Insurance Claims Modeling Project

Team 15

White Paper

Members:

Ayotunde Oluleke

Kayode Balogun

Chukwudi (Michael) Okereafor

Elizabeth Ajabor

Emmanuel Maisaje

Mentor:

Stefan Ferreira

Introduction

In insurance, claim estimation is an integral part of the premium pricing process. Being able to generate a fully transparent and highly accurate prediction for a given risk is indispensable to guarantee profitability for all insurers as it: 1) enables insurers to accurately price for various risk levels, 2) facilitates regulatory reporting, and 3) informs business strategy on how various market segments should be treated.

Predictive modeling is a revolutionary concept in claims handling, insurers may use predictive models to help identify potentially fraudulent claims. It also can be used to score claims based on the likely size of the settlement, enabling an insurer to more efficiently allocate resources to higher priority claims.

To stay competitive in the insurance market environment, a company's premium should be sensitive to the indices that can influence an insured subject's likelihood to end in a claim. They should also consequently respond to changes in the premiums being offered by competitors as well as the subjective circumstances that inform the risks associated with the insurance of a particular subject. In determining the premium pricing, proper care must also be taken to ensure that premiums are not so high that potential customers are driven into the hands of competing insurance service providers and not so low that the company finds it challenging to stay profitable and remain in business. This project is aimed to predict the claim severity for vehicle insurance using personal and vehicle information and improve upon the current model built by the Financial Services Team

Problem Statement

To develop a machine learning algorithm that uses the demographic and geographic information of potential clients that can accurately predict claim severity in order to propose a sensible and competitive premium amount.

Methodology

Our team has attempted to solve this problem by studying insurance terms and subject matter. We then proceeded to exploratory data analysis to get a better understanding of underlying correlations and patterns found in the data. For the modeling phase, we started off doing feature selection, feature engineering and data preparation to feed clean data into the model. Following this, we built 2 machine learning models; Our team resolved the issues outlined in the problem statement by developing a robust supervised model for the prediction of insurance claims. A linear regression model and an XGBRegressor model were trained to fit the particularities of different policymakers commonly found in industry. We used these models to predict claim severity for current policyholders, in order to set premiums for prospective policyholders.

Exploratory Data Analysis(EDA).

This process was carried on our dataset with purposes of getting a better understanding of underlying correlations and patterns found in the data.

We unveiled certain patterns of notable importance. Some of which include;

- Risk profiles within the Age range of 26 - 35years made the highest counts of claims (44%) when compared to five other age_ranges within policyholders that actually made claims (figure 1.1 and figure 1.2). This probably could be due to insufficient driving experience or other traits that are outside our scope. Meanwhile, our insight is the older the age, the less the claim amount counts.

- Single Risk profiles made a 56.4% count of claims when compared to three other marital_status of policy holders that actually claimed. We've discovered no proven parameter to justify this but propose that widowed risk profiles make the least claims (figure 2.1 and figure 2.2).

- As shown in figure 3.1/ figure 3.2, policyholders who work as educators made the highest claim counts, with human resource consultants having a high frequency of no_claim counts. Although we expected "driver'" as occupation that leads to this point, it only portrays an out of box phenomenon displayed within our dataset.
- Figure 4 shows the average claim amounts on four attributes of policyholders. It depicts that for the colour of vehicles, Burgundy made highest average claims, for marital status, singles made highest claims. Part-time workers have the highest average claim, as well as those in the medical industry.

Feature Engineering

To improve the predictive capacity of our model, we first took steps to reduce the dimensionality of the model. We used the correlation coefficient and variance threshold to identify the features that were less important in determining the target feature, and we removed those features from the model. We used a linear regression algorithm to create a base regression model and we compared the performance of a more advanced algorithm (XGBRegressor) against the linear model. The root mean squared error (RMSE) was used as a metric for the evaluation of the performance of both models. we'd compare "RMSE" outputs of our base regression model and xgbregressor models to laid down standards of under fitting, overfitting, or great performance.

Modeling

- Figure 4.1a displays a distribution of the Actual versus Predicted values on our test_set while training our model using XGBRegressor. Figure 4.1b shows the predicted vs actual values for the linear regression model, however this model looks good but predicts negative values. It wasn't immediately clear why this is the case, and there was unfortunately not enough time to resolve this issue, as we rather needed to focus on the stronger XGBoost model.

- Figure 4.2 shows an overview of which features were most important to our XGBoost Regressor's performance using Shapley Additive exPlanations (SHAP). It displayed the top 20 important features and the horizontal axis showed how much the features contribute (average impact) to the output predictor variable.

- Table 1 compares the performance of our Linear Regression and XGBoost Regressor model based on the Root Means Squared Error and the Mean Absolute Error.

Data Engineering

To improve the users experience while trying to make predictions, an app was created with a user interface that granted users the ability to input certain features and get a prediction of the average claim amount. This app was built using streamlit in and hosted on the cloud by using Amazon web Services like the S3 bucket and an EC2 instance.

Conclusion and Recommendations

This project corroborates the theory that there are features that affect the likelihood of an insured event ending in a claim more strongly than other features. The process of accurately predicting these features and developing models that improves the capacity of determining the appropriate premiums to be charged for insurance is of immense economic benefit. In this project we have created the model but we will conclude by focussing on how this model can be further improved.

Recommendations:

Here are some recommendations that can make the work we have here better;

Features Used

The strength of a predictive model depends highly on the quality and quantity of the data available. There are features that would very likely prove to be important, but those features were not added during the process of data collection. An example is the total Distance covered by the vehicle in its lifetime, If the vehicle has been in an accident before and the frequency of routine car maintenance or service trips. If these details are added during data collection, they will likely contribute reasonably to the accuracy of the model.

Process Automation

The process of data collection and modeling can be totally automated by creating systems for collecting data (eg. online survey forms that can be filled out to feed a database), so that the necessary data for the modeling process can be obtained more easily and accurately. These efforts will contribute to making the model better.

figure. 1.1

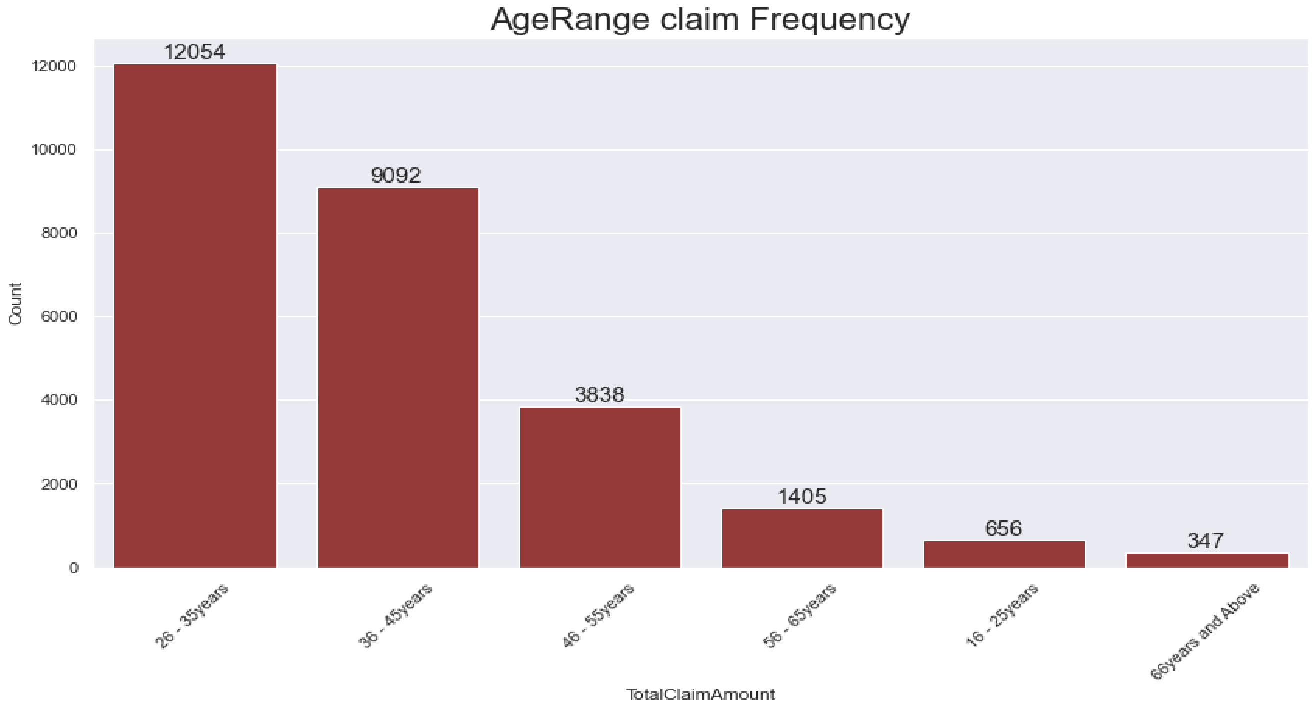


figure. 1.2

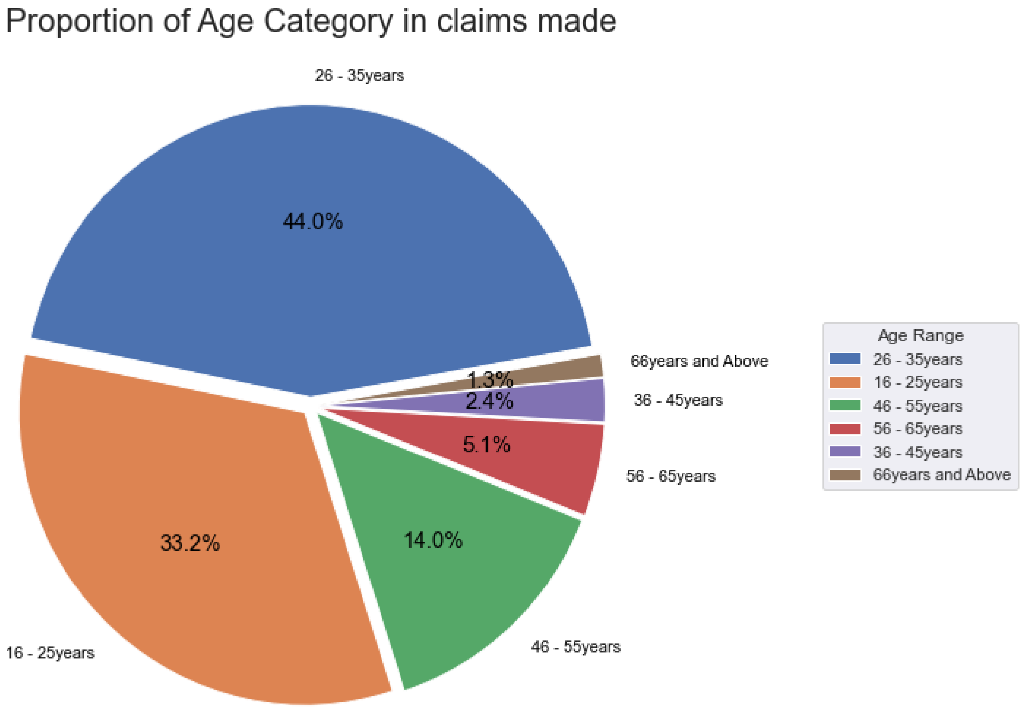


figure. 2.1

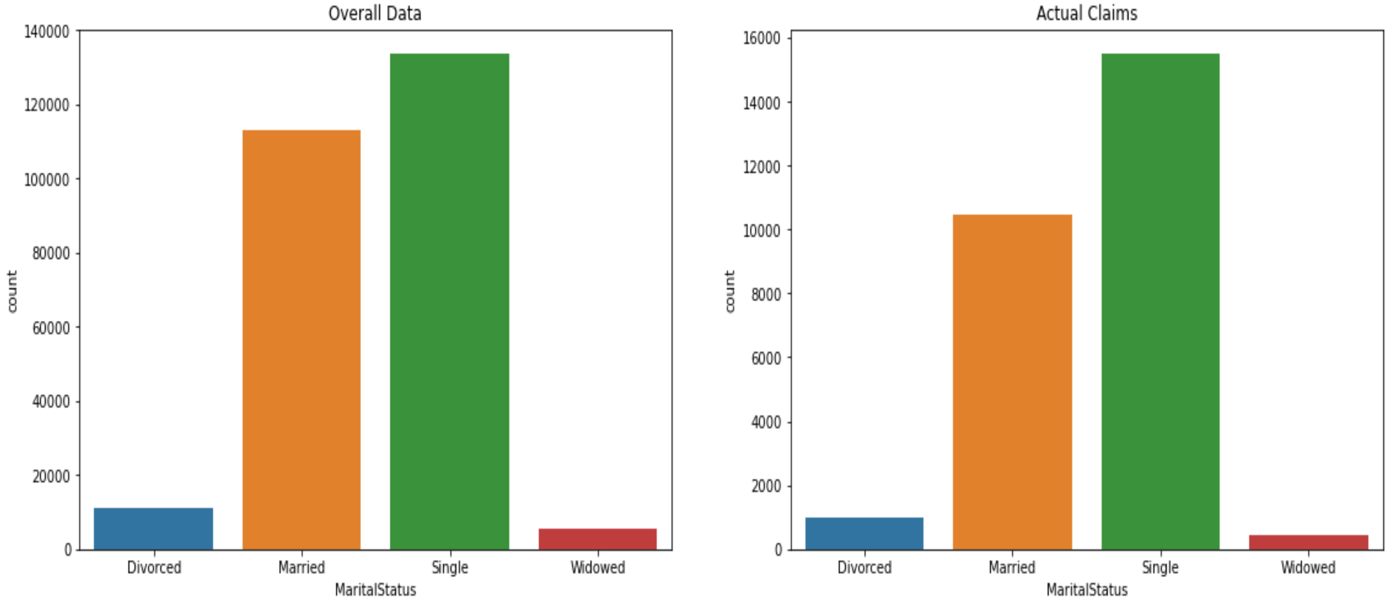


figure. 2.2

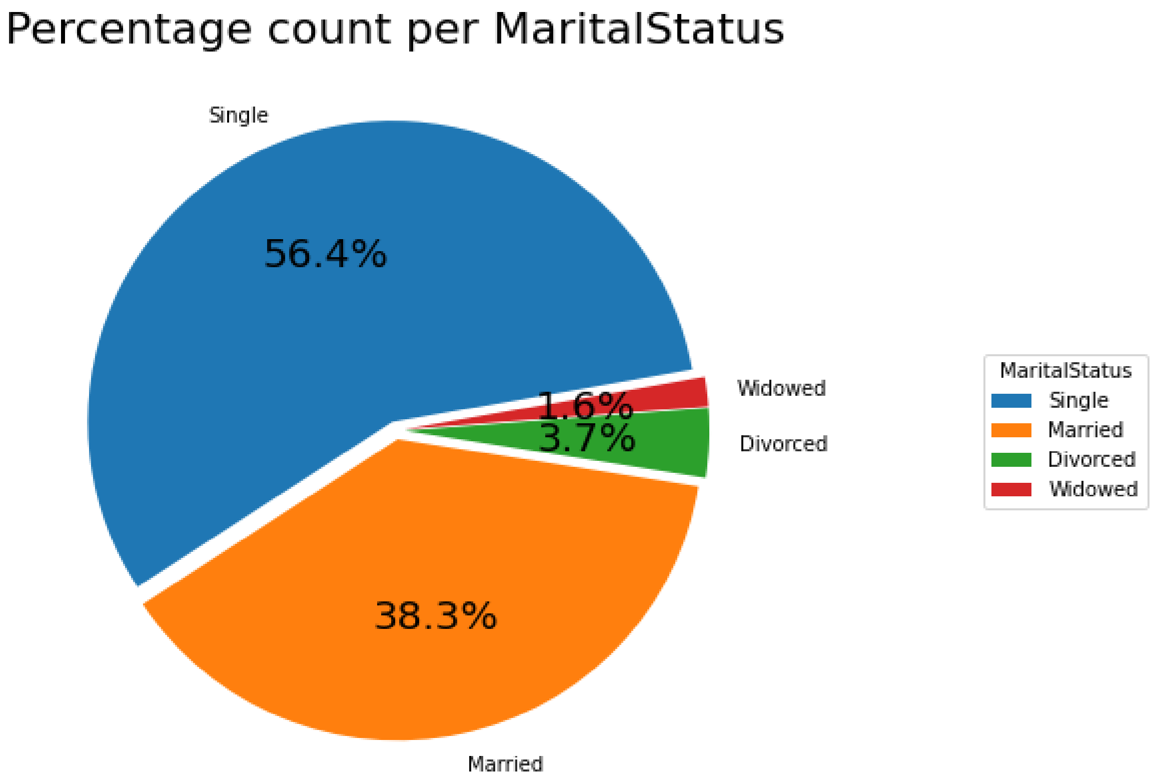


Figure 3.1

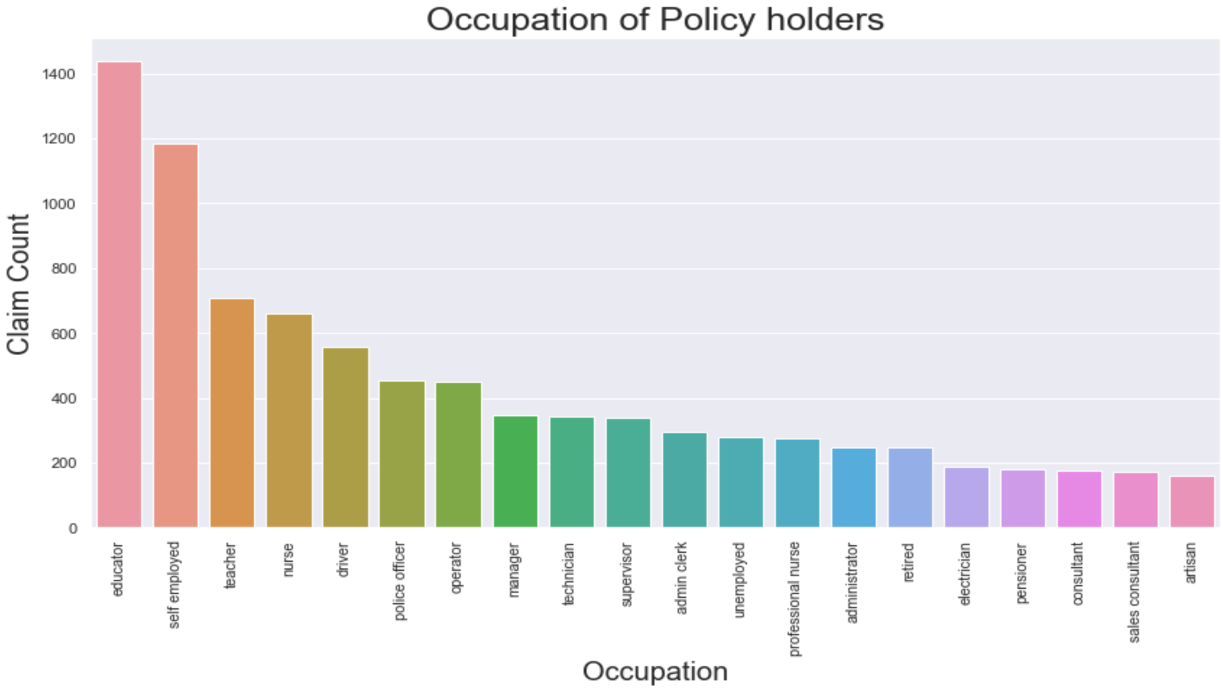


Figure 3.2

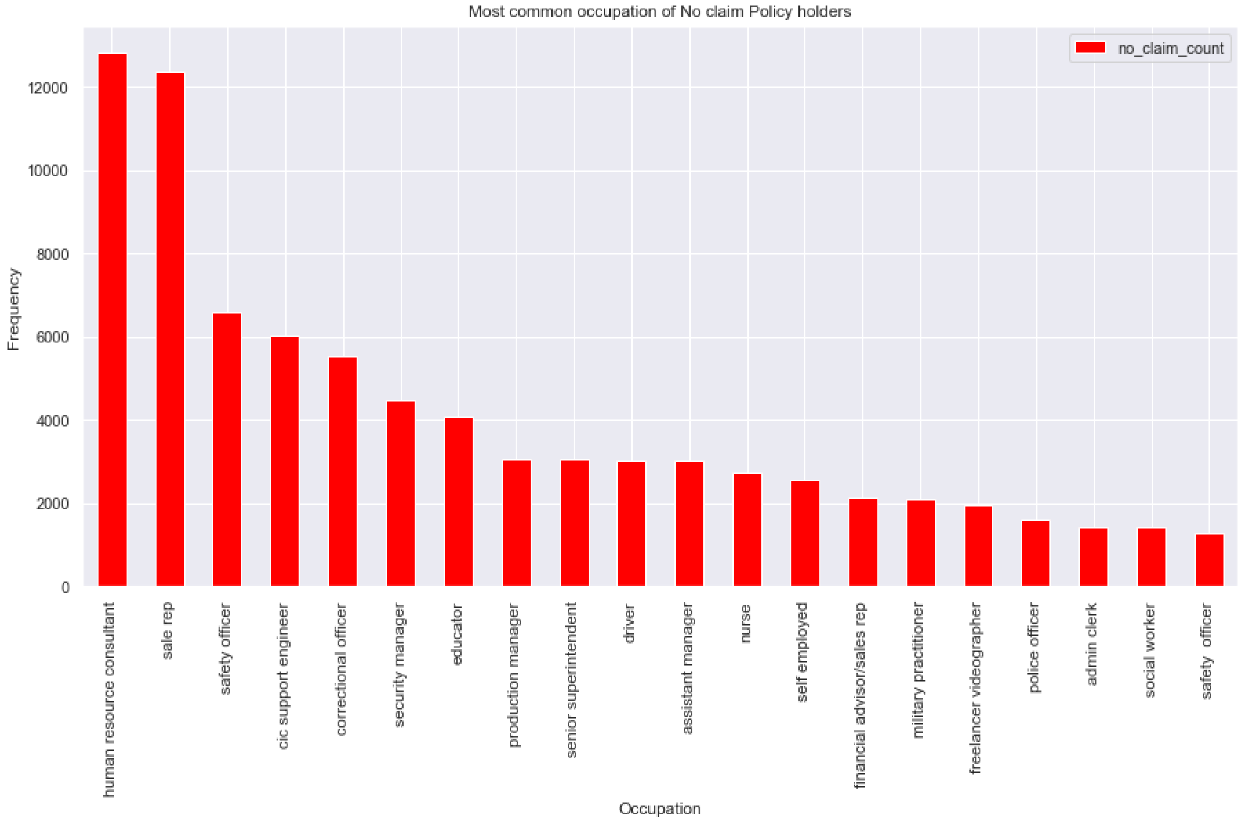


Figure 4

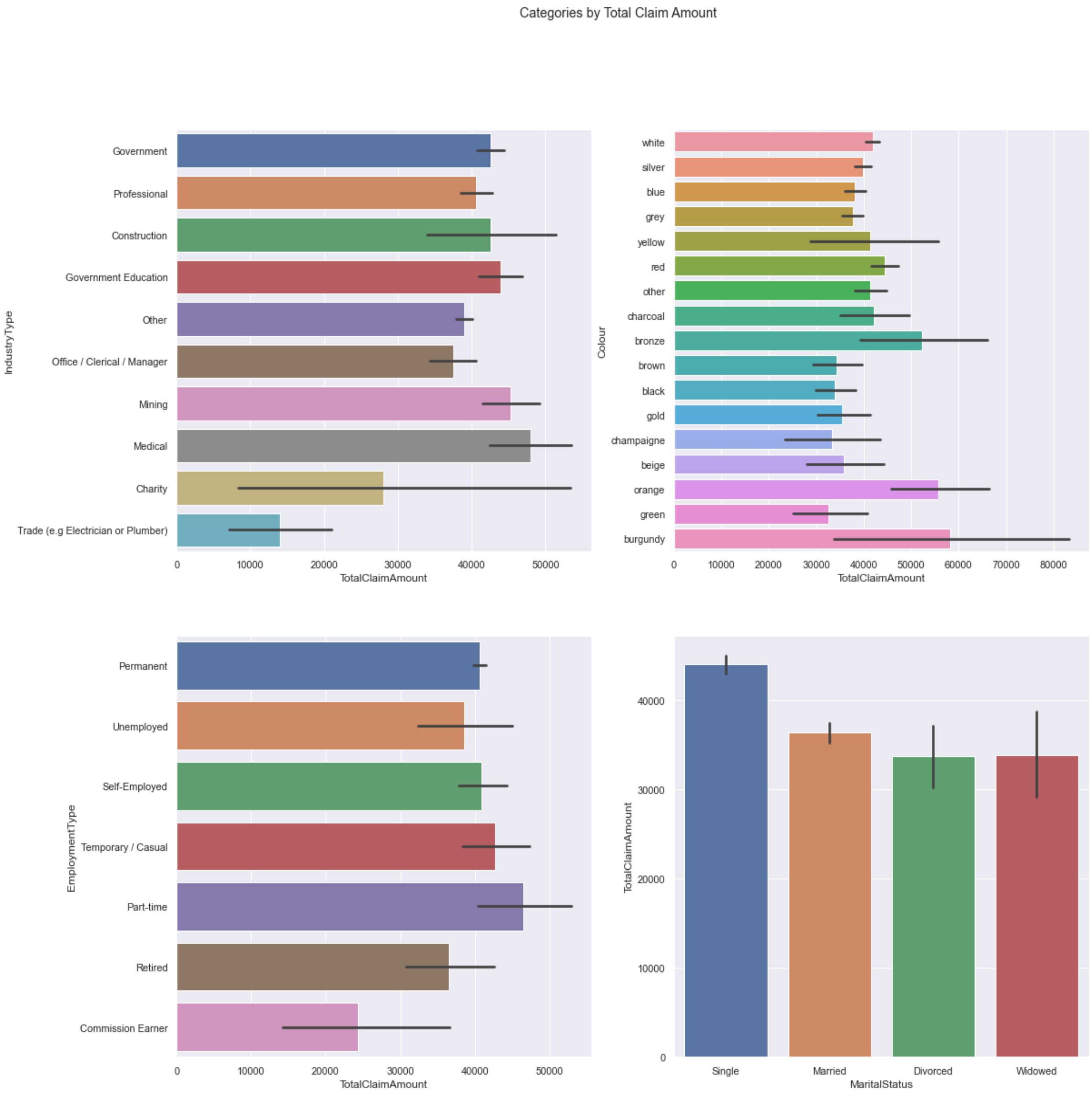

Categories by Total Claim Amount

Table 1

	Train MAE	Test MAE	Train RMSE	Test RMSE
Linear Regression	5949.07	5938.24	19438.84	20035.89
XGBoost Regressor	5792.14	5919.09	19174.31	20003.51

Figure 5.1a

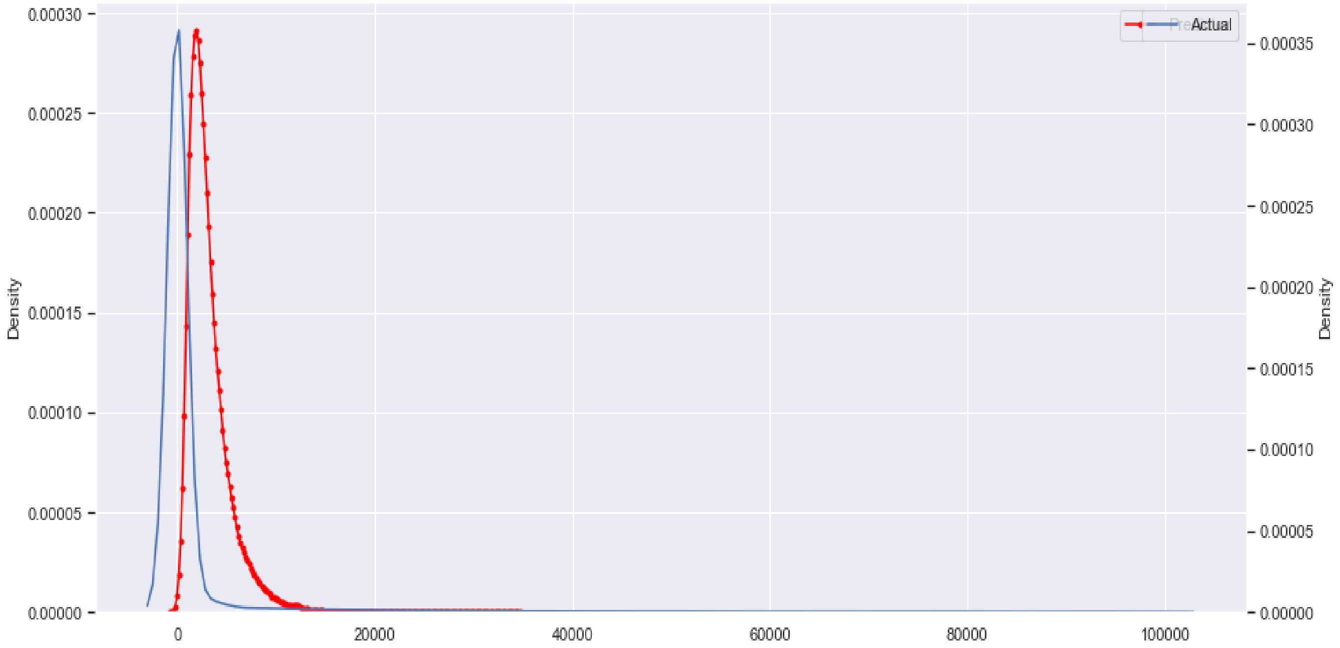


Figure 5.1b

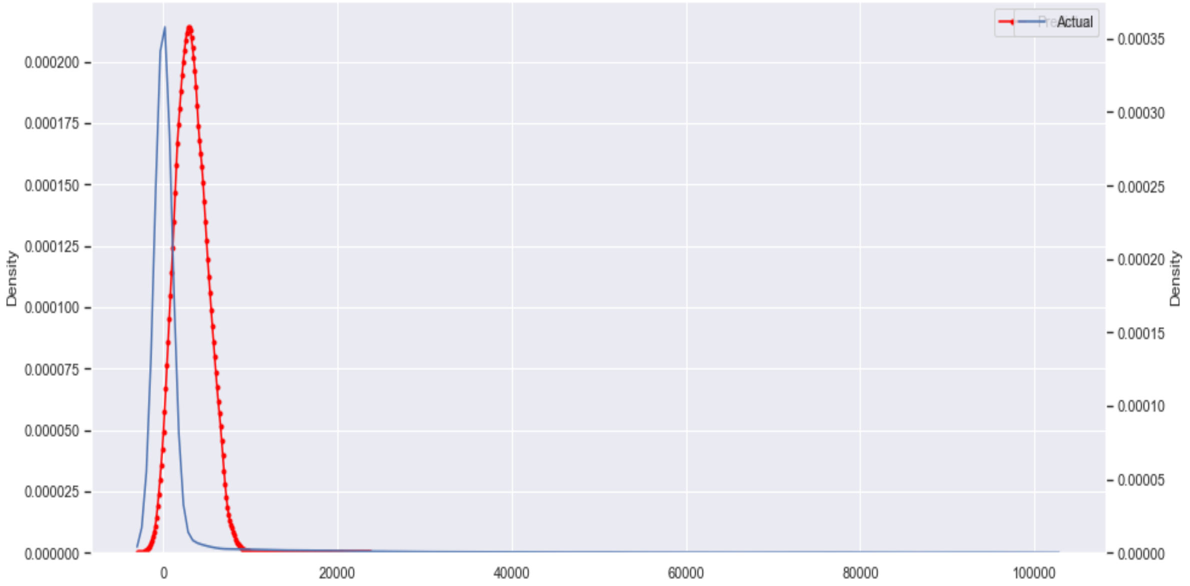


Figure 5.2

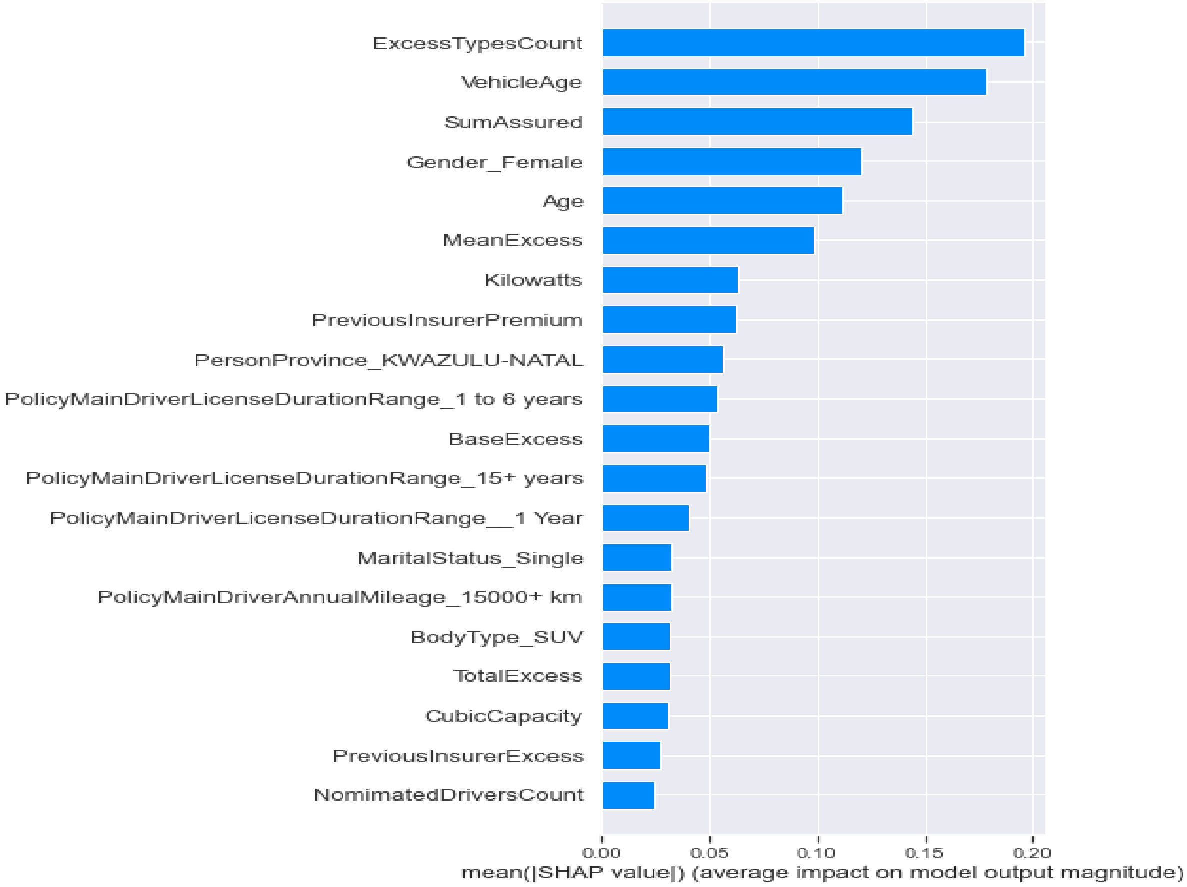
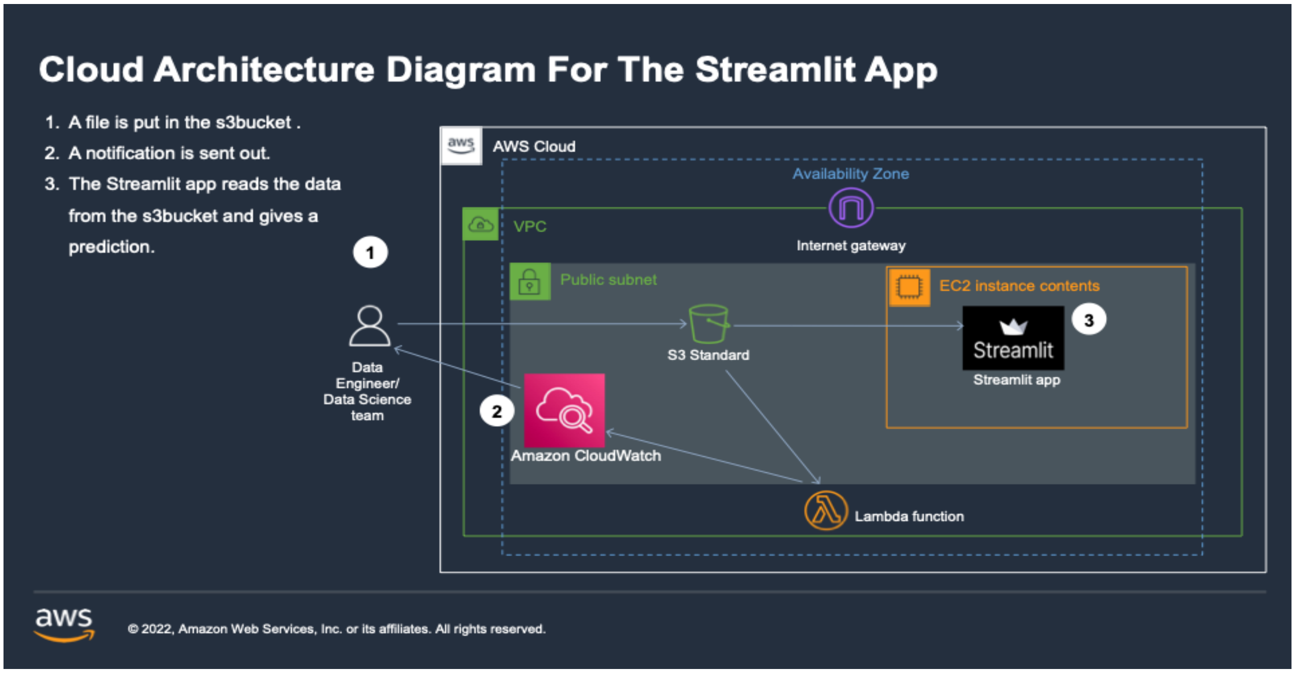


Figure 6



Cloud Architecture Diagram For The Streamlit App

1. A file is put in the s3bucket .
2. A notification is sent out.
3. The Streamlit app reads the data from the s3bucket and gives a prediction.

References

1. https://seaborn.pydata.org/tutorial.html
2. https://www.analyticsvidhya.com
3. www.datacamp.com
4. https://www.towardsdatascience.com
5. https://arxiv.org/pdf/2107.11059.pdf](https://arxiv.org/pdf/2107.11059.pdf