Motor Insurance Claim Frequency Prediction Using XGBoost

Original Research Article

Abstract

Insurance claim frequency modelling is an important task for non-life insurers, this together with other variables forms an important part of product pricing and risk management. The traditional frequency models such as Poisson, Negative Binomial and Zero Inflated models have several weakness such as scalibility issues, overdispersion and independence assumptions for large datasets and therefore not ideal to use when dealing with complex and unstructured data. Extreme gradient Boosting Algorithm (XGBoost) is an ensemble learning which has the capacity to effectively handle big complex and unstructured insurance data. XGBoost creates tree-based models by iteratively fitting decision trees to the residuals of the previous predictions, effectively reducing the error in each iteration. This research utilized and explored the XGBoost algorithm to process motor insurance claims large dataset in-order to predict the frequencies of insurance claims, that is 0,1,2,and 3 . Using this algorithm we aim to enhance the accuracy of predictions that will yield better estimates for improved risk assessment and pricing of insurance products. Cross validation was performed to assess the true performance of our model. Cross validation results showed that XGBoost models for the claim frequency had a RMSE estimate of 0.949, MAE of 0.7741 and RSQ 0.781. This demonstrated a strong predictive performance, with an RMSE of 0.949 and an MAE of 0.7741, indicating a low average error in the predictions.The RSQ of 0.781 suggests further that the model explained a significant proportion of variability of the insurance data. Our model was evaluated with a confusion matrix. The results of the confusion matrix showed that for the frequency 0 99.59% of cases were correctly predicted, frequency 1 94.01% were correctly predicted and frequency 2 84.80% and finally frequency 3 only 40.96% of the observations were correctly predicted.These results highlights the potential of XGBoost as a robust modeling technique for handling big data and accurately predicting insurance claim frequency. The results corroborate with other studies that XGBoost is an invaluable tool for insurance companies.

Keywords: Big data; Frequency; machine learning;ensemble learning; gradient boost; XGBoost

1 Introduction

Insurance industry/sector is composed of companies which offers risk management and mitigation services through the insurance products and contracts. The general idea behind the concept of insurance is that one party (insurer), guarantees a payment or compensation for the occurrence of uncertain future event that might cause financial loss to the insured. Meanwhile, the other party, the insured or the policyholder, remits a sum of money called premium to the insuring company, then the company in return offers the protection against the risk [1].

Insurance industry, like any business venture is often faced with challenges ranging from internally to externally. The key challenges faced internally might include; underwriting losses which is as a result of underpricing risks, whereby premium collected is less than benefits and expenses. Other challenges that are crucial are operational risks, investment volatility and catastrophic events.

The insurance sector in the recent years has continued to experience a shift in challenges due several factors both internally and externally. The challenges faced by insurance in the two categories are not unique and therefore they face similar challenges. Factors that have led shift in challenges include technological advancements, the changing customer preference and expectations, regulatory

changes, evolving risk landscapes, and competition [5].

Insurance industry has an obligation to the policyholders in form of benefits when they are due, therefore the sustatinability of the insurance company is paramount to the policyholder, shareholders and regulators. Sustainability of insurance company is dependent on several factors such as financial stability, solvency, investment strategy, regulatory compliance, product diversification, economic and market conditions among others.

Product pricing is an aspect for insurance as it directly influence sustainability and competitivenes of the company. Premiums pricing forms a component of the product pricing. To compute premiums actuaries take into consideration several factors including risk profile, claim history, type and value of the item, coverage and policy limits, risk factors among many other factors. There exist several methods for computing the ultimate premium that actuaries can adopt [2]. The simplest of this method is just a multiplication of the frequency of the claims that have occurred and the expected cost of the occurred claims.

Frequency of claims here can be defined as the number of claims received or reported to an insurance comapny from its pool of active policies within a specific time period, for general insurance it is typically one year. It is calculated by dividing the total number of claims received by the total number of exposure units/total number of policies. Several factors that can influence the frequency of the claims include; characteristics of the policy such as age, gender,geographical area, among others. The accuracy of frequency of claims predictiction is sometimes faced with some short coming and challenges such as the quality of data this might include incomplete and/ or innacurate data, changing trends such as the emergence of big data and other complex interactions that are hard to model.

This implies that actuaries need a clear understanding of the nature and behavior of frequency of the claims to accurately predict the frequency of future claims [4].The understanding of future claim frequency patterns enables the insurers manage risks that can eat into there profitability and reduce the chances of defaulting in payment of their claims.

To achieve this the insurance company analyze historical claim data, then use actuarial methods to predict the frequency of future claims.

There exists several methods for modelling claim frequency. Commonly used methods to model claim frequency distributions include the discrete random variable models such as binomial, geometric, negative binomial, and Poisson distributions. Additionally, other family of distributions for non-negative, integer-valued random variables such as (a,b,0) and (a,b,1) class of distributions would normally be used [6].

The growth and advancement of technology in many spheres of our lives has made it possible for insurance company to collect huge volume of data of different varieties under a high velocity and led to the emergence of big data. Therefore the current methods used for analysis such as statistical methods and actuarial formulas in insurance sector is becoming inadequate to solve the emerging problems and opportunities from the advancement in technology, particularly big data [3].

The use of Classification and regression trees (CARTs) a concept in Artificial Intelligence and machine learning has been proposed to address the short comings of generalized linear models (GLMs) oftenly used in frequency mdelling but this method has its own shortcomings such as high variance and lack of smoothness [8]. Therefore to improve the CARTs model, the model can be augmented by a range of ensemble methods which combine several trees so as to improve the trade off between

bias and variance [10].

Artificial Intelligence (AI) is a concept in computer science and engineering that describes how to make a computer program intelligent so that is able to do its function like a human mind [11]. In the words of Winston Henry AI is 'the study of the computations that make it possible to perceive reason and act' [12].

Machine learning has been described as a subfield of artificial intelligence which deals with the creation or development of algorithms and statistical and mathematical models that allows computer programs to learn from data without being programmed.Machine learning is a broad concept which is split up into two distinct types namely; Supervised and Unsupervised [13].

Ensemble learning a technique in machine learning which involves combining multiple base models to form an improve model with better performance compared to individual models. Extreme Gradient Boosting Trees (XGBoost) Algorithm is an example of ensemble learning algorithm used in regression tasks as well as classification tasks. Before we delve into this model, we first need to understand decision trees since its the building block for XGBoost.

Decision tree is an integral part of machine learning since it forms a basis or the building blocks for other learning methods such as gradient boost trees and random forests [14]. Decision trees are built from datasets and it comprises a root node, then the internal nodes and finally the leaf nodes [17]. The internal node is the rule that splits the input space into several nodes or terminal leaf/leaves according to some defined input attributes. The splitting criteria is characterized in two ways either by using gini impurity measure or by using the information gain criteria [18]. Decision trees can be pruned to reduce overfitting. Pruning methods includes; cost complexity , reduced error and minimum error pruning methods [15].

XGBoost is an ensemble ML technique that implements the gradient boosting approach. The technique used in this model is to combine the predictions of weak learners/model in this case decision trees to form a better/ strong model. The models are built sequentially with each new model improving the performance of the previous model by improving the prediction errors of the previous model. Its a gradient boosting model which implies that the model optimizes the loss function by reducing the residual errors of the previous model/tree [21].

Machine learning methods been incorporated to the actuarial research for a long time. There are two important applications of ML in actuarial field. The first part involves the application of machine learning classifiers into the problem. Frequency modelling usually falls under the classification part of ML, in this case Some of the applications include;

[9] Used XGBoost model to classify the risks faced by a life insurance company based on the historical data, they also performed a comparative study between XGBoost, decision tree, and random forest. From ther study XGBoost had the best evaluation metrics.[26] performed a classification analysis using decision trees, logistic regression and neural network classifiers. [29] performed a classification analysis using random forest classifier for insurance telematics data.

XGBoost has also been tested for its capability of handling missing data in insurance data. The simulation showed that XGBoost without the imputations gave a had a better accuracy as compared to the XGBoost with an imputation [23]. [24] developed an automated fraud detection system usint the extreme gradient boosting algorithm, the algorithm was also used to classify the different types of frauds according to their severity.

According to [30], who aimed to build a machine learning-based model that would provide more accurate estimations of the claims' cost and frequency in the Inpatient coverage for Multicare. Several algorithms were tested, including Linear and Logistic Regressions, Decision Trees, Random Forests, Gradient Boosting, and XGBoost. Their results were then compared to those of the current ARIMA model. The study showed that a machine learning technique, XGBoost, was more powerful than ARIMA, projecting 9% above the real costs compared to ARIMA's global error of -25%.

In their study [31] compared the relative performances of two machine learning techniques, logistic regression model and XGBoost in developing a model that predicts the occurence of accident claims for a telematics data. Their findings showed that logistic regression performed better that XGboost with regards to its interpretability and good predictive capacity.

[32] performed a comparative study of various machine learning approaches including XGBoost on diverse set of data. The models that were tested and assessed included, Support Vector Machines (SVMs), naive bayes, logistic regression, decision trees, boosted and bagged trees as well as boosted stumps. The study provided insights into the strengths and weakness of the different algorithms understudy. He also highlighted the XGBoost exceptional performance in the study as compared to the other algorithms.

[25] performed a comparative study on the machine learning predictive analytics employed in insurance premium prediction. Several methods were tested, they include; multiple linear regression, gradient boosting regression, random forest and decision tree. Gradient boosting was found to be the accurate with an accuracy of 87%

2 Methodology

2.1 Introduction

XGBoost was first initiated by Chen and Guestrin in 2016 , their idea was to create a scalable and efficient improvement of the gradient tree [21]. Their model utilized the idea behind the gradient boost method. The proposed model integrates multiple weak models by the use of residuals to build a stronger reliable model. The model has been widely used for classification tasks.

2.2 Model Outline

XGBoost utilizes the idea of boosting algorithm that constructs an ensemble of decision treesfrom the residuals of the model in a stage-wise manner. The general outline of the XGBoost objective function for classification tasks is given by:

$$\mathcal{L} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{2.1}$$

where:

- n is the dataset or appropriately the total number of training samples.
- $l(y_i, \hat{y}_i)$ is the loss function which measures how well the model performs in its predictions, it quantifies the difference between the true valuey_i and the predicted values $\hat{y}_i$.
- $\Omega(f_k)$ is the regularization term for the k-th tree f_k to prevent overfitting.
- K is the total number of trees in the model/ensemble.

2.3 Logistic Loss Function

For single binary classification tasks , the loss function borrows the idea of logistic regression and therefore uses the logistic loss as its loss function:

$$l(y_i, \hat{y}_i) = -\left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)\right] \tag{2.2}$$

where $\hat{y}_i$ is the predicted probability of the positive class.

For multiple binary classification tasks the loss function is an extension of the single binary classification logisti loss function as shown.

$$\mathcal{L} = \frac{1}{C} \sum_{j=1}^{C} \sum_{i=1}^{n} \left[y_{i,j} \log(\hat{y}_{i,j}) + (1 - y_{i,j}) \log(1 - \hat{y}_{i,j})\right] \tag{2.3}$$

where:

- n is the number of observation/instances.
- C is the total number of classes or labels in the data.
- $y_{i,j}$ is the true binary class for instance i and label j.
- $\hat{y}_{i,j}$ is the predicted probability of the positive class for instance i and label j.

$y_{i,j}$ is actual class label for each instance and class pair. It is 1 if the instance belongs to the class and 0 otherwise.
$\hat{y}_{i,j}$ is the predicted probabilty by the model that the instance belongs to the class.
$log(\hat{y}_{i,j})$penalizes incorrect predictions when the true label is 1, while $(1 - y_{i,j}) \log(1 - \hat{y}_{i,j})$ is used to penalize the incorrect predictions when the true label is 0.
The average over all the classes j ensures that the loss is normalized across the different classes shown by C.

2.4 Regularization Term

XGboost methodology adds a regularization term $\Omega(f_k)$ which is defined as:

$$\Omega(f_k) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} \omega_j^2 \tag{2.4}$$

where:

- T is the number of leaves in the k-th tree.
- γ is a parameter that controls the minimum number of samples required to split a node.
- λ is a parameter that controls the L2 regularization term on the weights.
- ω_j is the weight of the j-th leaf in the tree.

2.5 Model Training

The XGBoost methodology uses an additive strategy to train the model. This strategy start with a single leaf then add one tree at a time. The XGBoost methodology builds its model sequentially from the initial model, then adds the output of the decision trees built from the residuals of the previous tree.
The $f_t(x_i)$ are the functions with we need to learn, $f_t(x_i)$ each contains the leaf structure and leaf scores [33].

The final model will be given by the formula

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \rho f_t(x_i) \tag{2.5}$$

$\hat{y}_i^{(t)}$ is the predicted value at the t^{th} iteration, $\hat{y}_i^{(t-1)}$is the predicted value at the $t - 1^{th}$ iteration. ρ is called the learning rate and its used to scale the predictions of the current tree. $f_t(x_i)$ is the prediction of the t^{th} iteration.

2.6 Model Evaluation

Model evaluation in machine learning is the process of determining a model's performance via a metrics-driven analysis. For XGboost there are two ways that the model is evalauted; Online and Offline. Online means that the model is evaluated during the process of development while for offline the model is evaluated after the end of the process of model construction.

2.6.1 K Fold Cross Validation

Cross validation can be defined as an approach that is used to evaluate the performance of a learning algorithm. This is an online evaluation whereby the model is evaluated during the production of the model.
The method performs its objective by dividing or partitioning the dataset randomly into k other smaller dataset. The smaller sets of data are called the folds, if we have 5 partitions of the data then we say that we have 5-folds. The algorithm or the model is run and trained on each k-1 folds at a time with one fold put aside and then it will be tested in the next iteration. The process is repeated back and forth until each of the k folds has been used as the test set.
Once the cross validation has finished running you will have ended up with k different performance scores which can be summarised by using a mean and a standard deviation. This models are then evaluated using Root Mean Squared Error (RMSE) and R-squared.

$$\begin{aligned} \text{RMSE} &= \sqrt{\sum \frac{(\hat{y}_i - y_i)^2}{n}} \\ \text{R-Squared} &= 1 - \frac{\text{Total Explained Variation}}{\text{Total Variation}} \\ &= 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y}_i)^2} \end{aligned}$$

2.6.2 Confusion Matrix

Confusion matrix is an evaluation tool used in classification problems for the offline evaluation. It give rise to other metrics for evaluation, they include precision and recall, accuracy and F-1 score. Its a table in form of a matrix that summarizes how well the classification was done by the model. The matrix consists of predicted values from the model and actual values from the testing data.

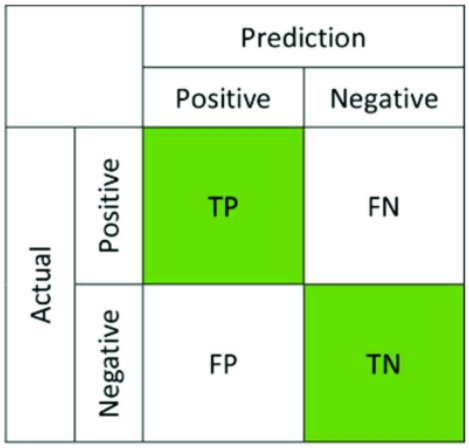


Figure 1: Confusion matrix

Metrics	Computing method
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
Specificity	$\frac{TN}{TN + FP}$
F1-score	$\frac{2 \times TP}{2 \times TP + FP + FN}$

Figure 2: Metrics and computing method

3 Results and Discussion

3.1 Introduction

The data used for this research was for a French Insurance company specializing in Motor Vehicle insurance. The data was freely and readily available in the Kaggle website [35]. The data was obtained in the form of Excel comma separated values (csv) from the website. The claim dataset was for the year 2015, January to December.
The data contained 42 feature/variables for 4000 policyholders under the motor insurance cover. These features characterize the policyholder (such as age, gender, education, job, marital status), the insurance policy (such as policy number,policy state, deductible,umbrella limit) and insured vehicle (such as age of the car, car brand, manufacture year, roads drive).

3.2 Descriptive Statistics

The target variable for the objective under study is the number of claims filed by a policyholder which is a non negative discrete random variable. Summary statistics for the frequency of claims in the dataset is shown herein;

Table 1: Summary of the frequency of claims.

Frequency of Claim	Number of Policyholders	Percentage
0	3108	77.71%
1	498	12.45%
2	314	7.84%
3	80	2.001%

From Table (2) we see that the 77.71% of the policy holders did not file a claim. The number of policyholders who filed one and two claims were 12.45% and 7.84% respectively. The number of

policyholders who filed for three claims had the least numbers with 2.01% of the total number.
We also computed the summary statistics for the claim numbers as shown.

Table 2: Summary Statistics of the frequency of claims per policy.

Variables	Mean	Median	Standard deviation	Variance	Min	Max
Frequency of claims	0.5295	0.00	0.8639	0.7463	0	3

Table (2) shows the mean, median, standard deviation, variance, the minimum and maximum values of the number of claims. The mean is 0.5295 which is a low value and closer to the minimum value of the frequency. The mean shows that the majority of the policyholder had filed a nil (0) and single claim (1). This shows that the claims for this insurer were infrequent but not necessarily neglible. The median for the dataset is 0.00, this number shows that there were many observations with zeros (0) and according to the definition of median value this means that atleast 50% of the total number of the claims had a frequency of 0.

The summary statistics shows that the variance of the frequency of claims is 0.7463 which is larger than its mean of 0.5295. This indicates that there is over-dispersion. The summary also shows that the minimum value observed is 0 and the maximum value is 3.

3.3 Distribution of the frequency of claims under Different Attributes of the Policyholders

We performed descriptive statistics on distribution of claims number, that is the number of claims per policy. We performed this to explore the relation of the claim number against the different sets of the features in our data.
The distribution in Table (2) of claim frequencies can conveniently be summarized in a histogram as shown below.

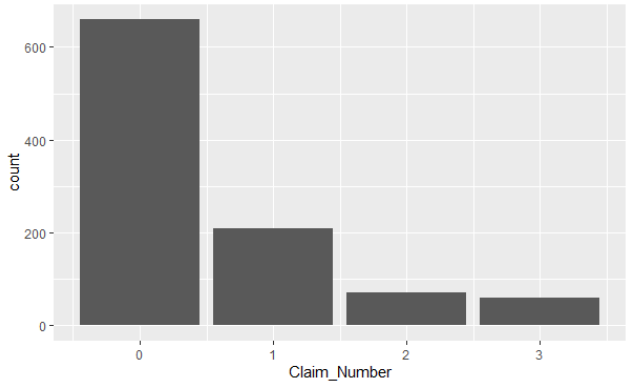


Figure 3: Distribution of frequency of claims

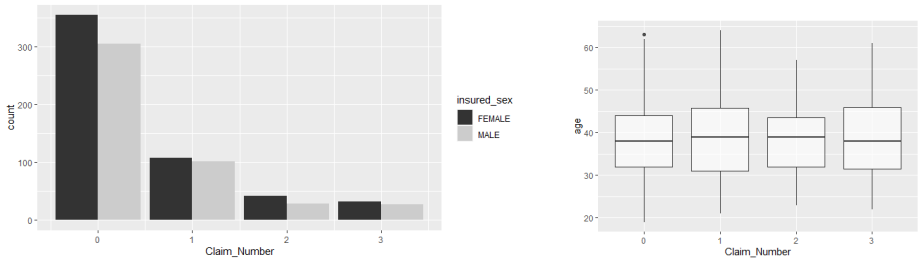
From the histogram (values scaled down) it can be seen that the frequency of claims is highly skewed to the right thus indicating a positive skewness, the data also exhibits a heavy-tailed distribution. The number of policy holders who did not file any claim comprised of the highest proportion of the total

policyholders and another important observation is that the number of zero counts is large. A small number of policyholders filed three claims. The number of policyholders who filed for one claim were more than those who filed for two claims thus resulting into a skewed distribution. This distribution is similar to other previous studies which showed that claim frequency distribution is usually highly skewed with most policyholders not filing a claim while the number of policyholders filing more than one claim tends to decreases as the number of claim increases. A study by [16], [27] and [28] from their studies also found out that the distribution of the insurance claim frequencies were positiveky skewed with heavy tails.
This result which shows that most of the policy holders for this insurer are low risk helps the insurer to price their products better and to set the reserves adequately.

3.4 Distribution of Frequency of claims according to Gender and Age

The barplot and boxplot in Figure (4) shows the claim number patterns according to gender and ages of policyholders. Figure (4)(a) shows that females had a higher count of insurance claims than males in the same category for all categories of claim numbers. Additionally, largest gender disparity was observed in claim number 0 category, where the number of females had not filed any claims as compared to the males by a significant margin. The general trend according to the barplot is that as the number of claims filed decreased as the number of claims increased.

The barplot in Figure (4)(b) shows the pattern exhibited by the frequency of claims across the different ages of the policyholders. The lowest age for a policyholder in this data was 19 years old, while the lowest age to file a claim was 21 as shown by the plot. The median age across the claim number categories was 40 with a consistent interquartile range and whiskers span. The notable observation is that there is a presence of an outliers for the age of those who had filed nil claims.



(a) Frequency of claims and Gender

(b) Frequency of claims and Age

Figure 4: Age and Gender Distribution

Distribution of the claim numbers according to age depicts an evenly distribution of claim numbers across the ages. The number of claims is distributed evenly within the 35-40 years old across the claim numbers.

The results of the distribution of the claim frequency according to gender and age is very important for insurers risk management techniques and premium claculation as well as fraud detection. This information can enable the insurer's to tailor their premiums to be gender specific so as to reflect the actual risk faced, the information can also be used by the insurers to offer cusstomized products for the different genders and ages thereby enhancing the competitiveness of its products in the market.

4 Frequency model development

XGBoost was used to perform the classification of the four claim frequencies and to predict the claim number frequencies. The steps taken are discussed below.

4.1 Initial Data Split

The dataset was split into two sets. One set called the training set was used to create the initial model and the size of this dataset was 70% of the total observation in the dataset. The other set called testing set was set aside to be used as an evaluation of the model created and it takes 30% of the total set.
The training dataset was used for hyper parameter tuning, model development and training . On the other hand the test dataset is used to evaluate the model trained on the training set to measure its performance or accuracy. The data was pre-processed to make the process less computer intensive, then data split for cross validation and finally parameter tuning process. After the whole proces was complete, the algorithm had developed a model for classification and prediction.

4.2 Pre-processing

The stage includes converting all the character variables into factors, then the factors are converted to numeric i.e dummy variables, combining low frequency values and remove predictors with no variance which provide no predictive information. The factor columns were converted to dummy using one-hot encoding. For the model in this study minimal data processing was done since XGBoost is a robust algorithm that can handled correlated and highly skewed data with minimal imputations.

4.3 Splitting for Cross Validation

Cross validation which is a technique used to evaluate Machine Learning models. The idea here is having several models on one partition of the data and then evaluate the performance of this models on the other remaining partition of the data. The method is important because its able to detect overfitting of the models. For our analysis we used k-fold stratified cross validation to split further our training set and test set. We also used cross validation to tune our hyper parameters. We split the training data into subsets of 10, which will be used in cross validation.
XGBoost model has eight (8) parameters that has to be chosen before deploying the model and we call this process tuning parameters. Parameters due for tuning are, tree depth, number of trees, learning rate, randomly selected predictors, minimal node size, minimal loss reduction, proportion of observations sampled and number of iterations before stopping. We used the grid search for the tuning of parameters in our model.

```
xgboost_model <-
  parsnip::boost_tree(
    mode = "classification",
    trees = 1000,
    min_n = tune(),
    tree_depth = tune(),
    learn_rate = tune(),
    loss_reduction = tune()
  ) %>%
  set_engine("xgboost", objective = "multi:softprob", num_class = <number_of_classes>)
```

Samples of the best hyper parameters after initial tuning.

Table 3: Samples of the best hyper parameters after initial tuning.

min_tree	tree-depth	learn-rate	loss_reduction	Mean	Standard Error
43	10	0.0000000	0.0000000	938.7877	62.7867
17	3	0.0000195	0.0000000	1149.2278	78.8470
7	7	0.0050908	0.0000061	1213.0258	83.7222
37	4	0.0034828	0.6319577	1229.1610	100.8766
12	11	0.0000000	0.0000000	1355.8749	103.1406
14	13	0.0168147	3.9205150	1216.3456	97.3786

The hyper parameters used in training the model are listed below;

Table 4: The hyper parameters to be used in training the frequency model.

min_tree	tree-depth	learn-rate	loss_reduction	Mean	Standard Error
7	6	0.0108	0.8205	0.8350	0.0309

The performance of the models created during cross validation (during the training phase) were compared using Root Mean Square Error (RMSE), R-squared (RSQ) and Mean Absolute Error (MAE). The model with the best metrics had the following is for the training dataset.

Table 5: The RMSE, RSQ and MAE for the frequency model for training data.

Frequency Model		
Metric	**Estimator**	**Estimate**
RMSE	Standard	0.949
RSQ	Standard	0.781
MAE	Standard	0.7741

RMSE value of 0.949 suggests that the average error between the predicted values and the actual values were relatively low and there it indicates a better model performance. An RSQ of 0.78 which is relatively a higher value indicates a better model performance and technically means that approximately 78.1% of the variability in the data is explained by the model. The MAE of 0.7741 indicates that the average absolute error between the predicted and actual values is 0.7741. In summary the table shows that the frequency model for the training data performed well.

For the test data set, we have the following metrics computed.

Table (5) and (6) contains the results for the RMSE and MAE and RSQ, the values are similar for both training and test data. This indicates that the magnitude of the prediction errors of the model is consistent across the two datasets. This consistency implies that the model's errors are not changing significantly when applied to new data and therrefore the model can be used for prediction.

Table 6: The RMSE, RSQ and MAE for the frequency model for the test data.

Frequency Model		
Metric	**Estimator**	**Estimate**
RMSE	Standard	0.849
RSQ	Standard	0.796
MAE	Standard	0.8741

4.4 Model Evaluation

The most efficient method for model evaluation in a classification problems include metrics such as confusion matrix,accuracy, precision, recall and F1-Scores [34]. The results of these evaluation are shown here.

Table 7: Results of the computed Confusion matrix for the four classifiers using the cross-validation approach.

		Predicted Values			
		0	1	2	3
Actual Values	0	3095(99.59%)	13(0.41%)	0.00%	0.00%
	1	48(3.12%)	461(94.01%)	14(2.88%)	0.00%
	2	0.00%	41(14.01%)	249(84.80%)	4(1.19)%
	3	31(38.81%)	14(17.23%)	4(2.88%)	32(40.96%)

Table (7) and (8) provides the frequency model's prediction performance in terms of the confusion matrices for all classification algorithm of the claim frequency which were obtained during the multi class classification. The confusion matrices shows the discrepancy within the predicted values from the model and the actual observations for individual frequency of claim classes in the dataset. The rows in the matrix do represents the actual number of observations for a specific frequency, while the columns indicate the predicted number of actual observations for that particular frequency.

The cells' values across the diagonal of the matrix imitate the accurate predictions (highlighted with colour in the table), the other cells away from the diagonal cells indicates the misclassifications which results in over estimation or under estimation of the model. The percentage (%) values in the brackets entry represents the ratio of predicted number of frequency divided by the actual number of frequency in the dataset. From the table we see that our model performed quite well in predicting the frequency of claims when the frequency was 0,1,and 2 and it performed not so well when predicting the number of claims to be 3.
Using the values of the confusion matrix in Table (7), we can compute further the precision, recall, Specificity and F1-Score.

Table 8: Outputs of the other metrics generated from the matrix.

Model	**Precision**	**Recall**	**Specificity**	**F1-Score**	**Accuracy**
XGBoost	85.59%	86.77%	78.59%	83.44%	75.89%

Shown in Table (8), the XGBoost has the highest precision score, indicating a huge proportion of

true and positive predictions. The model in addition exhibits a higher recall, meaning that it correctly identifys many claims that will occur. The high F1-Scores indicate that there is a good balance between precision and recall.
We can conclude that the XGBoost classifier is a reliable tool in claim frequency classification. This model can be used by insurance companies when they are performing frequencies prediction of their claims. Effective prediction is important as it helps when performing experience rating for insurers.

5 CONCLUSIONS

a In the section (1), a brief outline of the frequency of claims and its importance to insurance companies. This section also discussess the methods of frequency modelling both the traditional and the machine learning methods. In this section we also reviewed some literatures that describes the emergence, applications continued use of XGBoost model in insurance claim analysis.

b In section (2), we have shown the development of the XGBoost model for classification tasks. we have described the development of the loss function and regularization term that was used as our objetive function. We have also highlighted the evaluation methods that have been used during the cross validation as well as the confusion matrix which has been used to evaluate our classification model.

c In the section (3) We have performed a descriptive statistics of the data to understand the data patterns and characteristics, explore the relationships between the dependent and independent variables. From the analysis we found out that the data was positively skewed and heavy tailed. The data also had a higher variance compared to mean and therefore there was an overdispersion. From the summary statistics, it was observed that the highest number of policyholders did not file a claim while the least number of policyholders failed three claims. The results showed that there was an even distribution of the claim numbers across the ages, with the youngest policyholder being 18 years and the youngest policyholder to file a claim was 21 years old. The results also showed that the number of females who had filed for a claim exceed the males for the claim frequencies band.

c In the section (4) We described the steps involved in the XGBoost model development, preprocessing and cross validation. We also have shown the results of the cross validation, since cross validation was important to reduce or prevent overfitting. Cross validation was performed on both the training data and testing data. The K folds created during the cross validation were assessed via RMSE, MAE and RSQ. For the training data the RMSE, MAE and RSQ were 0.949,0.781 and 0.7741 respectively. This results showed that the model picked during cross validation was a model that could be relied upon for prediction purposes.
The XGBoost model constructed/trained was evaluated by another set of unseen data the testing data through cross validation. The RMSE, MAE and RSQ for the test data were 0.849, 0.796 and 0.8741 respectively, this values are similar for the training data. This consistency implies that the model's errors are not changing significantly when applied to new data and therrefore the model can be used for prediction. The model constructed was evaluated using a confusion matrix, the evaluation showed that the model performed well in its prediction. The results of the confusion matrix showed that for the frequency 0 99.59% of cases were correctly predicted, frequency 1 94.01% were correctly predicted and frequency 2 84.80% and finally frequency 3 only 40.96% of the observations were correctly predicted. The poor performance of the model to predict claim frequency 3 might have been due to the very small number of policyholders in the category.

.

References

[1] Parodi, Pietro, *Pricing in general insurance*, Chapman and Hall/CRC, 2023.

[2] Outreville, J. François, *Theory and Practice of Insurance*, Springer Science & Business Media, 2012,

[3] Alex Zarifis, Christopher P. Holland, and Alistair Milne. *Evaluating the Impact of AI on Insurance: The Four Emerging AI-and Data-Driven Business Models.* Emerald Open Research, 1:15, 2019.

[4] Grize, Yves L, *Applications of statistics in the field of general insurance: An overview*, International Statistical Review 83, no. 1 (2015): pp.135-159.

[5] Željko Šain and Jasmina Selimović, "Challenges in Insurance Industry," *Interdisciplinary Management Research*, vol. 471, pp. 479, 2009.

[6] Bjørn Sundt. *An Introduction to Non-Life Insurance Mathematics*, volume 28. VVW GmbH, 1999.

[7] Naman Kumar, Jayant Dev Srivastava, and Harshit Bisht. Artificial intelligence in insurance sector. *Journal of the Gujarat Research Society*, 21(7):79–91, 2019.

[8] Martin Eling, Davide Nuessle, and Julian Staubli. The impact of artificial intelligence along the insurance value chain and on the insurability of risks. *The Geneva Papers on Risk and Insurance-Issues and Practice*, pages 1–37, 2021.

[9] Orji, Ugochukwu, and Elochukwu Ukwandu. Machine learning for an explainable cost prediction of medical insurance. *Machine Learning with Applications*, 15 (2024): 100516.

[10] Syed Hasan Jafar and Shakeb Akhtar. AI in insurance. In *Artificial Intelligence for Business*, pages 164–173. Productivity Press.

[11] Margaret A. Boden. *Artificial Intelligence*. Elsevier, 1996.

[12] Patrick Henry Winston. Artificial Intelligence. *Addison-Wesley Longman Publishing Co., Inc.*, 1984.

[13] Aurélien Géron. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. *O'Reilly Media, Inc.*, 2022.

[14] Tom Hanika and Johannes Hirth. Conceptual views on tree ensemble classifiers. *International Journal of Approximate Reasoning*, 159:108930, 2023.

[15] Xiaobo Li and Richard C. Dubes. Tree classifier design with a permutation statistic. *Pattern Recognition*, 19(3):229–235, 1986.

[16] Omari, Cyprian Ondieki, Shalyne Gathoni Nyambura, and Joan Martha Wairimu Mwangi. Modeling the frequency and severity of auto insurance claims using statistical distributions. *Journal of Mathematical Finance*, 2018, 8, 137-160.

[17] Barry De Ville. Decision trees. *Wiley Interdisciplinary Reviews: Computational Statistics*, 5(6):448–455, 2013. Wiley Online Library.

[18] Steven W. Norton. Generating better decision trees. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, volume 89, pages 800–805, 1989.

[19] Peter Bühlmann and Bin Yu. Boosting. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(1):69–74, 2010. Wiley Online Library.

[20] Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, 7:21, 2013. Frontiers Media SA.

[21] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[22] Didrik Nielsen. *Tree Boosting with XGBoost - Why Does XGBoost Win "Every" Machine Learning Competition?* Master's thesis, Norwegian University of Science and Technology (NTNU), 2016.

[23] Rusdah, Deandra Aulia, and Hendri Murfi. XGBoost in handling missing values for life insurance risk prediction. *SN Applied Sciences* 2.8 (2020): 1336.

[24] Dhieb, Najmeddine, Hakim Ghazzai, Hichem Besbes, and Yehia Massoud. Extreme gradient boosting machine learning algorithm for safe auto insurance operations. *In 2019 IEEE international conference on vehicular electronics and safety (ICVES)*, pp. 1-5. IEEE, 2019.

[25] Jyothsna, Chaparala, K. Srinivas, Bandi Bhargavi, Akuri Eswar Sravanth, Atmuri Trinadh Kumar, and JNVR Swarup Kumar. Health Insurance Premium Prediction using XGboost Regressor. *International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, pp. 1645-1652. IEEE, 2022.

[26] Johannes Paefgen, Thorsten Staake, and Frédéric Thiesse. Evaluation and aggregation of pay-as-you-drive insurance rate factors: A classification analysis approach. *Decision Support Systems*, 56:192–201, 2013.

[27] Yip, Karen CH, and Kelvin KW Yau. On modeling claim frequency data in general insurance with extra zeros. *Insurance: Mathematics and Economics* 36, no. 2 (2005): 153-163.

[28] Gschlößl, Susanne, and Claudia Czado. Spatial modelling of claim frequency and claim size in non-life insurance. *Scandinavian Actuarial Journal*, no. 3 (2007): 202-225.

[29] Philippe Baecke and Lorenzo Bocca. The value of vehicle telematics data in insurance risk selection processes. *Decision Support Systems*, 98:69–79, 2017.

[30] José M. Maisog, Wenhong Li, Yanchun Xu, Brian Hurley, Hetal Shah, Ryan Lemberg, Tina Borden, Stephen Bandeian, Melissa Schline, Roxanna Cross, and others. Using massive health insurance claims data to predict very high-cost claimants: a machine learning approach. *arXiv preprint arXiv:1912.13032*, 2019.

[31] Jessica Pesantez-Narvaez, Montserrat Guillen, and Manuela Alcañiz. Predicting motor insurance claims using telematics data: xgboost versus logistic regression. *Risks*, 7(2):70, 2019.

[32] Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 161–168, 2006.

[33] Omer Sagi and Lior Rokach. Approximating XGBoost with an interpretable decision tree. *Information Sciences*, 572:522–542, 2021.

[34] J. D. Novaković, A. Veljović, S. S. Ilić, Ž. Papić, and M. Tomović, "Evaluation of classification models in machine learning," *Theory and Applications of Mathematics & Computer Science*, vol. 7, no. 1, p. 39, 2017.

[35] Aashish Jhamtani. *Automobile Insurance*. Kaggle, 2023. Available at: https://www.kaggle.com/aashishjhamtani/automobile-insurance. Accessed: 2024-05-04.