Vehicle Insurance Fraudulent Claim Detection Data Science Project

VISHAL RASTOGI

09/2022

PROBLEM STATEMENT

How can Data Science be leveraged to detect Vehicle Insurance Fraudulent claims to reduce losses on account of exaggerated and false claims about accidents, property damage and physical injury by developing machine learning models which can improve the accuracy as measured by F1 SCORE to above 0.8 in 6 months.

Scope:
- One CSV File
- Develop Models, only for accurately predicting false claims
- Only valid for data with same feature space

Constraints:
- Only one csv file
- Imbalanced data
- Only categorical features

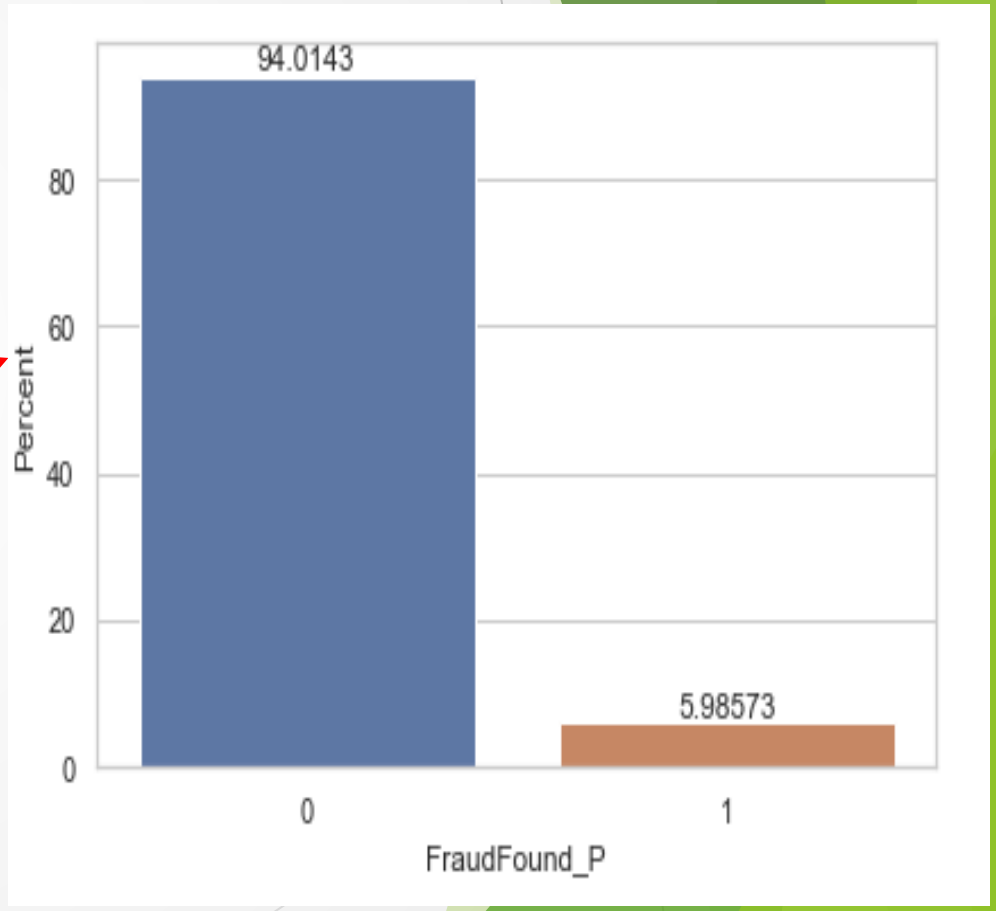
Stakeholders:
- Executive Management Team

Criteria for Success:
- Explore factors contributing to fraudulent claims
- Develop models to improve f1 score to 0.80

Key data sources
- Csv File https://www.kaggle.com/shivamb/vehicle-claim-fraud-detection
- Metadata

DATA WRANGLING

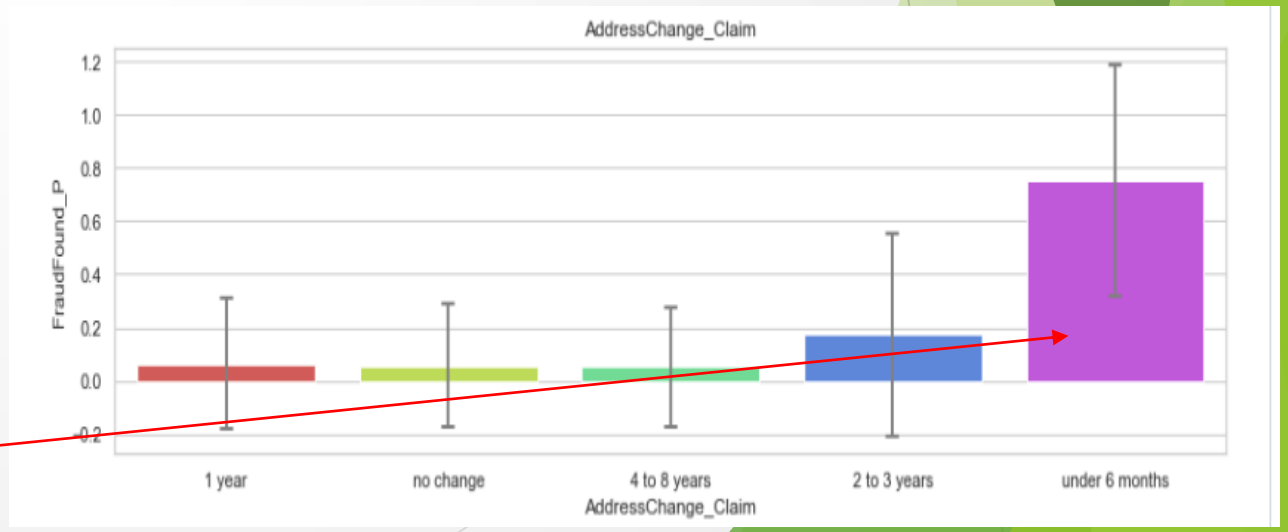
- Data checked for missing values, type of data and data discrepancies
- 15420 rows and 33 columns
- 320 instances when the age is recorded as 0
- All the columns except age in years is categorical
- 94.01 % of transactions are non-fraudulent and 5.99% transactions are fraudulent
- FraudFound_P selected as target feature
- Policy numbers looks randomly distributed and is just a serial number



Exploratory Data Analysis

Key findings

- Higher number of fraudulent transactions on Monday and Friday.
- Males contributed 88.62% to fraudulent claims
- **Utility vehicles have higher probability of fraudulent claims in vehicle categories**
- Rural areas have higher probability of fraudulent claims
- Highest probability of fraudulent claims is in higher end vehicles like Acura, BMW and Mercedes
- Age group 30-40 has higher involvement in fraudulent activities
- **Higher probability of fraud if address is changed within last 6 months**



Preprocessing and Training Data

Step 1

- Converted age column into 4 bins called age_bins
- Dropped ‘PolicyNumber’, ‘RepNumber’ and ‘Age’ columns
- Installed category_encoder, pycaret package,
- Dummy encoded all categorical features
- Used SMOTE , ADASYN, Random Over Sampler with PyCaret package
- None of the models and combination generalized well on unseen data

Dummy Encoding+SMOTE USING PYCARET

1. Training Results

	Model	Accuracy	AUC	Recall	Prec.	F1
rf	Random Forest Classifier	0.9665	0.9697	0.9342	0.9987	0.9654
et	Extra Trees Classifier	0.9656	0.9696	0.9333	0.9977	0.9644
gbc	Gradient Boosting Classifier	0.9651	0.9711	0.9311	0.9989	0.9638
ada	Ada Boost Classifier	0.9649	0.9702	0.9309	0.9987	0.9636
lightgbm	Light Gradient Boosting Machine	0.9641	0.9690	0.9304	0.9978	0.9629
qda	Quadratic Discriminant Analysis	0.9638	0.9727	0.9276	1.0000	0.9624
nb	Naive Bayes	0.9637	0.9730	0.9274	1.0000	0.9623
knn	K Neighbors Classifier	0.9634	0.9668	0.9301	0.9965	0.9622
lr	Logistic Regression	0.9626	0.9651	0.9251	1.0000	0.9611

2. Validation Results

```
predunseenlgbm=predict_model(tuned_smote_rf, data=Test)
```

	Model	Accuracy	AUC	Recall	Prec.	F1
0	Random Forest Classifier	0.9403	0.4983	0.0000	0.0000	0.0000

This model did not generalize at all on unseen data

Preprocessing and Training Data

Step 2

1. Weight of Evidence Encoder with Logistic Regression Class Weight Optimization

- Train Test Split with Stratified target column.
- Used PyCaret for optimized Logistic Regression Model

```
LogisticRegression(C=9.921, class_weight='balanced', dual=False,
                   fit_intercept=True, intercept_scaling=1, l1_ratio=None,
                   max_iter=1000, multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=1122, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

- Optimized class weight for f1 score using grid search CV:

{'class_weight': {0: 0.1293467336683417, 1: 0.8706532663316583}}

Precision, Recall ,F1 Scores

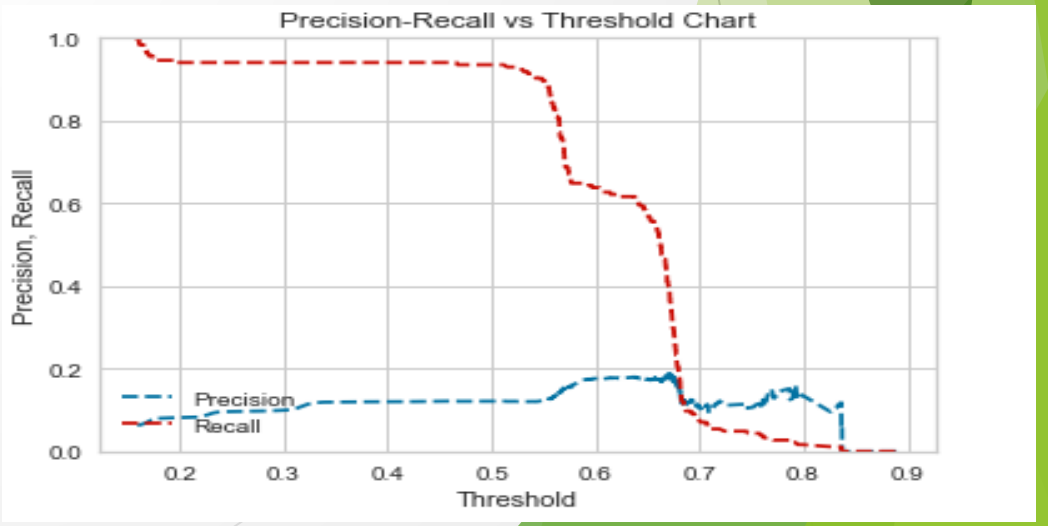
Train

	precision	recall	f1-score	support
0	0.96	0.87	0.92	11598
1	0.20	0.50	0.28	738

Test

	precision	recall	f1-score	support
0	0.96	0.86	0.90	2899
1	0.15	0.38	0.21	185

Precision Recall Threshold Chart



Preprocessing and Training Data

Step 2

1. CatBoost Encoder with Logistic Regression

Class Weight Optimization

- Train Test Split with Stratified target column.
- Used PyCaret for optimized Logistic Regression Model

```
LogisticRegression(C=9.921, class_weight='balanced', dual=False,
                   fit_intercept=True, intercept_scaling=1, l1_ratio=None,
                   max_iter=1000, multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=1122, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

- Optimized class weight for f1 score using grid search CV:

{'class_weight': {0: 0.06964824120603015, 1: 0.9303517587939698}}

Precision, Recall ,F1 Scores

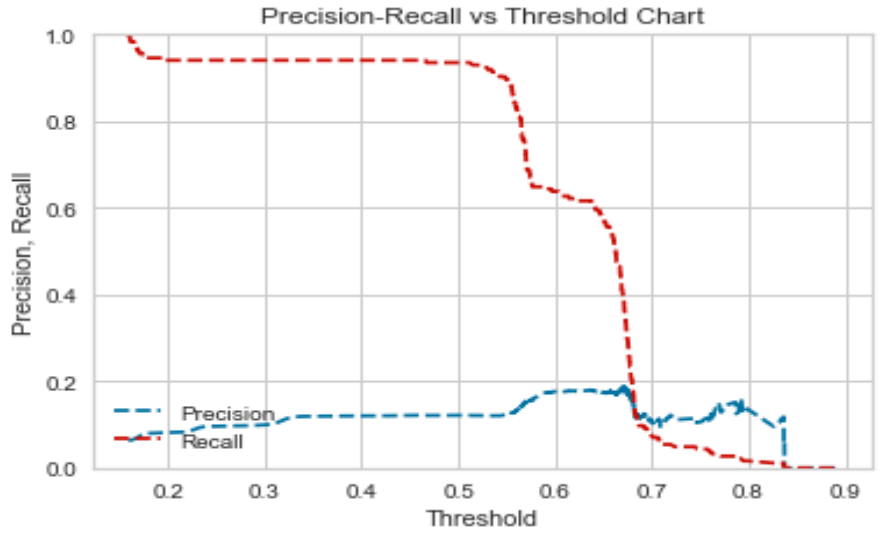
Train

	precision	recall	f1-score	support
0	0.99	0.64	0.77	11598
1	0.13	0.86	0.23	738

Test

	precision	recall	f1-score	support
0	0.99	0.57	0.72	2899
1	0.12	0.94	0.21	185

Precision Recall Threshold Chart



Modeling

Combination Modeling Approach:

- SMOTEN for balancing
- Weight of Evidence Encoder for encoding
- Logistic Regression with Class Weight Optimization for Modeling
- Optimized class weight for f1 score using grid search CV:

```
{'class_weight': {0: 0.4477386934673367, 1: 0.5522613065326634}}
```

Precision, Recall ,F1 Scores

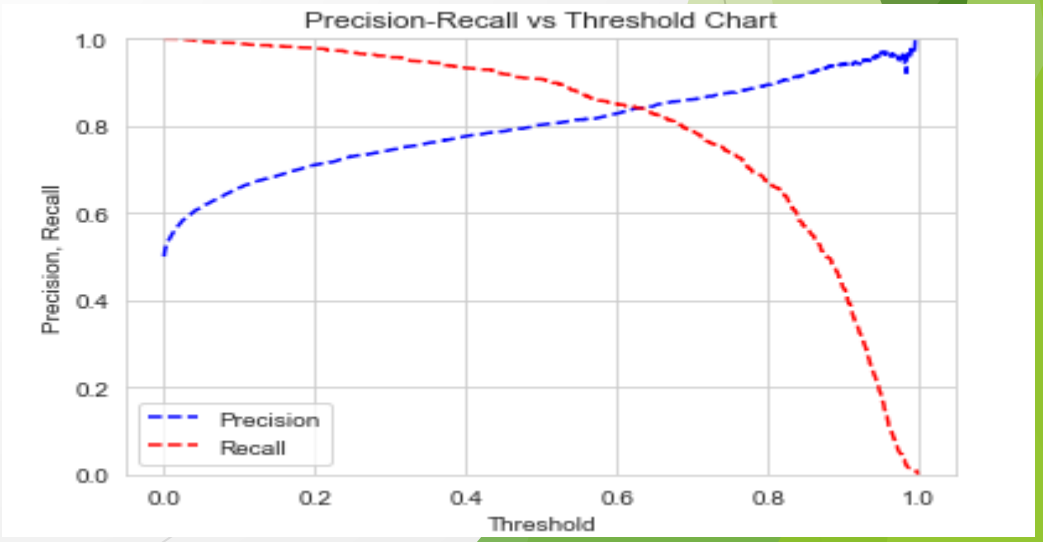
Train

	precision	recall	f1-score	support
0	0.89	0.78	0.83	11598
1	0.81	0.91	0.85	11597

Test

	precision	recall	f1-score	support
0	0.89	0.78	0.83	2899
1	0.80	0.91	0.85	2900

Precision Recall Threshold Chart



Conclusion

,

- SMOTEN balancing combined with weight of evidence encoder, and tuned Logistic Regression model with class weight optimization, gave the best results and F1 score of 0.85 is achieved on minority class on validation data.
- Dummy encoding approach did not generalize well due to too many features
- Logistic regression with optimized class weight without balancing , generalized well but F1 score was only 0.22
- Utility vehicles have higher probability of fraudulent claims in vehicle categories
- Higher probability of fraud if address is changed within last 6 months
- Higher end vehicles like Mercedes, BMW and Acura are more involved in fraudulent claims

THANK YOU