[Auto Insurance Fraud Prediction]

DSCI 4780/6780 Final Project Report

[Yeon-Joo Kang, Cassandra Lundberg, Paige Christian, Rubaiya Masnun]

[April 21st, 2023]

Problem Description

Problem

The problem we are trying to solve is to detect fraudulent auto insurance claims from a given dataset. Insurance fraud costs $308.6 billion every year from American consumers and 10% accounts for property-casualty insurance losses, including auto insurance [1]. Personal auto insurers have at least a $29 billion annual premium leakage, which also creates problems for consumers since 14% of individual auto premiums can be attributed to the cost of covering premium leakage [2]. Auto insurance fraud is a significant problem for both individuals and insurance companies, and detecting fraudulent claims is essential to prevent financial losses. It contributes to huge financial losses for insurance companies, and therefore, we are trying to solve this problem by developing a predictive model that can accurately identify fraudulent insurance claims to help insurance companies save money and prevent or reduce the occurrence of fraudulent activities.

Dataset

We used the auto insurance fraud dataset available on Kaggle [3], which contains 1000 insurance claims with 40 features for each claim, with a mix of both fraudulent and legitimate claims. The data contains various features such as the insured amount, age of the driver, incident type, incident location, policyholder information, and more. The 'fraud_reported' column was used for the target variable, as it indicates whether the claim was fraudulent or legitimate with "Y" and "N," respectively.

	months_as_customer	age	policy_number	policy_bind_date	policy_state	policy_csl	policy_deductable	policy_annual_premium	umbrella_limit	insured_zip
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
...	...	...	...	...	...	...	...	...	...	...
995	3	38	941851	1991-07-16	OH	500/1000	1000	1310.80	0	431289
996	285	41	186934	2014-01-05	IL	100/300	1000	1436.79	0	608177
997	130	34	918516	2003-02-17	OH	250/500	500	1383.49	3000000	442797
998	458	62	533940	2011-11-18	IL	500/1000	2000	1356.92	5000000	441714
999	456	60	556080	1996-11-11	OH	250/500	1000	766.19	0	612260

1000 rows × 40 columns

Table1. Dataset features

Proposed Analytics Solution

We planned to train the dataset with machine learning (ML) algorithms through an ensemble that contains Random Forest, Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbour (KNN), AdaBoostClassifier, and Logistic Regression models to identify patterns that can help distinguish between legitimate and fraudulent claims and predict fraud cases. After training, the model was assessed for its accuracy and effectiveness in detecting fraud.

Data Exploration and Preprocessing

Data Quality Report

The domains of each feature were identified first to determine whether the feature was a categorical or a continuous variable. And we also found some features have "?" value which is technically a Nan, therefore we converted "?" as Nan to accurately detect all missing values. Since feature "_c39" had all Nan for every claim, this feature was removed. We identified 22 categorical variables and 15 continuous variables and created data quality reports for each variable.

	Count	% of Missing	Card.	Mode	Mode Freq.	Mode %	2nd Mode	2nd Mode Freq.	2nd Mode Perc
policy_number	1000	0.0	1000	521585	1	0.100000	687755	1	0.100000
policy_state	1000	0.0	3	OH	352	35.200000	IL	338	33.800000
policy_csl	1000	0.0	3	250/500	351	35.100000	100/300	349	34.900000
policy_deductable	1000	0.0	3	1000	351	35.100000	500	342	34.200000
insured_zip	1000	0.0	995	477695	2	0.200000	469429	2	0.200000
insured_sex	1000	0.0	2	FEMALE	537	53.700000	MALE	463	46.300000
insured_education_level	1000	0.0	7	JD	161	16.100000	High School	160	16.000000
insured_occupation	1000	0.0	14	machine-op-inspct	93	9.300000	prof-specialty	85	8.500000
insured_hobbies	1000	0.0	20	reading	64	6.400000	exercise	57	5.700000
insured_relationship	1000	0.0	6	own-child	183	18.300000	other-relative	177	17.700000
incident_type	1000	0.0	4	Multi-vehicle Collision	419	41.900000	Single Vehicle Collision	403	40.300000
collision_type	1000	17.8	3	Rear Collision	292	35.523114	Side Collision	276	33.576642
incident_severity	1000	0.0	4	Minor Damage	354	35.400000	Total Loss	280	28.000000
incident_state	1000	0.0	7	NY	262	26.200000	SC	248	24.800000
incident_city	1000	0.0	7	Springfield	157	15.700000	Arlington	152	15.200000
incident_location	1000	0.0	1000	9935 4th Drive	1	0.100000	4214 MLK Ridge	1	0.100000
authorities_contacted	1000	0.0	5	Police	292	29.200000	Fire	223	22.300000
property_damage	1000	36.0	2	NO	338	52.812500	YES	302	47.187500
police_report_available	1000	34.3	2	NO	343	52.207002	YES	314	47.792998
auto_make	1000	0.0	14	Saab	80	8.000000	Dodge	80	8.000000
auto_model	1000	0.0	39	RAM	43	4.300000	Wrangler	42	4.200000
fraud_reported	1000	0.0	2	N	753	75.300000	Y	247	24.700000

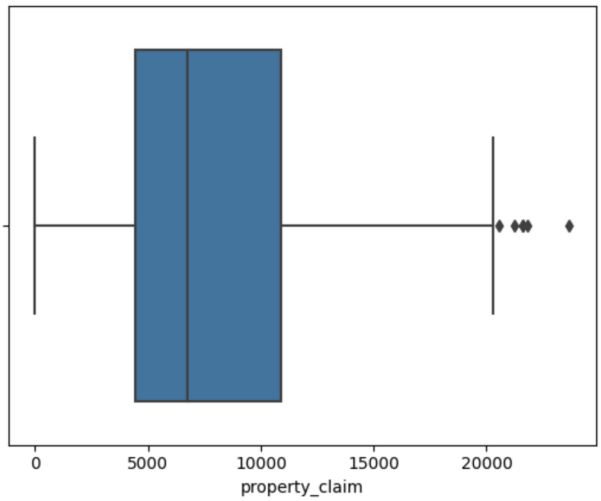
Table 2. Categorical variable data quality report

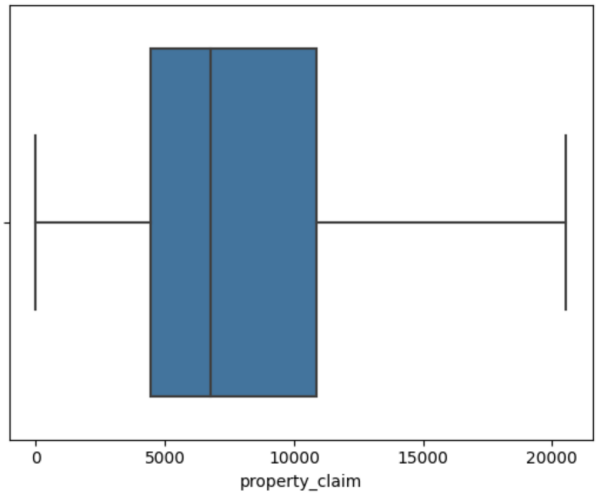
	Count	% of Missing	Card.	Min.	Q1	Median	Q3	Max.	Mean	Std.Dev.
months_as_customer	1000	0.0	391	0.00	115.7500	199.5	276.250	479.00	2.039540e+02	1.151132e+02
age	1000	0.0	46	19.00	32.0000	38.0	44.000	64.00	3.894800e+01	9.140287e+00
policy_annual_premium	1000	0.0	991	433.33	1089.6075	1257.2	1415.695	2047.59	1.256406e+03	2.441674e+02
umbrella_limit	1000	0.0	11	-1000000.00	0.0000	0.0	0.000	10000000.00	1.101000e+06	2.297407e+06
capital-gains	1000	0.0	338	0.00	0.0000	0.0	51025.000	100500.00	2.512610e+04	2.787219e+04
capital-loss	1000	0.0	354	-111100.00	-51500.0000	-23250.0	0.000	0.00	-2.679370e+04	2.810410e+04
incident_hour_of_the_day	1000	0.0	24	0.00	6.0000	12.0	17.000	23.00	1.164400e+01	6.951373e+00
number_of_vehicles_involved	1000	0.0	4	1.00	1.0000	1.0	3.000	4.00	1.839000e+00	1.018880e+00
bodily_injuries	1000	0.0	3	0.00	0.0000	1.0	2.000	2.00	9.920000e-01	8.201272e-01
witnesses	1000	0.0	4	0.00	1.0000	1.0	2.000	3.00	1.487000e+00	1.111335e+00
total_claim_amount	1000	0.0	763	100.00	41812.5000	58055.0	70592.500	114920.00	5.276194e+04	2.640153e+04
injury_claim	1000	0.0	638	0.00	4295.0000	6775.0	11305.000	21450.00	7.433420e+03	4.880952e+03
property_claim	1000	0.0	626	0.00	4445.0000	6750.0	10885.000	23670.00	7.399570e+03	4.824726e+03
vehicle_claim	1000	0.0	726	70.00	30292.5000	42100.0	50822.500	79560.00	3.792895e+04	1.888625e+04
auto_year	1000	0.0	21	1995.00	2000.0000	2005.0	2010.000	2015.00	2.005103e+03	6.015861e+00

Table 3. Continuous variable data quality report

Missing Values and Outliers

Missing values were found in 3 categorical features (collision type, property damage, police report availability) and handled with mode imputation to keep the central tendency of the data. Box plot was used to identify the outliers in continuous variables and found in 4 features (property claim, total claim amount, and policy annual premium). Outliers were handled using Interquartile range (IQR) clamp transformation to eliminate the effect of outliers on the variance of data.





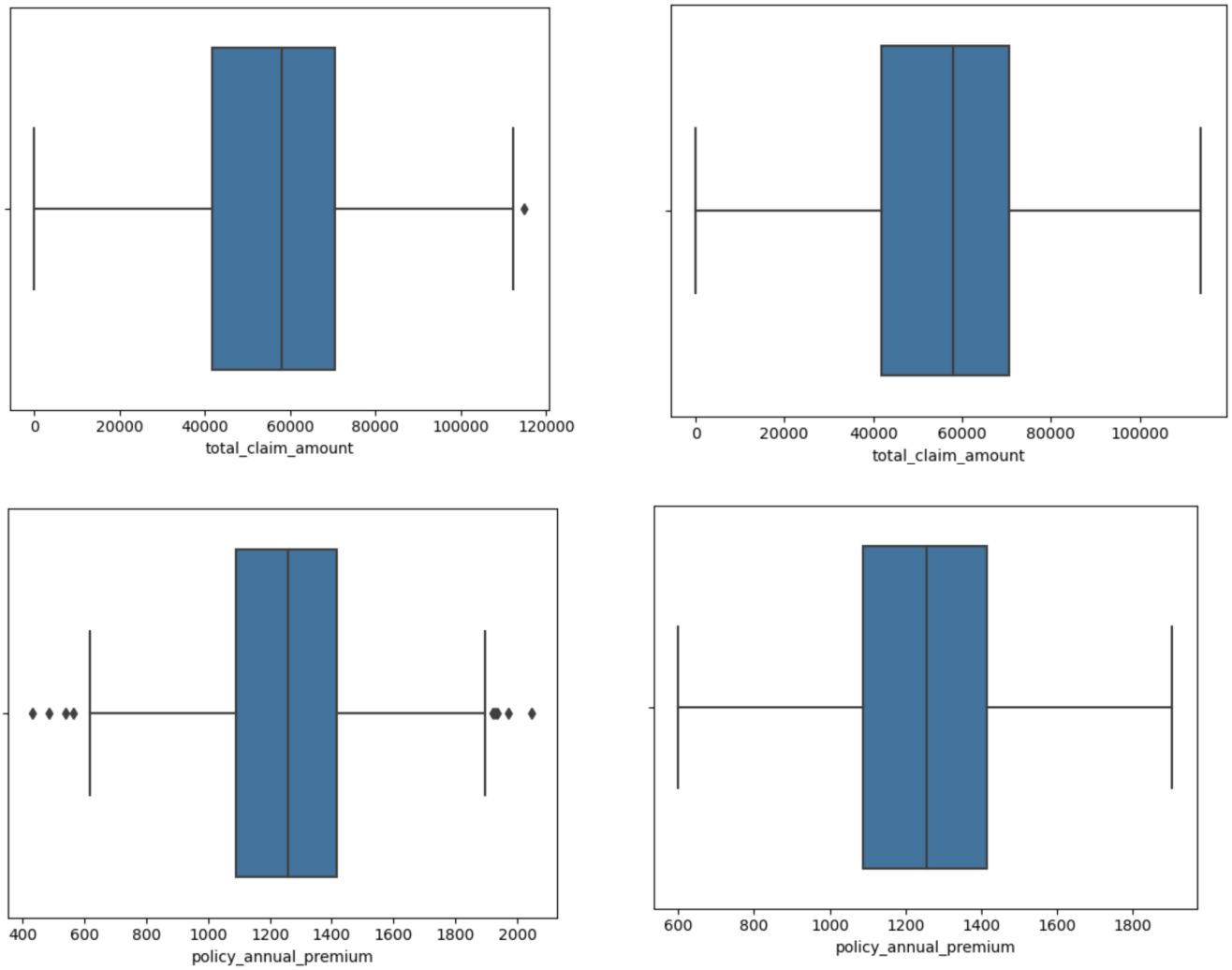


Figure 1. Outliers before and after IQR clamp transformation

Normalization

Min-max normalization was utilized to range the data from -1 to 1 (Table 4) since each feature of the data had different scales, and not all features were normally distributed.

	months_as_customer	policy_deductable	policy_annual_premium	umbrella_limit	capital-gains	capital-loss	incident_hour_of_the_day	number_of_vehicles_involved
0	0.369520	-0.333333	0.236530	-1.0	0.060697	1.000000	-0.565217	-1.000000
1	-0.048017	1.000000	-0.084994	-1.0	-1.000000	1.000000	-0.304348	-1.000000
2	-0.440501	1.000000	0.246082	-1.0	-0.301493	1.000000	-0.391304	0.333333
3	0.068894	1.000000	0.250069	-1.0	-0.026866	-0.123312	-0.565217	-1.000000
4	-0.048017	-0.333333	0.507929	-1.0	0.313433	0.171917	0.739130	-1.000000
...	...	...	...	...	...	...	...	...
995	-0.987474	-0.333333	0.089161	-1.0	-1.000000	1.000000	0.739130	-1.000000
996	0.189979	-0.333333	0.282346	-1.0	0.410945	1.000000	1.000000	-1.000000
997	-0.457203	-1.000000	0.200619	-1.0	-0.301493	1.000000	-0.652174	0.333333
998	0.912317	1.000000	0.159878	-1.0	-1.000000	1.000000	-0.826087	-1.000000
999	0.903967	-0.333333	-0.745906	-1.0	-1.000000	1.000000	-0.478261	-1.000000

1000 rows × 65 columns

Table 4. Scaled dataset with min-max normalization with range from -1 to 1

Feature Selection and Transformations

We used the heatmap to recognize the features correlated with each other for feature selection to reduce the redundancy of the data. Pearson's r and Chi-squared test were used for continuous and categorical features, respectively. Pearson's r range was set from -1 to 1 to examine both negative and positive relationships, and the significance level of the P value was set to 0.05 for the Chi-squared test.

From the continuous feature heatmap (Figure 2), age and month as customer were correlated with each other, and claim amounts (injury claim amount, property claim amount, vehicle claim amount, and total claim amount) were identified to have moderate to strong positive relationships. We decided to remove age and total claim amount from continuous features since the month as customer seemed more related to business, and the total amount claim was the sum of all three claim amounts (injury claim amount, property claim amount, and vehicle claim amount)

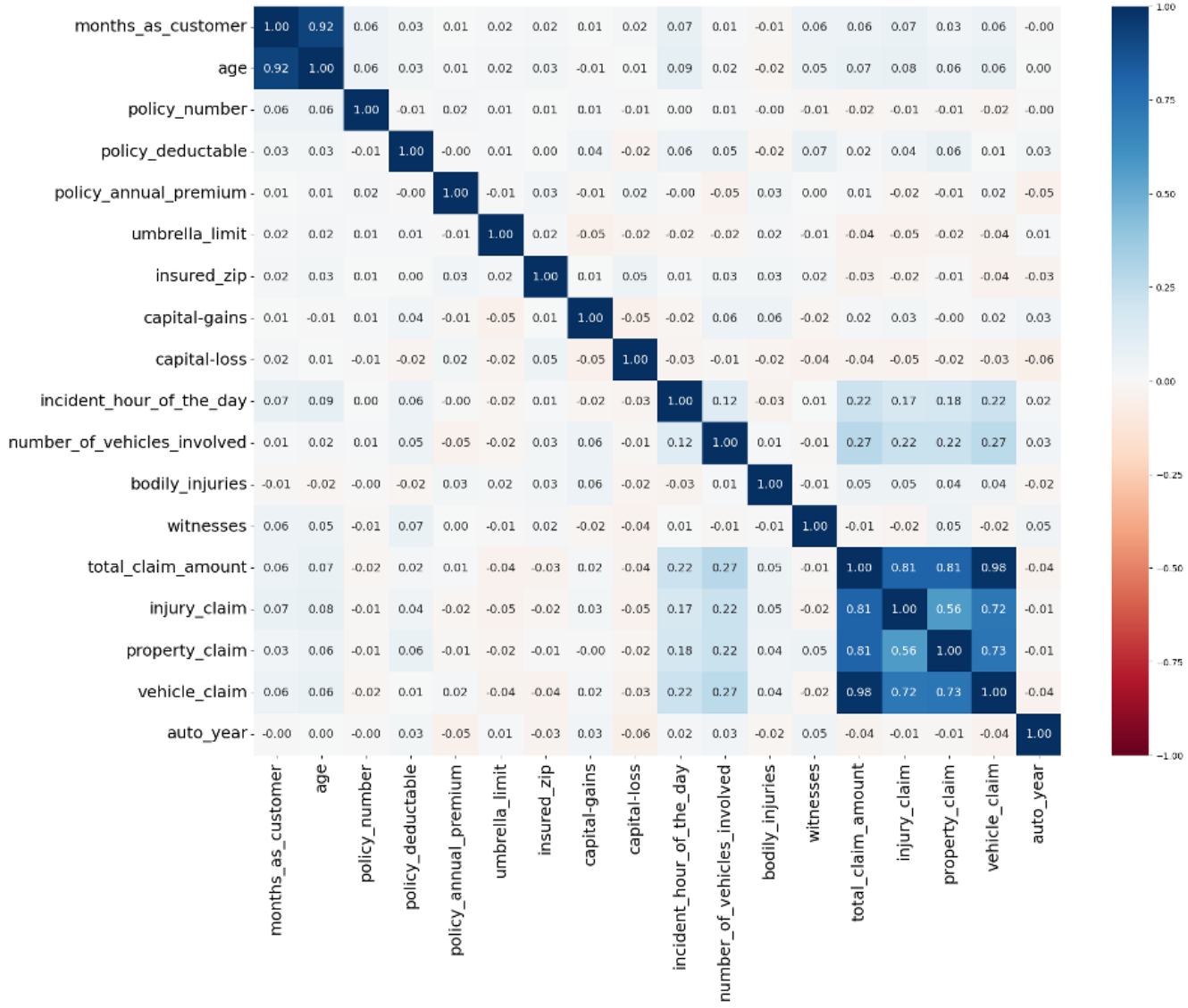


Figure 2. Continuous features Heatmap

For categorical features, there were some features with high cardinality, which did not provide sufficient information for the analysis. Therefore, we decided to remove features with less than 20% frequency (policy number, insured zip, and incident location) indicated as value "2" in the categorial heatmap (Figure 3). In addition to low-frequency features, incident type, collision type, incident severity, and authorities contacted were identified to have a statistically significant relationship between features in the categorical feature heatmap (Figure 3). However, we decided to keep all these variables since we had to drop some features with relatively high cardinality due to dummy code problems (auto make, auto model, and insured hobbies), which were somewhat related to these features.

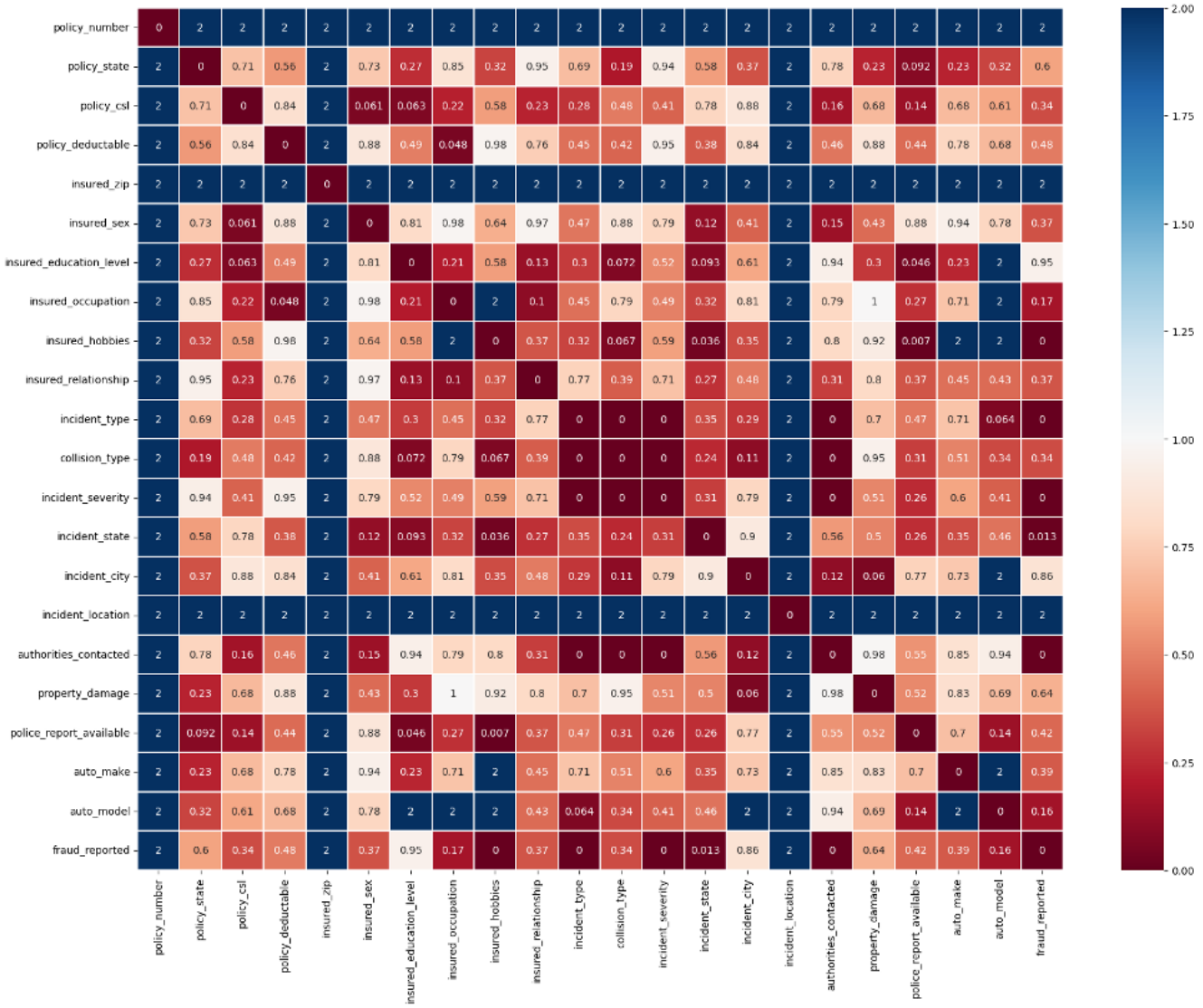


Figure 3. Categorical features Heatmap

Model Selection and Evaluation

Evaluation Metrics

To completely evaluate models on our dataset, we decided to first perform cross validation with five estimators on each model and calculate the accuracy mean score. Since accuracy alone cannot provide enough evidence to select the best model, we also found the precision score, recall score, f1-score, and the confusion matrix for each model. The important problem associated with our data is correctly identifying potential fraud cases within the data where there is a high cost related to false negatives. If more cases go undetected of fraud where the model is not predicting fraud cases out of all true fraud cases, then the model is not performing up to standards. More people will get away with fraud and the detection system is failing. So in the evaluation of models, recall score is the most crucial metric we rely on. This metric shows the percentage of fraud and non-fraud cases detected out of the total number of true fraud and non-fraud cases existing in the dataset.

Models

We evaluated Random Forest, Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbour (KNN), AdaBoost Classifier, and Logistic Regression models with our data. Decision Tree classifier is a tree-like hierarchical model which splits decision nodes based on Gini impurity input, and its output leaf nodes represent predictions of the target value for a particular combination of input feature values. KNN relies on distance for classification with the input consisting of the k closest in a data set and the output is a value classified by a majority vote of its neighbors. AdaBoost is the abbreviation of Adaptive Boosting and a meta-estimator as it additionally fits the classifier on the same dataset by correcting classified weights of instances. its input is a classifier on the original dataset and the output becomes a weighted combination of input classifiers. Random Forest is also a meta-estimator that fits the input of decision tree classifiers from sub-samples of the dataset and averages them to improve the prediction and yields class selected by most trees as the output. Logistic Regression fits the input probabilities of the features to a linear combination through the logistic function and yields log odds of the target value as the output which represents a likelihood of the event taking place. SVM classifies the data based on a hyperplane with the largest margin which is yielded by the dot product of support vectors from data as input.

Sampling and Evaluation Settings

When we created the training and testing datasets, we decided to split the training data into 80% and testing data into 20%. This gives enough data to the models for each to be trained effectively and make accurate predictions. After evaluating the six models, Support Vector Machine performed with the best recall, precision, and accuracy. However, we can hypertune this model to increase the accuracy and output a better recall for the purpose of detecting fraud.

Evaluation

During our model section, we chose to first assess the Decision Tree classifier since it has been proven to effectively predict the class of never-seen data using rule-based approaches. However during evaluations, this model computed low accuracy scores around 70% and held low precision and recall scores in determining fraud cases in particular . We decided to then evaluate how the k-nearest neighbor model would perform on this data. As we set the k neighbors to 3 as a base for observation, we proceeded to run the model with different values of k to improve performance metrics. We noticed how when k was 2 or lower, the predictions showed more errors to detecting fraud and more cases went unseen by the model. Yet, when k was increased to 4 or higher the model began to underfit more, in turn increasing the errors. As k-nearest neighbors also seemed to be slow, we decided to look further at other models. With the AdaBoost classifier being a prevalent technique for binary classification and using its ability to convert multiple weak learners to strong learners in prediction problems, we hypothesized this model would be better than using a decision tree classifier. Although our hypothesis was correct, AdaBoost still predicted too many fraudulent cases as non-fraud and performed worse than k-nearest neighbors in predicting non-fraudulent cases. We then decided to pursue another model which utilized decision trees, Random Forest classifier. This model combined the AdaBoosting strategy from before with the rule-based approaches from the Decision Tree classifier. However, after evaluation, this model performed the same in accuracy as AdaBoost and contained more errors in recall score to predicting fraud than both models. Since the Logistic Regression model has been proven to also be effective for binary classification and uses the sigmoid function to separate data into two classes, this model was hypothesized to help in distinguishing between fraudulent and non-fraudulent claims. However as the accuracy score was the best by far, we decided to evaluate one more model. In the evaluation of the Support Vector Machine on the data, this model performed the best accuracy, precision, and recall.

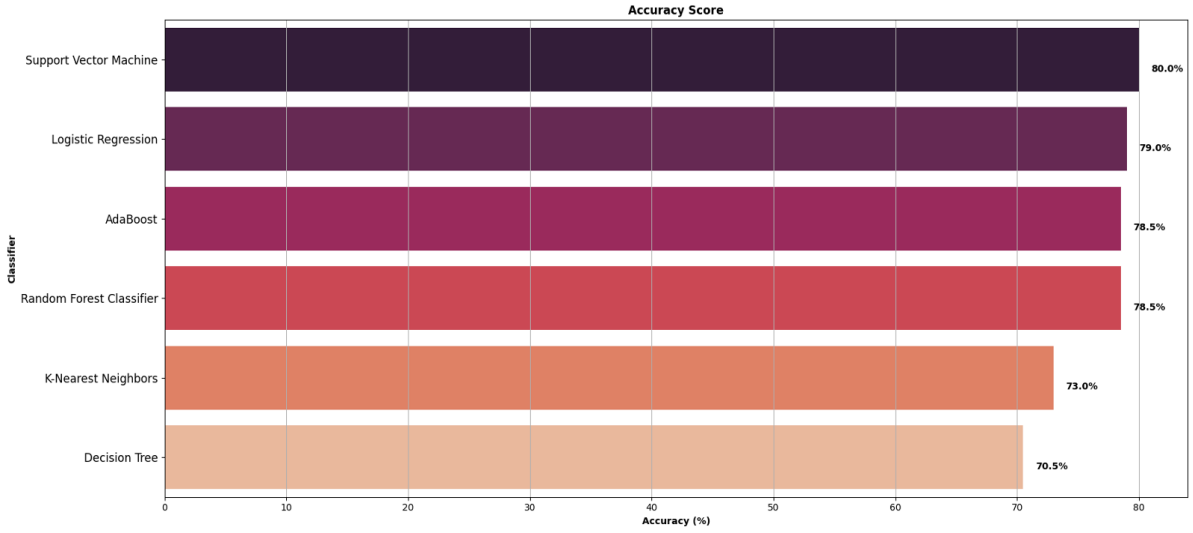
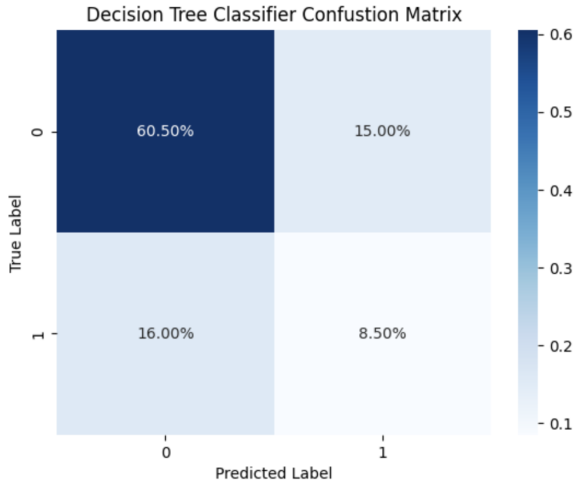


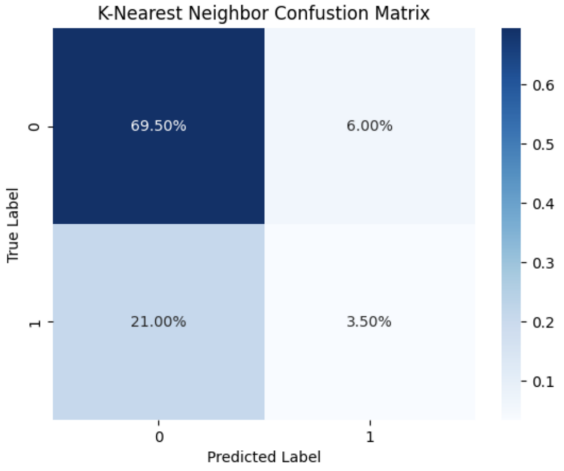
Figure 4. Accuracy score comparison of each model

(a)



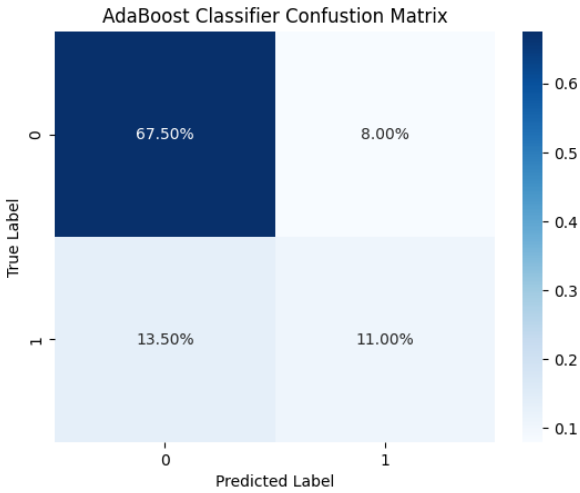
Classification Report	precision	recall	f1-score	support
N	0.79	0.80	0.80	151
Y	0.36	0.35	0.35	49
accuracy			0.69	200
macro avg	0.58	0.57	0.58	200
weighted avg	0.69	0.69	0.69	200

(b)



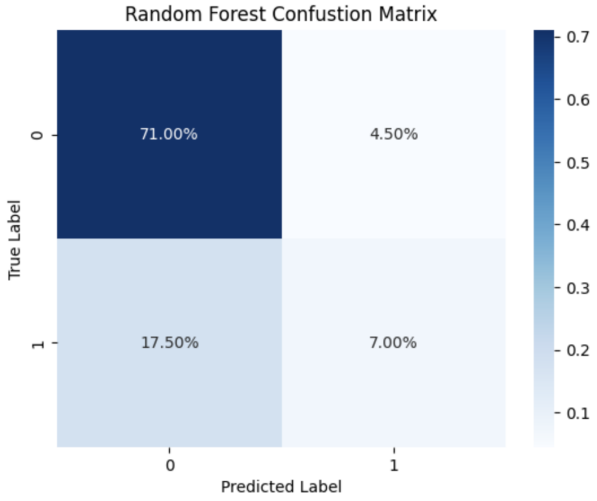
Classification Report	precision	recall	f1-score	support
N	0.77	0.92	0.84	151
Y	0.37	0.14	0.21	49
accuracy			0.73	200
macro avg	0.57	0.53	0.52	200
weighted avg	0.67	0.73	0.68	200

(c)



Classification Report	precision	recall	f1-score	support
N	0.83	0.89	0.86	151
Y	0.58	0.45	0.51	49
accuracy			0.79	200
macro avg	0.71	0.67	0.68	200
weighted avg	0.77	0.79	0.78	200

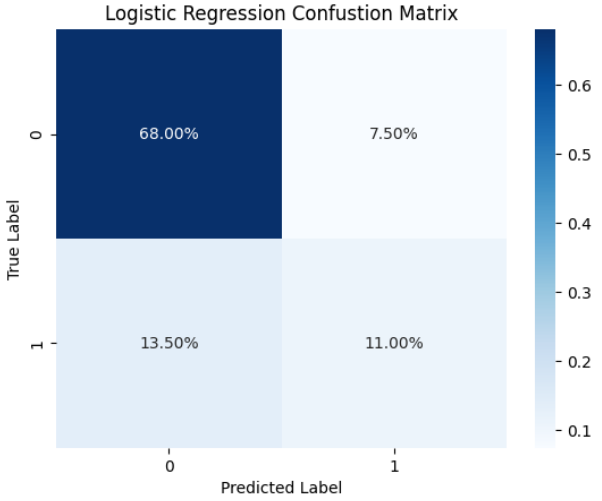
(d)



Classification Report

	precision	recall	f1-score	support
N	0.80	0.94	0.87	151
Y	0.61	0.29	0.39	49
accuracy			0.78	200
macro avg	0.71	0.61	0.63	200
weighted avg	0.75	0.78	0.75	200

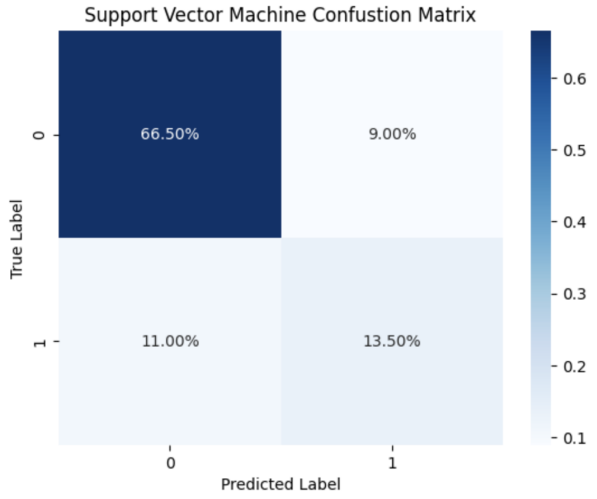
(e)



Classification Report

	precision	recall	f1-score	support
N	0.83	0.90	0.87	151
Y	0.59	0.45	0.51	49
accuracy			0.79	200
macro avg	0.71	0.67	0.69	200
weighted avg	0.78	0.79	0.78	200

(f)



Classification Report

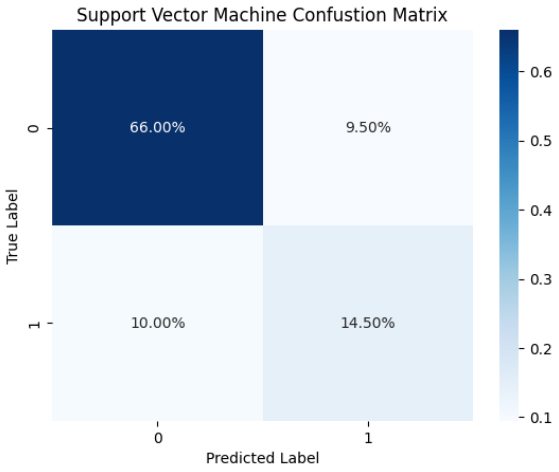
	precision	recall	f1-score	support
N	0.86	0.88	0.87	151
Y	0.60	0.55	0.57	49
accuracy			0.80	200
macro avg	0.73	0.72	0.72	200
weighted avg	0.79	0.80	0.80	200

Figure 5. Confusion matrix and classification report: (a) Decision tree, (b) KNN, (c) AdaBoost, (d) Rain Forest Classifier (e) Logistic Regression (f) SVM

Hyper-parameter Optimization

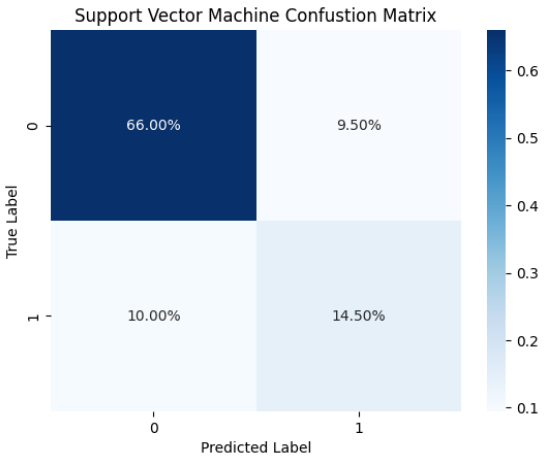
First we decided we needed to weigh the classes of fraud and non-fraud to tell the model which class has the most importance. We adjusted the weight of the fraud or 'Y' class to be 0.60 and the non-fraud of 'N' class to be 0.40. By weighing each class, the accuracy increased to 80.5%. If the weights of fraud increase further and non-fraud decrease further, this results in no change compared to the original model with no weights. With the weights added to the model, we also decided to change the kernel of the Support Vector Machine model. At first the kernel is linear, compared to the polynomial and the sigmoid kernel, this kernel does the best. However when compared to the radial basis function kernel, the accuracy, precision, and recall all stay the same. These hyper parameters seem to be the best for the support vector machine model and result in the best metrics for fraud detection.

(a)




```
Support Vector Machine
Accuracy Score:  0.805

Confusion Matrix:
 [[132  19]
 [ 20  29]]

Classification Report
              precision    recall  f1-score   support

           N       0.87      0.87      0.87       151
           Y       0.60      0.59      0.60        49

    accuracy                           0.81       200
   macro avg       0.74      0.73      0.73       200
weighted avg       0.80      0.81      0.80       200
```

(b)




```
Support Vector Machine
Accuracy Score:  0.805

Confusion Matrix:
 [[132  19]
 [ 20  29]]

Classification Report
              precision    recall  f1-score   support

           N       0.87      0.87      0.87       151
           Y       0.60      0.59      0.60        49

    accuracy                           0.81       200
   macro avg       0.74      0.73      0.73       200
weighted avg       0.80      0.81      0.80       200
```

Figure 6. SVM Confusion matrix: (a) with weights ('Y' = 0.60, 'N' = 0.40), (b) with weights and

Radial Basis Kernel

Results and Conclusion

In addition to ML model evaluation, we conducted principal component analysis (PCA) to identify the most informational features in predicting fraud. PCA (Table) revealed the three most contributed features were vehicle claim amounts, incident type (= single vehicle collision), and insured sex (= female). Since PCA did not reveal further information about continuous features, we created the box plot to determine whether a high or low amount of vehicle claims were more likely to be a fraud and conducted the independent t-test for its statistical significance. Box plot (Figure) revealed that a high amount of vehicle claims are more likely to be fraudulent, and the independent t-test confirmed the vehicle claim amount difference between fraud and legitimate claims is significant ($p < 0.0001$).

0	Rank0	vehicle_claim
1	Rank1	incident_type_Single Vehicle Collision
2	Rank2	insured_sex_FEMALE
3	Rank3	police_report_available_NO
4	Rank4	property_damage_NO
5	Rank5	policy_csl_100/300
6	Rank6	policy_csl_500/1000
7	Rank7	collision_type_Front Collision
8	Rank8	policy_csl_250/500
9	Rank9	authorities_contacted_Police

Table 5. Principal component analysis results

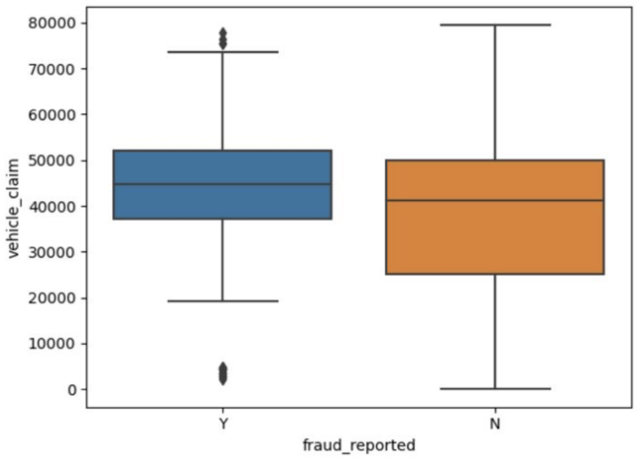


Figure 7. vehicle claim amount box plot

Based on our results, we suggest SVM as the model for predicting auto insurance fraud and claims with a higher amount of vehicle claims, single collisions, female drivers, no police report, and no property damage be flagged for closer investigation as these are more likely to have fraudulent claims.

Appendix A. Data Domains

	domain
months_as_customer	[328, 228, 134, 256, 137, 165, 27, 212, 235, 4...
age	[48, 42, 29, 41, 44, 39, 34, 37, 33, 61, 23, 3...
policy_number	[521585, 342868, 687698, 227811, 367455, 10459...
policy_bind_date	[2014-10-17, 2006-06-27, 2000-09-06, 1990-05-2...
policy_state	[OH, IN, IL]
policy_csl	[250/500, 100/300, 500/1000]
policy_deductable	[1000, 2000, 500]
policy_annual_premium	[1406.91, 1197.22, 1413.14, 1415.74, 1583.91, ...
umbrella_limit	[0, 5000000, 6000000, 4000000, 3000000, 800000...
insured_zip	[466132, 468176, 430632, 608117, 610706, 47845...
insured_sex	[MALE, FEMALE]
insured_education_level	[MD, PhD, Associate, Masters, High School, Col...
insured_occupation	[craft-repair, machine-op-inspct, sales, armed...
insured_hobbies	[sleeping, reading, board-games, bungie-jumpin...
insured_relationship	[husband, other-relative, own-child, unmarried...
capital-gains	[53300, 0, 35100, 48900, 66000, 38400, 52800, ...
capital-loss	[0, -62400, -46000, -77000, -39300, -51000, -3...
incident_date	[2015-01-25, 2015-01-21, 2015-02-22, 2015-01-1...
incident_type	[Single Vehicle Collision, Vehicle Theft, Mult...
collision_type	[Side Collision, ?, Rear Collision, Front Coll...
incident_severity	[Major Damage, Minor Damage, Total Loss, Trivi...
authorities_contacted	[Police, None, Fire, Other, Ambulance]
incident_state	[SC, VA, NY, OH, WV, NC, PA]
incident_city	[Columbus, Riverwood, Arlington, Springfield, ...
incident_location	[9935 4th Drive, 6608 MLK Hwy, 7121 Francis La...
incident_hour_of_the_day	[5, 8, 7, 20, 19, 0, 23, 21, 14, 22, 9, 12, 15...
number_of_vehicles_involved	[1, 3, 4, 2]
property_damage	[YES, ?, NO]
bodily_injuries	[1, 0, 2]
witnesses	[2, 0, 3, 1]
police_report_available	[YES, ?, NO]
total_claim_amount	[71610, 5070, 34650, 63400, 6500, 64100, 78650...
injury_claim	[6510, 780, 7700, 6340, 1300, 6410, 21450, 938...
property_claim	[13020, 780, 3850, 6340, 650, 6410, 7150, 9380...
vehicle_claim	[52080, 3510, 23100, 50720, 4550, 51280, 50050...
auto_make	[Saab, Mercedes, Dodge, Chevrolet, Accura, Nis...
auto_model	[92x, E400, RAM, Tahoe, RSX, 95, Pathfinder, A...
auto_year	[2004, 2007, 2014, 2009, 2003, 2012, 2015, 199...
fraud_reported	[Y, N]
_c39	[nan]

Reference

[1] The impact of insurance fraud on the U.S. Economy 2022. Coalition Against Insurance Fraud.

[2] The challenge of auto insurance premium leakage. Verisk insurance solutions

[3] Auto Insurance Claims Data. Kaggel