Measuring the effect of rain over Car Insurance Claims Frequency in Mexico City

Juan Jose Echevarria
University of Colorado
Boulder
Boulder, Colorado, USA

ABSTRACT

This project aims to enhance the customer service of an insurance company by analyzing the correlation between rainfall and the increase in car insurance claims. Recognizing that adverse weather conditions can lead to a surge in claims, this study focuses on quantifying the impact of rainfall on car insurance policies. By leveraging historical weather data and claim records, we will employ statistical techniques to identify patterns and trends. The findings will enable the company to predict periods of high claim activity and optimize resource allocation, thereby improving response times and customer satisfaction. This project seeks to not only improve operational efficiency but also enhance the overall customer experience by providing timely and relevant support.

Key findings

1. 78.18% of Mexico City's rainfall happens between July and October of each year
2. There is a clear difference in claims' frequency in rainy days.
3. The biggest difference in claims frequency is in the months of September and October. There is also a social element to claims frequency, in this case July and August are affected by school being on summer break, so the real difference can be seen when the kids go back to school.
4. During the COVID19 pandemic the was a general decrease of claims since people didn't leave their homes for an extended period. Frequency hasn't returned to pre-pandemic levels since most companies in Mexico implemented remote work, the norm in the country is at least 1 day of Home Office a week, which also helps to minimize current claims frequency.
5. Mexico City is divided in 16 "Alcaldias", like Counties in the U.S. This is also relevant because most of the offices are in certain areas of the city and residential areas are also in different counties, so people travel back and forth creating a change to having a car accident.

6. With the rainfall, claims and fleet information we have created a model to predict where in the city will most car accidents happen. All you need is the current fleet information and a trusty weather prediction of "Rain-No Rain"

Related Work

Currently, the insurance company is using empirical data and experience to try and satisfy the increased demand of claims adjusters during the rainfall months in Mexico City.

The idea is to come up with a better way of doing things while improving the customer service to our policy holders.

Proposed Work and Timeline

Research will follow the following steps:

1. RESEARCH ON PROBLEM-SPECIFIC TRENDS (1 WEEK):
    a. Historical analysis of trends in Claims occurred in Mexico City in the last 7 years (2017-2023).
    b. Examination of the evolution of the Insurance Company fleet during the same period.
    c. Identification of historical information on rainfall in Mexico City.
2. EXPLORATORY DATA ANALYSIS (EDA) (2 WEEKS):
    a. Comprehensive exploration of the three datasets to uncover insights.
    b. Identification of patterns, correlations, and outliers in historical data.
    c. Visualization of key EDA findings to inform subsequent modeling.
3. MODEL BUILDING (1 WEEK):
    a. Development of predictive models tailored to claims reporting with and without rainfall.
    b. Experimentation with various statistical algorithms and subsets of data.
    c. Fine-tuning model parameters for optimal predictive accuracy.
4. MODEL EVALUATION (1 WEEK):
    a. Rigorous evaluation of model performance using cross-validation techniques.
    b. Application of relevant evaluation metrics, focusing in accuracy.
    c. Sensitivity analysis to identify influential factors affecting predictions.
5. FINAL VISUALIZATIONS (1 WEEK):
    a. Creation of informative and intuitive visualizations.
    b. Clear presentation of predicted areas with the most frequency uprise.
    c. Visual representation of the research findings for stakeholders.

Evaluation

To ensure the accuracy and reliability of our predictive models, I employ a rigorous evaluation process. This process involves assessing the performance of our models, both quantitatively and qualitatively, to determine their effectiveness in making accurate predictions.

Quantitative Evaluation Metrics:

Accuracy: Accuracy measures the proportion of correctly predicted outcomes among all predictions. It provides an overall assessment of model correctness.

Experimental Setup

With the Data Bases available (Rainfall, Claims, Fleet) we were able to create joined subsets where we fitted Linear Regression Models to try and predict with an overall accuracy how many claims will happen by Alcaldia.

The models are based specifically over the number of cars insured by zone. In Non-Life Insurance we use "exposure" which is loosely defined as the number of vehicles insured for one year, we did this month by month with the changing fleet of vehicles using the following formula:

$$Exposure = \frac{Days\ of\ Policy\ In\ Force}{365}$$

Where, "Days in Force" means the number of days that the policy was at risk and any claims would be accepted by the insurance company.

For example, if a policy ends on June 17th and we were calculating the exposure of such June, the exposure would be 17/365. In the same example, if we did the math for exposure but for the previous May exposure would be 31/365; and for July it would be 0/365 since the policy expired a month before.

Exposure is calculated by policy including all its transactions like Issue, Cancellation, Renewal and Rehabilitation.

Frequency is defined by the formula:

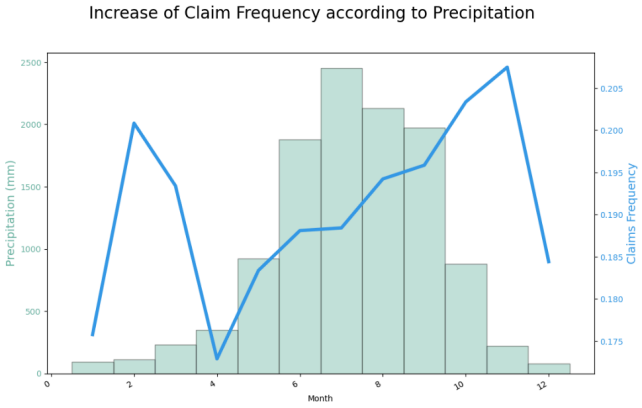
$$Frequency = \frac{\#\ of\ Claims}{Exposure}$$

Methods to Compare

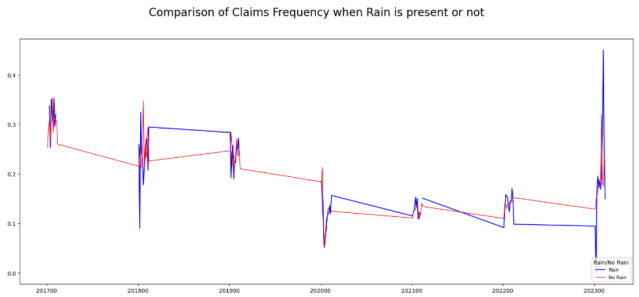
We used different methods and created 7 different models where the main difference is the data used between them, trying to create a prediction useful depending on if the Company doesn't have weather information (General Model) and is the Company has a trusted weather source (Rain Model/Dry Model). We also have an even further model to predict the number of claims using the current fleet exposure (General Alcaldia Model, Rain Alcaldia Model and Dry Alcaldia Model).

Key Results

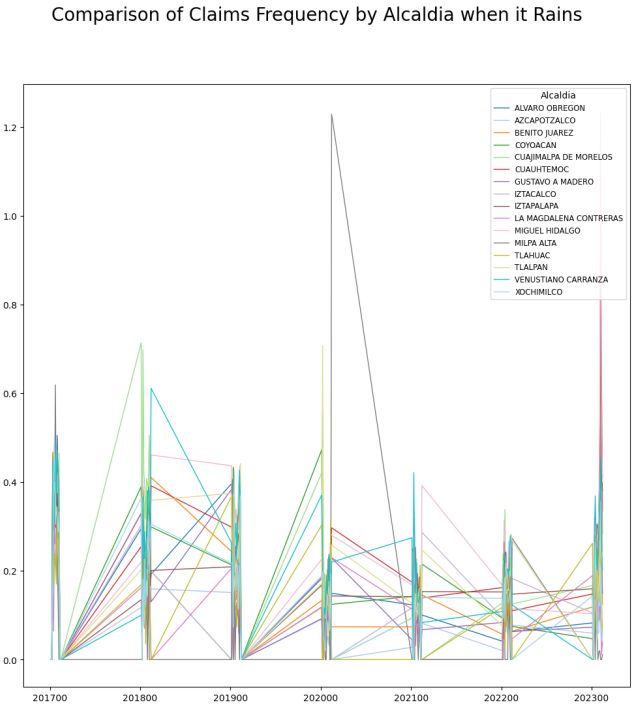
1. 78.18% of Mexico City rainfall happened between the months of July and September. If we check the claims' frequency on those months we can see and increase in the tendency, getting the highest point in October and then falling to restart the year:



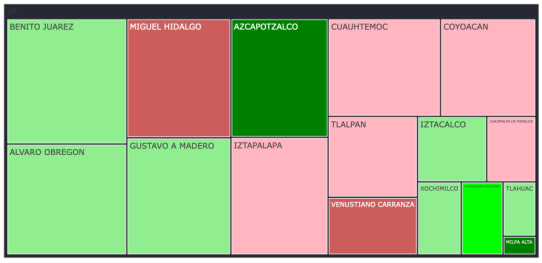
2. When we split the data between rain and non-rain days, we see that there's a big difference between the two. Also in this graph we see that claim frequency went on an all time low during the COVID pandemic in 2020 and most of 2021, recovering a little in 2022 but the introduction of remote work in Mexico has also modified the frequency:



3. Then we tried introducing the zone or county or “Alcaldia” feature to try and visualize the changes in frequency when it rains, as you can see there is a lot of chaos going on:

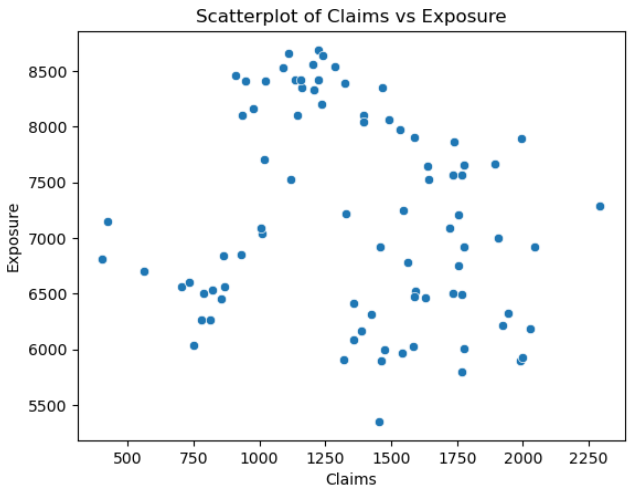


So then, we tried to measure the changes in frequency against the mean (18.86%), the size of each box represents the exposure by Alcaldia, so the bigger means more cars in the area and the color represents the difference against the mean, so the greener it is the frequency is smaller while the redder/pinker it is higher than the mean:

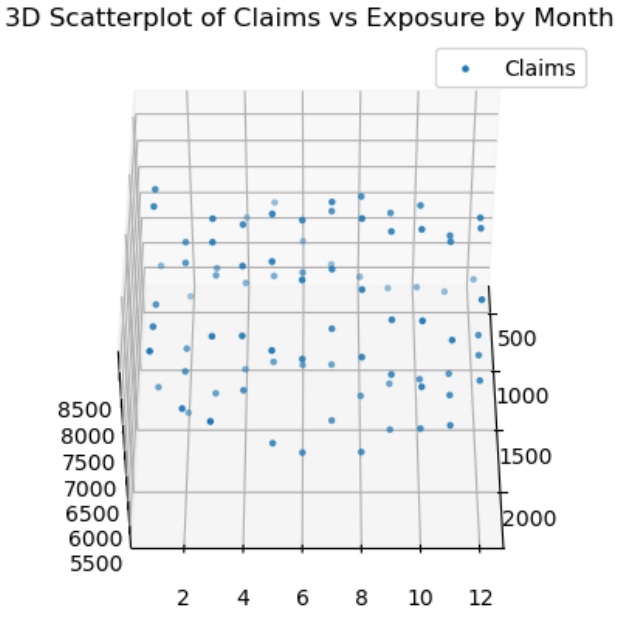


The results are somewhat expected, the main areas with offices/business are Miguel Hidalgo and Cuajimalpa de Morelos while the biggest residential areas are in Coyoacan and Tlalpan, the Mexico City airport is in Venustiano Carranza, all of these Alcaldias are red, meaning that they have the most frequency upside against its mean and it should be where the Insurance Company set the most claims adjusters, especially on heavy traffic hours like early in the morning or before/after business hours.

4. While we were trying to create our models we tried graphing the data as a scatterplot, just claims versus exposure since that’s how frequency is calculated:



The problem is that even with the same exposure the claims where all over the place, we introduced a third dimension with the month of claims occurrence, and we got something a lot better:



5. We created 7 models, they can be used by the Insurance Company depending on the data that they can get, especially rain forecast and fleet exposure.

   The first one is just using the claims and exposure data, there is no Rain information:

```
In [78]: reg = LinearRegression()
         reg.fit(X, Y)
         predictions1 = model.predict(X)
         print(
           'mean_squared_error : ', mean_squared_error(Y, predictions1))
         print(
           'mean_absolute_error : ', mean_absolute_error(Y, predictions1))
         print(reg.coef_)

         mean_squared_error :  165107.1309923975
         mean_absolute_error :  320.89826735960315
         [10.87344056 -0.07339983]
```

   It has a mean error of claims of 320.89, that seem good but we felt it was high.

   So, the second model we tried some Machine Learning, separating into Train and Test subsets and tried the regression again:

```
In [70]: X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2, random_state = 28)

In [76]: model = LinearRegression()
         model.fit(X_train,Y_train)
         predictions = model.predict(X_test)
         print(
           'mean_squared_error : ', mean_squared_error(Y_test, predictions))
         print(
           'mean_absolute_error : ', mean_absolute_error(Y_test, predictions))
         print(model.coef_)

         mean_squared_error :  180646.4538672362
         mean_absolute_error :  350.26342079650954
         [19.0820722  -0.09164548]
```

   The mean error went up to 350.26, which is worse from the previous model, it certainly has to do with the almost 2 years of abnormal frequency because of the pandemic, that's why we decide to continuing building from the first model.

   We then tried the model with our subset or just rainy days:

```
In [87]: model_rain = pd.read_csv('expuestos_siniestros_rain.csv')
         X_rain = model_rain[['mes', 'expuestos']].to_numpy()
         Y_rain = np.array(model_rain['siniestros'])
         reg_rain = LinearRegression()
         reg_rain.fit(X_rain, Y_rain)
         predictions_rain = reg_rain.predict(X_rain)
         print(
           'mean_squared_error : ', mean_squared_error(Y_rain, predictions_rain))
         print(
           'mean_absolute_error : ', mean_absolute_error(Y_rain, predictions_rain))
         print(reg_rain.coef_)

         mean_squared_error :  41115.129043096946
         mean_absolute_error :  148.21558548405412
         [6.42295346 0.17251117]
```

   This came back with a standard error of 148.21, so we are on the right path.

   Doing the same with non-rainy days we get a model with a standard error of 286.18, higher than the rainy days model but makes sense since just one third of the days of the year:

```
In [88]: model_dry = pd.read_csv('expuestos_siniestros_dry.csv')
         X_dry = model_dry[['mes', 'expuestos']].to_numpy()
         Y_dry = np.array(model_dry['siniestros'])
         reg_dry = LinearRegression()
         reg_dry.fit(X_dry, Y_dry)
         predictions_dry = reg_dry.predict(X_dry)
         print(
           'mean_squared_error : ', mean_squared_error(Y_dry, predictions_dry))
         print(
           'mean_absolute_error : ', mean_absolute_error(Y_dry, predictions_dry))
         print(reg_dry.coef_)

         mean_squared_error :  114725.21200249023
         mean_absolute_error :  286.1870061002958
         [3.04412632 0.15478095]
```

   Finally, we tried the model introducing the "Alcaldia" feature, this should give us a better result. We tried with the full dataset:

```
In [90]: model_alcaldia_tot = pd.read_csv('alcaldia_expuestos_siniestros_total.csv')
         X_alcaldia_tot = model_alcaldia_tot[['mes', 'alcaldia','expuestos']].to_numpy()
         Y_alcaldia_tot = np.array(model_alcaldia_tot['siniestros'])
         reg_alcaldia_tot = LinearRegression()
         reg_alcaldia_tot.fit(X_alcaldia_tot, Y_alcaldia_tot)
         predictions_alcaldia_tot = reg_alcaldia_tot.predict(X_alcaldia_tot)
         print(
           'mean_squared_error : ', mean_squared_error(Y_alcaldia_tot, predictions_alcaldia_tot))
         print(
           'mean_absolute_error : ', mean_absolute_error(Y_alcaldia_tot, predictions_alcaldia_tot))
         print(reg_alcaldia_tot.coef_)

         mean_squared_error :  2002.1088427942814
         mean_absolute_error :  34.44245836857302
         [0.58573569 0.8870982  0.16144312]
```

   The standard mean error improves to 34.44, since it's a monthly measure there is only a 1.13 claim difference by day and by Alcaldia since there are roughly 30.4 days a month.

Doing the same with the “Rain” subset and the “Alcaldia” feature:

```
In [91]: model_alcaldia_rain = pd.read_csv('alcaldia_expuestos_siniestros_rain.csv')
         X_alcaldia_rain = model_alcaldia_rain[['mes', 'alcaldia','expuestos']].to_numpy()
         Y_alcaldia_rain = np.array(model_alcaldia_rain['siniestros'])
         reg_alcaldia_rain = LinearRegression()
         reg_alcaldia_rain.fit(X_alcaldia_rain, Y_alcaldia_rain)
         predictions_alcaldia_rain = reg_alcaldia_rain.predict(X_alcaldia_rain)
         print(
           'mean_squared_error : ', mean_squared_error(Y_alcaldia_rain, predictions_alcaldia_rain))
         print(
           'mean_absolute_error : ', mean_absolute_error(Y_alcaldia_rain, predictions_alcaldia_rain))
         print(reg_alcaldia_rain.coef_)

         mean_squared_error :  356.8980084125096
         mean_absolute_error :  12.076300917763537
         [0.40275291 0.34752271 0.17230134]
```

The standard error goes to 12.07 on average by month and “Alcaldia” since there are 9.47 rainy days a month we could say that the model has a 1.27 claim error for Alcaldia.

And lastly, the same exercise but with the “Non-Rain” or “Dry” subset:

```
In [92]: model_alcaldia_dry = pd.read_csv('alcaldia_expuestos_siniestros_dry.csv')
         X_alcaldia_dry = model_alcaldia_dry[['mes', 'alcaldia','expuestos']].to_numpy()
         Y_alcaldia_dry = np.array(model_alcaldia_dry['siniestros'])
         reg_alcaldia_dry = LinearRegression()
         reg_alcaldia_dry.fit(X_alcaldia_dry, Y_alcaldia_dry)
         predictions_alcaldia_dry = reg_alcaldia_dry.predict(X_alcaldia_dry)
         print(
           'mean_squared_error : ', mean_squared_error(Y_alcaldia_dry, predictions_alcaldia_dry))
         print(
           'mean_absolute_error : ', mean_absolute_error(Y_alcaldia_dry, predictions_alcaldia_dry))
         print(reg_alcaldia_dry.coef_)

         mean_squared_error :  1101.6864842643924
         mean_absolute_error :  23.93021933935371
         [0.26709369 0.78843495 0.16708333]
```

The standard error falls to 23.93 or 1.14 claims by Alcaldia worse, since there are 20.93 Non-Rain days in Mexico City by month in average.

Discussion

Challenges

1. Getting current rainfall data to include in the model. The data went as far as January 2024. We had Claims and Fleet Exposure information up to May 2024 but couldn’t include the data since Rainfall was incomplete.
2. We need better rainfall data since we can only determine if it rained a certain date, but we don’t know when and where, this would complete the model.
3. Claims data includes when and where the accident happened but only by Alcaldia, having zip code could help create a spatial map of where does claims happen more often and could help the Company place their claims adjusters better.

Changes

1. All the preprocessing happened in Visual Fox Pro (SQL) because that’s where the company keeps it data, would like to have it all in Python.

Conclusion

We set out to find a model to predict the number of claims that would occur in a month.

We managed to find that two of the best predictors where the exposure of cars insured and the “Alcaldia” where the car is registered in. This is also influenced by the fact if it rained or not.

Our “Rain” Model predicts that in a year with the current fleet should have 4,865 claims in a rainy day with a mean error of 12 claims.

Our “Dry” Model predicts that in a year with the current fleet there should be 12,687 claims on non-rainy day with a mean error of 24 claims.

We also found that there are other social factors that affect claim frequency like pandemics or the kids school break.

Future work

1. Include in the model the day of the week, this also has a social element to it, it has been proven that claims occur different by day of the week.
2. Check on the result of frequency of February and March, there’s something there that we can’t identify, at least in this exercise.
3. Move preprocessing to Python (Jupyter Notebook)

References

Precipitation Information from the Government of Mexico City

(https://datos.gob.mx/busca/dataset/precipitacion-actual-y-acumulada-por-estacion**)**

Precipitation Information from CONAGUA

(https://smn.conagua.gob.mx/tools/RESOURCES/Diarios/)

ACM Template

(https://www.acm.org/publications/proceedings-template**)**

Check out my work in github: