CAR INSURANCE FRAUD

Aniket Pandey, Manmeet Singh Khanna

December 12, 2021

Abstract

Insurance companies have become prone to fraud claims in today's world. There is a huge amount of finance and energy spent in solving these cases individually. The company wishes to reduce amount of loss occurring due to fraudulent transactions. Checking each claim costs the company a certain amount based on the type of representatives used. For checking trends seen in case of fraud claims to better understand the data analytic models can reduce cost by a significant amount. In this project we showcase different Data Mining and Visualization techniques used for detecting and analyzing fraudulent claims. We used a data set containing all the necessary details about the accident and the individual involved in it along with their personal details. As the dataset is unbalanced which has a relatively low number of fraud claims, sampling has been done on claims using different sampling methods. We have found several important trends in fraudulent claims which has been visualized for a clear understanding moreover feature selection was vital for prediction as the data is very sensitive. Light GBM proved to be the best model for our dataset. It gave better result than other algorithms after under sampling the data. One Class SVM and Isolation Forest which are supposed to be good for less sample data and anomaly data didn't work very well for our dataset and gave a relatively less score. We even used Random Forest with various variation of sampling and got a better result than Isolation and One Class SVM.

1 Introduction

Fraud is one of the largest and most well-known problems that many insurance companies face. This project will focus on claim data of a car insurance company. Claims which are fraudulent can be highly expensive for companies. Therefore, it is important to know which claims are valid and which are not. It is not doable for insurance company to check all claims personally since this will cost too much time and money. In this project, we will take advantage of the largest asset which insurers have in the fight against fraud: Data. We will employ various attributes about the claims, insured cars and other circumstances which are included in the data by the insurer. Separating different groups of claims and the corresponding rates of fraud within those groups will provide us with new insights. Furthermore, we will be using Data Mining techniques to predict which claims are likely to be fraudulent. This information can narrow down the list of claims that need a further check. It will enable an insurer to easily detect fraudulent claims made by the policy holder. The major issue with fraudulent cases is that people who file a genuine case get delay in receiving their insurance money and a longer time is taken because of few cases where the fraud claims have been made.

2 Problem Statement

The aim of this project will be to build a model that can detect car insurance fraud. The challenge behind fraud detection is that frauds are far less common as compared to legitimate insurance claims. It's a challenging problem, given the variation in fraud patterns and relatively small ratio of known frauds in typical samples. While building any detection or prediction models, the savings from loss prevention needs to be balanced with the cost of false alerts. Data Mining techniques will allow for improving predictive accuracy, enabling loss control units to achieve higher coverage with low false positive rates. Insurance frauds cover the range of improper activities which an individual may commit to achieve a favorable outcome from the insurance company. This could range from staging the incident, misrepresenting the situation including the relevant actors and the cause of incident and finally the extent of damage caused. If this problem is addressed this will tackle the above stated problem and bring clarity to claims being made by genuine customers.

3 Literature Review

There have already been many studies made on detection of fraud however there are very few dedicated to Car Insurance Fraud. Most of the research papers cover a general approach to the overall insurance fraud. In my project I'll target a particular area of Car insurance and try to narrow down the research around it. Phua et al. proposed a hybrid fraud detection model on a publicly available labelled dataset, which bring together the characteristics of stacking and bagging ensemble. Initially, the stacking classifier employs a meta classifier to select the best learner model from a bunch of base classifiers. In another paper the idea of applying fuzzy logic for identification of fraudulent claims among the settled cases has been suggested in Pathak et al. If-Then Rule is applied to the training dataset and based on the instances, the degree whether it is a fraud or legal is predicted. This paper illustrated how S_Curves can be interpreted and avoids ambiguity based on the "Degree of Goodness" using Fuzzy Logic membership functions. This technique is used for high dimensional large datasets with a high and sufficient accuracy.

A statistical bivariate probity model for analyzing and identifying false insurance cases has been developed by Pinquet. Additionally, an asymmetric legit model by employing Bayesian method has also been proposed in Bermúdez et al. for finding out illegitimate claims in a Spanish automobile insurance dataset.

4 Methods and Techniques

4.1 Feature Engineering

After preprocessing data, we introduce some new relevant features:

- Claim Gap: The gap between the day of accident and day of claim.

$$Claim\ Gap = Claim\_Date - Accident\_Date$$

- Claim Amount Percent: The percentage of claim amount over price of vehicles.

$$Claim\ Amount\ Percentage = \frac{Claim\_Size}{Vehicle\_Price} * 100$$

Using the Claim Amount Percent, we found that customers having vehicle price between 30,000 to 39,000 and claim percentage greater than 100% are more likely to commit fraud.

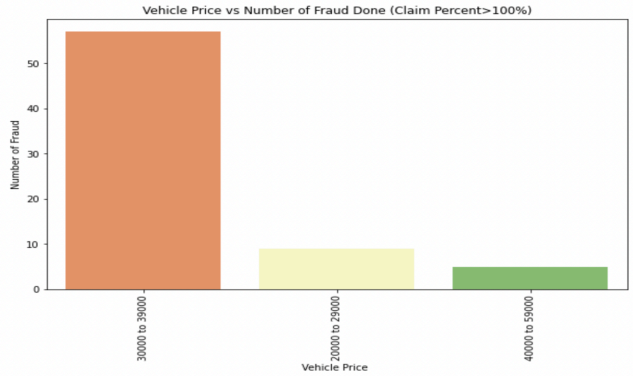


Figure 1: Vehicle Price vs Number of Frauds

4.2 Balancing our imbalanced dataset

Since our dataset was highly imbalanced, we had relatively a smaller number of fraud claims as compared to genuine claims therefore we decided to use sampling method to balance the imbalance. There are different algorithms present to balance the target dataset. We used the following algorithms:

1. Under Sampling
   - TomekLinks: Tomek links are pairs of very close instances, but of opposite classes. Removing the instances of the majority class of each pair increases the space between the two classes, facilitating the classification process. In this we find pairs of examples, one from each class; they together have the smallest Euclidean distance to each other in feature space.

   - ClusterCentroids: In this method we under sample the majority class by replacing a cluster of majority samples by the cluster centroid of a KMeans algorithm. This algorithm keeps N majority samples by fitting the KMeans algorithm with N cluster to the majority class and using the coordinates of the N cluster centroids as the new majority samples.

2. Over Sampling
   - SMOTE: Synthetic minority oversampling technique works by randomly picking a point from the minority class and computing the k-nearest neighbors of this point. The synthetic points are added between the chosen point and its neighbors. SMOTE algorithm works in 4 simple steps:

     1. Choose a minority class as input vector.
     2. Find its k-nearest neighbors.
     3. Choose one of these neighbors and place a synthetic point anywhere on the line joining the point under consideration and its chosen neighbors.
     4. Repeat the step until the data is balanced.

3. Combined Sampling
   - SMOTETomek: In this method first the SMOTE method is applied to oversample the minority class to a balanced distribution, then examples in Tomek Links from the majority classes are identified and removed.

5 Discussion and Results

5.1 Dataset

In this project, we have a dataset which was obtained from Kaggle and has the details of the insurance policy along with the customer details. It also has the details of the accident based on which the claims have been made. The given dataset contains 11,565 rows and 34 columns. The obvious con of this dataset is small size however the ability to work with what is available is crucial for any company looking to transition into leveraging data science, compared to a company that waits for the day when it has a huge data set, the team that started with a small data set and worked on it will more likely succeed earlier in its data science journey and reap its rewards. The column names like policy number, gender, marital status, age, fault, policy type, vehicle category, vehicle price, deductible amount, driver rating, age of vehicle, age of policy, police report, witness present, number of vehicles, base policy, claim size, etc.

	WeekOfMonth	WeekOfMonthClaimed	Age	FraudFound_P	PolicyNumber	RepNumber	Deductible	DriverRating	Year	ClaimSize
count	11565.000000	11565.000000	11560.000000	11565.000000	11565.000000	11565.000000	11565.000000	11559.000000	11565.000000	11565.000000
mean	2.784003	2.701167	39.899567	0.059230	7710.474449	8.469780	407.617812	2.493468	1994.865975	22955.978035
std	1.284854	1.258153	13.590556	0.236066	4453.762219	4.618952	43.397393	1.118102	0.801798	26988.811719
min	1.000000	1.000000	0.000000	0.000000	1.000000	1.000000	300.000000	1.000000	1994.000000	0.000000
25%	2.000000	2.000000	31.000000	0.000000	3851.000000	4.000000	400.000000	1.000000	1994.000000	4148.845001
50%	3.000000	3.000000	38.000000	0.000000	7712.000000	8.000000	400.000000	3.000000	1995.000000	8130.994563
75%	4.000000	4.000000	49.000000	0.000000	11542.000000	12.000000	400.000000	3.000000	1996.000000	46299.646944
max	5.000000	5.000000	80.000000	1.000000	15420.000000	16.000000	700.000000	4.000000	1996.000000	141394.159289

Figure 2 Statistics about Dataset

From the above table we can find few important statistics about Deductible and Claim Size. As seen the average claim size is around 22,955. Rest other columns statistics is not relevant to us.

5.1.1 Data Preprocessing

We preprocessed the data using the fastai library which is open source and available over net. It provides many important function ready to use for structuring the dataset. We used the following functions.

- train_cats(df): Change any columns of strings in a panda's dataframe to a column of categorical values. This applies the changes inplace.
- apply_cats(df, trn): Changes any columns of strings in df into categorical variables using trn as a template for the category codes.
- proc_df(df, y_fld=None, skip_flds=None, ignore_flds=None, do_scale=False, na_dict=None, preproc_fn=None, max_n_cat=None, subset=None, mapper=None): proc_df takes a data frame df and splits off the response variable, and changes the df into an entirely numeric dataframe. For each column of df which is not in skip_flds nor in ignore_flds, na values are replaced by the median value of the column.

5.1.2 Data Visualization

The Infographic_Visualization.ipynb consists of very detailed visualization of the complete dataset. Few of the important visualizations are mentioned in this report, other relevant visualization can be verified from the notebook. The visualization alone helped us detect and deduce a lot of important observation related to how and when fraud related to car insurance takes place.

```
customers[customers['FraudFound_P']==1]['WitnessPresent'].value_counts()
```

```
No     683
Yes      2
```

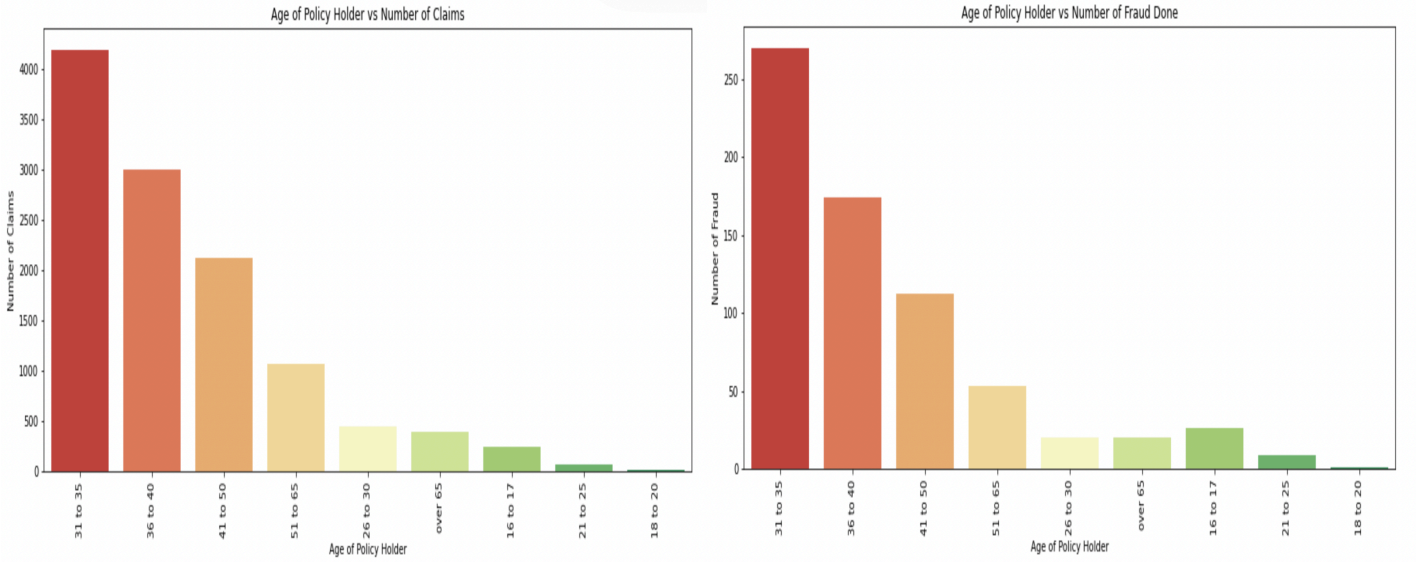
Figure 3 Fraud when Witness was Present

```
customers[customers['FraudFound_P']==1]['PoliceReportFiled'].value_counts()
```

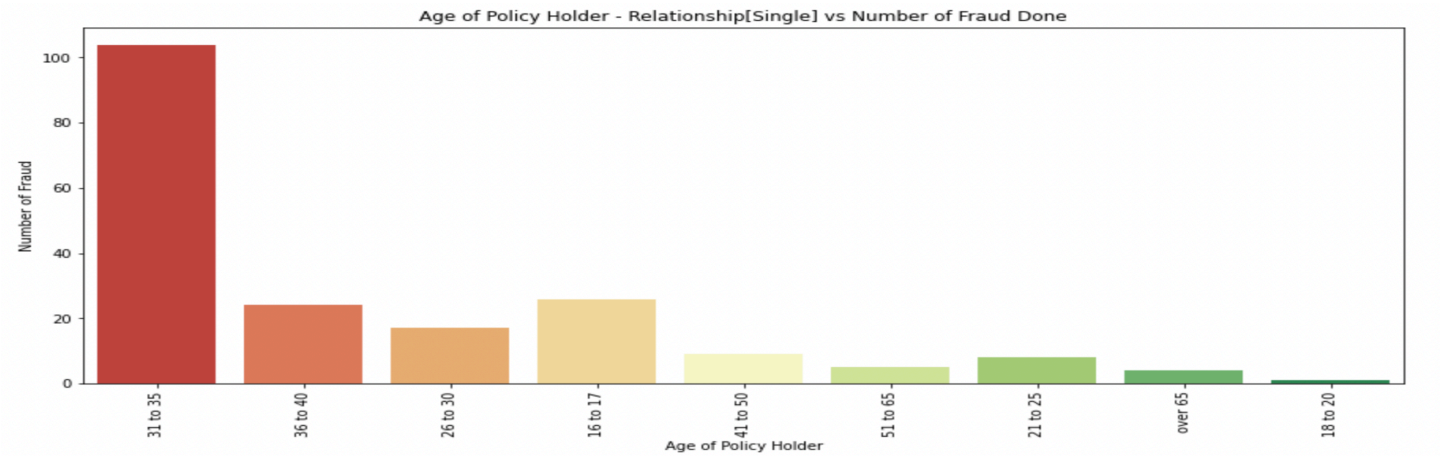
```
No     672
Yes     13
```

Figure 4 Fraud when Police Report was Filed

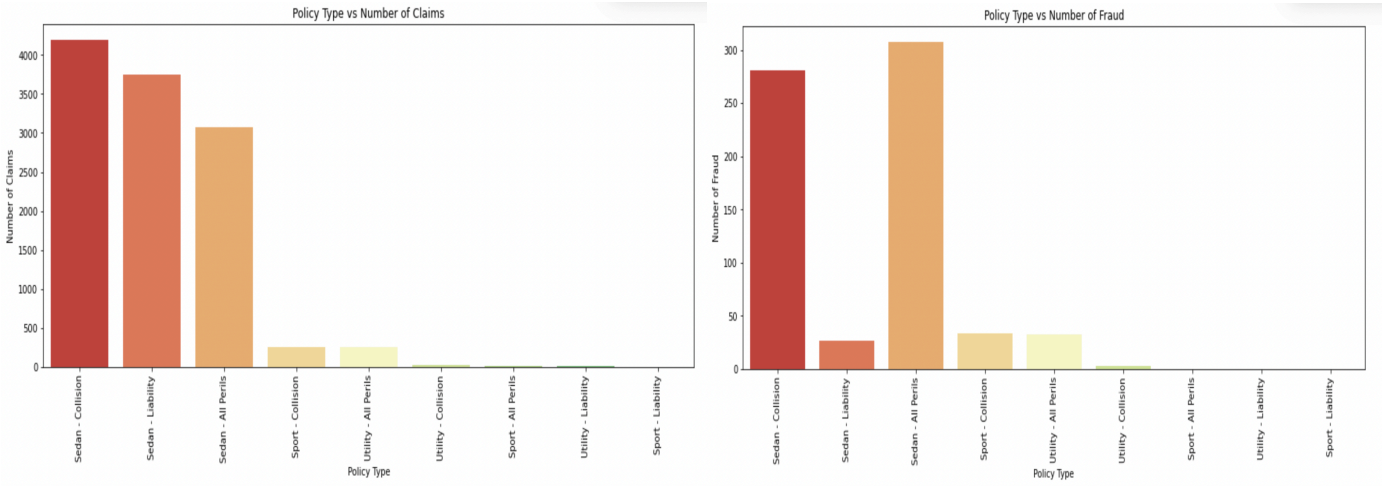
From the above two information it can be inferred that a smaller number of fraud claims were made in cases where Witness was present and Police report was filed.



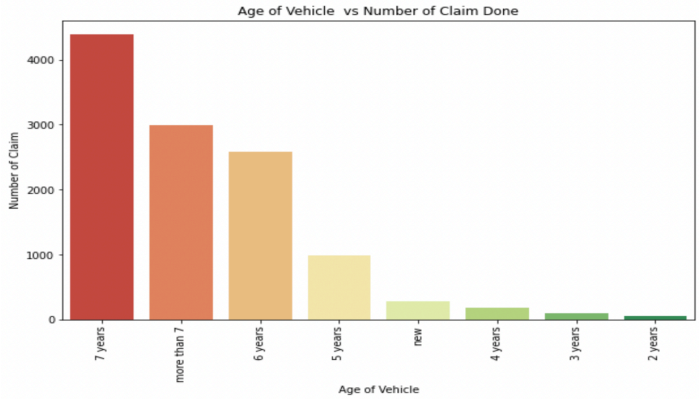
As seen in the chart people of age 31-35 made the most number of claims around 36.31% of the data and they were the ones who made the most number of fraud claims about 39.42%

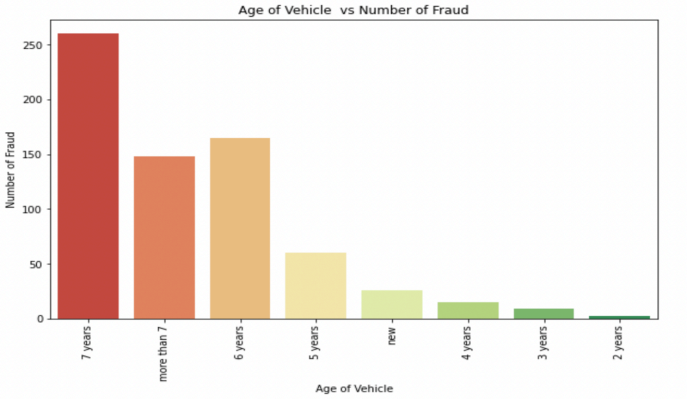


Around 52.53% of customers whose age range between 31-35 and were single made the most number of fraud claims and that too with a huge number of difference.

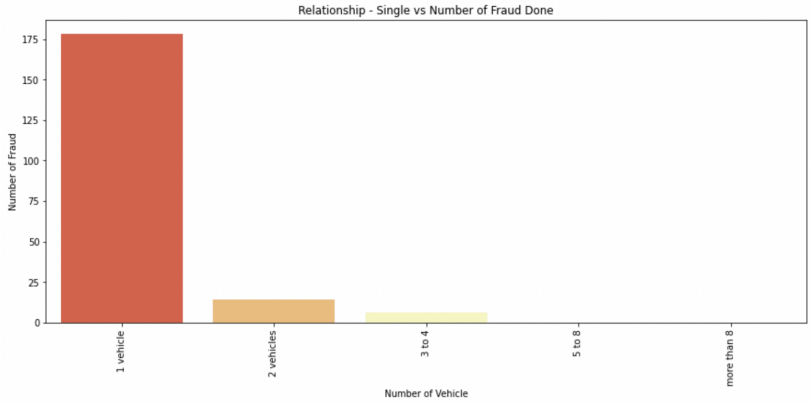


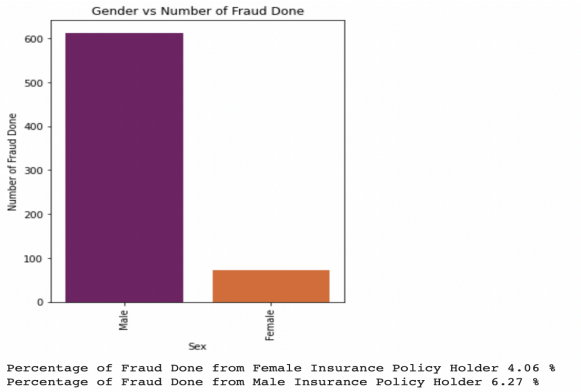
Maximum number of claims 36.20% and 32.36% were made for policy type Sedan Collision and Sedan Liability respectively. Additionally, Sedan All Perils and Sedan Collision recorded maximum number of fraud claims of 44.82% and 41.02% respectively.





Furthermore, the maximum number of claims and fraudulent claims were made for vehicles whose age was around 7 years.





Customers who were Single, had one vehicle and customers who were male made the most number of fraud claims as suggested by data.

5.2 Evaluation Metrics

We used the following standard evaluation metrics to evaluate our model correctness.

- F1 Score: this is the harmonic mean of precision and recall and gives a better measure of the incorrectly classified cases than the accuracy matrix.

$$F1\ Score = \left(\frac{Recall^{-1} + Precision^{-1}}{2}\right)^{-1}$$

- Precision: It is implied as the measure of the correctly identified positive cases from all the predicted positive cases. Thus, it is useful when the costs of False Positives are high.

$$Precision = \left(\frac{True\ Positive}{True\ Positive + False\ Positive}\right)$$

- Recall: It is the measure of the correctly identified positive cases from all the actual positive cases. It is important when the cost of False Negatives is high.

$$Recall = \left(\frac{True\ Positive}{True\ Positive + False\ Negative}\right)$$

- Accuracy: One of the more obvious metrics, it is the measure of all the correctly identified cases. It is most used when all the classes are equally important.

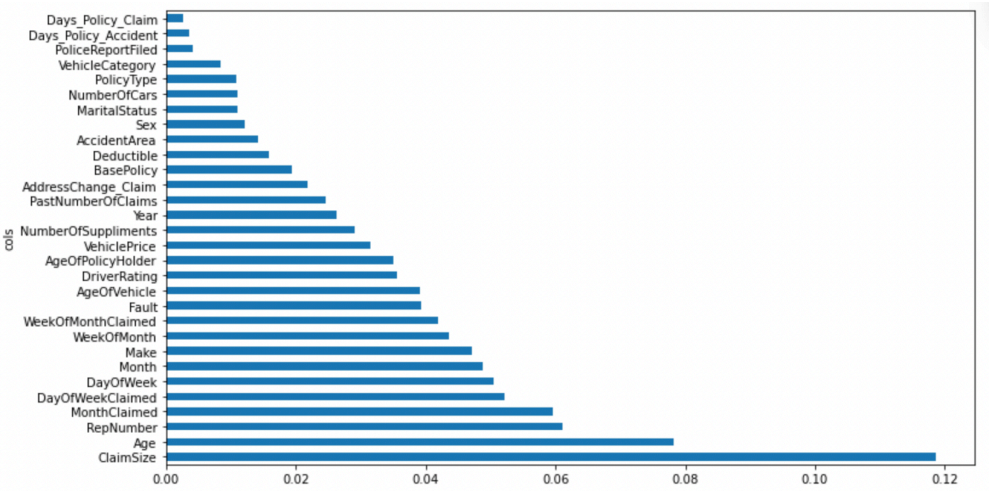
$$Accuracy = \left(\frac{True\ Positive + True\ Negative}{True\ Positive + False\ Positive + True\ Negative + False\ Negative}\right)$$

5.3 Experimental Result

We performed a variation of different algorithms with multiple parameters and following were the observations. The result of each is discussed below.

Random Forest Classifier: We used the sklearn.ensemble library for this. As we know that a forest is made up of trees and more trees means more robust forest. Similarly, a random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result.
We first applied a simple Random Forest Classifier with no variation and plotted the following feature importance table.

	cols	imp
15	PolicyNumber	0.135051
32	ClaimSize	0.093159
10	Age	0.080743
0	Month	0.056027
6	MonthClaimed	0.055724
16	RepNumber	0.055416
1	WeekOfMonth	0.042617
31	BasePolicy	0.040348
5	DayOfWeekClaimed	0.039187
3	Make	0.038213



As can be seen from the table Policy Number got the highest number of importance but we know that it has nothing to do with the fraud detection and it is just a random number which is given to each insurance so that it can be identified uniquely. And therefore, we decided to drop the Policy Number column and again plotted the graph which is displayed in the second image above and Claim Size got the highest importance.

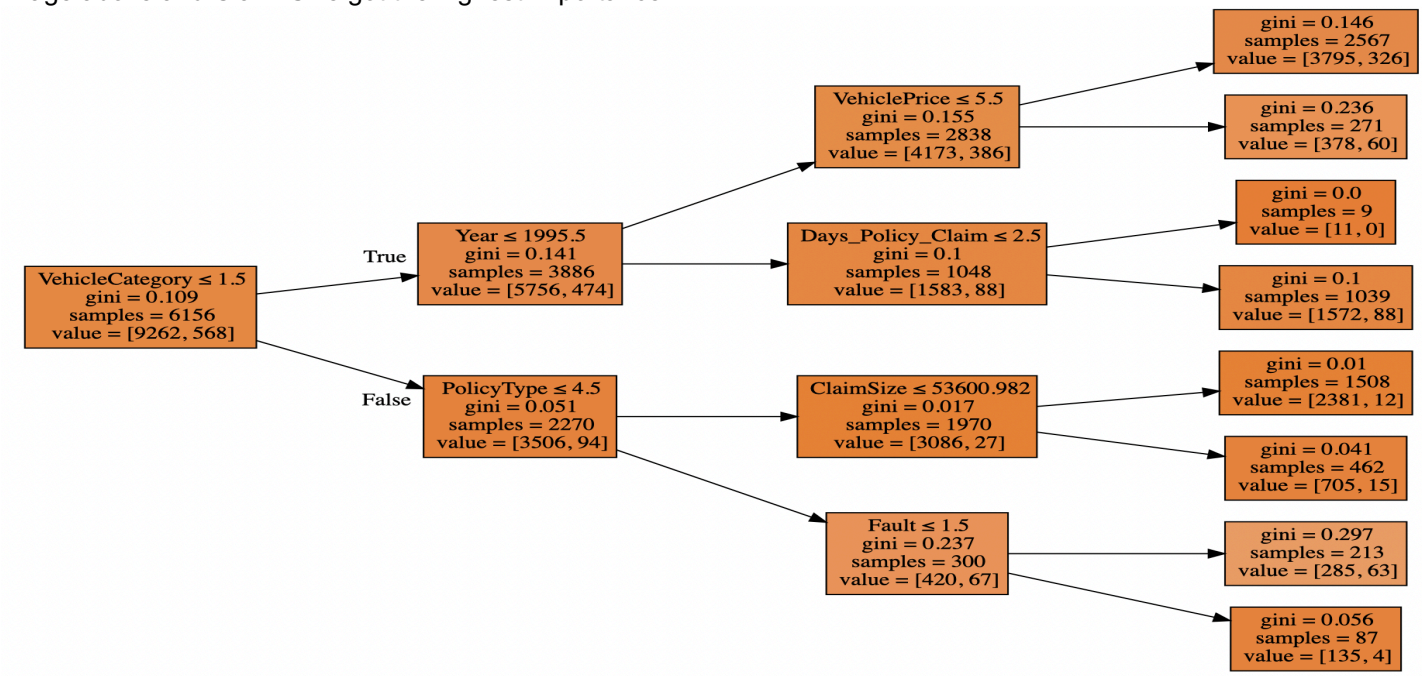
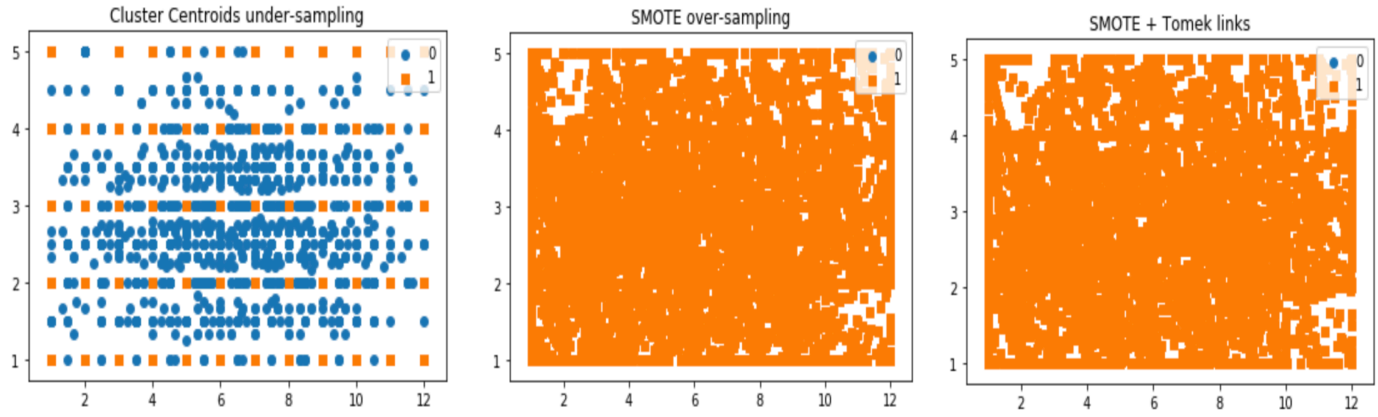


Figure 5 Random Forest split representation

We used the draw_tree(t, df, size=10, ratio=0.6, precision=3) function of the fastai library with the given parameters to draw a representation of a random forest in IPython which looks something like above which defines how our tree was created with respective gini index and tree split.
The next approach we used was under sampling, over sampling and combined sampling. We applied Random Forest Classifier after different variations of sampling. The sampling 2D graph of data used for Random Forest Classifier is as displayed below.



We couldn't find any concrete result from the above experiments moreover in over-sampling, a random set of copies of minority class examples is added to the data. This increased the likelihood of overfitting, especially for higher over-sampling rates. Moreover, it also decreased the classifier performance and increase the computational effort. The results were not satisfactory and therefore we decided to move with our next algorithms.

One Class SVM: It is majorly used where the sample is less and since we have less number of sample we decided to go with this model. One Class SVM works well with novelty detection. The idea of novelty detection is to detect rare events i.e., events that happen rarely, and hence, of which you have very little samples. The problem is then that the usual way of training a classifier will not work. So how do you decide what a novel pattern is? Many approaches are based on the estimation of the density of probability for the data. Novelty corresponds to those samples where the density of probability is "very low". How low depends on the application. The results were not at all satisfactory and we got a weighted average F1 Score of 0.36

Isolation Forest: Next, we decided to remove work by removing outlier. We could have either dropped them from sample or capped the values at some reasonable rate based on domain knowledge. Isolation Forests were built based on the fact that anomalies are the data points that are "few and different". In an Isolation Forest, randomly sub-sampled data is processed in a tree structure based on randomly selected features. The samples that travel deeper into the tree are less likely to be anomalies as they required more cuts to isolate them. Similarly, the samples which end up in shorter branches indicate anomalies as it was easier for the tree to separate them from other observations. After our experiment even this model failed to perform good with a weighted average F1 Score of 0.67

Light GBM: It is a gradient boosting framework that uses tree-based learning algorithms. Light GBM grows tree vertically which means that Light GBM grows tree leaf-wise. It will choose the leaf with max delta loss to grow. When growing the same leaf.
What did we do with the LGBM? We first under sampled our data using Cluster Centroids and then dropped the Policy Number column as we already know it is of no use to us and then trained the model. We kept the prediction threshold as 0.45. If the prediction was less than 0.45 then we classified, it as 'Not Fraud' else marked the set as 'Fraud'. Using LGBM we achieved a F1 Score of 0.80

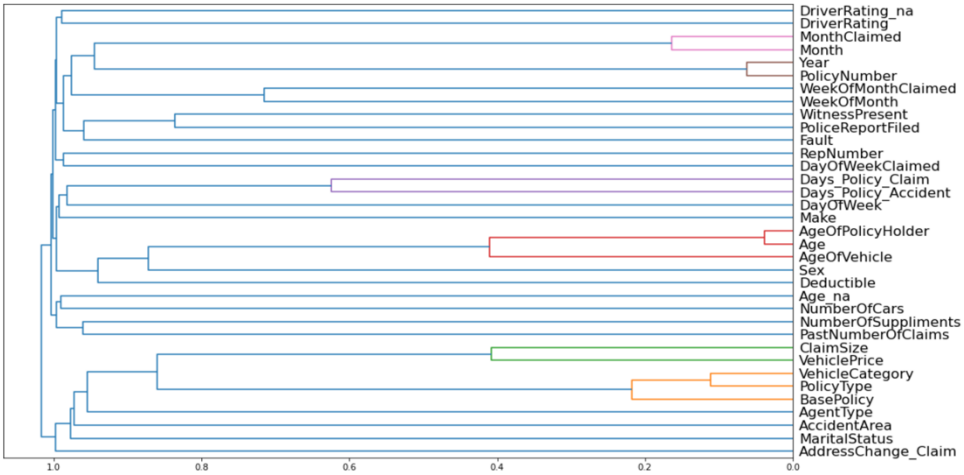


Figure 6 Correlation Hierarchy

We plotted a hierarchy based on correlation to find the roots of the forest formed by a cut by Light GBM and it is as displayed above. A Light GBM model with 32 leaves and learning rate 0.05 was used to give the max augmented score.

After performing various visualization and model training we got the following results:

Algorithm	Sampling	F1 Score
Random Forest	Cluster Centroids Under Sampling	0.57
	SMOTE Over Sampling	0.56
	SMOTE+Tomek Combined Sampling	0.61
One Class SVM	No Sampling	0.36
Isolation Forest	No Sampling	0.67
Light GBM	Cluster Centroids Under Sampling	0.80

Figure 7 Experiment Result

6 Conclusion

Detection of fraudulent claims in automobile insurance claims is a very challenging task as the claim data is highly skewed in nature. In this project, we found Light GBM to perform the best with an accuracy of 0.84 and F1 Score of 0.80.
We first preprocessed the data and then performed thorough visualization of the same. Initially we started with training the Random Forest model and found that Policy Number is getting more importance however it doesn't add value to our classification, so we dropped the Policy Number column and again trained the model and observed no growth. We then switched to sampling and performed all the three variations: under-sampling, over-sampling, and combined sampling which improved the result by a small margin. We then trained the model using One Class SVM and Isolation Forest because they are good algos in condition where we have less sample and more anomaly data respectively however for our data, they performed even poor than Random Forest. Moving forward we decided to test gradient boosting framework and used Light GBM which gave us the best result with cluster centroids under sampling.

6.1 Direction for future work

We can perform even better by stacking multiple models on top of each other and then classifying the set as Fraud or Not Fraud based on majority obtained from the stacked models. In future, other feature selection algorithms and under sampling techniques can be applied for further improving the performance of the proposed system. Besides, preprocessing the model can be optimized by applying various optimization techniques for enhancing the classifier performance. In addition, although this work focuses on a specific application of fraud detection, the present model can be effectively used for fraud detection in other applications and generic databases as well.

References

G. Kowshalya and M. Nandhini, "Predicting Fraudulent Claims in Automobile Insurance," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 1338-1343, doi: 10.1109/ICICCT.2018.8473034.

Pathak, J., Vidyarthi, N., & Summers, S. L. (2005). A fuzzy-based algorithm for auditors to detect elements of fraud in settled insurance claims. Managerial Auditing Journal, 20(6), 632–644. doi:10.1108/02686900510606119

Phua, C., Alahakoon, D., & Lee, V. (2004). Minority report in fraud detection: classification of skewed data. ACM SIGKDD explorations newsletter, 6(1), 50-59.

Pinquet, J., Ayuso, M., & Guillén, M. (2007). Selection bias and auditing policies for insurance claims. The Journal of Risk and Insurance, 74(2), 425–440. https://doi.org/10.1111/j.1539-6975.2007.00219.x

C. Bernard, S. Vanduffel, Mean–variance optimal portfolios in the presence of a benchmark with applications to fraud detection. https://doi.org/10.1016/j.ejor.2013.06.023

N. Dhieb, H. Ghazzai, H. Besbes and Y. Massoud, "Extreme Gradient Boosting Machine Learning Algorithm For Safe Auto Insurance Operations," 2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES), 2019, pp. 1-5, doi: 10.1109/ICVES.2019.8906396.

K. Supraja and S. J. Saritha, "Robust fuzzy rule based technique to detect frauds in vehicle insurance," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 2017, pp. 3734-3739, doi: 10.1109/ICECDS.2017.8390160.

https://medium.com/@pushkarmandot/https-medium-com-pushkarmandot-what-is-lightgbm-how-to-implement-it-how-to-fine-tune-the-parameters-60347819b7fc

https://www.kaggle.com/rafjaa/resampling-strategies-for-imbalanced-datasets/notebook?cellIds=42&kernelSessionId=1756536

https://medium.com/geekculture/insurance-claims-fraud-detection-using-machine-learning-78f04913097