FRAUDULENT CLAIM DETECTION – REPORT

Group Members:
- Anish Dhondi
- Ankit Shrivastava

🔎 1. Problem Statement

Global Insure faces significant losses due to fraudulent insurance claims. The existing manual review process is inefficient and expensive. This project aims to build a machine learning model to identify potentially fraudulent claims early, enabling faster decisions and reduced financial risk.

---

2. Methodology Overview

The solution followed a structured 8-step process:

1. **Data Preparation & Cleaning**
2. **Train-Validation Split (70:30)**
3. **Exploratory Data Analysis (EDA)**
4. **Feature Engineering**
5. **Model Building (Logistic Regression & Random Forest)**
6. **Cutoff Optimization**
7. **Validation Evaluation**
8. **Final Recommendation**

---

3. Data Cleaning Assumptions

- Column `_c39` was dropped as it had 100% NULL values.
- 91 Rows with missing `authorities_contacted` values or NULL values were removed.
- `umbrella_limit` values with negatives (one row with value = -1,000,000) were removed as negative insurance coverrage did not make sense.
- `insured_zip` was dropped after realizing it offered very little value as the incident_city column is already there. Also with so many unique values, if considered, `insured_zip` column would have made hundreds of dummy variables and confusing the analysis.
- `incident_location` was dropped because all its values were unique and not contributing to model and analysis.
- `policy_number` was retained for a long time but later excluded from modeling since it's a unique ID and not useful for modelling.
- `policy_bind_date` and `incident_date` were dropped after deriving useful features (incident_month, incident_dayofweek, policy_bind_year) from them.

---

4. EDA & Feature Engineering Assumptions

- `capital-gains` and `capital-loss` dropped due to low correlation with fraud.
- Rare values in `insured_hobbies` and `insured_occupation` grouped into "Other.
- Created new features: `policy_bind_year`, `incident_month`, `incident_dayofweek`, `injury_ratio`, `property_ratio`, `vehicle_ratio`.
- Dummy variables created for categorical columns
- MinMax scaling applied to numerical features.
- Class imbalance handled using `RandomOverSampler`.

---

5. Key Insights from EDA

CATEGORICAL VARIABLES

Category	Feature	Value(s)	Fraud Likelihood (%)	Interpretation
High Fraud Likelihood	insured_hobbies	chess, cross-fit	80.6%, 76.0%	Very high fraud rates; strong indicators of possible fraud
High Fraud Likelihood	incident_severity	Major Damage	58.40%	Claims reporting major damage are highly suspicious
High Fraud Likelihood	auto_model	Silverado, ML350, C300	53.8%, 46.7%, 46.2%	Certain high-end models show strong fraud patterns
High Fraud Likelihood	insured_education	MD, JD, PhD	33.3%, 29.3%, 29.2%	Higher degrees correlate with increased fraud rates
High Fraud Likelihood	insured_occupation	farming-fishing, exec-managerial	38.7%, 37.0%	Fraud more likely in specific occupations
High Fraud Likelihood	auto_make	Mercedes, Ford, Audi	39.6%, 35.0%, 33.3%	Certain car brands see higher fraud frequencies
High Fraud Likelihood	authorities_contacted	Other, Fire	33.6%, 29.1%	Unusual authority contact types linked with more fraud
High Fraud Likelihood	incident_state	OH, SC	50.0%, 32.7%	Certain states report significantly more fraud
High Fraud Likelihood	collision_type	Rear, Front	30.5%, 28.9%	Rear and front collisions are more associated with fraud
High Fraud Likelihood	insured_sex	MALE	30.20%	Male policyholders show a higher fraud ratio

NUMERICAL VARIABLES

Feature	Insight	Indicator Strength
Total Claim Amount	Higher in fraud cases (62K vs. 56K); strong and consistent difference.	Strong. Should be kept.
Injury Claim	Higher in fraud (8.5K vs. 8.0K); relevant and consistent.	Strong. Should be kept.
Property Claim	Higher in fraud (8.6K vs. 7.7K); noticeable impact.	Strong. Should be kept.
Vehicle Claim	Significantly higher in fraud (45K vs. 40K); key variable.	Strong. Should be kept.
Witnesses	Slightly more in fraud cases; may help support claims analysis.	Moderately Strong. Should be kept.
Bodily Injuries	Slight increase in fraud (1.07 vs. 1.01); mild	Moderately Strong. Should be kept.
Months as Customer	Slightly higher for fraud (212 vs. 200); trend is weak.	Moderately Strong. Should be kept.
Age	Almost equal in both groups (~39); not strong	Weak. Should be removed.
Policy Number	Randomly distributed; no signal expected.	Weak. Should be removed.
Policy Deductible	Almost identical across both groups.	Weak. Should be removed.
Policy Annual Premium	Very similar between groups (~1250); no clear difference	Weak. Should be removed.
Umbrella Limit	Mostly zero; heavy skew and no meaningful difference.	Weak. Should be removed.
Capital Gains	Similar means; no clear difference.	Weak. Should be removed.
Capital Loss	Very similar and heavily negative; not helpful.	Weak. Should be removed.
Incident Hour of the Day	Slightly earlier in fraud cases; slight pattern.	Moderately Strong. Should be kept.
Number of Vehicles Involved	Slightly higher in fraud; low variability overall.	Moderately Strong. Should be kept.
Auto Year	Almost identical means (2005); no usable variance.	Weak. Should be removed.

6. Model Building & Selection

☑ Logistic Regression

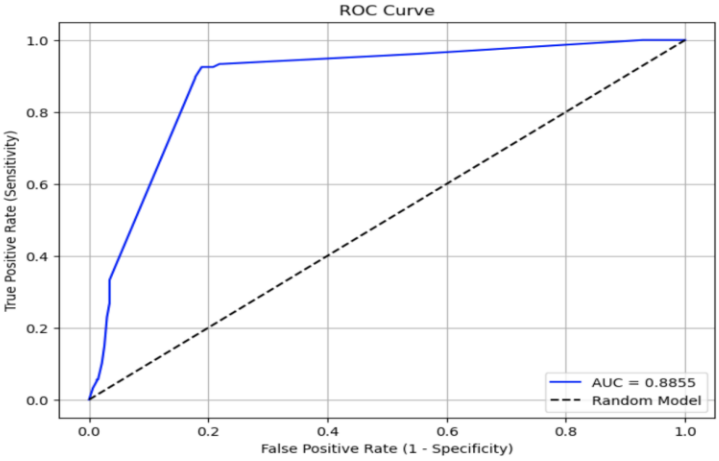
- **Feature Selection:** RFECV reduced variables to 7 final features:
    - `insured_hobbies_chess`
    - `insured_hobbies_cross-fit`
    - `insured_hobbies_video-games`
    - `incident_severity_Minor Damage`
    - `incident_severity_Total Loss`
    - `incident_severity_Trivial Damage`
    - `auto_model_92x`
- **Cutoff selected:** 0.4 based on sensitivity-specificity trade-off.
- **Scaling:** Applied MinMax scaler.

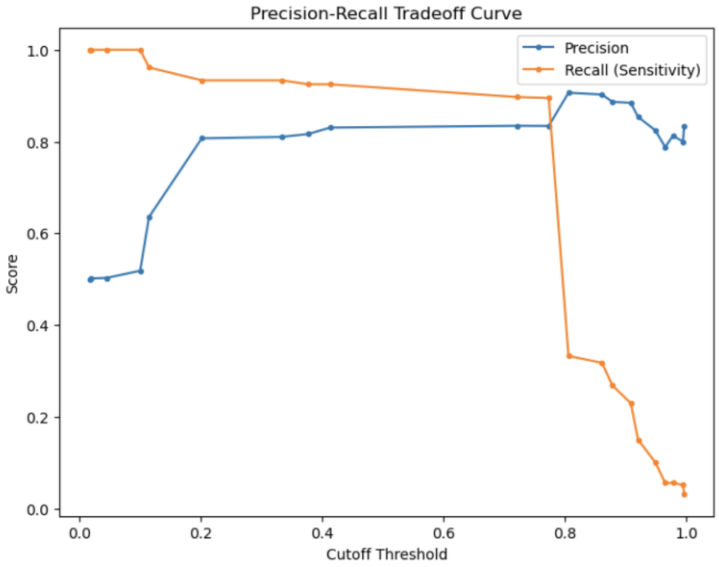
Comparison Between Initial and Optimal Cutoff probabilities

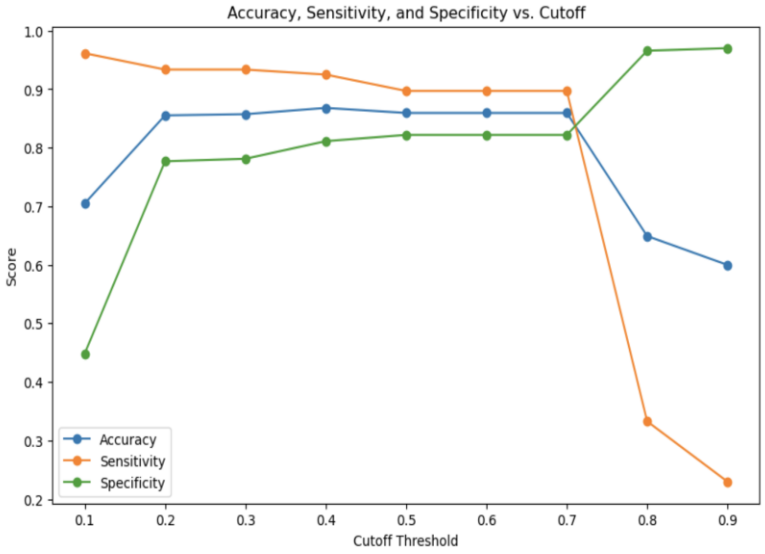
Metric	Cutoff = 0.5 (Initial)	Cutoff = 0.4 (Optimized)
Accuracy	0.8594	0.8680
Sensitivity	0.8970	0.9249
Specificity	0.8219	0.8112
Precision	0.8343	0.8304
F1 Score	0.8645	0.8751

Why is optimal cutoff probability = 0.4?

- Balances **high sensitivity (0.9249)** with good specificity (0.8112); better fraud detection than default 0.5 cutoff.
- **ROC Curve -**AUC = **0.8855**; steep rise near Y-axis shows strong performance even at lower thresholds.
- **Cutoff vs Metrics Plot -**At **0.4**, sensitivity and specificity are well-balanced; recall drops sharply beyond 0.7
- **Precision-Recall Curve -**At 0.4, both **precision (0.8304)** and **recall (0.9249)** are high ideal for minimizing false negatives.


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate (1 - Specificity)
AUC = 0.8855
Random Model


Precision-Recall Tradeoff Curve
Precision
Recall (Sensitivity)
Score
Cutoff Threshold


Accuracy, Sensitivity, and Specificity vs. Cutoff
Accuracy
Sensitivity
Specificity
Score
Cutoff Threshold

☑ Random Forest

- Initial results showed very high overfitting and by using Gridsearch CV we got a confirmation that hyperparameter tuning is required
- Gridsearch CV results

```
Cross-Validation Scores:
[0.89839572 0.9144385 0.92473118 0.89247312 0.93548387]
Mean CV Accuracy: 0.9131
```

- After hyperparameter tuning we got these results -
  - `max_depth`: 10
  - `min_samples_leaf`: 1
  - `min_samples_split`: 2
  - `n_estimators`: 200
  - Best Cross-Validation Accuracy: 0.9034

- Results of the model on training data after Hyperparameter tuning

Metric	Before Tuning	After Tuning
Accuracy	1	0.9796
Precision	1	0.9627
Recall	1	0.9979
Sensitivity	1	0.9979
Specificity	1	0.9614
F1 Score	1	0.9800

- Following features are finalized –

Feature List		
incident_severity_Minor Damage	incident_severity_Total Loss	injury_ratio
total_claim_amount	vehicle_claim	vehicle_ratio
insured_hobbies_chess	policy_annual_premium	incident_dayofweek
property_claim	injury_claim	incident_month
months_as_customer	incident_hour_of_the_day	property_ratio
age	policy_bind_year	insured_hobbies_cross-fit
witnesses	bodily_injuries	

7. Model Performance (Validation Set)

Metric	Logistic Regression (cutoff=0.4)	Random Forest (tuned)
Accuracy	86.80%	83.20%
Precision	64.30%	66.30%
Recall	**87.50%**	73.60%
Specificity	82.60%	**86.60%**
F1 Score	**74.10%**	69.70%

8. Final Recommendation

We recommend deploying the **Logistic Regression model (cutoff = 0.4)** due to:

- Higher recall (captures more fraudulent claims)
- Simpler, interpretable model
- Balanced trade-off between sensitivity and specificity

The model can be used to **flag high-risk claims for manual review**, improving fraud detection rates and reducing financial loss.

Q1: How can we analyse historical claim data to detect patterns that indicate fraudulent claims?

We used **Logistic Regression** and **Random Forest** to analyze labeled historical claim data.

Key steps included:

- Selecting **important features** that contribute most to fraud detection.
- Applying **resampling techniques** to address class imbalance and ensure fair model training.
- Measuring performance using **recall, precision, and F1 score** to assess fraud detection accuracy.

By comparing both models' responses to past claims, we identified **features** which correlate with fraudulent activity. This helped refine fraud detection

Q2: Which features are the most predictive of fraudulent behaviour?

From model training and importance analysis of the Logistic Regression model, the most predictive features included:

- `insured_hobbies_chess`
- `insured_hobbies_cross-fit`
- `insured_hobbies_video-games`
- `incident_severity_Minor Damage`
- `incident_severity_Total Loss`
- `incident_severity_Trivial Damage`
- `Auto_model_92x`

Q3: Based on past data, can we predict the likelihood of fraud for an incoming claim?

Yes, using historical insurance claim data, we trained models that can estimate the probability of a claim being fraudulent. Both the Logistic Regression and Random Forest models give us a probability score, which we can use to decide whether a new claim is likely fraud or not, based on a selected threshold.

In our evaluation, recall (or sensitivity) was important, since in fraud detection it's more critical to catch as many fraud cases as possible, even if it means flagging a few genuine claims by mistake.

The Logistic Regression model, when using a threshold of 0.4, gave us a recall of 87.5%, meaning it correctly identified most of the fraudulent claims. On the other hand, the Random Forest model

had a lower recall of 73.6%, which means it missed more fraud cases—even though it was a bit more precise and had higher specificity.

Considering this trade-off, we chose Logistic Regression for fraud prediction. It:

- Catches more fraud cases,
- Has a better balance between precision and recall (F1 score = 74.1%), and
- Is simpler and faster to deploy in real systems.

Q4: What insights can be drawn from the model that can help in improving the fraud detection process?

Logistic Regression works better for fraud detection since it catches more fraudulent cases with higher recall.

Random Forest is more precise, but it tends to miss fraud cases, making it less useful in situations where missing fraud is risky.

Adjusting the cutoff from 0.5 to 0.4 for Logistic Regression improves fraud detection by striking a balance between recall and specificity.

The most important features identified can be used to improve screening and fraud detection systems.