis

Metric	SVM	AdaBoost	Gradient Boosting
Total Estimated Cost	$3,122,000	$2,485,500	**$1,474,000**
Cost per Case	$1,923	$1,531	**$908**
Fraud Losses Prevented	$4,600,000	$5,250,000	**$6,300,000**
Net Savings	$1,432,000	$2,712,000	**$4,763,000**

Conclusion: The Gradient Boosting model is not only the most accurate but also the most financially viable, generating nearly **$4.8 million in net savings** compared to a baseline of doing nothing.

Chapter 7

Conclusion and Recommendations

7.1 Summary

This project successfully developed a machine learning solution to detect healthcare provider fraud. By leveraging comprehensive feature engineering and advanced ensemble modeling, we achieved a system that detects **83% of fraud cases** with a high degree of reliability.

7.2 Recommendations

1. **Deploy Gradient Boosting:** We recommend deploying the Gradient Boosting classifier as the primary screening tool.
2. **Tiered Investigation:**
   - **Priority 1 (Prob > 0.7):** Immediate audit.
   - **Priority 2 (Prob 0.4 - 0.7):** Automated review and flag for monitoring.
3. **Human-in-the-Loop:** The model should assist, not replace, investigators. Feedback from investigations should be used to retrain the model quarterly.
4. **Focus on "Clone" Providers:** High feature importance for unique diagnosis counts suggests that "cloning" claims (reusing the same diagnosis codes) is a major fraud signal. Investigators should specifically look for this pattern.

7.3 Future Work

- **Network Analysis:** Incorporate graph theory to detect rings of providers sharing the same beneficiaries.
- **Temporal Analysis:** Use Time-Series forecasting to detect sudden spikes in billing volume.
- **Unsupervised Learning:** Implement anomaly detection (e.g., Isolation Forest) to catch new, unknown fraud schemes that supervised models might miss.