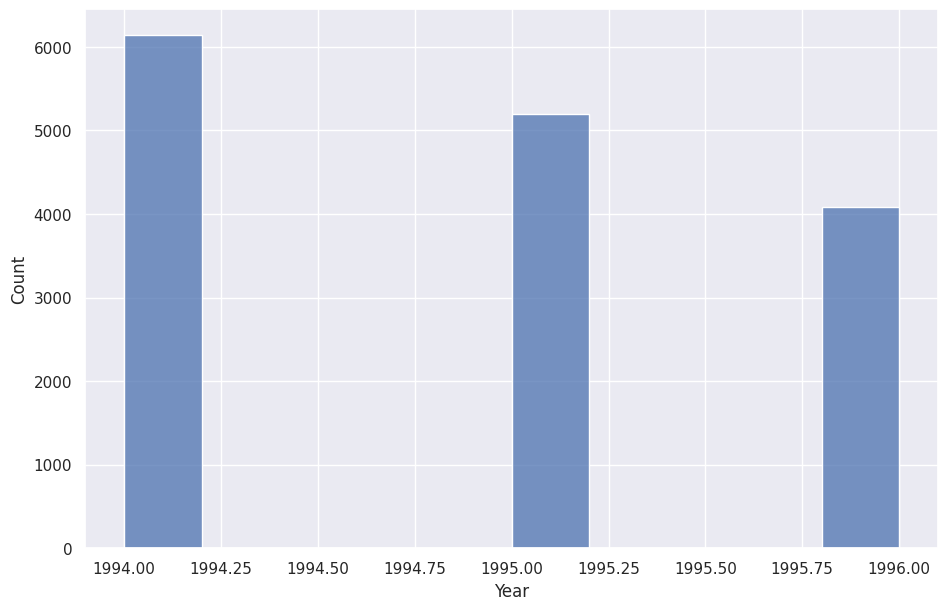
**Machine Learning - 2nd Assignment project**

**Implementation report :**

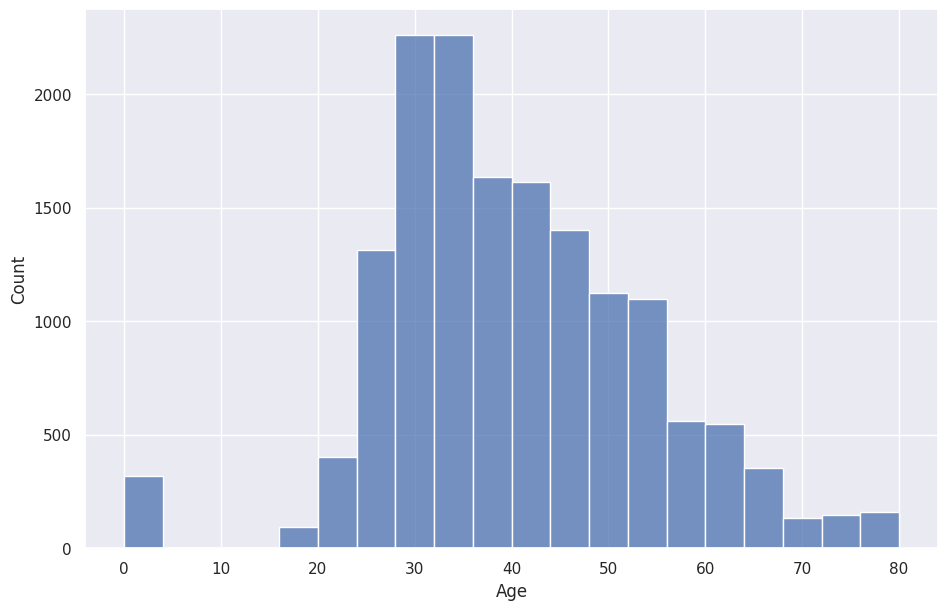
This project concerns predicting fraudulent cases among vehicle insurance claims using various machine learning models. This can be a challenging task, since fraud cases are quite rare and our dataset is imbalanced.

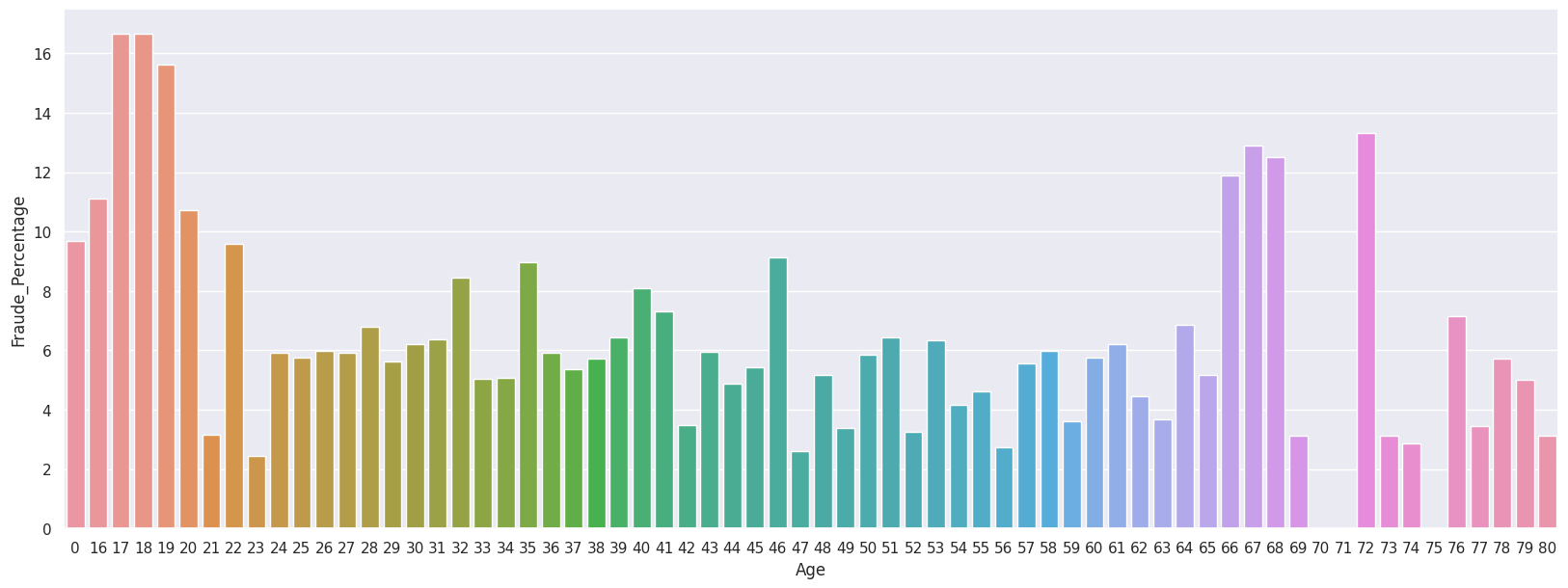
I have used a Weight-Based approach and under-sampling to overcome the issue, and estimators such as Logistic Regression, SVM, Decision-Trees, etc. have been used to evaluate the results.

Firstly, we focus on the information data provides us. We have plotted frequency of some variables from data and the percentages of fraud among their values :

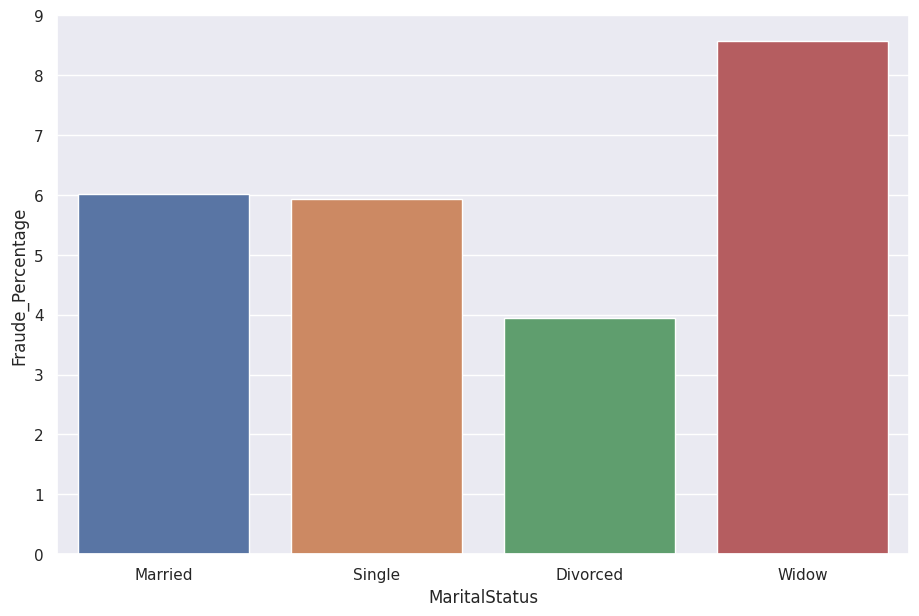
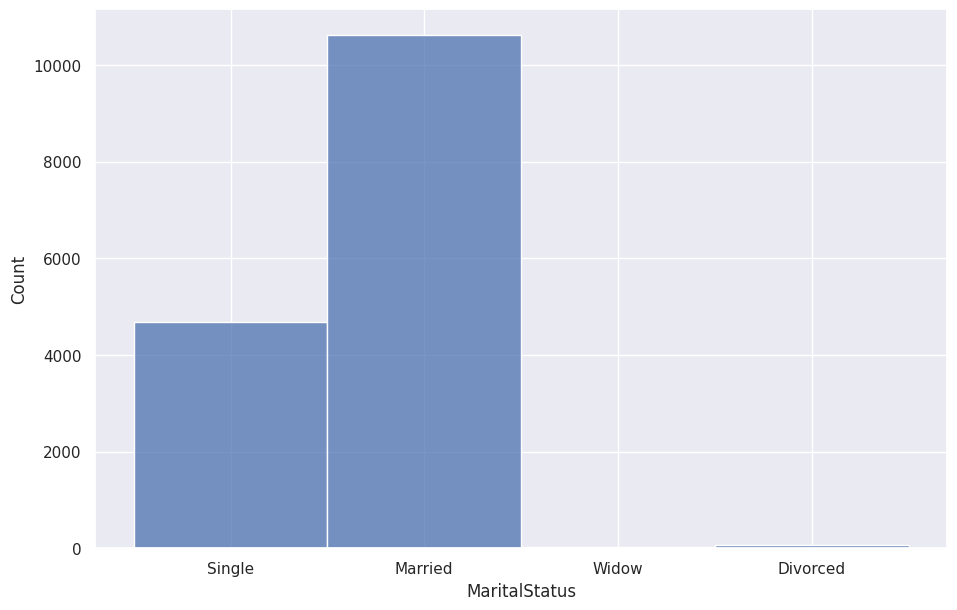


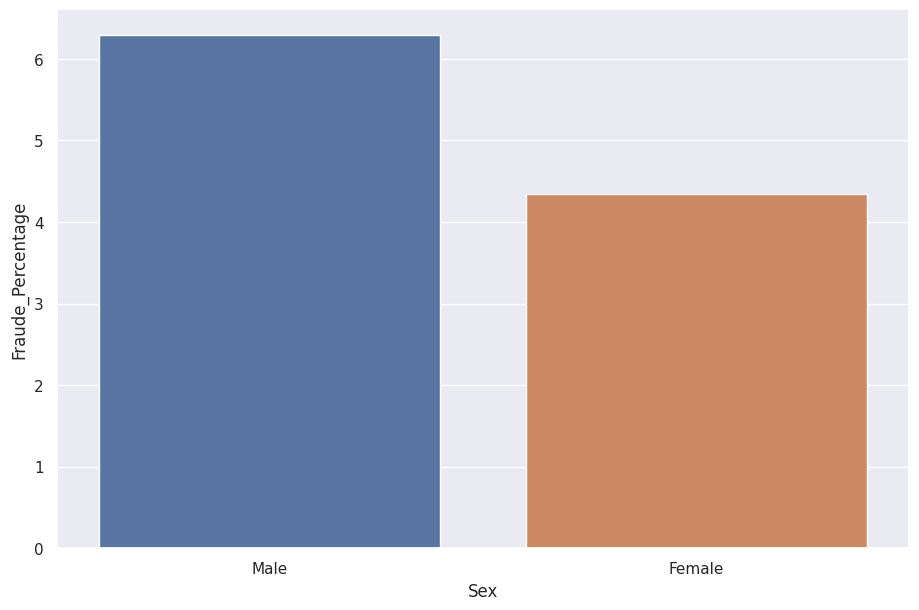
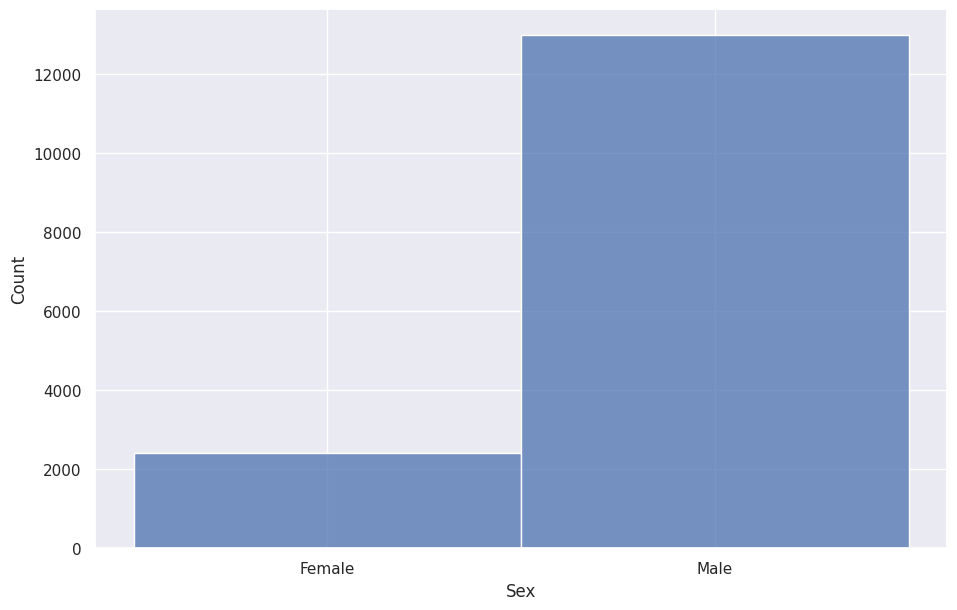
Our dataset only corresponds to data from years 1994,1995 and 1996.

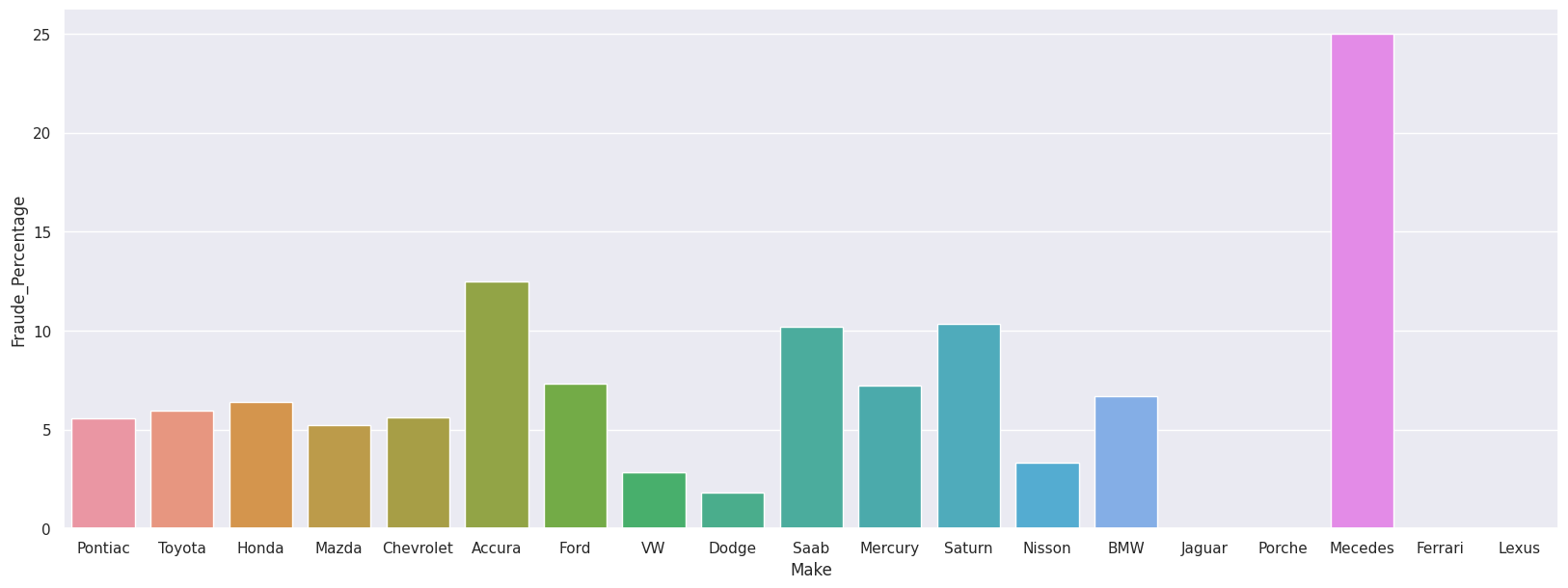
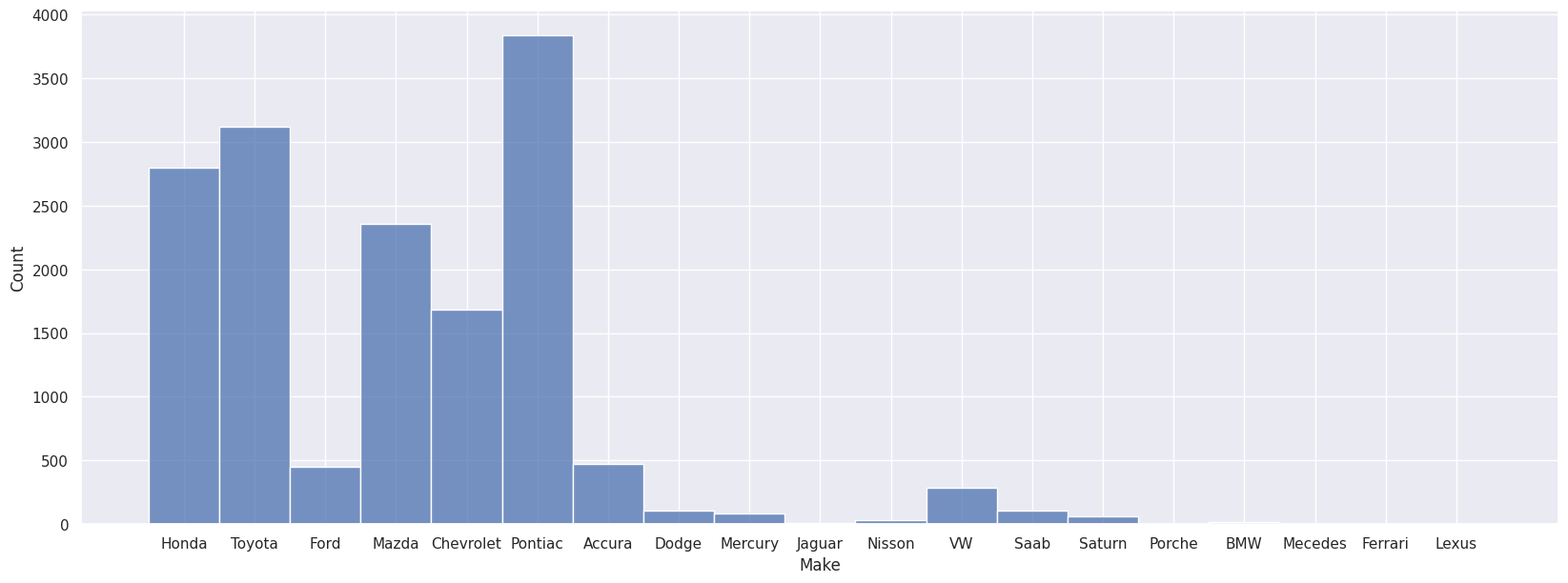


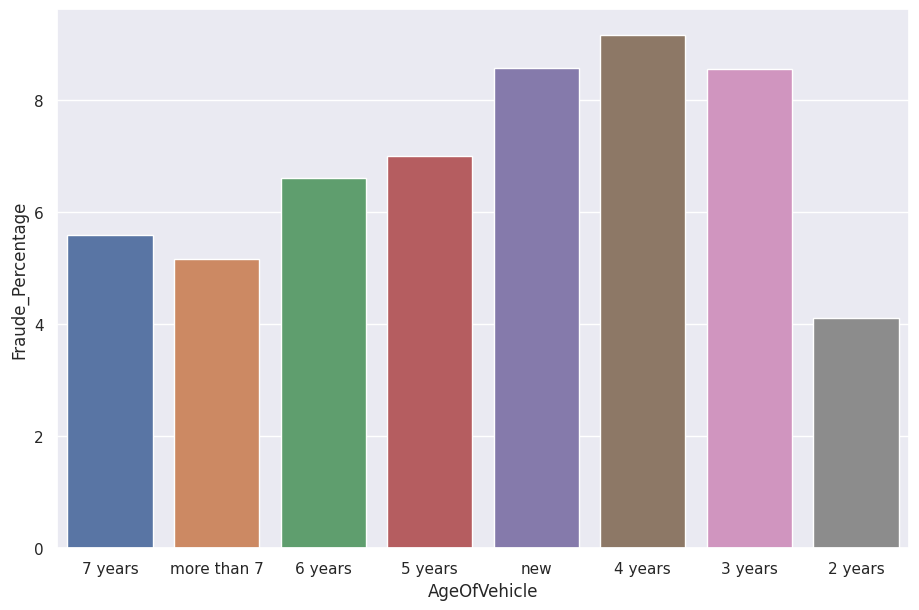
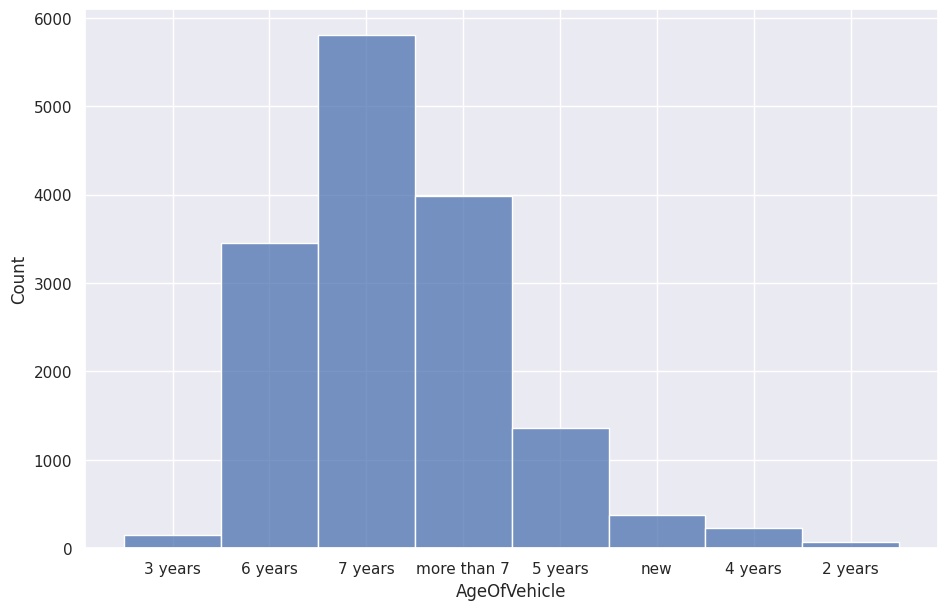


Most policyholders are between 30 to 40 years old, and they also have a higher percentage of fraud among them. Other high percentages belong to very sparse age brackets and are not reliable.

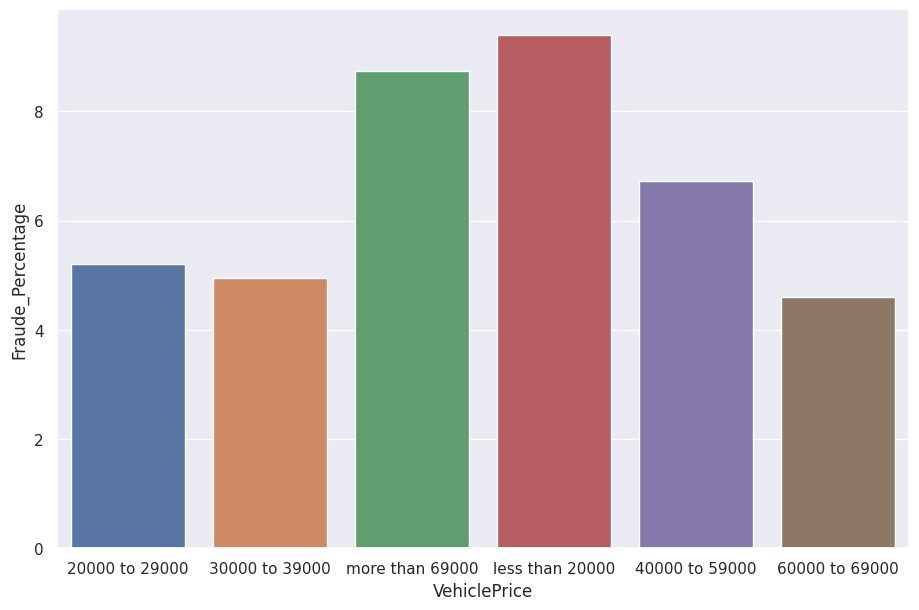
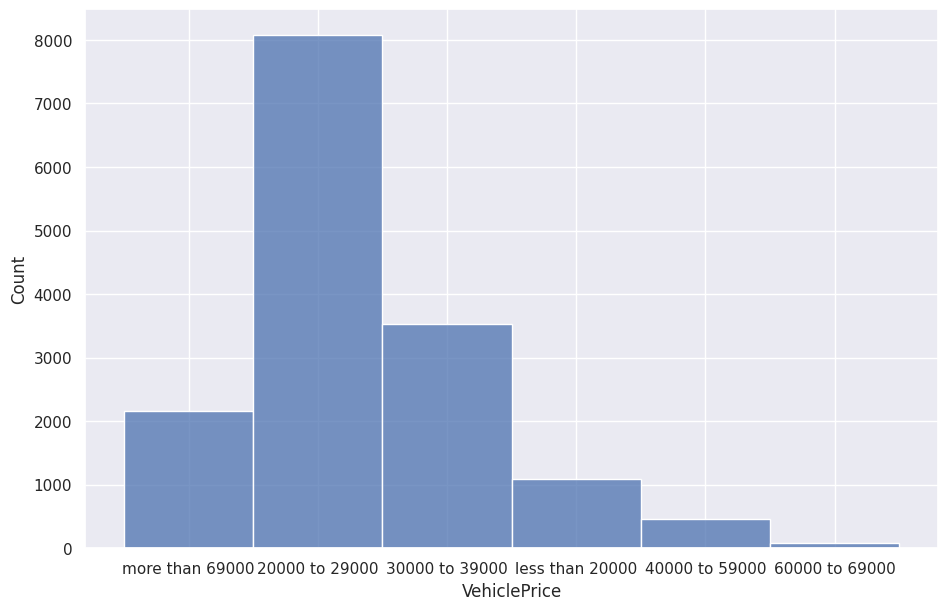
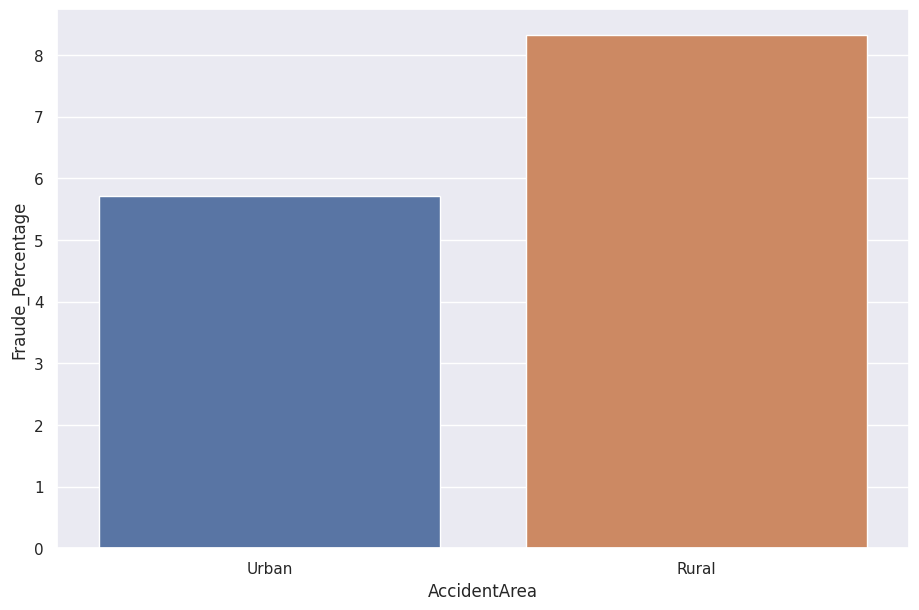
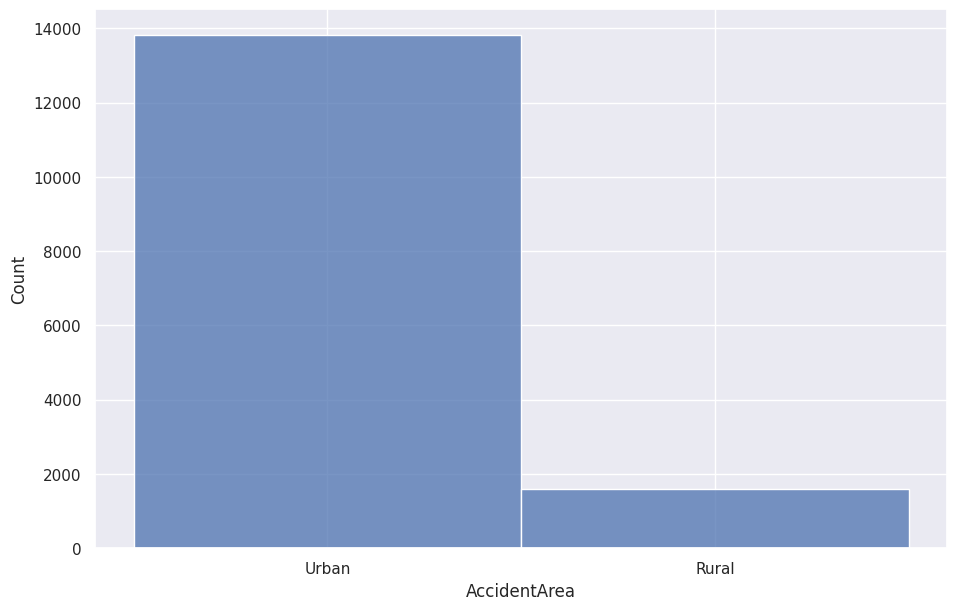
Most policyholders are either married or single, and they are relatively equally likely to commit fraud.

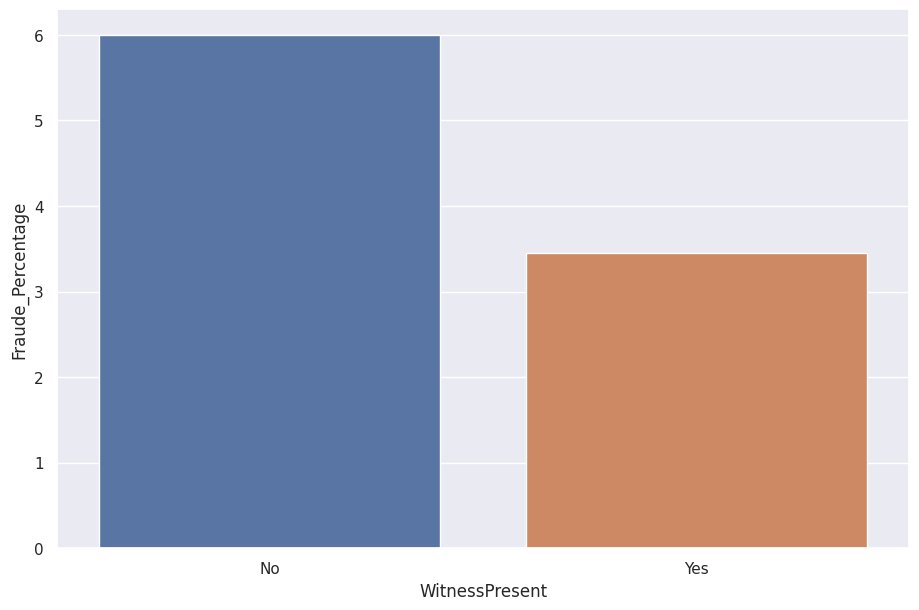
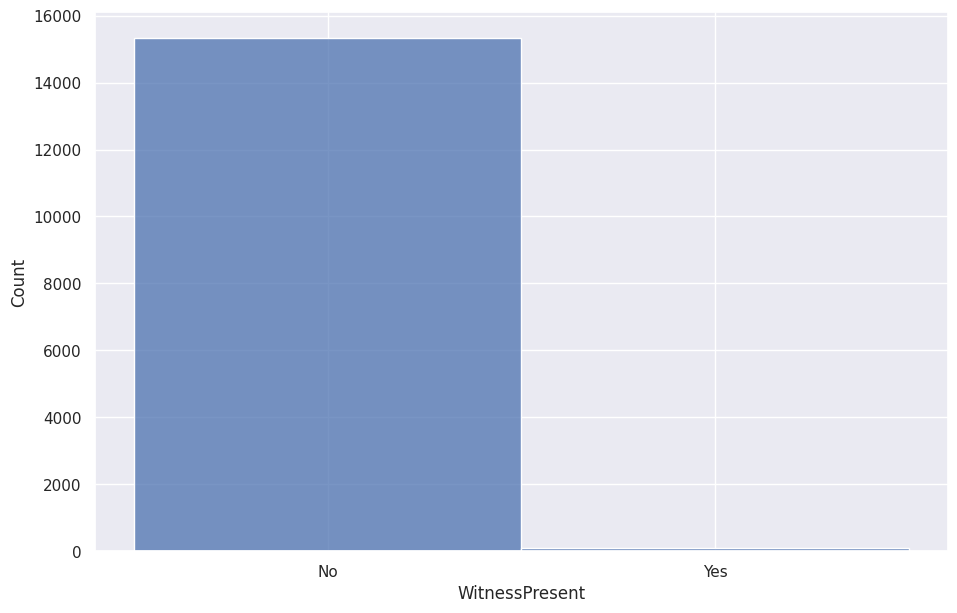
Men are the majority of policyholders and are also more likely to commit fraud.

Honda, Toyota and Pontiac are the most common brands among vehicles, and their owners are also more likely to commit fraud.



Highest percentage of fraud happens with vehicles that are up to 5 to 6 years old, and as their vehicle becomes older, the policyholders are less likely to commit fraud.

Fraud percentage among car owners whose vehicle costs more than 69000 dollars is far higher than other policyholders.Almost all the claims are requested from urban areas, but the percentage of fraudulent claims from rural areas are higher.

And finally, while claims in which a witness is present is less likely to be fraudulent, these claims are also nonexistent.

Now that we have some idea, what our dataset entails, we can move on to build predictive models. After trimming the dataset and removing absolutely irrelevant features such as PolicyNumber, RepNumber, Month, DayOfMonth and DayOfWeek, I have replaced integer values of age with categorical age brackets that are usable by the models. After making sure our data is categorical, I encoded the features as dummy variables, scaled them and now they are ready to be fed to the model.

In the first step, Logistic Regression, SVM, Decision-Trees, Random Forest and KNN models are used without any balance or weight, and the result is as follows. It should be kept in mind that since data is imbalanced, accuracy is a misleading metric score, and model performance is better captured by ‘ROC\_AUC’ or area under ROC curve :

**Logistic Regression :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 4340 | 1 |
| Fraud | 283 | 2 |

|  |  |
| --- | --- |
| Accuracy | 0.94 |
| Precision Score | 0.94 , 0.66 |
| Recall Score | 0.9997 , 0.007 |
| F1 Score | 0.97 , 0.14 |
| ROC-AUC Score | 0.5034 |

Because fraud cases are very uncommon, even though this model has not predicted even a single case of fraud, it still has a high accuracy. It, however, scores close to the minimum of 0.5 in ROC-AUC score which means it is terrible.

**SVM :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 4341 | 0 |
| Fraud | 285 | 0 |

|  |  |
| --- | --- |
| Accuracy | 0.94 |
| Precision Score | 0.94 , 0 |
| Recall Score | 1 , 0 |
| F1 Score | 0.97 , 0 |
| ROC-AUC Score | 0.5 |

Our regular unweighted SVM model is also completely useless, because it has not predicted a single case of fraud.

**Decision-Trees :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 4096 | 245 |
| Fraud | 223 | 62 |

|  |  |
| --- | --- |
| Accuracy | 0.90 |
| Precision Score | 0.95 , 0.20 |
| Recall Score | 0.94 , 0.22 |
| F1 Score | 0.95 , 0.21 |
| ROC-AUC Score | 0.580 |

Our decision-tree model, however, has a slightly better performance, but it is still very unreliable.

**KNN :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 4317 | 24 |
| Fraud | 279 | 6 |

|  |  |
| --- | --- |
| Accuracy | 0.94 |
| Precision Score | 0.94 , 0.2 |
| Recall Score | 0.99 , 0.02 |
| F1 Score | 0.97 , 0.038 |
| ROC-AUC Score | 0.508 |

Our KNN model also shows poor performance and misses most of the fraud cases.

**Random Forest :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 4341 | 0 |
| Fraud | 285 | 0 |

|  |  |
| --- | --- |
| Accuracy | 0.94 |
| Precision Score | 0.94 , 0 |
| Recall Score | 1 , 0 |
| F1 Score | 0.97 , 0 |
| ROC-AUC Score | 0.5 |

The Random Forest model also misses all the fraud cases.

Huge disparity between accuracy and ROC-AUC shows clearly that the data is imbalanced, therefore we utilize some techniques to make our models more adapted to our dataset. In our first approach, we add weights to fraud class so that our models are more significantly penalized for skipping fraud cases. Secondly, we try undersampling the dataset to provide our models with a balanced dataset for training. As we will see further, both of these methods are very effective in improving the performance of our models.

**Class Weighting**

For this step I have ran an intensive hyperparameter grid to calculate an effective weight for out Logistic Regression model and the result was :

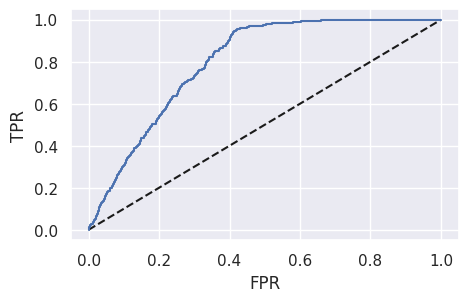
W = { 0:1 , 1:49.8}

Therefore we move on to create a model that penalizes skipping fraudulent cases over 50 times the skipping of non-fraudulent cases.

**Weighted Logistic Regression :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 2497 | 1844 |
| Fraud | 13 | 272 |

|  |  |
| --- | --- |
| Accuracy | 0.60 |
| Precision Score | 0.99 , 0.13 |
| Recall Score | 0.58 , 0.95 |
| F1 Score | 0.73 , 0.23 |
| ROC-AUC Score | 0.765 |



The results show a drastic improvement in the model’s effectiveness in detecting fraud. Although its accuracy is low, it detects almost all fraudulent claims and maintains a good balance in its predictions. Area under the ROC curve reflects the ROC-AUC score of the model.

**Weighted SVM :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 3494 | 847 |
| Fraud | 137 | 148 |

|  |  |
| --- | --- |
| Accuracy | 0.79 |
| Precision Score | 0.96 , 0.15 |
| Recall Score | 0.80 , 0.52 |
| F1 Score | 0.88 , 0.23 |
| ROC-AUC Score | 0.66 |

The performance of our SVM model is significantly improved under new class weights, and it is able to predict more than half of fraud cases, while not compromising many non-fraudulent claims.

**Weighted Decision-Tree :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 4133 | 208 |
| Fraud | 237 | 48 |

|  |  |
| --- | --- |
| Accuracy | 0.90 |
| Precision Score | 0.95 , 0.19 |
| Recall Score | 0.95 , 0.17 |
| F1 Score | 0.95 , 0.18 |
| ROC-AUC Score | 0.56 |

The performance of our decision-tree, however, is still very poor, and it requires more adjustments.

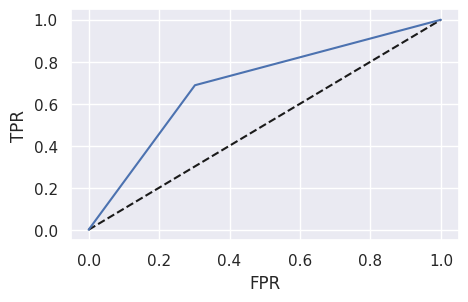
**Under-Sampling :**

Our other method of resolving data imbalance is to choose a smaller, more balanced sample from the data. We choose a subset of data for testing and training that contains as many non-fraudulent cases as fraudulent ones. Under this new sample performance of our decision-tree, KNN and random forest is improved noticeably.

**Under-sampled Decision-Tree :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 192 | 83 |
| Fraud | 87 | 192 |

|  |  |
| --- | --- |
| Accuracy | 0.69 |
| Precision Score | 0.69 , 0.70 |
| Recall Score | 0.70 , 0.69 |
| F1 Score | 0.69 , 0.69 |
| ROC-AUC Score | 0.69 |

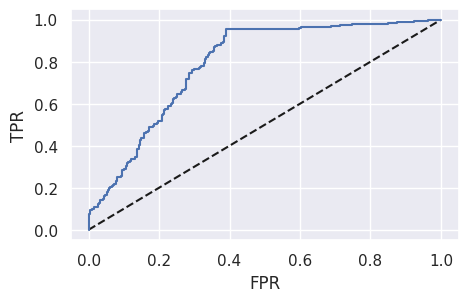


Under a more balanced subset of the data, decision-tree shows more balanced performance and is able to predict fraud cases more consistently.

**Under-sampled Random Forest :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 168 | 107 |
| Fraud | 14 | 265 |

|  |  |
| --- | --- |
| Accuracy | 0.78 |
| Precision Score | 0.92 , 0.71 |
| Recall Score | 0.61 , 0.95 |
| F1 Score | 0.74 , 0.81 |
| ROC-AUC Score | 0.78 |

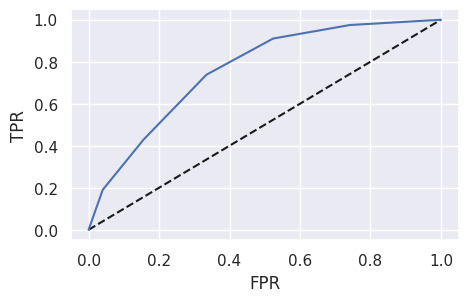


Performance of the random forest model is significantly better with under-sampling and it could reliably predict fraud cases. Even so, it sacrifices nearly half of its accuracy in predicting non-fraudulent cases.

**Under-sampled KNN :**

|  |  |  |
| --- | --- | --- |
|  | No Fraud Prediction | Fraud Prediction |
| No Fraud | 183 | 92 |
| Fraud | 73 | 206 |

|  |  |
| --- | --- |
| Accuracy | 0.70 |
| Precision Score | 0.71 , 0.70 |
| Recall Score | 0.67 , 0.74 |
| F1 Score | 0.69 , 0.71 |
| ROC-AUC Score | 0.70 |



Our KNN is also performing much better, though not as good as the random forest model. The area under the ROC curve also reflects this result.

**Summary**

Due to the high rarity of fraud cases, the data for vehicle insurance claims are imbalanced. This makes it difficult for predictive models to accurately predict fraud cases. Therefore, we have used class weighting and under-sampling to train models that are more sensitive to data from fraud claims and are more capable of recognizing them. Our weighted Logistic Regression and under-sampled Random Forest have shown the best performance with best ROC-AUC scores of 0.765 and 0.78 respectively.