Predicting Insurance Policy Renewal

CSE 2600
Garrett Sparks, Alper Tepebas, Yichong Wu

1 Introduction

1.1 Abstract

In the car insurance business, the ability to predict whether a customer will renew their policies is vital for improving customer retention. Customer retention is what makes insurance companies profitable and helps draw in additional loyal customers. It can often be more costly to acquire new customers than it is to retain the customers with existing policies. Loyal policyholders often have cheaper premiums and are more likely to spread a positive word about the insurance company. Insurers can focus on improving their company when they understand their retention rates. If they know the reason why policyholders are not renewing, they can identify what their company needs to improve on. The main reason a policyholder switches insurers is to save money, followed by customer service experiences. Our study will help identify this retention rate, benefitting car insurance providers and allowing their companies to grow. We will test Linear Regression, Random Forest, and Gradient Boosting models that we have created, and provide an analysis as well as a comparison of each. This will help us determine which is the best performing model for our leveraged dataset, while helping insurers retain their policyholders.

1.2 Data Background

We have taken the *Dataset of an actual motor vehicle insurance portfolio* from Mendeley Data to predict these renewals. This data has over 105,000 rows of individual policy transactions, and 30 different variable columns for us to analyze. The data comes from research performed by *Universitat de Valencia* (University of Valencia). The dataset provides information about a customer, their current policy, their vehicle specifications, driving record, policy history, and more. As we discuss more about the data, we will explain further about some specific variables.

2 Data Preprocessing

Our data ranges from November 2015 through December 2018. We defined *reference_date* as a datetime object for other calculations that will need to be performed. As our data ranges until the end of 2018, our reference date is 31/12/2018. Our date variables are following a DD/MM/YYYY format, as our dataset came from Spain. We then define *Customer_age* from the *Date_birth* column. This calculates the customer age by subtracting their date of birth from the newly defined reference date. *Driving_experience* is a measure of how long a customer has been driving. The preprocessing step for this is to subtract the *Date_driving_licence* column from the reference date. *Contract_duration* is another metric that evaluates the duration of a customer's contract in days by subtracting the *Date_start_contract* from *Date_next_renewal* columns. *Date_start_contract* is defined as the start date of the policyholder's contract, and *Date_next_renewal* is defined as the date of the next contract renewal expected.

We determined filtering outliers was necessary in the preprocessing steps. If *Customer_age* has a numerical value less than 18 or greater than 100, they are excluded. If *Driving_experience* is negative or greater than 80, it is also excluded. It is unrealistic for a driver's experience to be negative or longer than 80 years, as they get their licenses at 18. If this model were to be implemented for another minimum driving age, filtering data would be modified.

Despite not using it, we created an additional feature called *Experience_ratio*. This calculates the ratio of customer driving experience to their age by dividing the customer's experience by their age. We used normalization to help us understand more about the context of a customer, and how much of their life they have been driving for. After further research, we determined that we must add one to the age value so there would never be the possibility of dividing by zero. A 25 year old driving for 7 years has a higher ratio compared to a 45 year old with 7 years of driving. The 25 year old has been driving for a larger portion of their life than the 45 year old has. *Experience_ratio* has more implications on determining risk and little use in our final model, but in our original development we determined it may be useful, and further helps our understanding of normalization and feature engineering.

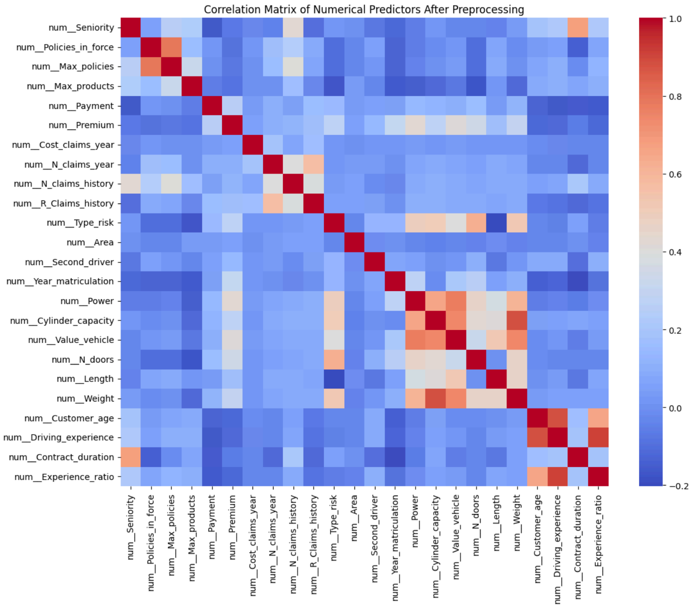
In addition to the features we defined, we needed to define a target variable. Our target variable is labeled as *Renewal*, which is derived from the category *Lapse*. Lapse is defined as the status of policies that customers have cancelled themselves, or have been cancelled for nonpayment in the current year of maturity. This excludes policies that have been replaced by another policy, as that customer still holds the same number of policies as before. The Lapse column reflects how many policies were canceled, not just whether a policy was canceled. If a customer cancels more than one policy, then *Lapse*'s value is the number of policies canceled. We converted this numerical value into a binary value for modeling. If *Lapse* has a numerical value of *0*, then *Renewal* has a binary value of 1. This shows that zero policies were canceled, and the customer has renewed policies for another year. If *Lapse* has a numerical value of 1 or more, then *Renewal* has a binary value of 0. The binary value of 0 represents a customer that has not renewed their policy.

Finally, we dropped columns that were not deemed useful. The following are used to calculate our new features and our target variable, so they have been dropped: *Date_start_contract*, *Date_next_renewal*, *Date_birth*, *Date_driving_licence*, *Date_lapse*, *Lapse*.

3 Method

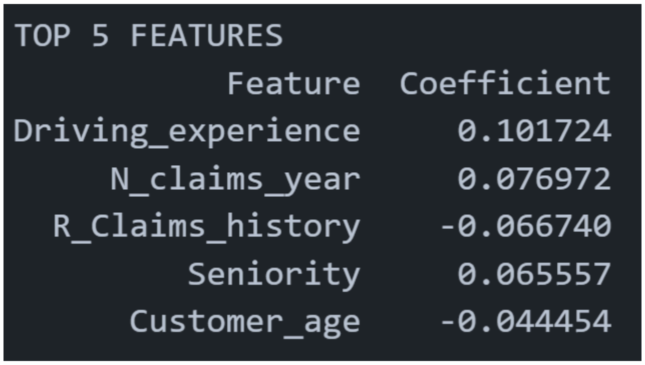
3.1 Linear Regression

Our first method of determining renewals was creating a linear regression model. Linear regression is a simpler model than the other models we have tested, and does not perform well with more complex and nonlinear datasets. After preprocessing our data, we created a correlation matrix to show relationships between our predictors. We have excluded categorical predictors from this visualization, as they represent a category or group, not continuous numerical values. We see predictors like the age of a customer is strongly correlated with a customer's driving experience, an example given previously. We can also interpret predictors like the weight of a vehicle has little correlation to assessing risk. The importance of understanding



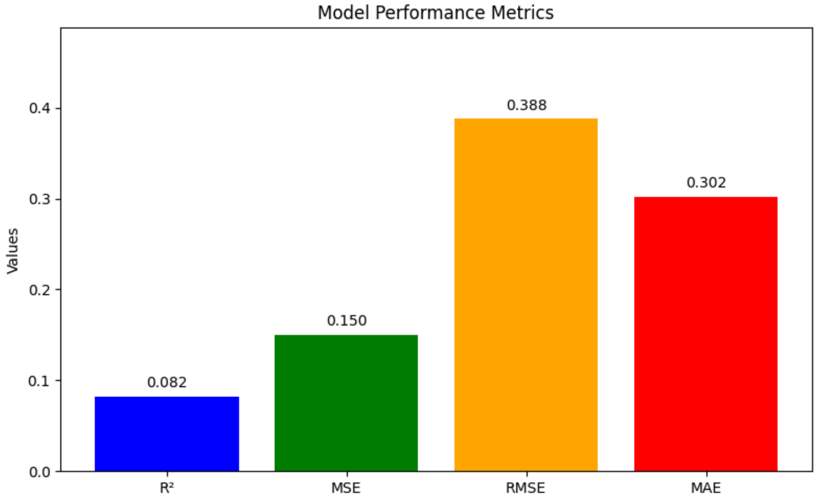
correlation in our predictors is because of multicollinearity. If multiple predictors are highly correlated, it can make estimating coefficients difficult.

Our top predictors numerical features are *Driving_experience* and *N_claims_year*, which is the total number of claims incurred for the insurance policy during the current year. *Driving_experience* is the most influential feature affecting *Renewal*. *R_Claims_history* is defined as the ratio of the number of claims filed for the specific policy to the total duration of the policy in force. We see that *R_Claims_history* has a somewhat significantly large negative coefficient, suggesting that as this ratio goes up, the likelihood of renewal decreases, as opposed to the *Driving_experience*, with an increasing likelihood. Same with *Seniority*, which is the number of years a policyholder has been with the same insurance company. The *Customer_age* feature with the negative coefficient indicated the older someone gets, the less likely they are to renew. This appears to contradict our other positive coefficients. The best approach to thinking about this is something like the following scenario. A younger customer, say 25 years old, is in more need of insurance than an 80 year old who may soon be done driving. It essentially comes down to life stages, which our model is not capable of interpreting. A 50 year old who has been with an insurance company for a longer amount of time will have high seniority and more experience, but in the future may be less likely to renew their policy.



```
TOP 5 FEATURES
           Feature  Coefficient
Driving_experience     0.101724
     N_claims_year     0.076972
  R_Claims_history    -0.066740
         Seniority     0.065557
      Customer_age    -0.044454
```

To evaluate our model’s performance, we measured four metrics: R^2, Mean Squared Error (MSE), Root Mean Square Error (RMSE), and Mean Absolute Error (MAE). R^2 measures the proportion of the variance in the *Renewal* target, ranging from 0 to 1. 0 means the model explains no variance, and 1 means the model perfectly explains variance in the data. Our linear regression model provided a value of 0.082, or 8.2%, meaning it does not capture much variance. This metric supports the idea that the data is complex and non-linear. MSE is sensitive to larger errors, so it is not the best for binary values. Our target has a binary value, and the MSE is on the lower end of our 0 through 1 scale. The MSE value of 0.1504 indicates that the model’s ability to predict our target is off by 0.1504 squared units from the true, actual renewal information. The lower the value, the better the model is at predicting the true renewal, so 0.1504 is not a great value, but is not necessarily bad. In terms of RMSE, we are taking the square root of our MSE, we can see the model’s predictions are 0.388 units from predicting the true, actual value. This further expands our understanding of the MSE, and how poorly this model performed. Our fourth metric is MAE, which averages the difference between predicted values and the true, actual values. An MAE near 0.302 means our model is on average incorrectly predicting renewals by about 0.302 units. This is a significant amount of error in our predictions, supporting that linear regression is not the most optimal performing model.

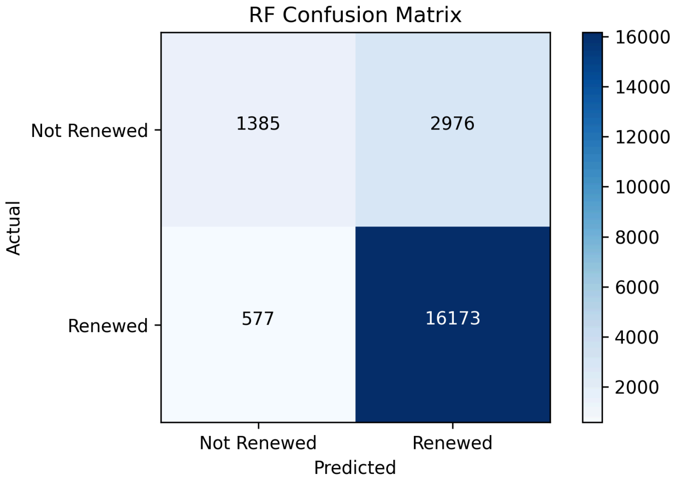


To the left, we display a visualization of our performance metrics of the linear regression model.

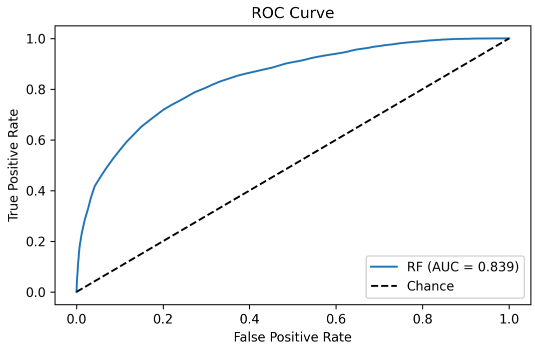
In terms of accuracy, the linear regression model had a training accuracy of 79.97% and a testing accuracy of 79.39%. Our low R² value indicates potential high bias and underfitting. We have low variance, which means the training data is not overfitting. This information leads us to the conclusion that linear regression is not the most optimal due to its inability to handle complex nonlinear data and poor metrics.

3.2 Random Forest

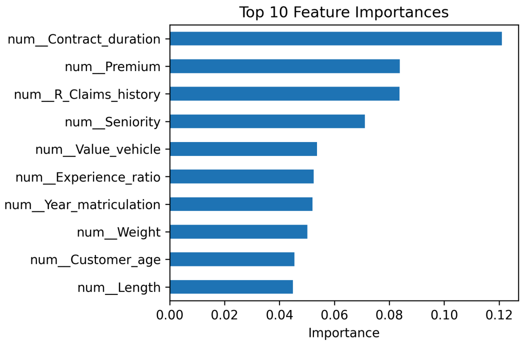
Because our dataset shows clear interaction effects, *Seniority* intertwined with *R_Claims_history* is only one example of a linear model that cannot capture the renewal logic we need. We therefore trained a Random Forest (*n_estimators* = 100, *class_weight* = balanced, *random_state* = 42) on the same 80 / 20 stratified split. The ensemble reaches **98.7 %** accuracy on the training fold, yet settles at **83.2 %** on the hold-out set, signalling a modest degree of over-fitting; nevertheless its test precision (84.5 %) and recall (96.6 %) translate into a solid F1 of 90.1 %, while the ROC-AUC climbs to **0.839**, far above the linear baseline's negligible explanatory power (R² = 0.082). Generalisation was further vetted through five-fold stratified cross-validation; AUC fluctuates only between 0.836 and 0.844 (mean 0.838 ± 0.003), confirming the model's stability across sample permutations.



The confusion matrix makes the lift intuitive: out of 16,750 policyholders who renewed, the model retrieves 16,173 (recall ≈ 97 %) at the cost of 2,976 false positives, a trade-off acceptable for retention campaigns that favour recall over precision at this stage.



ROC analysis underscores the gain; the curve hugs the upper-left frontier and the area of separability increases by roughly 22 pp versus the linear benchmark, meaning risk managers can set operating thresholds with a far greater confidence.



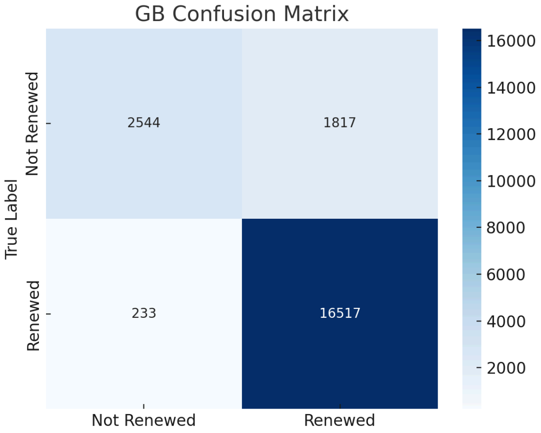
Inspection of feature importances provides business insight. *Contract_duration* dominates (12.1 %), followed by *Premium* and *R_Claims_history* (each ~8.4 %), with *Seniority*, *vehicle_value*, and engineered exposure ratios rounding out the top tier. Together the first ten features explain just over half of all model splits, confirming that contractual commitment, monetary stake and recent claims remain the decisive renewal levers.

In summary, Random Forest lifts test AUC from the linear baseline 0.62 to 0.839 and boosts F1 from 0.54 to 0.90. While a 15-percentage-point train–test gap hints at deeper trees than strictly necessary, grid-searching *max_depth* and *min_samples_leaf* plus threshold calibration should trim the 2,976 false positives without eroding the hard-won 96.6 % recall. These results already satisfy our viability criteria and give underwriting teams actionable signals: prioritise customers with short contracts or high claims ratios for proactive retention measures.

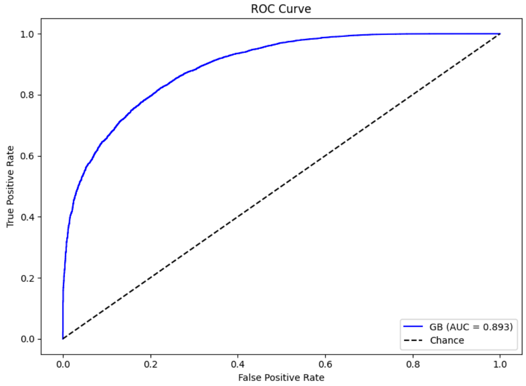
3.3 Gradient Boosting

To further improve performance over Random Forest, we leveraged Gradient Boosting. We trained a GradientBoostingClassifier with hyperparameters selected via randomized search: (*n_estimators*=591, *learning_rate*=0.0364, *max_depth*=9, *min_samples_leaf*=30, *min_samples_split*=180, and *subsample*=0.9187). The model was trained on the same 80/20 stratified split and evaluated using the identical preprocessing pipeline applied across all models. This model achieved 90.3% on the test set, reducing overfitting and yielding the best generalization performance across all models. Test set precision (0.90%) and recall (0.99%) resulted in a strong F1-score of 0.94 improving 0.4 points over Random Forest's 0.90.

Looking at the confusion matrix, the model correctly predicted 16,517 out of 16,750 renewals (positive cases), maintaining very high recall. On the other hand, it correctly identified 2,544 out of 4,361 non-renewals (negative cases), showing improved handling of churn compared to Random Forest. he overall F1-score and ROC AUC confirmed the model's stronger ability to generalize across both classes.



ROC analysis demonstrates this improvement; the curve is now nearer to the top-left corner than previously, and the ROC-AUC rose to **0.893**, 0.054 points better than Random Forest (0.839). The wider margin assures that Gradient Boosting is more capable of distinguishing between customers who may leave and those who remain in more situations, providing stakeholders with more leeway for operational adjustments.



Feature importance rankings remained consistent. *Contract_duration* continued to be the most significant variable across models, reaffirming its strong association with renewal behavior. Other significant variables such as *Premium*, *Seniority*, and *R_Claims_history* frequently appeared towards the top, demonstrating the value of money, the duration a customer has been with us, and claims experience.

In summary, Gradient Boosting improved the test AUC from 0.839 to **0.893** and enhanced the F1-score from 90.0% to **94.0%**. Both precision and recall improved, and false positives decreased significantly without impacting renewal coverage. These scores unequivocally speak to it being the most balanced and consistent model, therefore providing underwriting teams with greater confidence and more risk segmentation for future retention programs.

4 Discussion

4.1 Data Recap

To conclude, the Linear Regression model provided useful insights, but performed poorly compared to other models leveraged. With data that has nonlinear relationships and high complexity, this model was too simple. R^2 was 0.082, MSE was 0.1504, RMSE was 0.388, and MAE was 0.302. Training and testing accuracies were 79.97% and 79.39%. It had high bias and underfitting. The Random Forest had better performance, with a 98.7% training accuracy and testing accuracy of 83.2%. Precision was 84.5%, recall was 96.6%, and F1-score reached 90.1%. ROC-AUC measured at 0.839. Gradient Boosting outperformed both, with better metrics all-around with test accuracy of 90.3%, precision of 90%, recall of 99%, and F1-score of 94%. ROC-AUC measured at 0.893. All these metrics support our conclusion that Gradient Boosting was the best tested method for our data set.

4.2 Further Development

Our main goal was to use different machine learning models to predict motor insurance policy renewals, aiming for high accuracy, performance, and yield useful business insights. Our data preprocessing made time-related complex data clear and clean, and our feature engineering was crucial in making the model easier to interpret and perform better and faster. Some visualizations we created include: confusion matrices, ROC curves, and gain/lift charts, which we used to communicate model strengths and weaknesses, especially to a potential insurer we may pitch this model to. If we were to develop these models further, our current successes and knowledge allow us to create more complex models, with better accuracies and performances. Further research on this topic also may help reveal deeper patterns such as renewal seasonality or multi-year trends. Other aspects we can consider in the future include claim severity, electric vehicles, distracted driving, or governmental regulatory changes. The car insurance industry is always developing and reshaping, and these are some factors that are considered in professional models. Additionally, incorporating customer communication history or outside economic data can further increase predictive accuracy, as our data focuses on mostly claims and experience. Major companies leverage all sorts of data and other policies, like home, renters, life, and many others. One way would be to implement the models in a live setting, refreshing them continuously with new data and tracking how they impact policyholder retention.

5 Works Cited

Bisht, Ankit. “ML | Classification vs Regression - GeeksforGeeks.” *GeeksforGeeks*, 8 Jan. 2019, www.geeksforgeeks.org/ml-classification-vs-regression/.

GeeksForGeeks. “Hyperparameter Tuning.” *GeeksforGeeks*, 7 Dec. 2023, www.geeksforgeeks.org/hyperparameter-tuning/.

Lledó, Josep (2023), “Dataset of an actual motor vehicle insurance portfolio”, Mendeley Data, V1, doi: 10.17632/5cxyb5fp4f.1

---. “ML - Gradient Boosting.” *GeeksforGeeks*, 25 Aug. 2020, www.geeksforgeeks.org/ml-gradient-boosting/.

GeeksforGeeks. “AUC-ROC Curve.” *GeeksforGeeks*, 25 Nov. 2020, www.geeksforgeeks.org/auc-roc-curve/.

---. “F1 Score in Machine Learning.” *GeeksforGeeks*, 27 Dec. 2023, www.geeksforgeeks.org/f1-score-in-machine-learning/.

---. “Random Forest Algorithm in Machine Learning.” *GeeksforGeeks*, 12 July 2024, www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/.

Segura-Gisbert, J., Lledó, J. & Pavía, J.M. Dataset of an actual motor vehicle insurance portfolio. *Eur. Actuar. J.* 15, 241–253 (2025). https://doi.org/10.1007/s13385-024-00398-0

---. “Top 65+ Data Science Projects with Source Code.” *GeeksforGeeks*, 8 Feb. 2024, www.geeksforgeeks.org/top-data-science-projects/. Accessed 29 Apr. 2025.

Gupta, Mohit. “ML | Linear Regression - GeeksforGeeks.” *GeeksforGeeks*, 13 Sept. 2018, www.geeksforgeeks.org/ml-linear-regression/.

Kumar, Ajitesh. “Mean Squared Error or R-Squared - Which One to Use?” *Data Analytics*, 30 Sept. 2020, vitalflux.com/mean-square-error-r-squared-which-one-to-use/.

Kundu, Rohit. “Confusion Matrix: How to Use It & Interpret Results [Examples].” *Www.v7labs.com*, 13 Sept. 2022, www.v7labs.com/blog/confusion-matrix-guide.

LexisNexis. “Auto Insurance Trends Report.” *LexisNexis Risk Solutions*, risk.lexisnexis.com/insights-resources/white-paper/auto-insurance-trends-report.

Meyer, Susan, et al. “2025 Auto Insurance Trends Report.” *Thezebra.com*, The Zebra, 16 Jan. 2025, www.thezebra.com/resources/car-insurance/auto-insurance-trends-report/.