Machine Learning Techniques in Automobile Domain

Akash Manjunatha
School of Computing
National College of Ireland
Dublin, Ireland
x21141797@student.ncirl.ie

***Abstract*—Five distinct machine learning modeling techniques will be used in this research to three data sets, two of which are regression and one classification predicting information related to the automobile domain: Random Forest regressor and classifier, as well as k-nearest neighbors (KNN) and XG boost, are used for both datasets. Support Vector Machine (SVM) is used for classification, lasso regression is used for regression datasets. A total of four algorithms are used on each dataset and for evaluation k-fold cross-validation, hyperparameter tuning with grid and randomsearchcv, feature importance are used for regression and for the classification to balance the outcome, the oversampling method is utilized, and tuning is done to the best model to produce the best output. On both regression datasets, XG boost outperformed with 92 and 93 percent accuracy in predicting automobile prices. XG boost surpassed random forest in the classification, predicting 91 percent of the time who will file a false insurance claim.**

Index Terms— KNN, SVM, AUC, RF, RMSE, R2

I. INTRODUCTION

Machine learning is a subbranch of artificial intelligence (AI) [1] that automatically develops and learns from past data. Machine learning is increasingly being used in our daily lives, such as picture recognition, speech recognition, driverless cars, and so on. In this project, some machine learning methods are employed to solve real-world challenges.

Over the last few years, due to the rise in the price of cars and the economic slump people have not been able to afford a brand-new car. Anticipating low-budget used car prices is challenging, this is where we need a platform that helps common people to predict car prices and used car firms to identify the price that has been quoted for the proper price.

One of the most significant aspects of insurance services is the insurance claim. The amount of money necessary to remedy the harm is referred to as claim severity [2]. Using a variety of indicators, the company needs a strategic model to anticipate the likelihood of a claim being submitted in the following year. In this project Several machine learning techniques are used to solve classification and regression problems. Models are created utilizing the gathered data, which are then compared and evaluated using multiple evaluation metrics to determine which model is the best.

A. Research question and approach summary.

1) Research question:

a) How to Predict used car prices using historical data of cars?

b) Do the additional features of cars help in predicting the car prices better?

c) Who are the target customers who are likely going to falsely claim the vehicle insurance?

The following approach is followed to answer the question

2) Regression dataset Car Price prediction Poland and Pakistan.

a) Feature engineering: For both datasets During the pre-processing period, columns are encoded when appropriate, the part manufacturer name and location are filtered, and unnecessary information is deleted. Prediction of car prices- Pakistan has a collection of vehicle accessories in one location; we divided those things and grouped them into three categories based on our subjective knowledge: entertainment, safety, and luxury for evaluation.

b) Cleaning: For the Poland dataset, we dropped the car price below 1970 as they termed vintage cars to have liner values for further processing, Outliers are removed according to the size of the data set and applied IQR method where ever necessary and mostly used trim method by seeing box plot to save loss of data

c) Transformation: For both the data set target variable is featured scaled [22] using box cox transformation Scaling was done on dependent variables using Minmax scalar for just one dataset (Car price prediction-Pakistan) that was run through the only algorithms that required scaling, Lasso[23] and KNN [24], and the results were compared to the unscaled dataset (Car price prediction-Poland)

d) Feature selection: The least correlation with the target and the largest multi-correlated values are examined using a correlation plot also VIF has been checked and those columns are eliminated only after running through the model.

e) Model building and evaluation: For the lasso regression [26] along with and without transformation of the dependent variable, for both the dataset we performed different evaluation processes one dataset with randomsearchcv and another with gridsearchCV to compare. For the KNN also with and without scalar transformation on dependent variables and evaluated with different k fold to find the better outcome.

For random forest for both the data set is compared running through all variables and only with important variables later evaluated with the best model with hyperparameter tuning

For XG boost both the data sets are compared run with and without hyperparameter tuning [8]

3) Classification Car Insurance Claim dataset

a) The same process has been carried out here too encoding multiple data with qualitative details, outliers with trimming, and correlation.

b) Modeling and evaluation: to compare and contrast all three data sets, we use three common algorithms (XG

boost, KNN, and random forest), which work well with both regression and classification, as well as another algorithm SVM, all of which are run with and without over sampling[9], with the best model with recall and area under curve selected and tuned to get the best result.

II. INITIAL LITERATURE REVIEW

Determining resale value is a difficult task. It is common knowledge, that the value of second-hand cars is governed by several factors, the most important of which is the car's age save for high-end historical cars, model, number of kilometers driven, car maker, and engine volume (horsepower) Other criteria include the fuel type, country of origin, color, body type, and safety features such as ABS and airbags.

In this research paper, Sameerchand Pudaruth [7] focused on the price prediction of second-hand cars in a small country. Various machine learning algorithms, such as naive Bayes, decision trees, k-nearest neighbors, and multiple linear regression, are employed in the work. to create various models. He used historical data from daily newspapers to compile his findings. To find the optimal model, each one is reviewed and compared. The study concludes that utilizing the KNN is a good idea.

[6] Support Vector Machines, Random Forest, and Artificial Neural Networks were used to create the three ML models in this research article. They collected the data from the autopijaca.ba website. To choose the best predictive model, all of the models are compared and evaluated. On test data, the top model had an accuracy of 87.38 percent.

[8]Linear regression, KNN, Random Forest, XG boost, and Decision tree were among the supervised machine learning algorithms employed in this study. To develop a statistical model for predicting the cost of an old car, a formative comparison was plotted between each model, with random forest regressor test accuracy being highest at 93.11 percent with the lowest Root Mean Squared Error and linear regression test accuracy being the lowest at 73.46 percent with the highest Root Mean Squared Error. However, because the number of observations was low, the dataset for generating a strong inference was rather modest. More data can lead to more accurate predictions.

In this research, regression methods such as lasso, multiple, and regression trees are utilized to construct a statical model. It is observed that the rates of error for lasso and multiple regression are not significantly different, implying that a higher machine learning approach would be a better choice.[26]

In this research, they used OLX data from an Indian notable automobile selling to forecast the price and then employed additional tree regression and random forest with the Scikit-Learn package. Learn to forecast the price regardless of dataset size. Both models predicted precise results, and randomsearchcv was used to evaluate them. we have used both grid and random searchCV in our dataset[27].

Several factors influence the cost of car insurance. And these variables would have an impact on the price of the purchaser's insurance coverage. One of these variables is credit history; previous study alarms that the person with a bad credit score are more intend to claim, make no payments, and commit deception. Other factors such as DUI, speeding violations, and previous accidents will have a significant impact on insurance claims, and would potentially put the insurance company in financial trouble[9].In this research paper over-sampler, and SMOTE methods are used to handle the heavy imbalanced data.

overall eight different classifiers are used to forecast claims occurrence using insurance data[10], The random forest model outperformed the other eight versions.

To forecast the model, three algorithms are used namely Multinomial Logistic Regression, Artificial Neural Networks, and Decision Trees. The results showed that Artificial Neural Networks had a 61.71[11] percent total classifier accuracy. The comparative investigation revealed the whole pattern between the decision tree and the neural network, proving that policies categorized incorrectly by one are classified properly by the other. This could indicate that combining the models will improve classification performance.

III. METHODOLOGY

In this research, we have used KDD (Knowledge Discovery in Databases) method. There are various steps involved in this process: Data Selection, Pre-processing and EDA, Data preparation, Model training, and Evaluation performance/Interpretation, Fig. 1 shows the flow chart of the same.

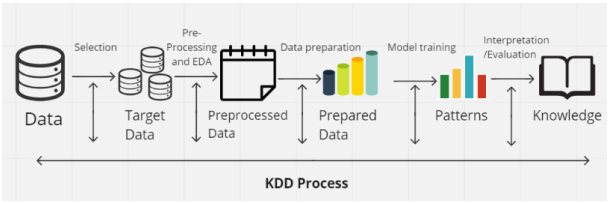


Fig. 1. KDD Flow Chart.

A. *Car Price Prediction (Poland Dataset-1)*

Step 1: Data Selection: Data is selected from the source Kaggle data, it was compiled on January 2022 from the well-known Polish car-sale company otomoto, number of records: 117,927, number of column 10, The goal of using this dataset is to examine how well a model performs when there are fewer dependent variables vs another dataset with more variables.[3] data described in Fig. 2

Column	Description
Mark	Carbrand
Model	Car model
Generation_name	Gen name of car
Year	Car Year
Mileage	Car Mileage in Kilometers (KM)
Vol_engine	Auto Engine Size
Fuel	Engine Type
City	Locality in Poland
Province	Region of Poland
Price	Price in PLN (approx. 1USD=1PLN)

Fig. 2. Column description.

Step 2: Pre-processing and EDA: In pre-processing null values are checked and it is found that there are 30 thousand missing values in the generation column as the column is unnecessary for the prediction, we removed it and no duplicates are found in the dataset. The fuel column had six types of qualitative variables that are encoded and later it is one hot encoded to improve the prediction results. The

column mark car brand had 23 unique values, the model column had just its variants, city and provinces are dropped for further analysis

Exploratory data analysis is performed to check for insights, A scatter plot is plotted against the price and the year(Fig. 3) the car was purchased; as the year decreases, the price increases, which makes sense for used cars; also, the cars between 1960 and 1970 are price tagged more, possibly because they are treated as vintage cars; we dropped those values to have liner values for further processing.

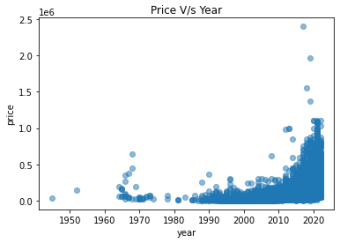


Fig. 3. Price VS Year Plot.

Outliers: Box plots Fig 4 is used to look for outliers and skewness in variables such as years, price, engine volume, and mileage. Initially, the IQR method is used to remove the outliers, which resulted in the removal of 22909 values in the data, which was a huge amount of data to be dropped off, as a result, we removed the outliers by trimming their values manually approximately 1960 number outliers' rows were removed from the dataset.

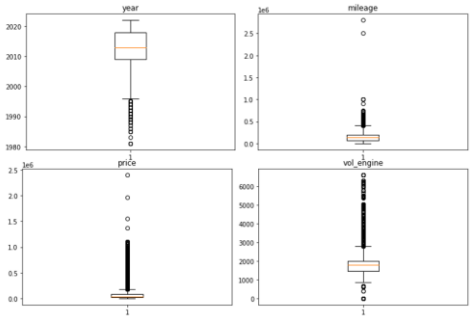


Fig. 4. Box Plot.

The skewness value for the price and engine volume was found to be more than one as we are using the regression algorithms in the evaluation it is necessary to transform the variables box cox transformation used to get the normalized values, For the engine value, we tried many transformation methods but the best was found to be box cox, at last, so we utilized the same transformation, shown in Fig. 5.

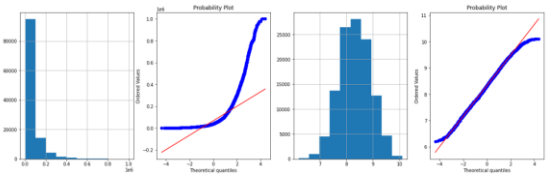


Fig. 5. Left: Before and Right: After Transformation plot.

The correlation plot Fig. 6 is used to check for the collinearity between the target price and the dependent variables as well as multicollinearity among the dependent variables. Mileage and year, as well as fuel types 3 and 6 (gasoline and diesel), are found to be negatively correlated with more than 0.5 value. as we have a few number of columns for the prediction and also algorithms like lasso regression have the ability to remove useless variables from the equation to build the model so we ignored them.

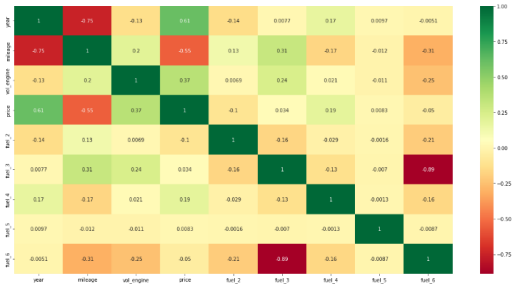


Fig. 6. Correlation Plot.

Step 3: Data Preparation: Data set is divided into the test and train split using sklean-train_test_split subset in 75:25 ratio, we decided to apply lasso regression, random forest, KNN, and XGboost regressor, to the dataset, and all these algorithms are tested and results are measured in terms of accuracy, Mean square error, RMSE (Root mean square error), R2 and adjusted R2

Step 4: Model Training and Evaluation:

Model 1: Lasso regression: as we include all the dependent variables, the lasso eliminates the unnecessary variables and reduces the variance in the model.

For the first model, we set the lasso alpha parameter as 1 or a full penalty which gives us a decent RMSE value of 0.34 but an accuracy of 64 by changing the alpha values we can get the better results by changing the value from 1 to 0.5(changes the slope of best fit line to fit) accuracy jumped to 81% to obtain the best outcome We employ Random search hyperparameter tuning, which picks a different parameter from the ones we provide it at random, based on a random search, it believes that the higher the entropy, the better the result.

Improve model performance: the sci-kit library built-in algorithm random searchCV is used and the number of iterations passed is 10 In the parameter distribution we pass the alpha dictionary to find the best alpha value bypassing this the best estimator alpha was found to be 0.05, best score 0.86 When the alpha value is added to the model, the accuracy is determined to be 86% and the RMSE is 0.2, which is deemed to be a superior model overall.

Model 2: Random Forest: It is widely used for both classification and regression models initially

Predictions on the testing data were used to evaluate this model. The random forest model had an accuracy of 84 percent and an RMSE of 0.23, and the number of estimators was initially set to ten. The model was run through all of the dependent variables in the model.

Improve model performance: to improve the model efficiency, we check the number of important features affecting the price by using sklearn library feature importance, Fig. 7 it is found only variables year and volume of the engine are affecting the price so we consider only

relatively important variables in the next step and the model is tuned to check the best fit, the result of the fit inserted for new model evaluation (n_estimator = 200 and random state = 0) result was found to be fruit full accuracy increased by 91 percentage and R-mean square reduced almost half the value predicted earlier 0.172

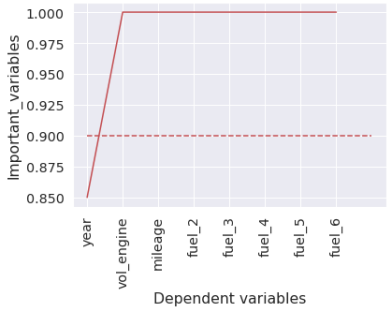


Fig. 7. Important features Plot.

Model 3: KNN: It can be employed for both regression and the classification example, its goal is to find out the new unknown data point to which it belongs by looking up all of its neighbors the employed model with no parameter gives the accuracy of 68 percentage and MSE value of 0.1.

Model improvement: we perform the manual tunning by sending k value from 1 to 15, the result is evaluated by selecting the least RMSE score and Fig. 8. is plotted RMSE vs k from the plot it can be seen that k =4 has the least score and gives the best outcome we employed the same on the model the accuracy improved by two percentage and MSE was found to be 0.313

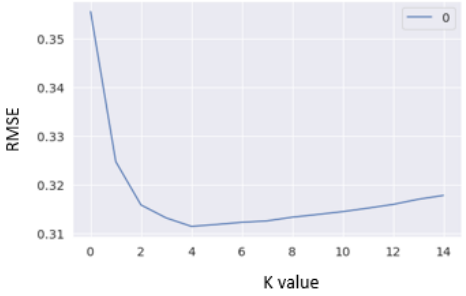


Fig. 8. RMSE Vs K-value plot.

Model4: XG boost: Boosting is an ensemble strategy that produces numerous individual models sequentially, similar to training. Each successive model tries to correct the mistakes of the prior batch. XG boost regressor is applied on the dataset by randomly setting parameters column sample by the tree to 0.7, learning rate to 0.1, depth to 6, and estimator to 100, the model with the set parameter gave the accuracy of 0.91 percentage and 0.028 mean square error which found to be the best so far.

Improving model performance: we employ gridsearchCV to find out the best-fit parameters and We imply the same on the model to get the least MSE value, Grid search cv make a list of all the possible combination parameter and runs through the model and picks up the best parameter so as we study the model employing those values already, we pass three different possible combinations in parameters except for the estimator 10 and 100. grid picks the best-fit parameters as shown in Fig. 9. The updated model now has an accuracy of 92 percent and the lowest and best RMSE of 0.16.

```
[15:21:33] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in fa
[15:21:40] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in fa
Best parameters: {'colsample_bytree': 0.7, 'learning_rate': 0.1, 'max_depth': 10, 'n_estimators': 100}
Lowest RMSE:  0.16163745420604594
```

Fig. 9. Tuned Result.

Step 5: Interpretation of results:

Car price prediction(Poland)				
Evaluation	Lasso Model		KNN	
	Before tuning	After Tuning	Before tuning	After Tuning
Accuracy	0.81	0.86	0.68	0.7
MSE	0.06	0.04	0.1	0.09
RMSE	0.25	0.21	0.32	0.313
R Square	0.81	0.86	0.9	0.83
Adjusted R2	0.81	0.86	0.9	0.83
Evaluation	**Random Forest**		**XG Boost**	
Accuracy	0.84	0.91	0.91	0.92
MSE	0.05	0.02	0.02	0.02
RMSE	0.23	0.17	0.16	0.16
R Square	0.84	0.91	0.91	0.93
Adjusted R2	0.84	0.91	0.91	0.93

Fig. 10. Final Model Summary Car price prediction Dataset 1.

The table Fig. 10 shows that the Tuned XG boost outperformed all other models in predicting car prices, with a 92 percent accuracy and a 0.02 least RMSE. R squared value was also discovered to be superior to the other algorithms. The random forest results were found to be the next best.

B. Car Price Prediction (Pakistan Dataset-2)

Step 1: Data Selection: Data is selected from the source GitHub, it was scaped from Pak wheels, number of records: 46,024, number of column 16, The goal of using this dataset is to examine how well a model performs when there are more dependent variables vs another dataset with fewer variables.[5] data description is shown in Fig. 11.

Column	Description
Ad No	Unique ID
Name	Name of the car includes the make, model, year and variant of a car
Price	Listed price of the car
Model Year	Model year of the car
Location	Locality of the car owner
Mileage	How much the car has travelled
Registered City	Registration city of the car
Engine Type	Engine type is broken down in 1 to 3 where: 1 -> Petrol 2 -> Diesel 3 -> Hybrid
Engine Capacity	Engine capacity of the car
Transmission	Transmission is broken down in 1 to 2 where: 1 -> Automatic 2 -> Manual
Color	Colour of the car
Assembly	Assembly Imported or Local
Body Type	Body type is broken down in 1 to 6 where: 1 for Hatchback 2 for Sedan 3 for Suv 4 for mini van 5 for Crossover 6 for Van
Features	includes extra features in car
Last Updated	Updated Date
URL	URL

Fig. 11. Column description.

Step 2: Pre-processing and EDA: The null values detected in the features are eliminated as these are qualitative data to perform further analysis

Feature engineering: part manufacturer name and location are filtered and extraneous information is removed during the pre-processing period.

The column features had various automotive accessories listed in one place; we separated those items, which added an additional twenty-eight columns to the data frame, so we categorized those features into three categories based on subjective knowledge: entertainment, safety, and luxury.

First, if those traits were present in that row, we programmatically assigned them a binary value of one, and then we added them to the category column to check if those values are helpful for predication, We hot encoded those categorical values and plotted the correlation matrix, which revealed that they are least connected to price prediction, and we also investigated using the same value on models and discovered that these feature values are not useful, therefore we dropped them for future investigation.

A few columns contain qualitative information that must be transformed into binary for the computer to recognize them as categories. those are one hot encoded for further analysis.

Data Cleaning and transformation: The outliers present in the price columns are removed by the IQR method and other columns' outliers are trimmed per the values displayed in the box plot Fig.12, the target column is transformed using box cox.

We found the multicollinearity between Engine Capacity & Body Type_2 with more than 0.6 value, removed the body type column, and dropped other unnecessary columns present in the data frame.

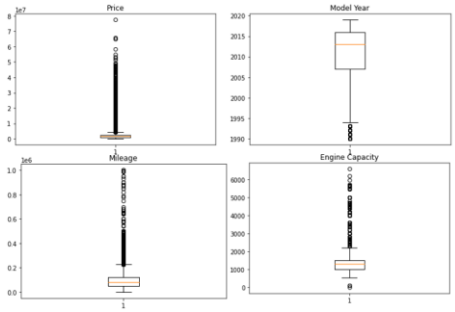


Fig. 12. Tuned Result.

EDA is used to extract interesting insights: as the year progresses, the price of the car rises; as the number of kilometers driven decreases, the price of the car rises; petrol cars are the most popularly listed on the website; and, surprisingly, there are more unregistered cars listed in the website than registered ones.

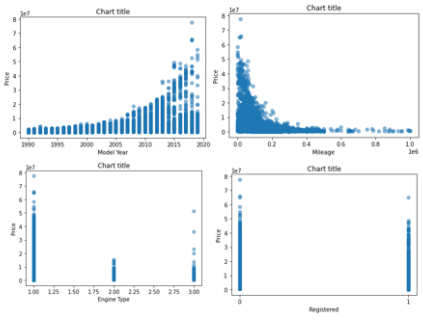


Fig. 13. Price VS Model Year, Milage, Engine type, and Registration Plot.

Step 3: Data Preparation: Data set is divided into the test and train split using sklean-train_test_split subset in 75:25 ratio, same algorithms that we used prior is used for this data set too but during the division of training and testing we apply min-max scalar transformation to the dependent variables for only on the lasso regression[23] and KNN algorithm [24] it is necessary to transform the data with different units to make the system to understand and to avoid being biased by large numbers, here, an attempt has been made to see how the model performs with the transformation and to compare it to the first dataset(Car price prediction Poland), which was processed without alteration. results are measured in terms of accuracy, Mean square error, RMSE (Root mean square error), R2, and adjusted R2.

Step 4: Model Training and Evaluation:

Model 1: Lasso regression: before applying the regression to the model we transform training and test data, when we set the alpha value to zero, the model becomes the least squared method; when we change the lambda value, we discover the best line of fit with reduced variance.

For the first model we set the lasso alpha parameter as 0.1 which gives us the RMSE value of 5.8 and 85 percent accuracy, as discussed earlier by changing the lambda value we can find the best fit for this time we employ grid searchCV to find the alpha value unlike the random searchCV Grid search cv make a list of all the possible combination parameter and runs through the model and picks up the best parameter

Improve model performance: We use the sklearn library built-in algorithm Grid search cv for this We pass three random values of 0.0001, 0.01, and 0.0005 in the parameter tuning, we use k-fold evaluation process and the number of iterations passed is 10 In the parameter distribution we pass the alpha dictionary to find the best alpha value bypassing this the best estimator alpha was found to be 0.0001, best score found to be 0.86 When the alpha value is added to the model, the accuracy is determined to be 86.6 and the RMSE is 5.6.

Model 2: KNN: We preserve the test and training data in a transformed state for the KNN assessment as well. When we run the model with the random parameter value set, the accuracy was found to be 90% and the RMSE value was 4.7, which was lower than the tuned random forest value, we re-ran the training split to pick a random number and re-ran the model to see whether it was due to overfitting. The value was the same, therefore we fine-tuned the model to get the least RMSE value and the best fit k value. Fig. 14 depicts after 8 all the values are susceptible to choose, we run the tuned model the accuracy jumped two percent and the RMSE value was found to be 4.17

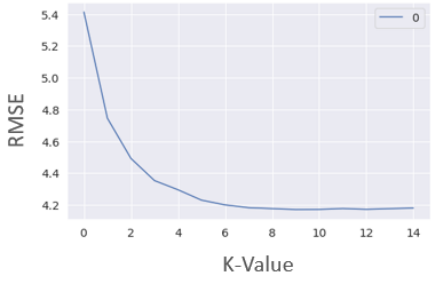


Fig. 14. RMSE Vs K-value plot.

Model 3: Random forest: This model was evaluated based on its test data predictions. The random forest model was found to have a 60 percent accuracy and a 9.7 RMSE. and the number of estimators is set to 10(The algorithm generates the number of trees before averaging for prediction), random state to 42(randomness of sample

controlled by this), depth to 2 and min split 5 initially the model was run through all the variables (8) in the model[25].

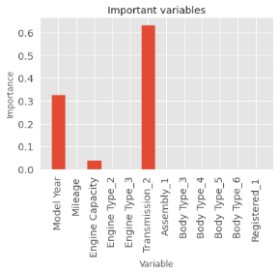


Fig. 15. Important features Plot.

Improving model performance: To improve model efficiency, we use the sklearn library feature importance to check the number of important features affecting the price. As shown in Fig 15, the variables model year, engine capacity, and transmission 2 volume of the engine all affect the price, so we only consider the most important variables for future modelling. and to tweak the model, we assess the best fit, the result of the fit inserted for new model evaluation(n estimator = 200 and random state = 0) result was felt to be fruitful and best so far the accuracy went up to 91 percent, and root mean square dropped about half the value predicted earlier 4.5.

Model 4: XG boost: XG boost regressor is applied to the dataset without setting any parameters, the model gave the accuracy of 92% percentage and 4.2 RMSE which was found to be the best so far.

Improving model performance: we employ gridsearchCV to find out the best-fit parameters and imply the same on the model to get the least RMSE value, Grid search cv makes a list of all the possible combination parameters and runs through the model, and picks up the best parameter.

Implementing hyperparameter tuning, we pass three different possible combinations in parameters max depth 3,6,10, learning rate 0.01, 0.05, and 0.1, and the estimator 10 and 100. Finally grid pics the best-fit parameters as shown in Fig. 16. The updated hyper-parameter tuned model produced an accuracy of 93.4 percent and the lowest and best RMSE of 3.9. found to be the best value so far.

```
[14:21:03] WARNING: /workspace/src/objective/regression_obj.cu:152: reg:linear is now deprecated in fa
Best parameters: {'colsample_bytree': 0.7, 'learning_rate': 0.1, 'max_depth': 10, 'n_estimators': 100}
Lowest RMSE:  4.009128941637115
```

Fig. 16. Tuned values.

Step 5: Interpretation of results:

Car price prediction (Pakistan)				
Scaled/Evaluation	Lasso Model		KNN	
	Before tuning	After Tuning	Before tuning	After Tuning
Accuracy	0.85	0.866	0.9	0.92
MSE	34.15	31.79	22.5	17.39
RMSE	5.84	5.6	4.74	4.17
R Square	0.85	0.86	0.95	0.93
Adjusted R2	0.85	0.86	0.95	0.93
Without Scaled/Evaluation	Random Forest		XG Boost	
Accuracy	0.6	0.91	0.92	0.93
MSE	94.09	20.29	17.85	15.73
RMSE	9.7	4.5	4.22	3.96
R Square	0.59	0.92	0.92	0.95
Adjusted R2	0.59	0.92	0.92	0.95

Fig. 17. Final Model Summary Car price prediction Dataset 2.

Fig. 17 with a 93 percent accuracy and a 15.75 least RMSE, Tuned XG boost surpassed all other models in predicting car pricing, according to the table. The R squared value was also found to outperform the other techniques. The KNN findings came in second place.

C. Car Insurance Claim(Dataset-3)

Step 1: Data Selection: Data is selected from the source Kaggle data, number of records: 10,001, number of column 19, The goal of using this dataset is to examine how well a model performs in detecting false claims and also it relates to the other two datasets which come under automobile domain [4].

Step 2: Pre-processing and Exploring the data:

We check for null values in the dataset during pre-processing, and we find 900+ null values in the credit score and annual mileage columns. We also eliminate 12 duplicate values identified in the data set. To fill the credit score null value, the plot Fig. 18 is checked, and it can be seen that the Poverty class has the lowest score, while the upper class has the highest score. To fill null values, we group by income categories and take the median of each and fill the same to the grouped null values of those categories in the credit score null values, whereas in annual mileage median is used to fill the null values.

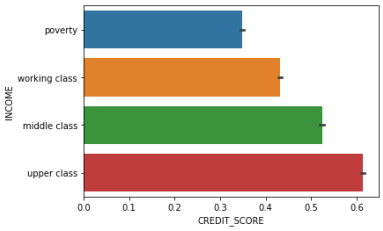


Fig. 18. Income VS Credit score bar plot.

Outliers: Because the dataset has less number of rows, we only removed 300 extreme outliers by looking at the box plot. The dataset contains a greater number of qualitative data columns, encoding was necessary. We binary encoded four columns and label encoded another four.

There are a greater number of columns, we used a correlation plot to check for multicollinearity. We then eliminated the least correlated and one highly correlated column before continuing with the model-building procedure.

EDA: Some interesting insights are discovered in the dataset Fig. 19 customer with an increase in the driving experience are less likely to claim or fraud There is more number of female customers than males in the dataset, men are quite likely to claim insurance than women.

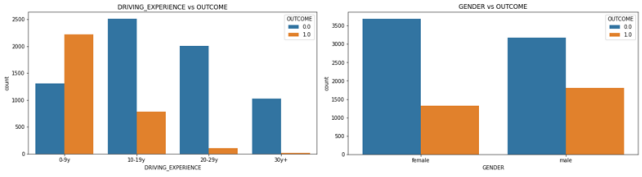


Fig. 19. Left Driving_Experience Vs Outcome, Right Gender Vs Outcome.

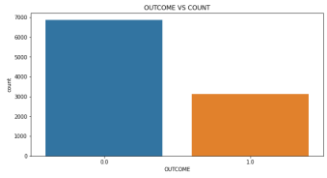


Fig. 20. Outcome count.

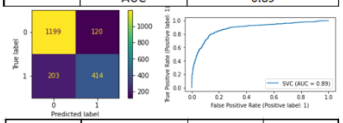
Step 3: Data Preparation: By seeing the plot Fig. 20, we can say that outcome is quite imbalanced. After the cleaning, we divide the data into 80% for training and 20% for testing as we have less data size. The random state has been set to 42 which controls the shuffling before applying split and for result reproducibility and Random oversampling technique is used to increase the number of defaulter/claims count to balance the data and analyse (Oversampling is because we have less data size), the number of claims increased from 2998 to 6679.

Step 4: Model Training and Evaluation: Model Training: In this dataset four algorithms are used: Support Vector Machine (SVM), KNN, Random Forest classifier, and XG boost. The results of each algorithm are compared with and without oversampling, and hyperparameter tuning is applied for the best results. For evaluation, the accuracy scores on the training set and test set, recall, precision, F1 score, and AUC were computed and printed.

Model 1: SVM: Without oversampling: The SVM model has been evaluated on the testing dataset and the model correctly predicted 414 customers who are going to claim out of 617 test data and 1199 good customer who is not going to false claim out of 1319 numbers. Since our data set is imbalance instead of accuracy we focus on the precision, recall, and F1 score values of the same found to be 78, 67, and 72 respectively (In the evaluation more focus is given to the one who is going to fraud claim) and area under the curve found to be 0.89

With oversampling: As we can see due to the presence of less claimed sample there is a huge difference between the claimed and without claimed recall, the random sampler is used to equate the outcome and the train and test split is again applied and run through the same model the result was found to be increased to 83,82 and 82 for precision, recall and F1 score and AUC of the curve increased to 0.9

SVM	Evaulators	Not Claimed	Claimed
Without Oversample	Precision	0.86	0.78
	Recall	0.91	0.67
	f1-score	0.88	0.72
	Training accuracy	0.83	
	testing accuracy	0.83	
	AUC	0.89	



SVM	Evaulators	Not Claimed	Claimed
With Oversample	Precision	0.82	0.83
	Recall	0.83	0.82
	f1-score	0.83	0.82
	Training accuracy	0.83	
	testing accuracy	0.83	
	AUC	0.9	

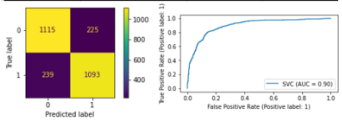


Fig. 21. Result summary of Model 1.

Model 2: Same methodology used in the KNN algorithm too without Oversample: Out of 617 test data, the model properly predicted 399 customers who will file a claim, which is found to be less than the SVM.

With Oversample: Out of 1332 test data, the model predicted 1104 customers who would file a claim. The training and testing accuracy of the model was found to be 82 and 81 percent, respectively. The model increased the recall value from 65 to 83 percent, and the area under the curve was determined to be 0.9 percent.

KNN model	Evaulators	Not Claimed	Claimed
Without Oversample	Precision	0.85	0.77
	Recall	0.91	0.65
	f1-score	0.88	0.7
	Training accuracy	0.83	
	testing accuracy	0.82	
	AUC	0.89	
With Oversample	Precision	0.83	0.81
	Recall	0.81	0.83
	f1-score	0.82	0.82
	Training accuracy	0.82	
	testing accuracy	0.81	
	AUC	0.9	

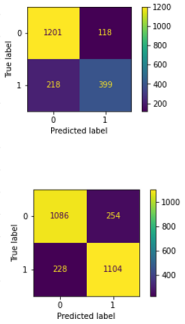


Fig. 22. Result summary of Model 2.

Model 3: Random Forest classifier Without Oversample: The model has been evaluated on the testing dataset and the model correctly predicted 404 customers who is going to claim out of 594 test data. There is an improvement in the claimed percent recall when compared to the prior model. This model is promising, with the superior training and test accuracy of 87 and 84 percent, respectively, and an AUC curve of 0.89.

With Oversample: Both claimed and unclaimed accuracy, recall, and F1 score gives a nearly identical percentage of results in this evaluation and is judged to be best equally 0.85 value and AUC jumped to 0.93, According to this, the classifier can almost correctly distinguish between claimed and unclaimed values.

RandomForest Classifier		Not Claimed	Claimed
Without Oversample	Precision	0.87	0.77
	Recall	0.91	0.7
	f1-score	0.89	0.73
	Training accuracy	0.87	
	Testing accuracy	0.84	
	AUC	0.89	
With Oversample	Precision	0.86	0.85
	Recall	0.84	0.85
	f1-score	0.85	0.85
	Training accuracy	0.87	
	Testing accuracy	0.85	
	AUC	0.93	

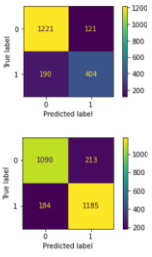


Fig. 23. Result summary of Model 3.

Model 4: XG boost: For without oversample evaluation XG boost give the best recall result for the claimed percent prediction of 73 which is determined to be superior to the random forest, which predicted 70 percent.

With oversample: The model did not perform well when compared to the Random Forest classifier, thus, hyperparameter tuning is performed, grid searchCV with 5 cross-fold validation and different number estimator is set to

improve the model, With an AUC of 0.94 and an F1 score of 0.89, the algorithm produced the best recall result, accurately predicting 91 percent of default claims and having the lowest false negative value.

XG boost	Evaulators	Not Claimed	Claimed	Hyper parameter Tuned result	
Without Oversample	Precision	0.88	0.79		
	Recall	0.91	0.73		
	f1-score	0.89	0.76		
	Training accuracy	0.84			
	testing accuracy	0.85			
	AUC	0.91		Not Claimed	Claimed
With Oversample	Precision	0.83	0.84	0.9	0.86
	Recall	0.84	0.83	0.84	0.91
	f1-score	0.84	0.83	0.87	0.89
	Training accuracy	0.83		0.93	
	testing accuracy	0.83		0.87	
	AUC	0.91		0.94	

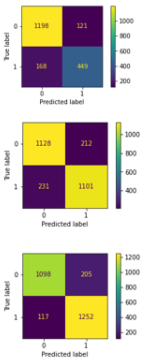


Fig. 24. Result summary of Model 4 and Confusion matrix: top right for with-out oversample, middle for with oversample, bottom for Tuned result.

Step 5: Interpretation of results:

From the Table Fig. 24 the Tuned XG boost yielded the best recall result, correctly predicting 91 percent of default claimants and having the lowest false-negative value of any model with an AUC of 0.94 and F1 score of 0.89.

IV. CONCLUSION AND FUTURE WORK

Regression datasets 1 and 2: The table shows that the Tuned XG boost outperformed all other models in both datasets, with the accuracy of 92 and 93 percent in predicting car price and least RMSE of 0.02 and 15.73, respectively. The question at the outset, whether the extra features help predict better results, can be answered by saying yes, at least by one percentage, also R squared value for dataset-2 was found to be superior to the dataset-1. The random forest results are not to be overlooked, as they are found to be the next best with a 1-2 percent difference. A noteworthy observation is that the random forest took less time than the XG boost, as grid tuning for the XG boost took almost 10 minutes to complete, assuming how much longer it would take with big data.

Final Model Summary of two Dataset				
Car price prediction(Poland)-Dataset-1				
Evaluation	Without scaling		Random Forest	XG Boost
	Lasso Regression	KNN		
Accuracy	0.86	0.7	0.91	0.92
RMSE	0.04	0.09	0.02	0.02
MSE	0.21	0.313	0.17	0.16
R Square	0.86	0.83	0.91	0.93
Adjusted R2	0.86	0.83	0.91	0.93
Car price prediction(Pakistan)-Dataset-2				
Evaluation	With scaling		Random Forest	XG Boost
	Lasso Regression	KNN		
Accuracy	0.866	0.92	0.91	0.93
RMSE	31.79	17.39	20.29	15.73
MSE	5.6	4.17	4.5	3.96
R Square	0.86	0.93	0.92	0.95
Adjusted R2	0.86	0.93	0.92	0.95

Fig. 25. Final Model Summary Car price prediction Dataset-1 and 2.

Classification dataset: For algorithm lasso and KNN, we ran the model without scaling for dataset 1 and did for dataset 2. The observation found in lasso regression does not change much, but for the KNN, we can find a dramatic rise in the result from 70% to 92 %, making it the second-best result in dataset 2. which depicts scaling is necessary for KNN finding the nearest neighbors and predicting the precise result.

Each algorithm's results were compared with and without oversampling, and recall, precision, F1 score, and AUC have calculated the results from the oversample were determined to be superior and for the best model XG boost result, hyperparameter tuning has been used. Table Fig. 26 shows a summary of all the best model outcomes.

Final model summary (with oversample)-Dataset 3								
Algorithms/Evaulation	SVM		KNN		Random forest		XG Boost Tuned	
	Not Claimed	Claimed	Not Claimed	Claimed	Not Claimed	Claimed	Not Claimed	Claimed
Precision	0.82	0.83	0.83	0.81	0.86	0.85	0.9	0.86
Recall	0.83	0.82	0.81	0.83	0.84	0.85	0.84	0.91
f1-score	0.83	0.82	0.82	0.82	0.85	0.85	0.87	0.89
cohen kappa score	0.65		0.63		0.7		0.76	
AUC	0.9		0.9		0.9		0.94	

Fig. 26. Final Model Summary Car insurance claim Dataset-3.

maybe due to the balanced dataset, all models appear to have predicted superior results along with an AUC greater than 0.9. The best model XG boost has a test accuracy of 0.87 and a training accuracy of 0.93. The high accuracy score suggests that there is little prejudice. The difference in accuracy between train and test is about 0.06 points. As a result, this model is stated to have low variance, which could be attributed to oversampling. Tuned XG boost has AUC of 94%, cohen kappa score of 0.76 which is close to one and has the best recall, properly forecasting 91 percent of default claimants and having the lowest false negative, which aids the company in anticipating who will default rather than wasting money on who is the ideal client.

Future work in the regression dataset would have been better if there were more dependent columns in the first dataset, and in the second dataset, the features column had different accessories list classifying those by rating would have produced fruitful results, and different cleaning and filling NA methods could have been used for better accuracy. Different samping approaches can be used in the classification dataset for better results, Only an imbalanced dataset with varied tuning settings could be used to evaluate the dataset.

V. REFERENCES

[1] By:IBM Cloud Education, “What is machine learning?,” IBM, 15-Jul-2020.[Online]. Available: https://www.ibm.com/cloud/learn/machine-learning.

[2] K. C. Dewi, H. Murfi, en S. Abdullah, “Analysis Accuracy of Random Forest Model for Big Data--A Case Study of Claim Severity Prediction in Car Insurance”, in 2019 5th International Conference on Science in Information Technology (ICSITech), 2019, bll 60–65.

[3] A. Glotov, “Car prices Poland,” Kaggle, 20-Jan-2022. [Online]. Available:https://www.kaggle.com/aleksandrglotov/car-prices-poland.

[4] S. Roy, “Car Insurance Data,” Kaggle, 05-Jul-2021. [Online]. Available: https://www.kaggle.com/sagnik1511/car-insurance-data.

[5] A.AliDD, “Asadalidd/PKWHEELSSCRAPER: Scrapy based ad scraper from Pakwheels.com,” GitHub, 11-Jul-2020. [Online]. Available:https://github.com/AsadAliDD/pkwheelsscraper.

[6] GEGIC, E., ISAKOVIC, B., KECO, D., MASETIC, Z. and KEVRIC, J., 2019. Car Price Prediction using Machine Learning Techniques. TEM Journal, 8(1), pp. 113-118.

[7] Sameerchand Pudaruth. Predicting the price of used cars using machine learning techniques. 4(7):753–764.

[8] P. Gajera, A. Gondaliya, en J. Kavathiya, “Old Car Price Prediction With Machine Learning”, Int. Res. J. Mod. Eng. Technol. Sci, vol 3, bll 284–290, 2021.

[9] M. Hanafy en R. Ming, “Classification of the Insureds Using Integrated Machine Learning Algorithms: A Comparative Study”, Applied Artificial Intelligence, bll 1–32, 2022.

[10] M. Hanafy en R. Ming, “Machine learning approaches for auto insurance big data”, Risks, vol 9, no 2, bl 42, 2021.

[11] K. Weerasinghe en M. C. Wijegunasekara, “A comparative study of data mining algorithms in the prediction of auto insurance claims”, European International Journal of Science and Technology, vol 5, no 1, bll 47–54, 2016.

[12] K. C. Dewi, H. Murfi, en S. Abdullah, “Analysis Accuracy of Random Forest Model for Big Data--A Case Study of Claim Severity Prediction in Car Insurance”, in 2019 5th International Conference on Science in Information Technology (ICSITech), 2019, bll 60–65.

[13] Analytics Vidhya. 2022. KNN - The Distance Based Machine Learning Algorithm. [online] Available at: <https://www.analyticsvidhya.com/blog/2021/05/knn-the-distance-based-machine-learning-algorithm/>.

[14] "KNN-The Distance Based Machine Learning Algorithm", AnalyticsVidhya,2022.[Online].Available:https://www.analyticsvidhya.com/blog/2021/05/knn-the-distance-based-machine-learning-algorithm/.

[15] "How Lasso Regression Works in Machine Learning", Dataaspirant, 2022. [Online]. Available: https://dataaspirant.com/lasso-regression/.

[16] J. Brownlee, "What is a Confusion Matrix in Machine Learning", Machine Learning Mastery, 2022. [Online]. Available: https://machinelearningmastery.com/confusion-matrix-machine-learning/.

[17] "How To Interpret R-squared in Regression Analysis - Statistics By Jim", Statisticsbyjim.com, 2022. [Online]. Available: https://statisticsbyjim.com/regression/interpret-r-squared-regression.

[18] "Evaluating linear regression models using RMSE and R²", Medium, 2022. [Online]. Available: https://medium.com/wwblog/evaluating-regression-models-using-rmse-and-r%C2%B2-42f77400efee.

[19] "AUC-ROC Curve in Machine Learning Clearly Explained - Analytics Vidhya", Analytics Vidhya, 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/.

[20] M. Hanafy en R. Ming, “Improving imbalanced data classification in auto insurance by the data level approaches”, International Journal of Advanced Computer Science and Applications (IJACSA), vol 12, no 6, bl 2021a, 2021.

[21] K. C. Dewi, H. Murfi and S. Abdullah, "Analysis Accuracy of Random Forest Model for Big Data – A Case Study of Claim Severity Prediction in Car Insurance," 2019 5th International Conference on Science in Information Technology (ICSITech), 2019, pp. 60-65, doi: 10.1109/ICSITech46713.2019.8987520.

[22] J. Brownlee, "How to use Data Scaling Improve Deep Learning Model Stability and Performance", Machine Learning Mastery, 2022. [Online]. Available: https://machinelearningmastery.com/how-to-improve-neural-network-stability-and-modeling-performance-with-data-scaling/.

[23] "Why is scaling required in KNN and K-Means?", Medium, 2022. [Online]. Available: https://medium.com/analytics-vidhya/why-is-scaling-required-in-knn-and-k-means-8129e4d88ed7.

[24] "When and why to standardize or normalize a variable? | Data Science and Machine Learning", Kaggle.com, 2022. [Online]. Available: https://www.kaggle.com/questions-and-answers/59305.

[25] "RandomForest| Introduction to Random Forest Algorithm", Analytics Vidhya, 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/.

[26] P. Venkatasubbu και M. Ganesh, ‘Used Cars Price Prediction using Supervised Learning Techniques’, Int. J. Eng. Adv. Technol. (IJEAT), τ. 9, τχ. 1S3, 2019.

[27] A. Pandey, V. Rastogi, και S. Singh, ‘Car’s selling price prediction using random forest machine learning algorithm’, στο 5th International Conference on Next Generation Computing Technologies (NGCT-2019), 2020.