Car Insurance Predictive Model

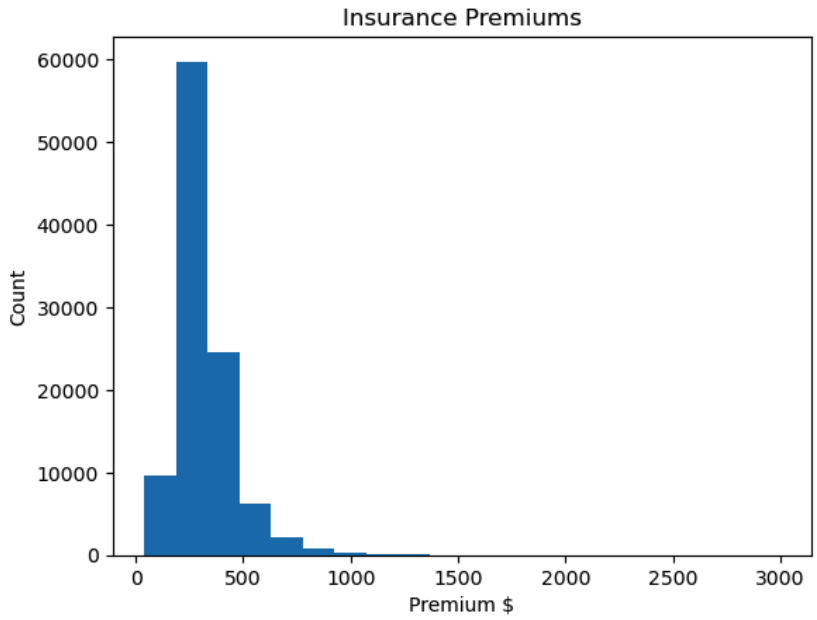
The automotive industry is a massive market that many people interact with every day. Largely, these interactions are driven by the purchase and sale of cars. With purchasing a new car, there are lots of obvious decisions surrounding the make, model, color, and other features based on customer preferences. However, one of the less contemplated aspects is understanding the subsequent costs that come with your car. The price is one thing, but an often overlooked cost inherent with owning a car is the insurance associated. In this project, I will build out a model that can predict insurance costs based on a variety of car features .This will assist any customer looking to understand their all in costs for a car, ultimately allowing them to make a more informed decision and avoid unexpected costs.

Data

The dataset I will be using to build this model is one from a motor vehicle insurance portfolio, accessible through Mendeley Data with over 100k entries (Lledó, Josep; Pavía, Jose M. (2024)). It has 30 columns including features such as Insurance Premium, Number and Cost of claims, Renewal Date, Value of the vehicle, Power, and other car features/ policy details. After importing, I dropped one irrelevant column and created an ‘age’ feature based on the date of birth of the insured person resulting in a final column count of 30.

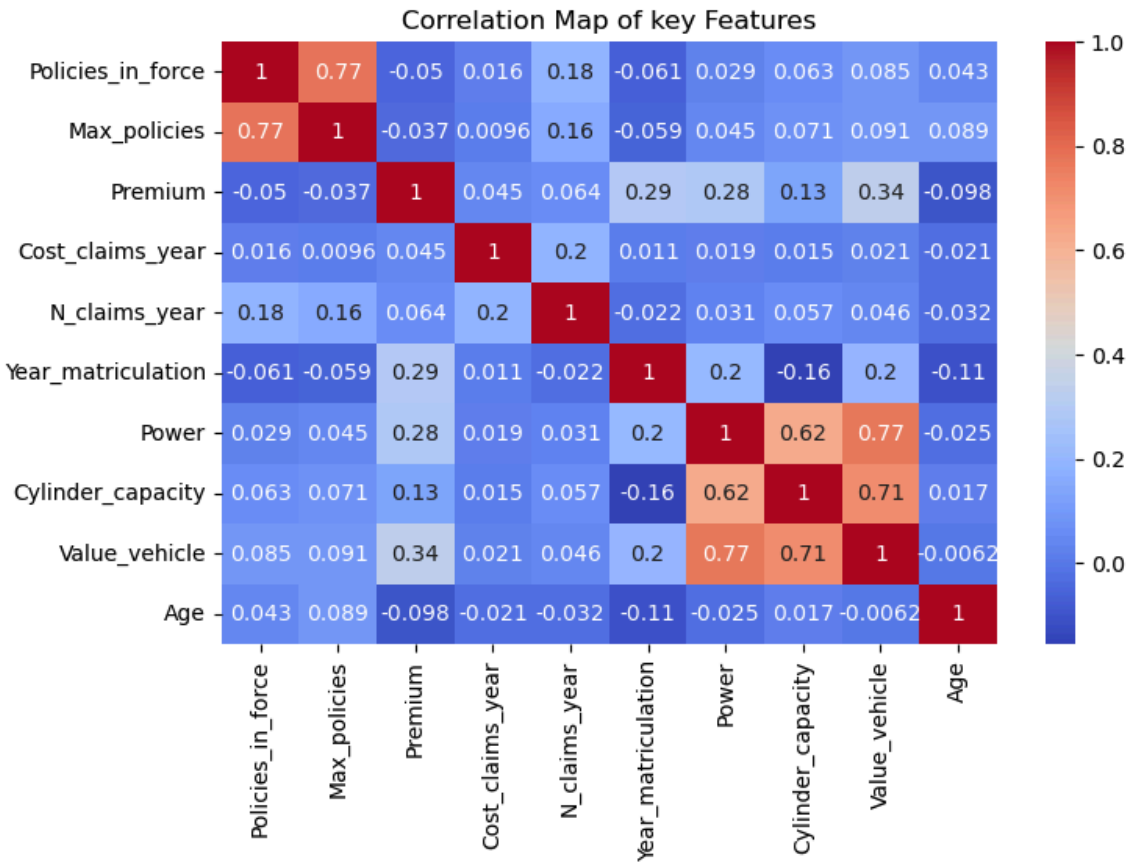
Exploratory Data Analysis

Figure 1:



My target variable of Insurance Premiums was quite dispersed and skewed to the right with a mean of $320 and a median of $300. Creating a correlation map from the dataset made it clear that there were few variables that were more strongly correlated with the Premium. Figure 2 below shows a condensed correlation map between features of high importance. Figures 3 and 4 are scatterplots of the Premium plotted against two of the more correlated features.

Figure 2:



Looking at this correlation heatmap we notice a number of collinear features… As such, in modeling, I'll be keeping an eye to see if condensing features further will improve the scores. Furthermore, I noticed that Power and Vehicle Value seem to have the highest correlation with Premium. This led to me examining them further in Figures 3 and 4 below.

Figure 3:

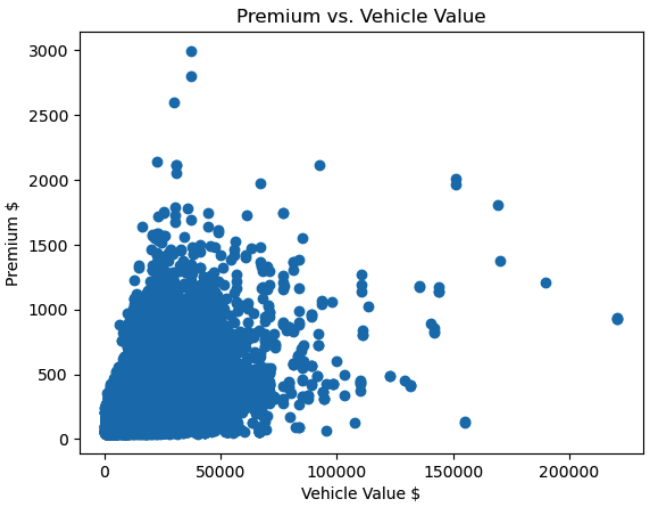
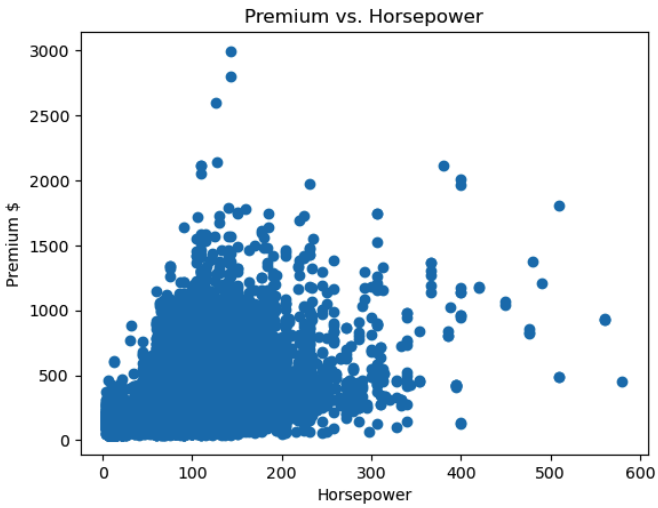


Figure 4:



Upon examination, neither of these features appeared to show strong correlation and had relatively low Pearson correlation coefficients (with p values being returned as 0.0). Another feature that showed decent correlation was the Year of Registration, which also appeared to show an exponential relationship with Premium. My next step was to compare Year of Registration to the log of the Premium to see if that correlation would improve. However after doing this, it did not improve enough for it to be a factor (Figures 4 & 5). Figure 6 shows a list of features, ordered by level of importance based on my Random Forest Model. For the most part, this reinforces what the correlation coefficients were showing with the exception of Age. I also tried running some single variable regressions with Premium to confirm the relatively weak relationship with some of these independent variables. R2 scores came back lower than the correlation coefficients (.16 for Vehicle Value and .08 for Year of Registration).

Figure 4:

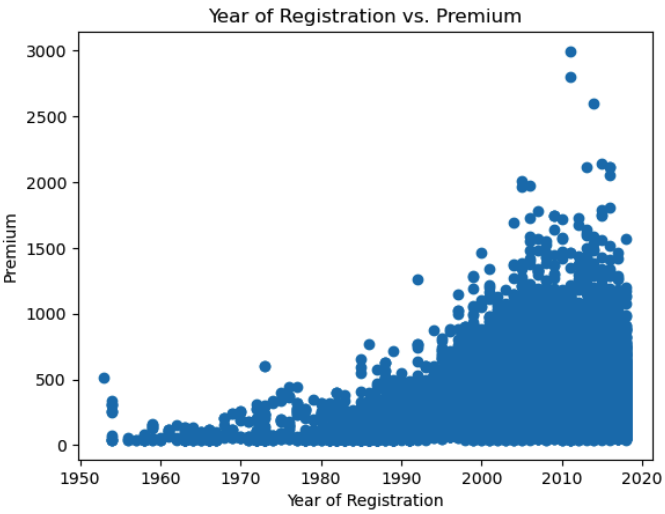


Figure 5:

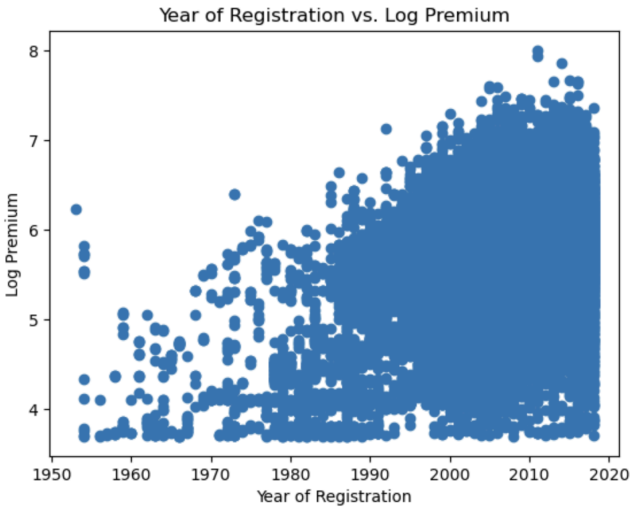


Figure 6:

	Feature	Importance
17	Value_vehicle	0.154670
14	Year_matriculation	0.128611
22	Age	0.121584
0	ID	0.085314
11	R_Claims_history	0.061122
21	Weight	0.054609
20	Length	0.053521
15	Power	0.048825
16	Cylinder_capacity	0.047924
2	Seniority	0.042417
10	N_claims_history	0.038449
7	Payment	0.035794
8	Cost_claims_year	0.021196
3	Policies_in_force	0.020461
13	Second_driver	0.017971
4	Max_policies	0.015217
6	Lapse	0.012341
1	Distribution_channel	0.010572
12	Area	0.010332
9	N_claims_year	0.006840
19	Type_fuel	0.004965
18	N_doors	0.003895
5	Max_products	0.003372

Modeling

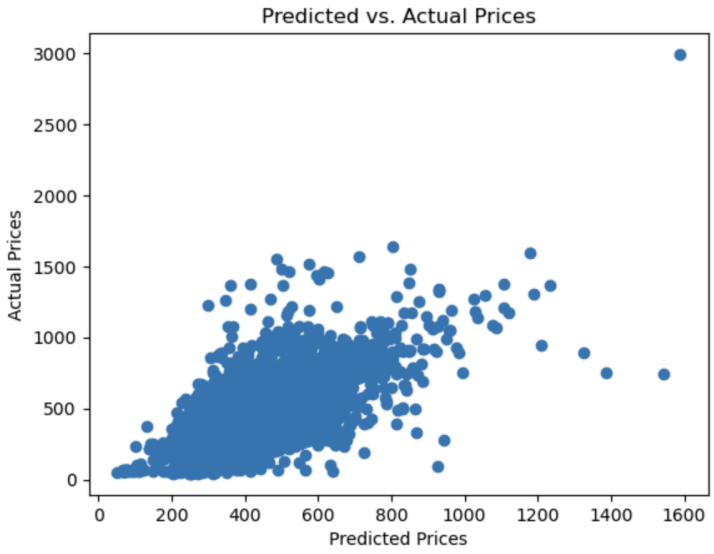
The error metrics I used to evaluate my model included the R2 score, and RMSE as this would tell me on average how far off in price I was from actuals. I first ran a few OLS models, testing with scaled data and with limiting the features as different variations. These models performed relatively poorly with R2 scores in the .25 range. I then ran a Lasso model to see if weighing the features helped improve scores at all. With this, I ran a GridSearchCV to evaluate the best hyperparameters to include for the model. Even with this, my model was still coming out to approximately .25 R2 score, with a RMSE of almost $120. Given that the mean premium was $320, this seemed like potentially too much error to be a useful predictor. The last two models I ran were tree based models. First, I tried a Random Forest model that yielded much different results than the Linear Regression models, showing an R2 score on the test data of .57 and a RMSE of approximately $85 (Root Mean Percentage Error of .42). Lastly, I ran a Gradient Boost Regressor model which scored much better than the Linear Regressors, but slightly worse than the Random Forest. See Figure 7 for a list of the results and parameters.

Figure 7:

	Model	Hyperparameters	R2_score	RMSE
1	OLS - All Features	Default Hyperparameters	0.255	N/A
2	OLS - All Features (Scaled)	Default Hyperparameters	0.255	N/A
3	OLS - Top 3 Features	Default Hyperparameters	0.174	N/A
4	Lasso Regression - All Features	aplha = .205	0.255	119.29
5	Random Forest Regressor - All Features	n_estimators: 1000, max_depth: 25	0.572	89.5
6	Gradient Boost Regressor - All Features	n_estimators: 189, learning_rate: .078, max_depth: 9	0.542	90.46

I was encouraged to see the jump in predictive power when switching to the tree based models. Figure 8 shows a scatterplot of predictions vs. actuals on the test data through the Random Forest model.

Figure 8:



Conclusion

Given that the average price of car insurance in this dataset is $320 and my RMSE is $85 (approximately 25% of the average cost), I have concerns about the usability of this model in practice. I do however, believe further feature engineering/ modeling can be done to increase the predictive power of this Tree based model. If I were able to increase the R2 score closer to what I was seeing on the training data and show an RMSE under $50, I believe this model could be useful for understanding rough estimates of the insurance associated with a car. One possible next step would be to subset the data by car type and build models specifically for each one to see if that improves the score, or if there are any car types or makes for which the model would be more useful. I could also run a clustering algorithm on the dataset and use those clusters as subsets for further modeling.