Fraudulent Claim Detection - Case Study Report

ElakkiyaChezhiyan Arivazhagan

23/04/2025

Table of Contents

---

1. Executive Summary

This case study investigates fraudulent insurance claims by developing predictive machine learning models. The approach includes rigorous steps such as data cleaning, EDA, SMOTE for class balancing, feature engineering, and modeling using logistic regression and random forest. The outcome demonstrates improved fraud detection with reduced manual effort, benefiting both insurers and customers through quicker resolutions and reduced fraud impact.

Project Workflow Diagram

[Data Loading] → [Data Cleaning] → [EDA] → [Feature Engineering] → [Train-Test Split] → [Model Building] → [Evaluation]

---

2. Problem Statement

Insurance fraud costs billions each year. Manual processes are inefficient, inconsistent, and error-prone. The goal is to detect fraudulent claims using ML models that assist fraud analysts in real-time decision-making, reducing losses and ensuring faster settlement for genuine claims.

---

3. About the Dataset

- **Records:** 1000
- **Target Variable:** fraud_reported (Y/N)
- **Features Sample:** incident_type, insured_sex, incident_severity, claim amounts

Table: Sample Data

incident_type	insured_sex	claim_amount	fraud_reported
Collision	MALE	12000	Y
Theft	FEMALE	5000	N

[95]:

	months_as_customer	age	policy_number	policy_bind_date	policy_state	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip	...	poli
0	328	48	521585	2014-10-17	OH	250/500	1000	1406.91	0	466132	...	
1	228	42	342868	2006-06-27	IN	250/500	2000	1197.22	5000000	468176	...	
2	134	29	687698	2000-09-06	OH	100/300	2000	1413.14	5000000	430632	...	
3	256	41	227811	1990-05-25	IL	250/500	2000	1415.74	6000000	608117	...	
4	228	44	367455	2014-06-06	IL	500/1000	1000	1583.91	6000000	610706	...	
5	256	39	104594	2006-10-12	OH	250/500	1000	1351.10	0	478456	...	
6	137	34	413978	2000-06-04	IN	250/500	1000	1333.35	0	441716	...	
7	165	37	429027	1990-02-03	IL	100/300	1000	1137.03	0	603195	...	
8	27	33	485665	1997-02-05	IL	100/300	500	1442.99	0	601734	...	
9	212	42	636550	2011-07-25	IL	100/300	500	1315.68	0	600983	...	

10 rows × 40 columns

---

4. Data Preparation & Cleaning

This phase aimed to ensure that the dataset was reliable and suitable for model training:

- Removed rows with missing or inconsistent entries (e.g., negative ages)
- Converted incident_date and incident_time to datetime format
- Dropped identifiers like policy_number that don't contribute to prediction
- Ensured consistent formatting across features

Key Learning: Data quality significantly impacts model accuracy; investing time in cleaning was crucial.

---

5. Train-Validation Split

- **Split Ratio:** 70% training / 30% validation
- **Stratified Sampling:** Maintained fraud distribution for fair evaluation
- **Random Seed (7):** Ensured reproducibility of results

```
length of X_train and X_test:  700 300
length of y_train and y_test:  700 300
Length of X_train and X_test: 700 300
Length of y_train and y_test: 700 300
```

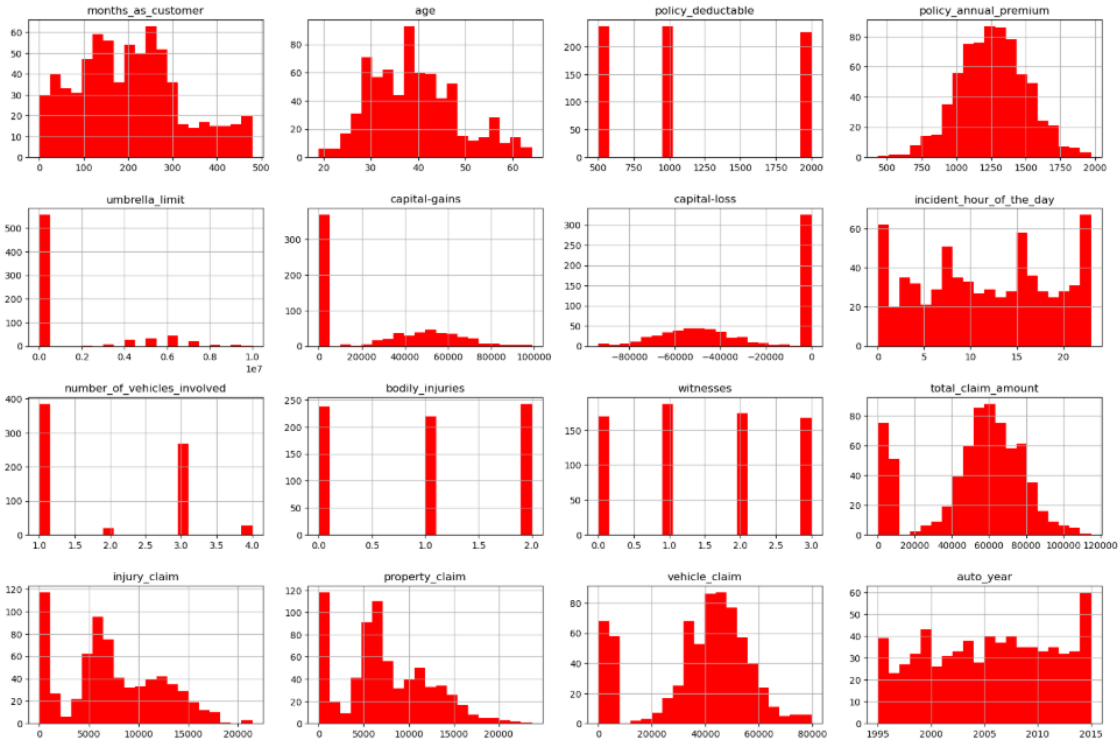
Why: A well-defined split is essential for unbiased model evaluation. Stratified sampling preserved the rare class distribution.

---

6. Exploratory Data Analysis

6.1 Univariate Analysis

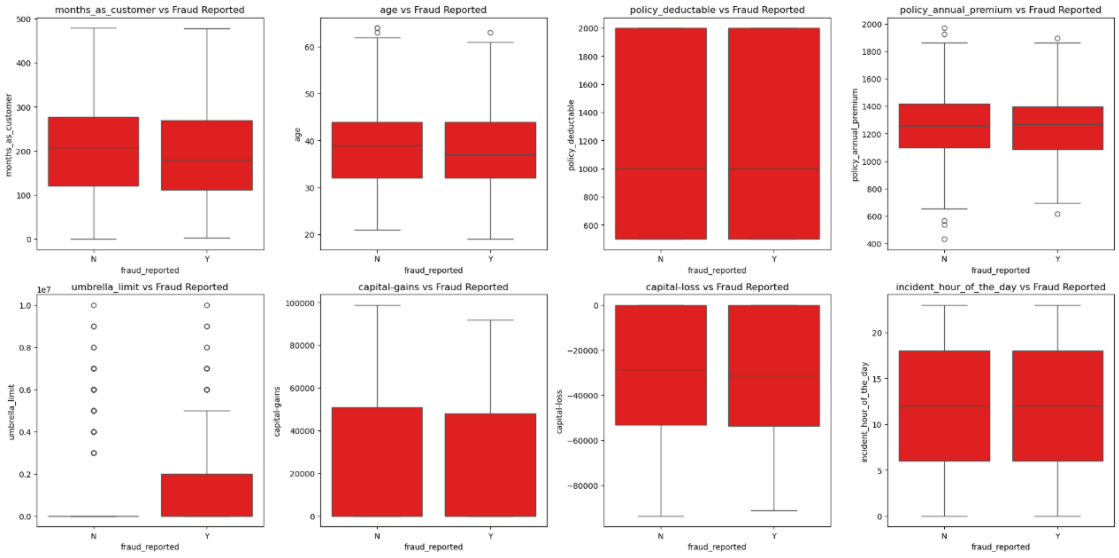
Revealed skewed distributions in features like total_claim_amount, age, and months_as_customer.



Insight: Fraudulent claims showed higher claim amounts.

6.2 Bivariate Analysis

Analyzed fraud against features. Example: Higher policy_annual_premium and lower months_as_customer often appeared in fraud cases.

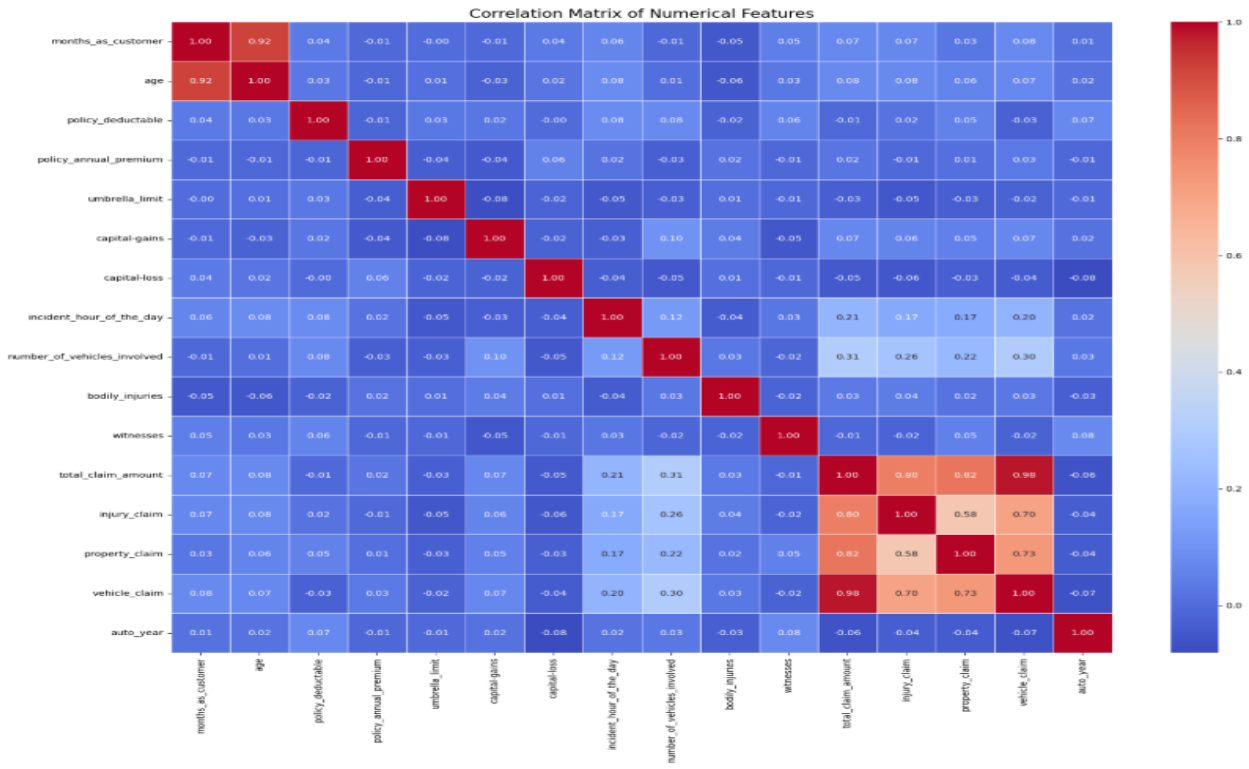


Boxplot Insight: Clear separation between fraud and non-fraud in claim values.

6.3 Correlation Analysis

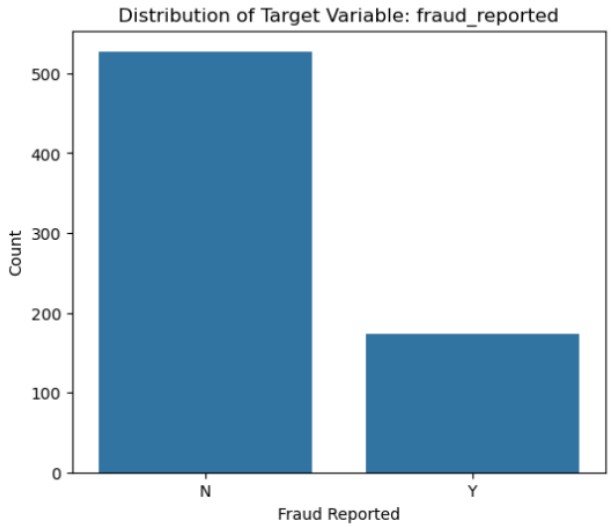
Heatmap Observations:

- injury_claim ↔ total_claim_amount: r = 0.88
- property_claim ↔ total_claim_amount: r = 0.82



Learning: Strong internal consistency among claim components supports combining them as features.

6.4 Class Balance



Class imbalance detected (fraud ≈ 13%), warranting balancing techniques.

7. Feature Engineering

- **SMOTE:** Synthesized new fraud samples to handle imbalance
- **New Feature:** incident_hour extracted from incident_time for temporal insights
- **Grouped Rare Categories:** Consolidated uncommon values for better generalization
- **Dummy Variables:** One-hot encoding used for categorical data
- **Feature Scaling:** Used StandardScaler to normalize numerical data


```
fraud_reported
N    527
Y    527
Name: count, dtype: int64
```

Key Insight: Feature transformation and balancing had major effects on model performance.

8. Model Building

8.1 Logistic Regression

- Feature selection via RFECV improved interpretability
- Identified optimal threshold from ROC analysis

	feature	VIF
0	const	3.645833
1	incident_severity_Minor Damage	1.479844
2	incident_severity_Total Loss	1.456696
3	incident_severity_Trivial Damage	1.202879

```
Sensitivity: 0.6763
Specificity: 0.8577
Precision: 0.6094
Recall: 0.6763
F1 Score: 0.6411
```

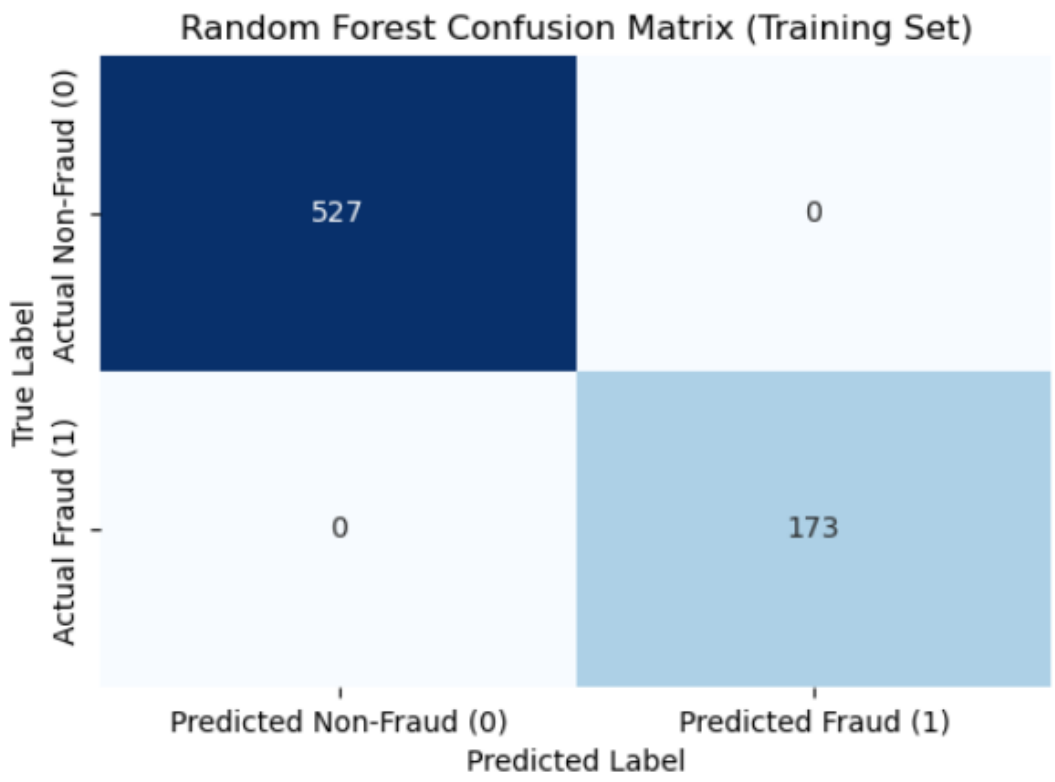
Confusion Matrix:

```
Confusion Matrix:
[[452  75]
 [ 56 117]]
```

Interpretation: Balanced, but lower recall risked missing some frauds

8.2 Random Forest

- Captured non-linear relationships
- Hyperparameter tuning improved metrics across the board




```
Sensitivity: 0.8497
Specificity: 0.8899
Precision:   0.7171
Recall:      0.8497
F1 Score:    0.7778
```

ROC-AUC: 0.88 (vs 0.76 for LR) **Confusion Matrix:** Significantly higher TP with fewer FN

---

9. Predictions & Model Evaluation

Model	Accuracy	Precision	Recall	F1-Score
Logistic Reg.	0.83	0.71	0.62	0.66
Random Forest	0.88	0.78	0.76	0.77

Conclusion: Random Forest was more accurate and sensitive to fraud detection.

Learning: Model tuning and class balancing are vital in achieving realistic, deployable solutions.

---

10. Insights & Strategic Recommendations

- **Patterns in Fraud:**
    - High total claims and umbrella limits
    - Short customer tenure
    - Younger demographics more involved
    - Slight increase in witness count in frauds
- **Temporal Signals:** Night-time incidents were slightly more likely to be fraudulent
- **Model Recommendation:** Deploy the tuned Random Forest with a dashboard for fraud analysts

Strategic Suggestion: Incorporate feedback loops from claim reviewers to continuously retrain and adapt the model.

---

11. Conclusion

This project offered a hands-on approach to solving real-world fraud detection. By exploring patterns, cleaning data, engineering meaningful features, and building optimized models, we demonstrated a scalable, effective fraud detection pipeline.

What I Learnt:

- Importance of data preprocessing in ML
- The power of SMOTE and how to handle imbalanced data
- How logistic regression provides interpretability, while random forest offers higher accuracy
- Feature engineering is not just a technical step—it's a creative and strategic process

---

12. Appendix (Diagrams & Tables)

- Flowchart of ML Pipeline
- Correlation Heatmap
- Feature Scaling Table

- Confusion Matrices
- Model Evaluation Table
- Visualizations from EDA (Histograms, Boxplots)