

Estimation of the Number of Claims Using Poisson and Quasi-Poisson Distribution

A Dissertation as a Course requirement for
Master of Sciences
Data Science and Computing

Bala Nagendra Babu Vinodula

Registration Number - 20226



SRI SATHYA SAI INSTITUTE OF HIGHER LEARNING
(Deemed to be University)

Department of Mathematics and Computer Science
Muddenahalli Campus
April 2022



SRI SATHYA SAI INSTITUTE OF HIGHER LEARNING

(Deemed to be University)

Department of Mathematics and Computer Science
Muddenahalli Campus

CERTIFICATE

This is to certify that this Dissertation titled **Estimation of Number of Claims using Poisson and QuasiPoisson Distribution** submitted by Bala Nagendra Babu Vinodula, Regd No.20226, Department of Mathematics and Computer Science, Muddenahalli Campus is a bonafide record of the original work done under our supervision as a Course requirement for the Degree of Master of Science Data Science and Computing.

Countersigned by

Dr Pallav Kumar Baruah
Supervisor

Dr Rita Gupta
Head of the Department


Sri Satya Sai Baba Mudigonda
Joint Supervisor

Place: Muddenahalli
Date: 25th April 2022

Vidyagiri, Prasanthi Nilayam - 515 134, Andhra Pradesh, India
Tel: +91 8555 287235 | hoddmacs@sssihl.edu.in | www.sssihl.edu.in

DECLARATION

The Dissertation titled **Estimation of Number of Claims using Poisson and QuasiPoisson Distribution** was carried out by me under the joint supervision of Sri. Pallav Kumar Baruah and Sri Satya Sai Baba Mudigonda, Department of Mathematics and Computer Science, Muddenahalli, as a Course requirement for the Master of Science Data Science and Computing degree and has not formed the basis for the award of any degree, diploma or any other such title by this or any other University.



Place: Muddenahalli
Date: 25th April 2022

Bala Nagendra Babu Vinodula
20226
II MSc Data Science and Computing
Muddenahalli

ACKNOWLEDGEMENTS

Firstly, I offer my gratitude to our beloved Swami for constantly guiding me and guarding me through His divine presence.

I would like to thank Sri Pallav K. Baruah Sir and Sri Satya Sai Mudigonda Sir for giving me a great opportunity to excel in my learning through this project.

I thank my mother and brother for the love and constant motivation and for creating a study ambience at home.

I thank Sri Phani Krishna Kandala, Sri Sai Aditya TVS, and Dr Rohan Yashraj Gupta for guiding the project, providing valuable inputs for the project's structuring. And also Sri Akash Saahu and Sri Nagateja Mariyala for resolving queries and knowledge support.

I would like to express my thanks to my sisters, Jyothi Lakshmi and Aishwarya, for their love and guidance.

I thank my friends and classmates M R Santhan and Krishnakanth for being with me and supporting me throughout the project period.

I thank Vigneshwaran S for providing valuable inputs and motivation.

I would like to express my gratitude to all my classmates for the support provided both academically and emotionally.

ABSTRACT

Actuarial science primarily deals with the insurance industry, which has a wide range of applications and plays a crucial role in helping firms manage risk. The effectiveness of insurance claim frequency analysis can help decide premium pricing, benefiting both the customer and the business with cheap premium rates and substantial profits. By identifying areas of weakness or prospective concerns that may be readily remedied, data science can also help enhance the efficiency of insurance claims processes. Simple instances can be prioritised for a rapid settlement, while more complex situations can be identified by claims assessors using data analytics for further inquiry.

The frequency severity model is one of the most common approaches to determining the price of insurance products. This study focuses on modelling the frequency of insurance claims, using a generalised linear model to quantify the frequency for the motor insurance line of business. By correctly predicting the number of claims, insurers can estimate the premiums accurately.

We will examine the findings of generalised linear models with Poisson and Quasi-Poisson distributions in this study to see how effective they are at predicting the claim frequency of the data set.

Key Words: Claim Frequency, Claim Severity, Generalised Linear Models, Poisson and Quasi-Poisson Regression

Table of Contents

ABSTRACT	5
Table of Contents	6
Project Repository	9
Chapter 1: Introduction	10
1.1 Actuarial Science	10
1.1.1 Insurance	10
1.1.2 General insurance	10
1.1.3 Frequency and Severity	10
1.1.4 Automobile insurance	11
1.1.5 Pricing in Automobile Insurance	11
1.1.6 Generalized Linear Models	11
1.2 Motivation	11
1.3 Challenges and opportunities	12
1.3.1 Business Perspective	12
1.3.2 Technology Perspective	12
1.4 Scope of the study	13
1.5 The uniqueness of the project	13
1.6 Objectives of the study	14
Chapter 2: Literature Review	15
2.1 Introduction	15
2.2 Literature review	15
2.2.1 Insurance terminology	15
2.2.2 Premium principle	16
2.3 Special Families of Distributions	17
2.3.1 Poisson Distribution	17
2.3.2 Quasi Poisson Distribution	17
2.4 Generalized linear models	18

2.5 Research Papers	19
2.5.1 modelling claim numbers when there are more zeros in the data	19
2.3.9 Claim frequency and claim size in non-life insurance using spatial modelling	20
2.6 Comparative study	20
Chapter 3: Methodology	22
3.1 Linear Regression	23
3.2 Exponential Family	23
3.3 Generalised Linear Models and Exponential Family functions	25
3.4 Poisson and Quasi-Poisson Distribution	25
3.5 Estimation of count variable as a Regression Problem	26
3.6 Estimation of count variable as a Classification Problem	27
3.7 Feature Selection	28
3.7.1 Recursive Feature Elimination	29
Chapter 4: The Dataset, EDA, Pyspark and Preprocessing	30
4.1 The Dataset	30
4.2 The Exploratory Data Analysis(EDA)	31
4.2.1 Speed Limit	32
4.2.2 Vehicles	32
4.2.3 Weather A and B	33
4.2.4 Regions	34
4.2.5 Light	35
4.2.6 Road Character	35
4.2.7 No of Lanes	36
4.2.8 Road Surface	37
4.3 Pyspark	38
4.4 Data Preprocessing	39
4.5 One Hot Encoding	40
Chapter 5: Regression Methods	42
5.1 Statistical Distributions	42

5.2 Response Variable	42
5.3 Interpretation of the GLM output	43
5.3.1 The Poisson Model	43
5.3.2 The Quasi Poisson Model	45
5.3.3 Prediction	47
5.4 Summary	49
5.5 Logistic Regression	49
5.6 Synthetic Minority Oversampling Technique(SMOTE)	50
Chapter 6: Machine Learning	54
6.1 Supervised Learning	54
6.2 Unsupervised Learning	55
6.3 Neural Networks	56
6.4 Activation Functions	58
6.5 Forward and Backpropagation	60
6.6 Model Explanation	60
Chapter 7: Future Work and Conclusion	63
7.1 Future Work	63
7.2 Conclusion	63
References	64
Appendix	65

Project Repository

The following links contain the project files:

1) The main repository:

<https://github.com/nagendranice/Estimation-of-number-of-claims>

2) The exploratory data analysis:

<https://github.com/nagendranice/Estimation-of-number-of-claims/blob/main/cas-eda.ipynb>

3) GLM modeling

<https://github.com/nagendranice/Estimation-of-number-of-claims/blob/main/cas-modeling.ipynb>

4) Logistic Regression and SMOTE

<https://github.com/nagendranice/Estimation-of-number-of-claims/blob/main/cas-modeling1.ipynb>

5) Neural Network model

<https://github.com/nagendranice/Estimation-of-number-of-claims/blob/main/cas-nn.ipynb>

Chapter 1: Introduction

1.1 Actuarial Science

In insurance, banking, and other businesses and professions, actuarial science is the discipline that uses mathematical and statistical methods to analyse risk. Actuaries, in general, use rigorous mathematics to model topics of ambiguity.

1.1.1 Insurance

Insurance is a legal contract between a policyholder and an insurer, where the policyholder is given financial protection if he incurs any loss. The policyholder pays a definite amount called premiums to the insurer.

1.1.2 General insurance

These insurance contracts cover the losses incurred from an event to the policyholder under the contract terms. It is also known as non-life insurance or property and casualty insurance. The types of general insurance available in India are:

- Automobile insurance
- Health insurance
- House insurance
- Travel insurance

1.1.3 Frequency and Severity

The total number of claims made by a policyholder to an insurance company during the term of the contract is referred to as the frequency. The severity of each claim is represented by its cost. If a claim is of high severity, the insurance company pays out a large sum, whereas claims of low severity result in moderate losses for the insurance company. This paper estimates claim frequencies for a policyholder using various models.

1.1.4 Automobile insurance

This is a yearly contract, and at the end of the year, the policyholder can either renew the contract or switch to another insurer. In this contract, the insurance company pays the policyholder if the car sustains damage due to an accident or theft. It also covers third-party damage, such as injury or death caused by the policyholder's driving. Our work focuses primarily on predicting claim frequencies in the automobile insurance industry.

1.1.5 Pricing in Automobile Insurance

The premiums charged to policyholders are determined by various factors. These are unrelated variables. The insurance company collects specific data from the policyholder and estimates the risk posed by the policyholder to determine the premium to be charged. The premiums are determined by models built with complex algorithms that calculate the frequency and severity of claims. If an insurer believes that a particular group of policyholders is less risky, they charge lower premiums and vice versa.

1.1.6 Generalized Linear Models

(GLMs) link the variables to be predicted to the independent variables, also known as explanatory variables, for which underwriting variables are used. GLMs are generalised linear regression models where the dependent variables can have an exponential distribution.

1.2 Motivation

Pricing engines created with the help of machine learning by the actuaries help businesses improve the loss ratios and the rate of profitability. Engines with built-in machine learning improve over time at assigning risk and claim frequency.

1.3 Challenges and opportunities

- The claim frequency is highly positively skewed (right-skewed), with a sharp peak and a long tail to the right.
- Claim frequency often has an excessive number of zero outcomes causing it difficult to model.

1.3.1 Business Perspective:

Every department in the organization's top priority is to manage the losses effectively and work toward a surge in profits. All the insurance companies make use of actuaries' expertise in this field as the models they develop would be used to forecast the number of claims and even the losses those claims may incur. Every model depends on a number of independent variables and a dependent variable which in our case is the number of claims. Some of the independent variables include types of risks and even the location and type of company or person who bought the policy with us and the number of claims filed. The actuaries use the past data of the organization to find a pattern or trend which would help the company boost its profits and reduce the losses.

1.3.2 Technology Perspective:

Predictive modelling in the insurance industry helps actuaries other than all other analysts improve their business operations. Before the advent of these techniques in the industry, dependence is mostly on the human experience rather than on the past numbers. The usage of modern technological tools in the company saves both time and human force resources. Predictive modelling has supplied insurance businesses with a set of tools for a range of purposes, ranging from pricing to underwriting and claim administration. Furthermore, the impact of predictive modelling is influenced by the quality of the data used to build the model.

1.4 Scope of the study:

The premium pricing decides whether a company runs into profits or incurs losses. High premiums will shoo away customers from buying whereas low premiums lead to less amount to cover the claims that will be filed. Furthermore, risk factors must be selected in such a way that it is legitimate to charge various groups of consumers varying prices based on them. Insurance pricing models will aid in the resolution of this issue, and frequency modelling is a component of insurance pricing. This study is limited to non-life insurance frequency modelling of insurance claims. The study will be able to reveal the perception of customers regarding non-life insurance. And this will help businesses understand the customers' demands better and work on them accordingly.

1.5 The uniqueness of the project:

This project mainly concentrates on the estimation of the number of claims i.e. claim frequency. Main work consists of the following,

- Creation of models to estimate number of claims.
- Applying data science in the actuarial science domain
- Scaling the created models towards Bigdata platforms.
- Estimation of SeriousInjuryCount with New Zealand motor crash analysis data using both Big Data and Machine learning techniques and developed packages from above.
- Development of Generalized linear models with various distribution that fits the data.
- Comparative study of different models created using various distributions such as Poisson and Quasi-Poisson.
- Usage of Artificial Neural Networks for creation of better models

1.6 Objectives of the study:

- Analyzing the insurance data by data visualizations.
- Recognizing relationships between the response and potential explanatory variables
- Using the Poisson and Quasi-Poisson Distributions, fit generalized linear models (GLM) to estimate the number of claims.
- Fine-tuning the models obtained from the above.
- Applying Logistic regression as a classification method
- Using [SMOTE\(Synthetic Minority Oversampling TEchnique\)](#)
- Comparing and analyzing the results from the different models.
- Looking at the performance of the Neural Network model
- Conclusions from the analysis.

Chapter 2: Literature Review

2.1 Introduction

In this chapter, The discussion is about the research papers and all other materials which referred as part of this study. The insurance and mathematical theory were obtained in part from Arthur Charpentier, Computational Actuarial Science. This book deals with computational aspects of insurance in the R programming environment “Introduction to Linear Regression Analysis by Douglas Montgomery, Elizabeth A. Peck, and G. Geoffrey Vining”[1] was the second most important source of mathematical theory. This provided a general understanding of regression analysis and the development of statistical models. It also served as a useful go-to resource when additional knowledge on specific topics was required to move forward with the project work.

2.2 Literature review

2.2.1 Insurance terminology

The following is a brief explanation of common terminology of insurance,

- Claim: A claim is when a customer to an insurance company has an accident or damage and they want to use their insurance. They report it to their insurance company, asking for reimbursement.
- Policyholder: A customer of an insurance company. It can be an individual or a company.
- Claim cost: Refers to the costs associated with a claim.
- Premium: The price that a customer pays for their insurance. Is usually paid on a yearly basis.

The business idea of an insurance company is to provide their customers with protection against financial risk in exchange for a fee, the premium. The risk is thus

transferred from the policyholder to the insurance company. By insuring a large number of customers, the insurance company's loss consists of a large sum of many small, roughly independent losses. As a result of the law of large numbers, the loss of an insurance company is much more predictable than the loss of an individual. Since an insurance company has a chance to predict their losses to some extent, they have the possibility of making a profit by charging premiums that cover their losses, other costs and give space for a certain return.

It has shown to be most advantageous for insurance companies to charge risk correct, or fair, prices. This means that the premium that each customer pays are dependent on their individual risk for the insurer. Simplified, you can say that a customer with higher expected claim cost should pay more for their policy premium than a customer with lower expected claim cost.

2.2.2 Premium principle

The expected value concept is a typical premium principle: the price of premium involved in annual risk. S is $E(S) = (1 + \theta) E(S)$, where S is the annual random loss and $\theta > 0$ signals some loading. Let (N_t) denote the number of claims that happened during the time range $[0; t]$. The amount of the i th claim is denoted by (Y_i) . The overall loss throughout the time $[0; t]$ is then

$$S_t = \sum_{i=1}^{N_t} Y_i \text{ with } S_t = 0 \text{ if } N_t = 0$$

If N and Y_i 's are independent and if Y_i 's are Identical Independent Distribution, then

$$\pi = E(S) = E(N) * E(Y)$$

“The pure premium is the product of two terms:

- Annual claims frequency $E(N)$,

- Claim severity $E(Y)$.”[2]

Claim Severity is the average loss associated with a single claim, i.e.

$$\text{Claim Severity} = \frac{\text{Total claim amount}}{\text{claim amount}}$$

2.3 Special Families of Distributions

- Poisson distributions – Easy to work when count variables are involved as response variable.
- Quasi Poisson distributions – Slight variation of Poisson distribution, this one deals with the overdispersion in the data.

2.3.1 Poisson Distribution

“The Poisson distribution is a continuous probability distribution with one parameter: lambda parameter which is the expected value of X. It is associated with the exponential distribution.

Poisson distribution has a parameter, named as λ where;

- λ = Average of the data parameter

It is characterized by mean and variance, $\mu = \lambda$ and $\sigma^2 = \lambda$ respectively. The corresponding probability density function of the distribution is

$$f(k; \lambda) = \Pr(X = k) = \frac{\lambda^k \cdot e^{-\lambda}}{k!}$$

, where

- k is the count of occurrences i.e. 1,2,3,4...
 - e represents the value 2.71828
 - And ! represents the factorial " [3]

2.3.2 Quasi Poisson Distribution

A Quasi Poisson distribution is a discrete distribution of random variable y whose natural algorithm is Poisson distribution. It is the generalization of the Poisson distribution where the dispersion is greater than 1, unlike in Poisson distribution where the dispersion parameter is 1. Dispersion is the ratio between the expected variance and the expected mean which is 1 in the Poisson distribution.

This distribution is used when the count variable of the data has overdispersion. Quasi Poisson distribution adjusts for this overdispersion in the regression models and gives better p-values and a better Adjusted R square value.

2.4 Generalized linear models

Generalized linear models (GLMs) link the variables to be predicted to the independent variables, also known as explanatory variables. GLMs are generalized linear regression models in which the dependent variables can have an exponential distribution.

The modelling differs from traditional regression modelling in two significant ways:

- The response distribution is chosen from the exponential family. As a result, the response distribution does not have to be normal or close to normal, and it can be explicitly nonnormal.
- The explanatory variables are linearly related to a transformation of the mean of the response.

GLMs are the extended version of the traditional linear model, and it is defined as below:

$$Y_i = \sum \beta_k x_{ik} + \epsilon_i \quad \forall i \in [1, n]$$

here n denotes the number of observations in the data.

A GLM is made up of three main components:

- Given the values of the explanatory variables, a random component gives the conditional distribution of the response variable Y_i (ith of n observations). Thus, the random component specifies the distribution of $E[y_i | \{x_1, \dots, x_k\}]$, in here k is the explanatory variables count that are present in the data. The Poisson distribution belongs to the “exponential family”.
- A *linear predictor* is given by

$$\eta_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \dots + \beta_k X_{ik}$$

- A linearizing link function that connects the random and systematic components in a smooth and invertible way. In other words, the link function converts the response variable's expected value, $\mu_i = E[y_i | x_1, \dots, x_k]$, to the linear component. As a result,

$$g(\mu_i) = \eta_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \dots + \beta_k X_{ik}$$

- The link function has the invertible property, so, we get $\mu_i = g^{-1}(\eta_i)$

Therefore, a GLM can be understood as either “a linear model of the transformation” of a dependent variable's predicted value or a nonlinear regression model for the dependent variable.

2.5 Research Papers

2.5.1 modelling claim numbers when there are more zeros in the data

“On some occasions, claim frequency data in general insurance may not follow the traditional Poisson distribution and in particular, they are zero-inflated. Extra dispersion appears as the number of observed zeros exceeding the number of expected zeros under the Poisson or even the negative binomial distribution assumptions. This paper presents several parametric zero-inflated count distributions, including the ZIP, ZINB, ZIGP and ZIDP, to accommodate the excess zeros for

insurance claim count data. Different count distributions in the second component are considered to allow flexibility to control the distribution shape. The generalized Pearson χ^2 statistic, Akaike's information criteria (AIC) and Bayesian information criteria (BIC) are used as goodness-of-fit and model selection measures. With the presence of extra zeros in a data set of automobile insurance claims, our result shows that the application of zero-inflated count data models and in particular the zero-inflated double Poisson regression model provides a good fit for the data.”[4]

2.3.9 Claim frequency and claim size in non-life insurance using spatial modelling

“Models for claim frequency and average claim size in non-life insurance are discussed in this work. The inclusion of both variables and geographic random effects allows for the modelling of a spatial dependency pattern. The quantity of claims is modelled using a Poisson distribution, while claim size is modelled using a Gamma distribution. In contrast to the traditional compound Poisson model, we allow for claim size and frequency dependencies. The parameters are computed using Markov Chain Monte Carlo in a completely Bayesian approach (MCMC). The topic of model comparison gets a lot of attention. We recommend using correct scoring procedures based on the posterior predictive distribution for comparing models in addition to the deviance information criterion and the predictive model choice criterion. We use machine learning to a large data set from a German vehicle insurance business. The incorporation of geographic variables enhances the models for both claim frequency and claim size, as well as the accuracy of total claim sizes forecasts. Furthermore, we find substantial correlations between the number of claims and the magnitude of the claims. From an actuarial standpoint, both spatial and the number of claims effects are evaluated and quantified.”[5]

2.6 Comparative study:

The use of Poisson and Quasi Poisson distributions for response variables can be demonstrated in our study. Both of these distributions are widely renowned for their ability to accurately predict the count response variable. To apply to real-world data, Quasi Poisson regression can be used because the dispersion parameter is usually greater than 1. Because the sample size is limited, both models perform well on smaller data sets. Apart from both the models, the logistic regression can also be applied considering the response variable as a multiclass categorical variable. This can be considered as a classification problem. The imbalance of data can be adjusted using the synthetic minority oversampling technique and the modeling can be done on that data. And the neural networks model applied on this data can be compared with its accuracy and the better root mean square error in the case of generalized linear models.

Chapter 3: Methodology

In this study, the primary focus is on the modelling claim frequency. The claim frequency is said to be the distribution of number of claims in the given period of time. The time period can range from quarter to one year. It is important to forecast the number of claims as the trends and patterns found can be used in business logic. Apart from the data preprocessing and the exploratory data analysis, for modelling purposes, two approaches have been made. Among which one deals with the regression type of modeling and the other deals with classification type.

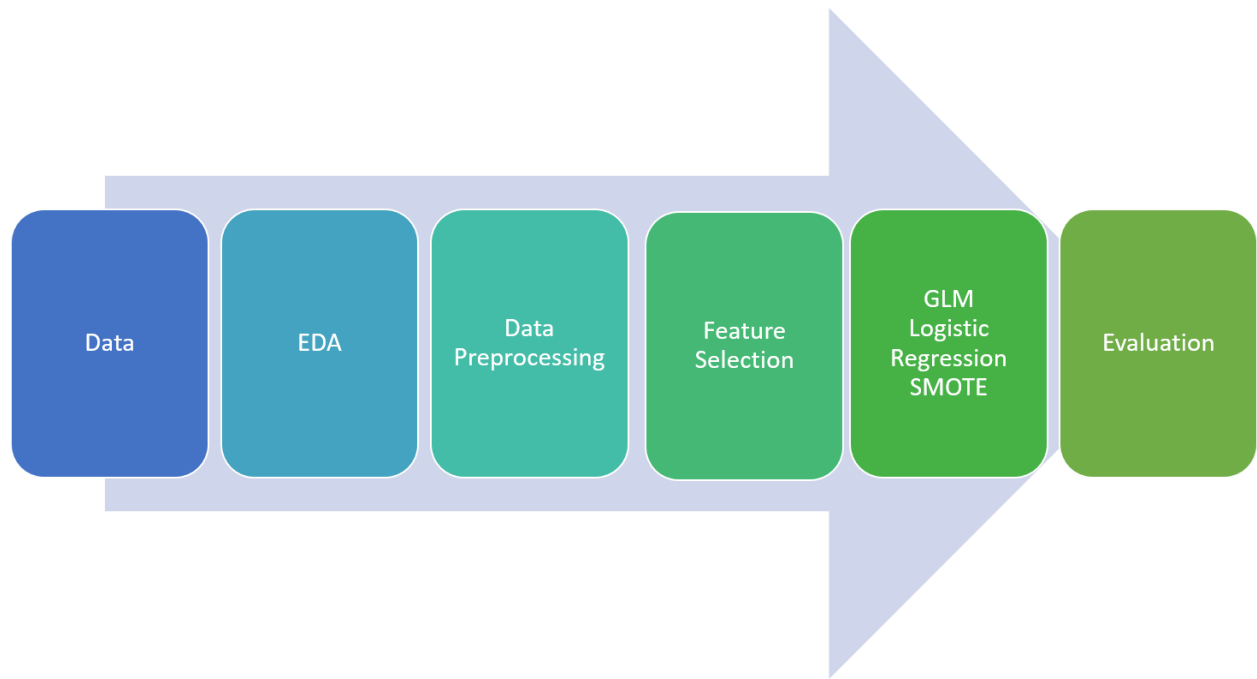


Figure 3.1: Schematic Diagram of the Study

Under the regression type modelling, the generalized linear models using the Poisson and Quasi Poisson Distributions are implemented. And under the classification type modelling, the Logistic Regression with multinomial class has been implemented. And in case of obtaining results in which a single class dominating the whole expected values can be adjusted using the Synthetic Minority Oversampling Technique.

As illustrated in the Figure 3.1, The data is collected and stored in a file. Usually it is in a csv file format. Once the data is ready, the exploratory data analysis can be performed which gives the visualised insights into the data that been dealing with. The next step would be data preprocessing which consists of handling of null values and missing data.

Next follows the feature selection, i.e. selecting of important features from all features in the data which will be forwarded to be used in modeling. And then the regression techniques such as generalized linear models and classifications techniques such as Logistic regression and incase of data imbalance, the SMOTE(synthetic minority oversampling technique) can be applied. Next comes the phase of training and testing followed by the evaluation phase.

Before entering into the main parts, the basics of linear regression followed by exponential family and its relevance here can be looked at in the next subsections.

3.1 Linear Regression

A linear model is where an x (input variable) and y (output variable) assumed to have a linear relationship. That y can be determined using a linear combination of the input variables is more detailed (x).

The procedure is known as “simple linear regression” when there is only one input variable (x). When there are several input variables, the procedure is referred to as multiple linear regression in the statistics literature.

To construct or train the linear regression equation using data, various strategies can be utilised, the most common of which is termed Ordinary Least Squares. Ordinary Least Squares Linear Regression, or simply Least Squares Regression, is a term used to describe a model created in this manner.

3.2 Exponential Family

In probability and statistics, an exponential family is a parametric set of probability distributions with a specified form, as defined below. This particular form was chosen

for its mathematical simplicity (it has numerous useful algebraic features) and generality (exponential families are a good set of distributions to study).

The term "exponential class" is sometimes used in place of "exponential family" or the older term Koopman–Darmois family.

The phrases "distribution" and "family" are often used interchangeably: A parametric family of distributions is referred to as "a distribution" (for example, "the normal distribution," which means "the family of normal distributions"), whereas the set of all distributions is referred to as "the set of all distributions."

The exponential family includes many of the most common distributions. The following are some examples of exponential families:

- Normal
- Chi-squared
- Bernoulli
- Poisson
- Gamma
- Exponential
- Quasi-Poisson

A single-parameter exponential family is a collection of probability distributions whose probability density function (or probability mass function, in the case of discrete distributions) can be written as:

$$f_X(x | \theta) = h(x) \exp[\eta(\theta) \cdot T(x) - A(\theta)]$$

$h(x)$ should be a non-negative function and all other functions are the known functions.

3.3 Generalised Linear Models and Exponential Family functions

The distribution functions employed in generalised linear models, which include many of the most regularly used regression models in statistics, are based on exponential families.

The standard linear regression model assumes that study variables have a normal distribution, whereas nonlinear logistic and Poisson regressions use Bernoulli and Poisson distributions, respectively. The research variable might follow several probability distributions, such as exponential, gamma, inverse normal, and so on, much like in logistic and Poisson regressions.

The exponential family of distributions is an example of such a distribution family. Based on this distribution, the generalised linear model unites linear and nonlinear regression models. It is assumed that the research variable's distribution belongs to the exponential family of distributions.

3.4 Poisson and Quasi-Poisson Distribution

Dispersion:

It is a general term in statistics to describe the spread of the data. For example, the variance and the standard deviation are the measures of dispersion.

Overdispersion:

It is the presence of greater variability in the data than what we would expect. This might be due to missing explanatory variables in our model.

Underdispersion:

It is the presence of smaller variability in the data than what we would expect.

Quasi-Poisson model:

This model will result in the same coefficients as the Poisson model, but with different standard errors and p-values since it adjusts for under - or overdispersion.

Note that the quasi-Poisson model cannot be fitted with the classical maximum likelihood method. The standard AIC value can therefore not be computed.

Assumptions of Poisson Distribution:

- K is the count of event occurrences i.e. 1,2,3...
- The events are assumed to be independent where the probability of one event does not affect the other event.[
- Occurrence of the event at a given rate is assumed to be constant in number.
- No two events can occur at the same time.[6]

When these conditions are satisfied, then k is a Poisson random variable, and the distribution of k is said to be a Poisson distribution.

3.5 Estimation of count variable as a Regression Problem

The normal linear regression is all about finding the linear relationship between the response variable and all other explanatory variables. In this study, the response variable that has been dealt with is a discrete variable. Therefore, It is only logical to use the Generalised Linear Models. That is the very reason, as to why the estimation of the number of claims is considered as a regression problem.

In economics and the social sciences, modelling count variables is a common problem. Because empirical count data sets frequently display over-dispersion and/or an excessive number of zeros, the conventional Poisson regression model for count data is often of little utility in these disciplines. The former problem can be solved by modifying the basic Poisson regression model in numerous ways, for as employing sandwich covariances or estimating a new dispersion parameter (in a so-called quasi-Poisson model)

- Poisson regression using the Generalised Linear models with log as the link function
- Quasi-Poisson regression with the adjusted over or underdispersion parameter to get better standard error values and thereby achieving the proper z-values and also the p-values.
- GLM can be done using the statsmodels package in the python for Poisson Distribution
- Comparison is done using the RMSE(Residual Mean Square Error)

3.6 Estimation of count variable as a Classification Problem

It is only prudent and logical to perform classification for a problem that contains the discrete values for its response variable. If we consider a column containing its values as natural numbers. And when we perform the regression and obtain values containing float values. It just doesn't seem right in the case.

So we go for a classification technique which gives a proper solution for our estimation of the count variable. Let's say the Logistic Regression.

Logistic modelling statistics is a statistical model that estimates the chance of one event occurring by making the event's log odds a linear combination of one or more independent variables. Logistic regression is a method of estimating the parameters of a logistic model in regression analysis. In binary logistic regression, there is a single binary dependent variable, coded by an indicator variable, with two values labelled "0" and "1," and the independent variables can be either binary variables or continuous variables. Regression analysis is a method for estimating the parameters of a logistic model. In binary logistic regression, there is a single binary dependent variable with two values labelled "0" and "1," coded by an indicator variable, and the independent variables might be binary variables or continuous variables. [7]

In our study,

We use Logistic Regression as a classification technique to classify the seriousInjuryCount we are dealing with where there are nearly 10 classes present in the response variable.

3.7 Feature Selection

After understanding the basics of linear regression and generalized linear models, the next part will be the feature selection. The real world datasets contain lots of columns i.e. attributes are more in number. Apart from the given columns, after onehot encoding, i.e. the technique where the categorical columns are transformed into the numerical columns, the features count increases automatically. While modeling, only the features which play vital role in creation of the model are to be considered. And for that task, From Supervised techniques, Recursive Feature Elimination technique powered by the Wrapper method i.e. drawing out the important features is being used here.

The main purpose of feature selection is to have fewer input variables for model prediction. The primary difference between dimensionality reduction and feature selection is that one reduces the whole input variables into small se. As a result, dimensionality reduction is a sort of feature selection rather than an alternative to it.

Feature selection:

Both Supervised and Unsupervised techniques can be used for feature selection from the data. The supervised technique uses the response variable and removes the redundant features in the data. Three types of supervised feature selection techniques are:

Intrinsic: feature selection happens during the training

Filter: Relationship with the response variable basis is used here

Wrapper: Looks for features which are performing well

RFE i.e. Recursive Feature Elimination comes under the wrapper technique. We make use of this technique in our data.

3.7.1 Recursive Feature Elimination

RFE, or Recursive Feature Elimination, is a well-known feature selection algorithm.

RFE is a popularly used technique owing to its easy to use installation and usage, and it is good at recognizing the best features of the train dataset which contribute more to the prediction of the response variable. because it's simple to set up and use, and it's good at identifying which features (columns) in a training dataset are more or more relevant in predicting the target variable.

When utilizing RFE, there are two crucial configuration options: the number of features to choose from and the algorithm used to help choose features. Both of these hyperparameters can be investigated, albeit their correct configuration does not have a significant impact on the method's performance.

RFE is an algorithm used for selecting features from a given dataset with a wrapper. This means that in the core of the technique, a different machine learning algorithm is given and used, which is wrapped by RFE and used to help choose features. Filter-based feature selections, on the other hand, score each feature and select the features with the highest (or lowest) score.

RFE is a wrapper-style feature selection technique that internally employs filter-based feature selection.

RFE works by searching for a subset of features in the training dataset, starting with all of them and successfully deleting them until the target number remains. This is accomplished by fitting the model's fundamental machine learning algorithm, sorting features by relevance, and deleting the least significant characteristics.

Out of 67 attributes in the dataset, Apart from the response variable, top 10 important features can be considered.

Chapter 4: The Dataset, EDA, Pyspark and Preprocessing

4.1 The Dataset

Crash Analysis System(CAS) Data

This data comes from the Waka Kotahi Crash Analysis System (CAS), which keeps track of all road accidents that the New Zealand Police report to us. CAS covers crashes on all New Zealand highways and sites where the general public has lawful access to drive a car.

The information is updated once a month, in the first week of the month.

Data is available beginning on January 1, 2000. Non-personal crash variables are included in the dataset.

See the charts in the 'Attributes' section below for a brief summary of the data. These will provide details on each of the dataset's attributes (variables).

Each chart is dedicated to a single variable and displays all data (without any filters applied).

Since the dataset contains 72 attributes, the description about each variable has been mentioned in the [Appendix](#) part of this report.

The CAS dataset contains 776878 rows and 72 columns among which there are 50 column whose datatype is float64 and 2 columns are of int64 type and the remaining 20 columns are of object type.

First few rows of the data are shown in the below figure:

	X	Y	OBJECTID	advisorySpeed	areaUnitID	bicycle	bridge	bus	carStationWagon	c
0	1772561.0	5896382.0	1	NaN	525420.0	0.0	NaN	0.0		1.0
1	1836757.0	5859311.0	3	NaN	534300.0	0.0	NaN	0.0		2.0
2	1762088.0	5912507.0	4	NaN	519500.0	0.0	NaN	0.0		1.0
3	1753522.0	5911939.0	6	NaN	518902.0	0.0	NaN	0.0		2.0
4	1761364.0	5914259.0	7	NaN	520202.0	0.0	NaN	0.0		2.0

Figure 4.1: First 5 rows of the dataset

4.2 The Exploratory Data Analysis(EDA)

EDA is the statistical way of assessing the datasets to analyse the important properties using various graphs, histograms, bar plots and scatterplots. This comes in handy when we need to explore the data to get a clear idea of what we are dealing with. This comes before the modelling and hypothesis testing.

In this the barplots, histograms, pie charts, heatmaps and many more can be done. Seaborn package and matplotlib pyplot package can be used in python to get these wonderful visualizations.

These visualisations help the business people to look at the picture and understand the trend or the pattern that's been recognized.

EDA gives the viewer a understanding of how each attribute is behaving when it is on its own and the behaviour of the variable when it is compared with the response variable that will be dealt with later in the study.

4.2.1 Speed Limit

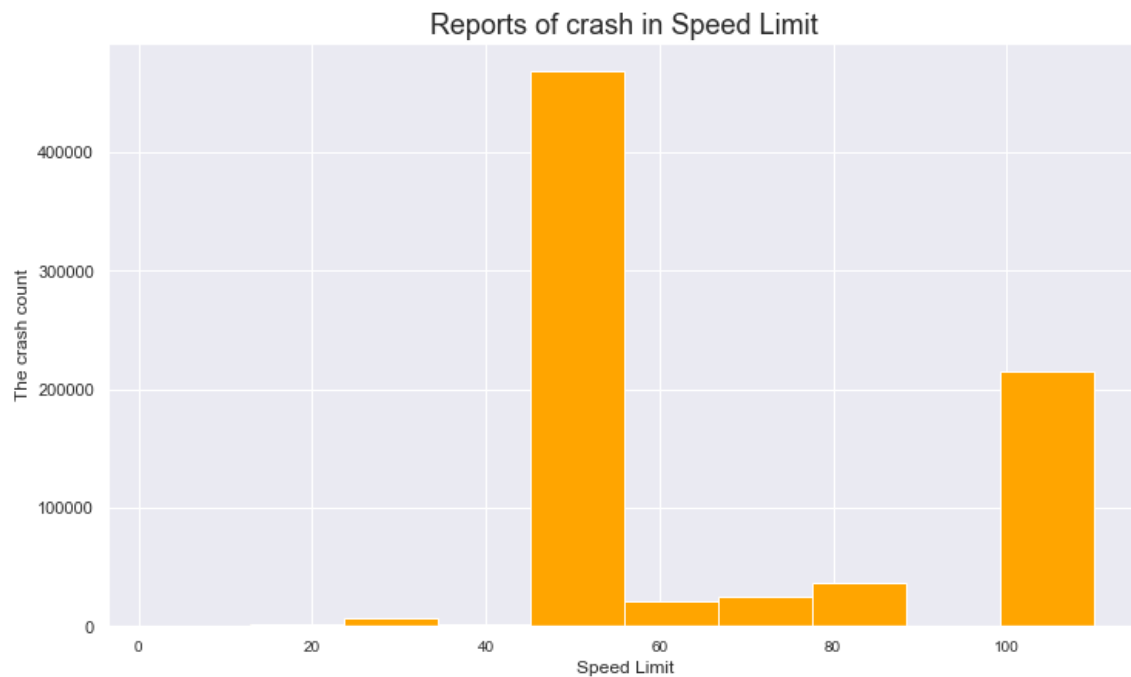


Figure 4.2: Speed Limit

As we can see in the above figure, More than half of the crashes have happened in the range of 45 kmph(Kilometers Per Hour) and 55 kmph. And nearly 2 lakh crashes are in the range of 100 kmph and above.

4.2.2 Vehicles

The below figure shows various vehicles present during the crash at the site where the mishap happened. There is more number of carStationWagon and the number is nearly 3 lakh whereas other vehicles which are more in number are SUVs and they are nearly less than 100 thousand. All other taxis, motorcycles, van or utility are significant in number.



Figure 4.3: All vehicles

4.2.3 Weather A and B

Weather A is with respect to the raining conditions during the crash and in most of the crash cases the weather is fine and around 1 lakh crashes happened when there was light rain. Weather B is with respect to the weather where strong winds and mist are involved. In our case most of the data with respect to Weather B is missing which is evident in the plot where almost all the values are null.

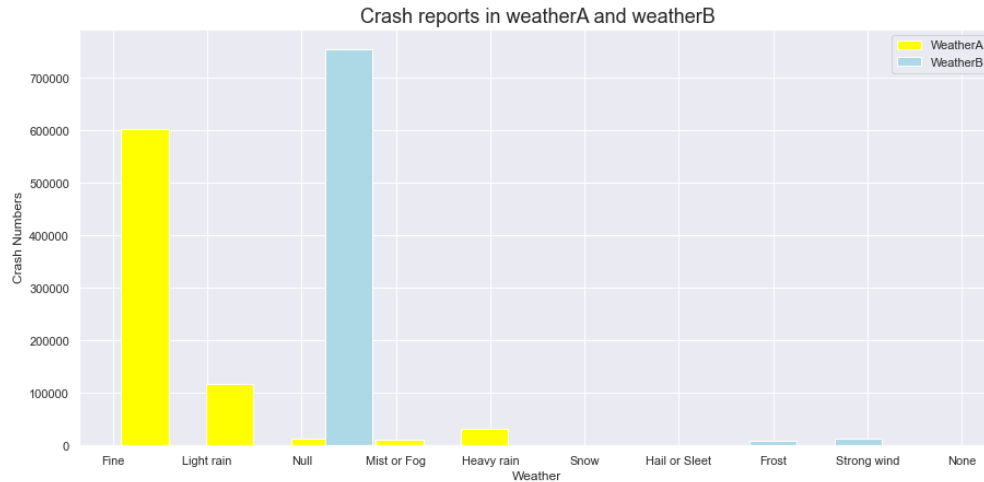


Figure 4.4: Weather A and Weather B

There are more crashes in the Auckland Region compared to any other region and the next region where more crashes happened is in Otago Region. And even the Waikato and Canterbury regions have a significant amount of crashes.

4.2.4 Regions

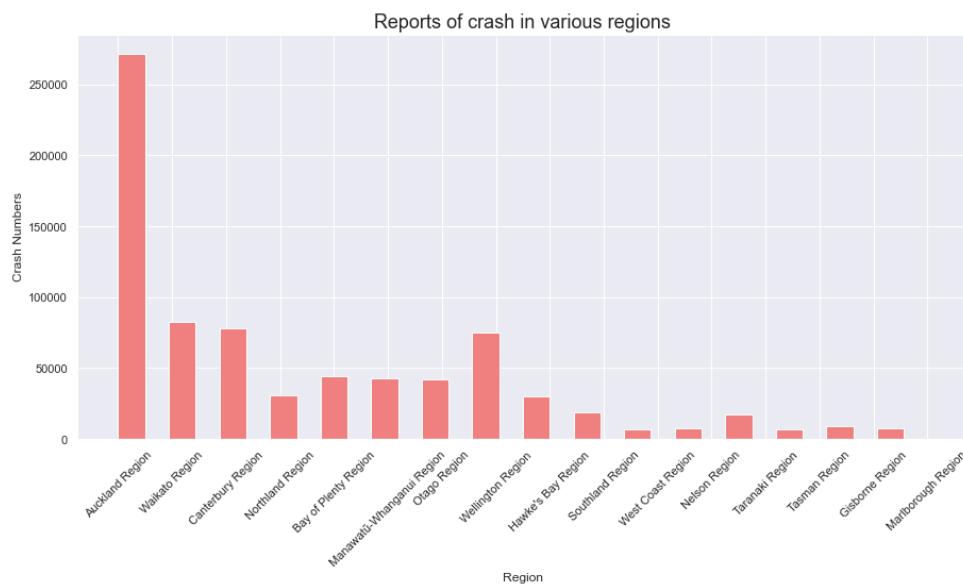


Figure 4.5 Region wise crashes

4.2.5 Light

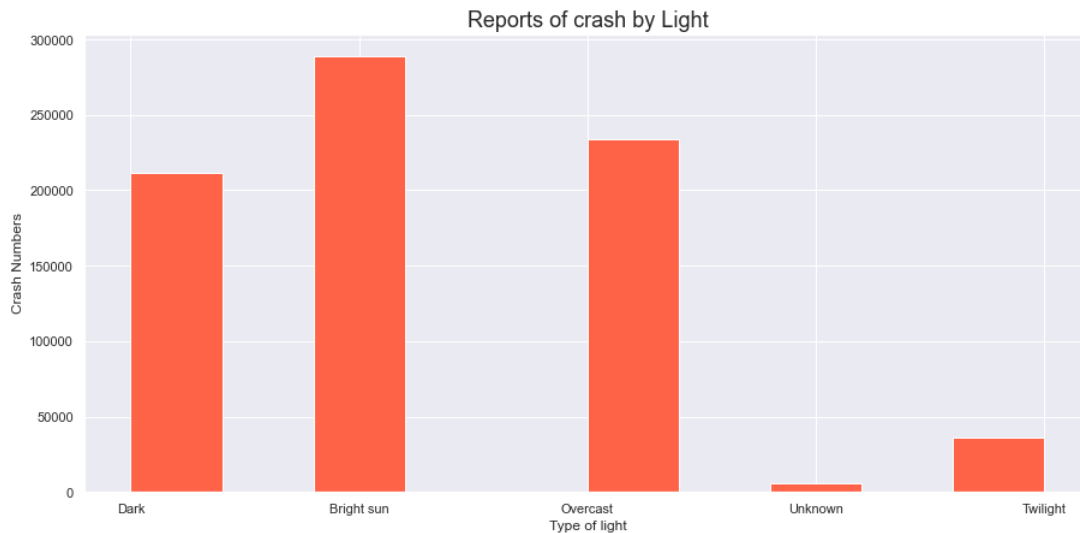


Fig 4.6: Reports of Crash by Light

As shown in the above figure, More crashes have happened during the broad daylight and the next more number of crashes during the Overcast and also when the light is very low, that is During night time.

4.2.6 Road Character

From the above figure, we can see that nearly 15000 crashes have happened in the Bridge area and around 10000 in the Motorway ramp.

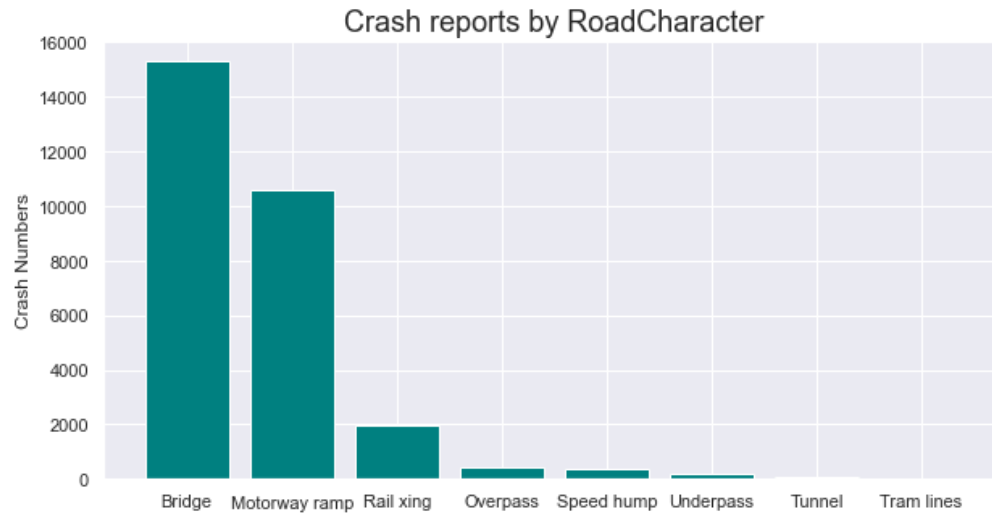


Fig 4.7: Road Character

4.2.7 No of Lanes

More number of crashes i.e. nearly five and a half lakh crashes happened in the Road Lanes where there are only 2 lanes present.

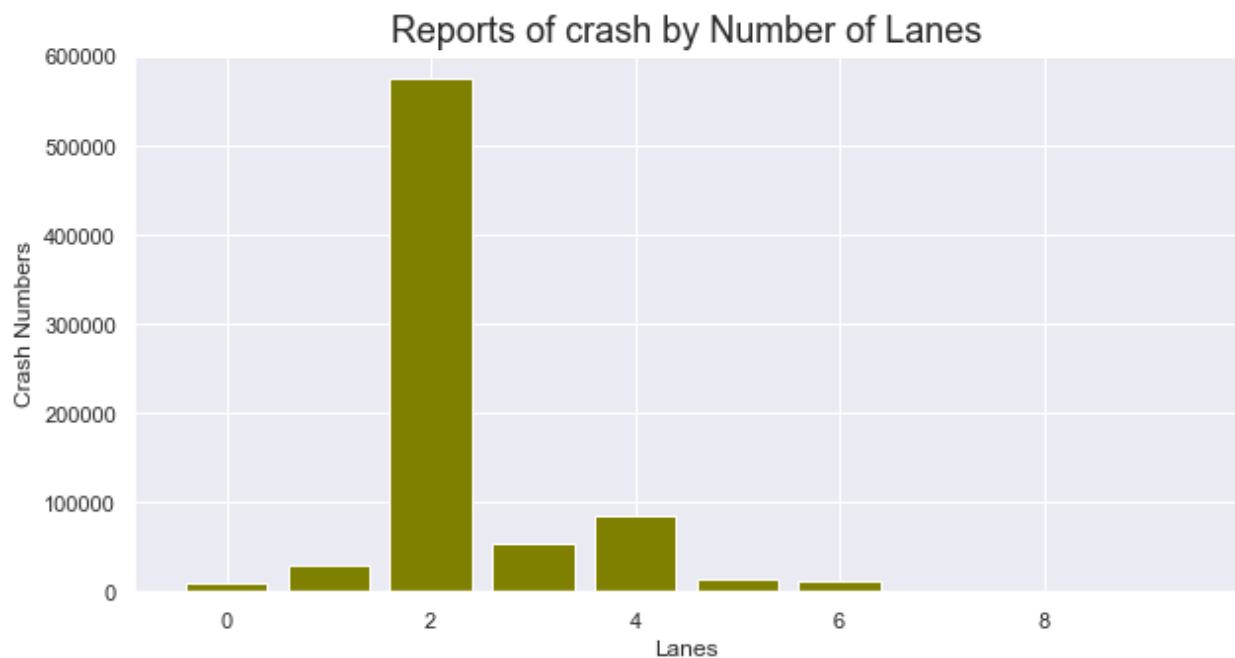


Figure 4.8: Number of Lanes

4.2.8 Road Surface

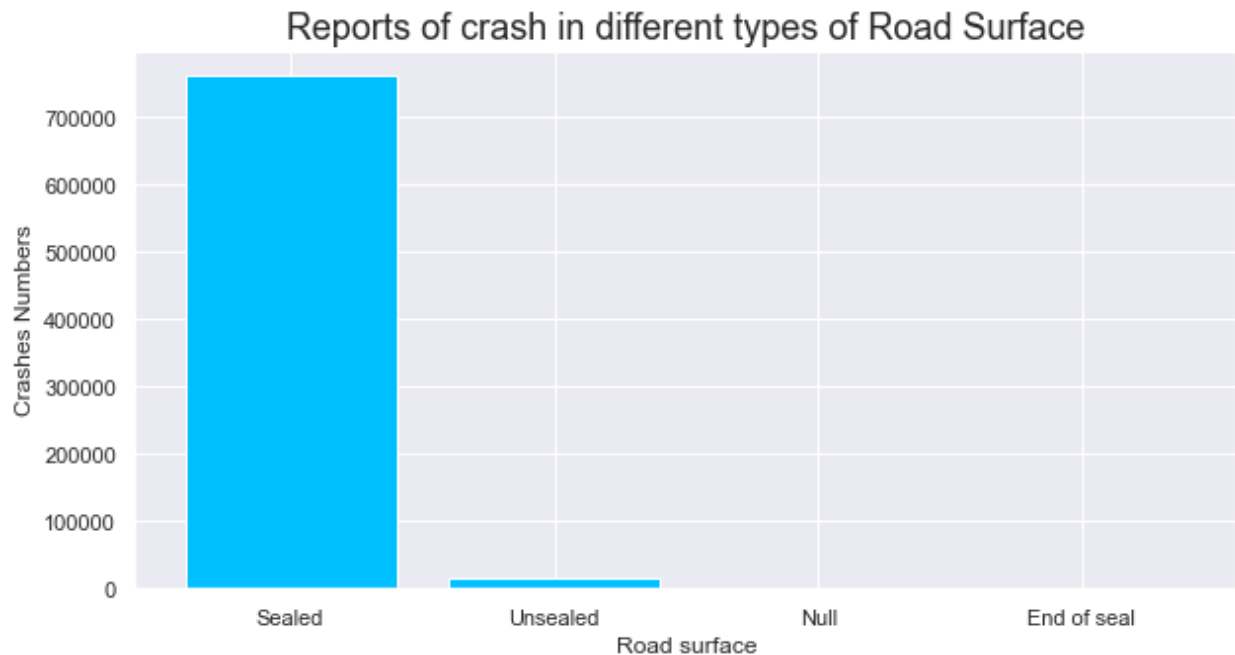


Fig 4.9: Road Surface

Sealed roads are those which have one of the pavement treatments i.e. built using either tar or cement. And unsealed roads are the ones which have no pavement area and a road which does not have a hard, smooth surface of tarmac, concrete, etc.

And these are the important attributes whose visualization helps in the modeling part as to which are the most important attributes of all. The other variables too can be visualized, The important features have been included in the report here.

The next part that's been dealt with is about the preprocessing that needs to be done before entering into the phase of modeling.

4.3 Pyspark

“Pyspark has a lot of potential when it comes to handling resources efficiently. Here the data processing is done simultaneously across multiple nodes in the clusters. In here, for preprocessing, pyspark has been used for its speed and efficiency.

Apache Spark is based on the Scala. Pyspark supports the integration of python API in the spark. Spark helps in interacting with RDDs using Scala programming and it works as an interface between both. Py4j comes in handy here.



Figure 4.10: Pyspark and python

Py4J is a famous library which is integrated within PySpark and allows python to communicate with JVM objects from time to time. PySpark consists of a few libraries for writing efficient programs. And there are several external libraries that also support the pyspark: Some libraries include:

- PySparkSQL
- MLlib
- GraphFrames

Pyspark aids parallel processing by allowing huge datasets to be processed in clusters and nodes at the same time. Workload distribution leads to more efficient output generation, which saves time and reduces computation energy usage.

RDDs are the building blocks of a spark application, and Pyspark makes use of them. Resilient Distributed Datasets (RDD) is an acronym for Resilient Distributed

Datasets. It indicates that data is distributed across numerous nodes in a cluster and that it is fault resilient.”[11]

4.4 Data Preprocessing

Before we move on to modeling, we must first wrangle our data. We'll remove a few columns that have a lot of missing data or whose value is the same across all rows because they don't contribute much to our model.

Filling missing or null numbers using a mean or average is one option for dealing with them. Zeros can also be used to fill in for missing values or null values in some instances.

In our dataset, we did the following steps in pyspark:

1. Replacing Nulls with NaNs

Replacing Nulls with NaN

```
df.replace("Null", np.NaN, inplace=True)  
df.head()
```

Figure 4.11: NaN replacement

2. Removing columns with more than 90 percent null values

```
df.drop(["OBJECTID", "advisorySpeed", "crashRoadSideRoad", "tlaId", "meshblockId", "intersection", "areaUnitID", "temporarySpeedLimit",  
df.drop_duplicates(inplace=True)
```

Figure 4.12: Removing columns

3. Null values

advisorySpeed	96.229910
areaUnitID	0.014932
bicycle	0.000644
bridge	60.034652
bus	0.000644
carStationWagon	0.000644
cliffBank	60.034652
crashLocation2	0.118551
crashRoadSideRoad	100.000000
crashSHDescription	0.000129
debris	60.034652
directionRoleDescription	0.010169
ditch	60.034652
fatalCount	0.017506
fence	60.034652
guardRail	60.034652
...	...

Figure 4.13: Percentage of Nulls

4. Handling missing values

```
df['minorInjuryCount'].fillna(0,inplace=True)
df['fatalCount'].fillna(0,inplace=True)
df['bicycle'].fillna(0,inplace=True)
df['bus'].fillna(0,inplace=True)
df['carStationWagon'].fillna(0,inplace=True)
```

Figure 4.14 : Handling of missing values

4.5 One Hot Encoding

One hot encoding is a process of converting the categorical columns into numerical columns before performing regression on them or before the data is used for modeling purposes.

In this study, there are nearly 72 columns in the dataset before preprocessing is done. Among which there are nearly 20 variables which are not numerical. And out of 72 columns, in the data preprocessing, about 11 columns have been removed as there were more than 90 percent null values in the respective columns.

After performing one hot encoding, it can be done using the `dmatrix` library in python before passing it to the GLM, the number of columns will be increased naturally. The number of columns or attributes are at 90 after the one hot encoding, where it converts region, flathill, light, road character and road surface to numerical columns thereby increasing the count of columns.

Chapter 5: Regression Methods

5.1 Statistical Distributions

In our analysis, we are using two statistical distributions which are apt for modeling the count variable `seriousInjuryCount` in our dataset. The two distributions are:

1. Poisson Distribution
2. Quasi-Poisson Distribution

Dispersion :

If our data shows overdispersion, the use of a Poisson model will underestimate the standard errors which will result in too low p-values with an increased risk for a type I error.

Quasi-Poisson:

The Quasi-Poisson model will result in the same coefficients as the Poisson model, but with different standard errors and p-values since it adjusts for under or overdispersion

5.2 Response Variable

The dependent variable used for modeling from the CAS dataset is regarding the non-life insurance part. The number of claims estimation is to be done. In the dataset, the variable with the name '`seriousInjuryCount`' is to be considered as claim number, As the other variables '`minorCount`', '`fatalCount`', and '`nonInjuryCount`' are less or no claims will be reported, the amount that has to be paid is fixed and no claim will be generated out of this, respectively.

Therefore, the variable '`seriousInjuryCount`' is considered as the response variable in this study.

5.3 Interpretation of the GLM output

Obtaining output and Interpreting it are two different tasks, where the former deals with getting the desired output whereas the latter deals with what exactly the output is trying to convey about each of the values obtained.

The important features have been selected using the [RFE](#) method. And it's output is shown in the below figure.

```
Feature[Feature.Decision==True].Feature
4          light[T.Overcast]
8    region[T.Canterbury Region]
31         roadLane[T.2-way]
37         streetLight[T.Off]
45    trafficControl[T.Unknown]
53          bicycle
56    carStationWagon
66          motorcycle
67    NumberOfLanes
77          speedLimit
84           tree
Name: Feature, dtype: object
```

Figure 5.1(a): Feature selection using RFE

5.3.1 The Poisson Model

The following is the interpretation of each part of the output:

Coefficients:

"The coefficient estimate indicates the average change in the log odds of the response variable associated with a one-unit increase in each predictor variable." [12]

In our model, coefficients are all about how the dependent variable varies with respect to each of the independent variables. Here the features: motorcycle, Light Overcast, Roadlane 2-way have more effect on the dependent variable seriousInjuryCount compared to all other independent variables.

Generalized Linear Model Regression Results						
Dep. Variable:	seriousInjuryCount	No. Observations:	620746			
Model:	GLM	Df Residuals:	620735			
Model Family:	Poisson	Df Model:	10			
Link Function:	log	Scale:	1.0000			
Method:	IRLS	Log-Likelihood:	-1.5826e+05			
Date:	Tue, 19 Apr 2022	Deviance:	2.3989e+05			
Time:	12:19:28	Pearson chi2:	1.19e+06			
No. Iterations:	7					
Covariance Type:	nonrobust					
	coef	std err	z	P> z	[0.025	0.975]
light[T.Overcast]	-0.2765	0.011	-25.651	0.000	-0.298	-0.255
region[T.Canterbury Region]	0.0332	0.015	2.276	0.023	0.005	0.062
roadLane[T.2-way]	-0.4753	0.013	-37.324	0.000	-0.500	-0.450
streetLight[T.Off]	-0.5908	0.014	-43.661	0.000	-0.617	-0.564
trafficControl[T.Unknown]	-0.0729	0.010	-6.955	0.000	-0.093	-0.052
bicycle	0.5614	0.017	32.302	0.000	0.527	0.595
carStationWagon	-0.6926	0.006	-110.271	0.000	-0.705	-0.680
motorcycle	0.8069	0.011	75.481	0.000	0.786	0.828
NumberOfLanes	-0.5099	0.007	-77.369	0.000	-0.523	-0.497
speedLimit	-0.0033	0.000	-20.192	0.000	-0.004	-0.003
tree	0.3755	0.018	21.204	0.000	0.341	0.410

Figure 5.1(b): Poisson Model

Standard Error:

“The standard error gives us an idea of the variability associated with the coefficient estimate. We then divide the coefficient estimate by the standard error to obtain a z value.”[12]

In our model, we have all the standard error values to be around 0.01 and it is not very large which implies that the coefficient estimate is near to precision.

P-values:

“The p-value $\Pr(>|z|)$ tells us the probability associated with a particular z value. This essentially tells us how well each predictor variable is able to predict the value of the response variable in the model.”[12]

We can determine the significance level based on our own preferences be it 1%, 5% or 10%.

As a 0.05 level of significance is common across models implemented using the Generalized linear models, we have all the p-values less than this threshold of 0.05, we

take it that the independent variables used in this model are quite significant and are contributing well to the regression model.

Null and Residual Deviance:

“The null deviance in the output tells us how well the response variable can be predicted by a model with only an intercept term.

The residual deviance tells us how well the response variable can be predicted by the specific model that we fit with p predictor variables. The lower the value, the better the model is able to predict the value of the response variable.”[12]

Dispersion Parameter:

As the difference between estimated variance and estimated mean is considered as the dispersion, in this case, we consider dispersion to be 1 as the mean and the variance of the Poisson model assumes to be equal. Therefore, it makes the dispersion value to be 1.

5.3.2 The Quasi Poisson Model

And now we look at the Quasi Poisson model, Which will have similar coefficients but will have different standard errors and also better p-values. Here we take into consideration the dispersion parameter, as it is what makes the values differ from the Poisson model.

Since all the p-values are zero or less than 0.05, we consider all the variables to be significant for our model. And the dispersion value here is 0.359, as it is less than the normal value of 1, This is a case of underdispersion.

In the Poisson model, the model overlooks this dispersion value, as it considers it to be 1 irrespective of the original value since it upholds the assumption that the mean and variance are equal. Thereby the chances of type I error are maximum.

QuasiPoisson GLM Model Summary.		
Name	Parameter Estimate	Standard Error
Intercept	-3.82	0.02
regionC	0.18	0.01
roadLane2	0.61	0.01
streetLight0	-0.27	0.01
trafficControlU	0.05	0.01
bicycle	0.97	0.01
carStationWagon	-0.28	0.00
motorcycle	0.98	0.01
NumberOfLanes	-0.12	0.00
speedLimit	0.02	0.00
tree	0.50	0.01

Figure 5.2: The Quasi Poisson Model

As we can observe, the Coefficients of both the models' outputs are very similar as we had discussed earlier. The varying points are only the standard errors and p-values.

```
qmodel.dispersion_
array(0.35853054)

qmodel.p_values_
array([0.00000000e+000, 2.24841314e-092, 0.00000000e+000, 2.28301346e-230,
       5.10059348e-012, 0.00000000e+000, 0.00000000e+000, 0.00000000e+000,
       2.22435976e-211, 0.00000000e+000, 0.00000000e+000])
```

Figure 5.3: The dispersion value and p-values

5.3.3 Prediction

Once the model is obtained, the next part is to look at the predictions using the test data. As the model is already trained, we also compare the actual values and the predicted values.

	Actual	Predicted	Residuals
549049	0.0	0.046926	-0.046926
444549	1.0	0.069334	0.930666
518123	0.0	0.053422	-0.053422
437517	0.0	0.069366	-0.069366
750131	0.0	0.024980	-0.024980
772252	0.0	0.560544	-0.560544
440714	3.0	0.254505	2.745495
25970	0.0	0.031563	-0.031563
313461	0.0	0.034000	-0.034000
531636	0.0	0.028879	-0.028879

Figure 5.4 Actual vs Predicted

Randomly, 10 rows are selected and these are the actual and predicted values of the Poisson model obtained before. Here, Predicted values are looking to be near zero in almost all the cases. So, now I will look at the residual plot for more close look at how residuals are distributed.

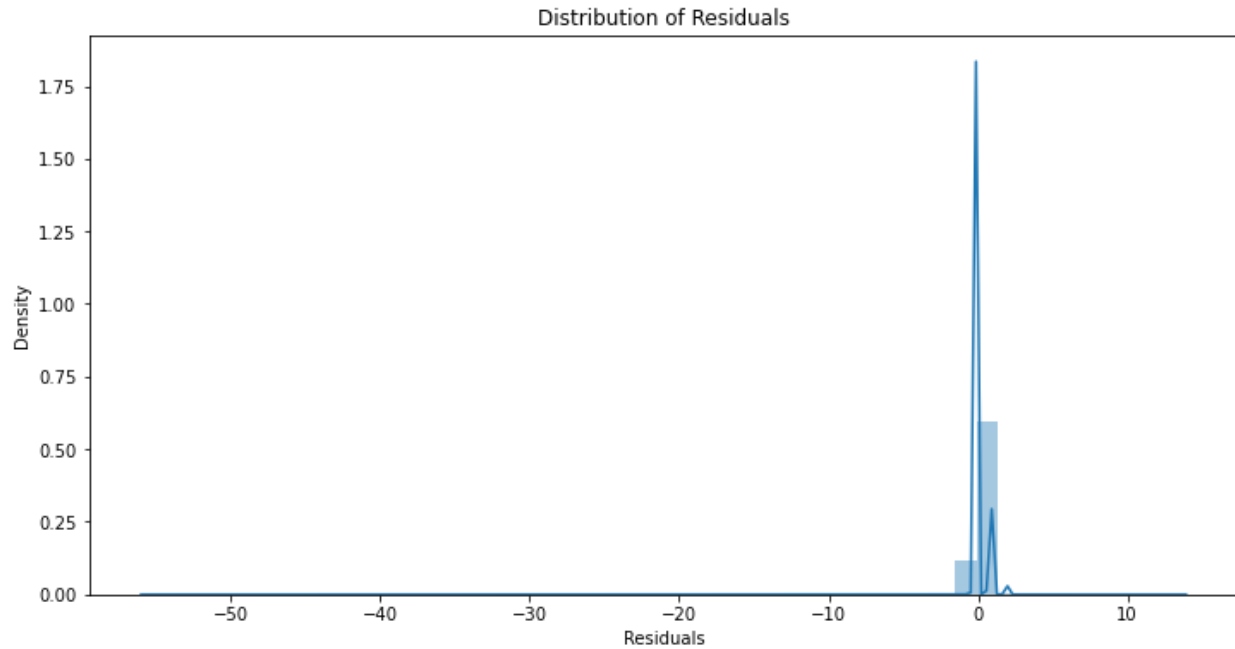


Figure 5.5: Residuals plot

AIC is the Akaike Information Criterion value. It is used to look at the good model among the models created. It uses the formula:

$$AIC = 2k - 2\ln(\hat{L})$$

Here, k is the estimated number of parameters and (\hat{L}) is the maximum likelihood function of the model.

Among many models implemented these two models are by far the best ones compared to any other models, as the AIC value for these two models is less compared to the others.

5.4 Summary

Between the two models i.e. Poisson model and the Quasi-Poisson model, even though there is a close similarity between the distributions, Both the models are performing well with respect to their RMSE.

```
from sklearn.metrics import mean_squared_error
ypred = model_poisson.predict(X_train_poi)
rmse = np.sqrt(mean_squared_error(y_train_poi,ypred))
rmse
0.3146239568440867
```

Figure 5.6: RMSE

RMSE is the Root Mean Square Error which shows the difference between actual values and the predicted values. RMSE in the range between 0.2 and 0.3 is said to a better model.

5.5 Logistic Regression

For logistic regression, the data has been divided into train and test using the sklearn method `train_test_split`. Then using the Logistic Regression method, model has been created. The next step is to fit the model. Then follows the prediction.

And this is the accuracy of the model:

```
print('Accuracy of logistic regression classifier on test set: {:.2f}'.format(model.score(X_test, y_test)))
```

Accuracy of logistic regression classifier on test set: 0.94

Figure 5.7: Accuracy

It is important to look at all the metrics before concluding something about the model, so in here we look at the classification report that can be generated using the `classification_report` method of `sklearn`.

	precision	recall	f1-score	support
0.0	0.94	1.00	0.97	218976
1.0	0.39	0.01	0.03	12140
2.0	0.00	0.00	0.00	1282
3.0	0.00	0.00	0.00	292
4.0	0.00	0.00	0.00	76
5.0	0.00	0.00	0.00	32
accuracy			0.94	232798
macro avg	0.22	0.17	0.17	232798
weighted avg	0.91	0.94	0.91	232798

Figure 5.8: Classification Report

From the fig 6.5, It can be observed that classes other than 0 are less in number.

5.6 Synthetic Minority Oversampling Technique(SMOTE)

Imbalanced classification has the drawback of having too less instances of the class whose occurrences are very less compared to other classes for a model to train and learn the threshold of the values.

Oversampling the examples in the minority class is one technique to tackle this problem. This can be accomplished by simply duplicating minority class samples in the training dataset before fitting a model. This can help to balance the class distribution, but it doesn't give the model any extra information.

Synthesizing new instances from the minority class is an improvement over replicating examples from the minority class. This is a sort of data augmentation that works well with tabular data.

SMOTE functions by choosing the instances in the attribute space that are near together, marking a threshold in the feature space between the examples, and marking a

new sample at a location along the marked line.

To be more precise, a case from the minority class is selected at random first. Then k of the nearest neighbours (typically $k=5$) are found for that example. An arbitrary neighbour is chosen, and a synthetic example is created.

In our study, we do have imbalanced data, as there are more zero value counts in the `seriousInjuryCount` variable.

Before SMOTE	After SMOTE
<pre>0.0 730095 1.0 40297 2.0 4349 3.0 865 4.0 259 5.0 126 Name: seriousInjuryCount, dtype: int64</pre>	<pre>seriousInjuryCount 0.0 511119 1.0 511119 2.0 511119 3.0 511119 4.0 511119 5.0 511119 dtype: int64</pre>

Figure 5.9: seriousInjuryCount values before and after SMOTE

From Fig. 4.7, We can see that, there is a significant increase in the minority classes values after applying the SMOTE, now all the classes look balanced as we can see in the histograms below.

The more data in obtained using the SMOTE can be helpful in the training of the model as there are biased amount of minority classes in the data. This helps in training of the data in a better way.

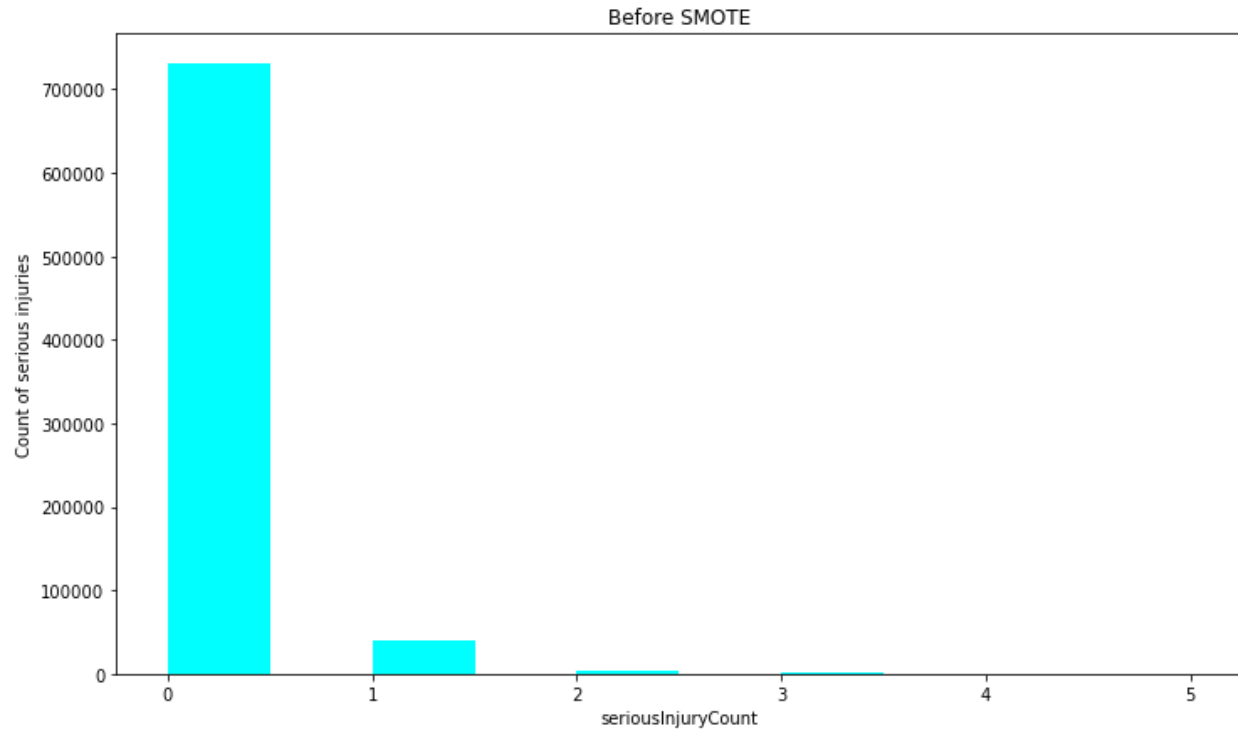


Figure 5.10: Histogram of seriousInjuryCount Before SMOTE

From fig.4.8, We can see that nearly 90 per cent above values have value count as zero in the response variable we are considering here.

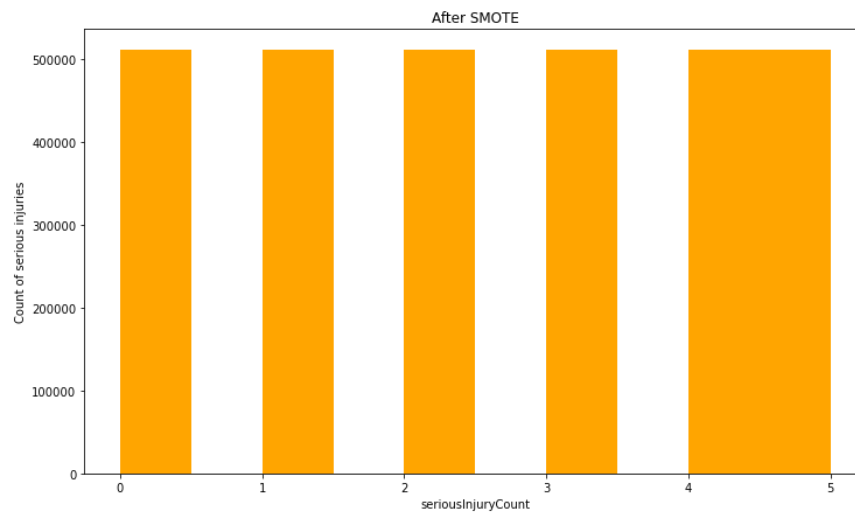


Figure 5.11: Histogram of seriousInjuryCount After SMOTE

From Fig 4.9, It can be clearly understood the performance of the SMOTE algorithm on the imbalanced data where the response variable contains more number of one class counts compared to others.

This data can be used for modeling and results are better compared to the modeling that has been done before applying this technique on the data.

Chapter 6: Machine Learning

Apart from the two techniques that have been discussed in the previous chapter, in this chapter, the use of Artificial Neural Networks can be observed. To begin with, the machine learning techniques are broadly divided into two, namely, Supervised and Unsupervised learning. The study of various automated systems that may learn and evolve on their own given experience and data is known as machine learning (ML). It's viewed as a form of artificial intelligence. Machine learning algorithms create a model based on training data and use it to make predictions or judgments without having to be explicitly programmed. Machine learning algorithms are used in a variety of applications, such as biomedicine, spam filtering, and voice recognition when traditional algorithms are difficult or impossible to build.

Machine learning, on the other hand, is not all statistical learning. Algorithmic statistics, which aims at making predictions using computers, is a subset of machine learning. Machine learning benefits from mathematical optimization research because it provides techniques, theory, and application fields. Data mining is a related field of study that focuses on exploratory data analysis using unsupervised learning. Some machine learning methods use data and neural networks to resemble the operation of a biological brain. When used to come up with solutions, machine learning is also known as predictive analytics.

6.1 Supervised Learning

Supervised learning algorithms build a statistical equation of a set of data that includes both the desired inputs and outputs. The data is known as training examples, and it is composed of a number of training occurrences.

Each training sample contains one or even more inputs and the desired output is a supervisory signal.

In the mathematical model, each training sample is expressed by an array or vector, sometimes referred to as a feature vector, and the training set is represented by a matrix. By repeatedly optimizing the process parameters, supervised learning

approaches build a function that may be used to anticipate the output associated with new inputs.

If the technique employs an optimal function, it will be able to accurately estimate the output for inputs that were not part of the training data. Over time, an algorithm that learns to perform the task improves the accuracy of its outputs or predictions.

Types of Supervised Learning:

Linear Regression, Decision Trees, Support Vector machines, K nearest neighbours and Naive Bayes

6.2 Unsupervised Learning

Unsupervised learning is a type of algorithm that learns patterns from unlabelled data. The idea is to force the system to construct a compact internal picture of its environment through emulation, which is a popular tactic for people to learn, and then generate inventive content from it.

Contrary to supervised learning, in which data is labelled by experts, such as "cat" or "mat," unsupervised methods show self-organization that captures patterns as probability densities or a blend of neural characteristic choices.

The other levels of the supervision spectrum are reinforcement learning, wherein the machine is given only a quantitative performance measure as direction, and semi-supervised learning, in which only a tiny portion of the data is labelled.

Unsupervised learning, also known as unsupervised machine learning, uses machine learning techniques to evaluate and cluster unstructured data. These algorithms identify interesting insights or data categories without the need for human intervention. Because of its efforts to identify similarities and contrasts in information, it is the greatest choice for exploratory data analysis, cross-selling methods, customer segmentation, and image classification. [8]

Three main types of Unsupervised Learning are:

- Clustering
- Association
- Dimensionality Reduction

6.3 Neural Networks

“Artificial neural networks (ANNs) are machine learning modules that are at the core of deep learning techniques. Their name and construction were inspired by the human brain, as they resemble how biological neurons interact with one another.

A node layer consists of an input layer, one or more hidden layers, and an output layer in artificial neural networks (ANNs). Each node, or artificial neuron, is connected to the next and has a weight and threshold associated with it. If a node's output exceeds a certain threshold, the node is activated, and data is sent to the network's next tier.”[9]

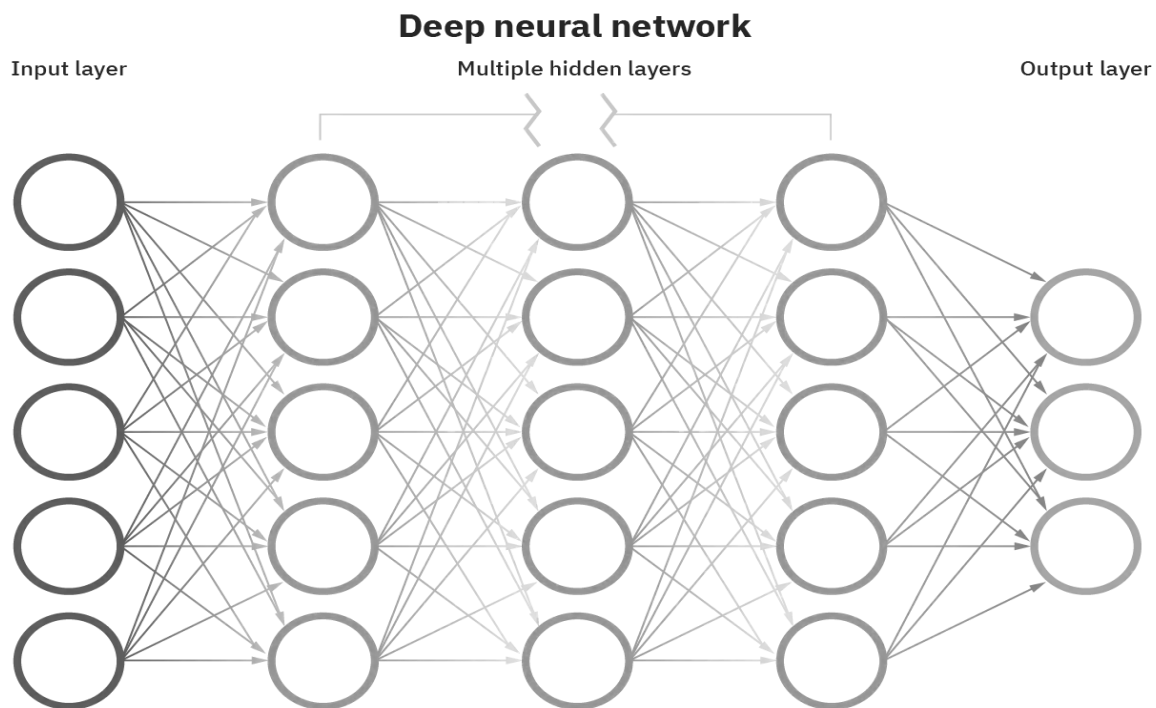


Figure 6.1: Neural Network

Neural networks use training sets to learn and improve their precision over time. These machine learning, once fine-tuned for accuracy, become powerful technologies in computer science and artificial intelligence, allowing us to efficiently categorize and

cluster data. Voice recognition and picture recognition activities can take mins rather than hours when compared to manual identification by human operators. One of the most well-known neural networks is Google's search algorithm.

And in our study, we implement a Network Model.

The Model has

Model: "sequential_1"

Layer (type)	Output Shape	Param #
normalization (Normalization)	(None, 67)	135
dense_5 (Dense)	(None, 256)	17408
dropout (Dropout)	(None, 256)	0
dense_6 (Dense)	(None, 128)	32896
dropout_1 (Dropout)	(None, 128)	0
dense_7 (Dense)	(None, 64)	8256
dropout_2 (Dropout)	(None, 64)	0
dense_8 (Dense)	(None, 32)	2080
dropout_3 (Dropout)	(None, 32)	0
dense_9 (Dense)	(None, 2)	66
Total params: 60,841		
Trainable params: 60,706		
Non-trainable params: 135		

Figure 6.2: The Neural Network Model

The model has 4 fully connected layers consisting of 256,128,64, and 32 units respectively. In between these layers we employed dropout layers to prevent overfitting. A single layer with 2 units activated by Softmax is used after the fully connected layers to classify with the help of a sparse categorical cross entropy loss.

6.4 Activation Functions

The activation function is a non-linear change that we apply to the input before passing it to the next layer of neurons or converting it to output. An artificial neural network's activation function is very essential. They essentially determine whether or not a neuron should be activated.

Two main functions we use here are:

Sigmoid function

$$S(x) = \frac{1}{1 + e^{-x}}$$

A sigmoid function is a limited, differentiable, real function that has a non-negative derivative at each point and exactly one inflexion point and is defined for all real input values. The terms "sigmoid function" and "sigmoid curve" both refer to the same thing. [10]

Sigmoid Function

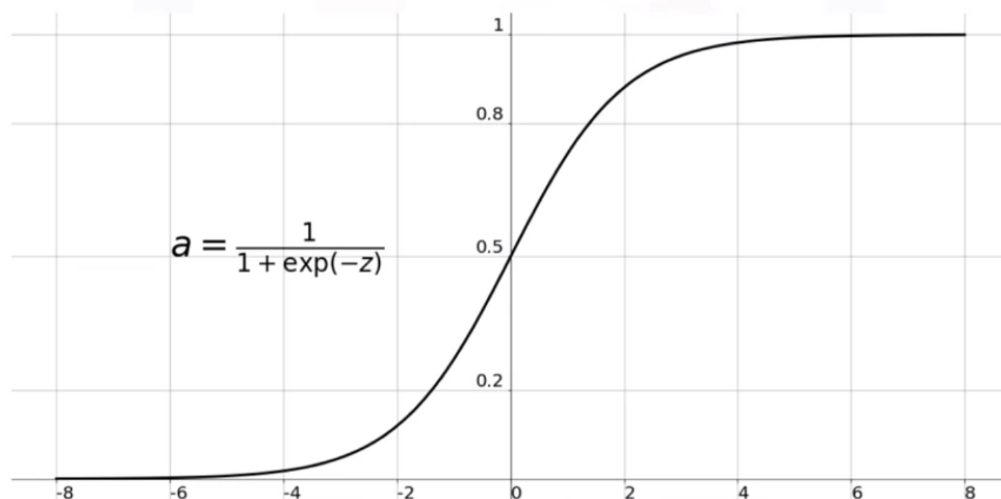


Figure 6.3: Sigmoid Function

ReLU function

ReLU function is the rectified linear unit function.

$$f(x) = \max(0, x)$$

Rectifying activation functions were employed to differentiate particular excitation and unspecific inhibition as they convert all the negative values to the zero which makes sure of the presence of values greater than or equal to 1.

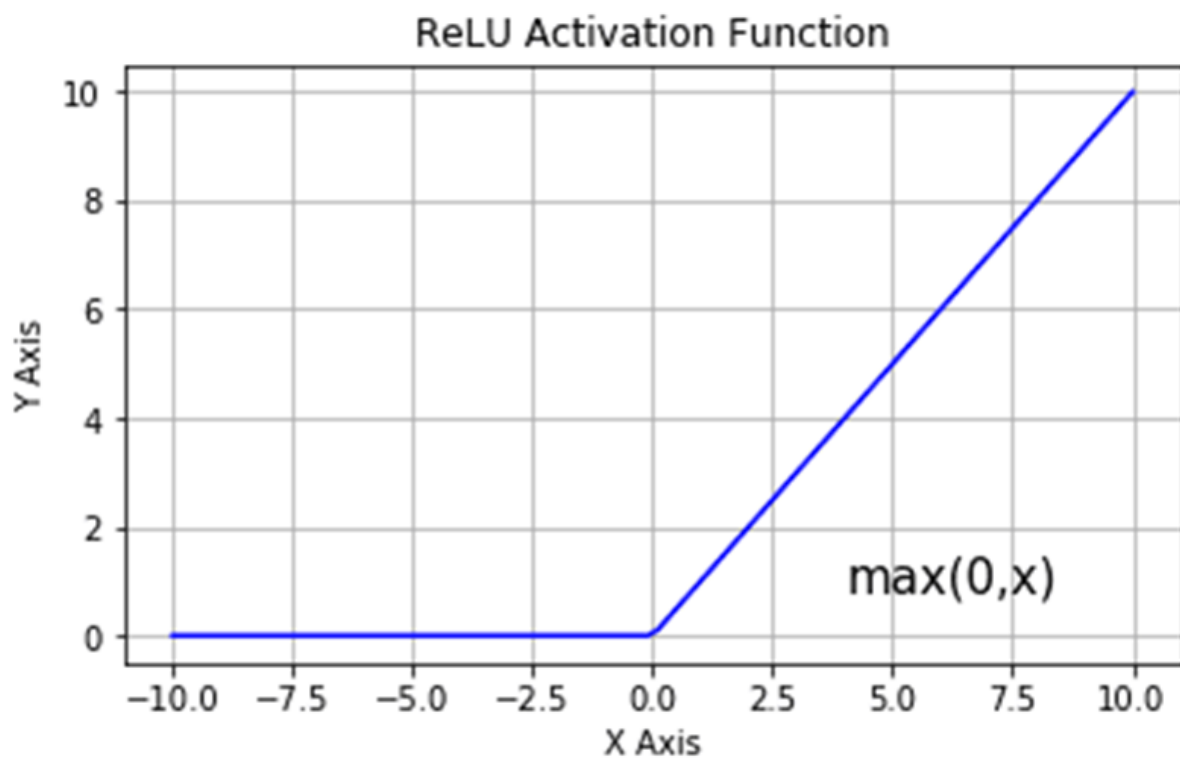


Figure 6.4: ReLU activation function

This function makes sure of having all positive values in the input that is sent to the each layer.

6.5 Forward and Backpropagation

The initial information is provided via the input X , which is subsequently propagated to the hidden units in all the layers to generate the output y . The activation functions, width and depth applied on each layer are all determined by the network's architecture. The number of hidden layers is measured in depth. We don't control the dimensions of the input or output layers, thus width is the number of units (nodes) on each hidden layer. Rectified Linear Unit, Sigmoid, Hyperbolic Tangent, and other activation functions are only a few examples. Deeper networks outperform networks with more hidden units, according to research. As a result, it's always better to train a wider network, and it won't hurt (with diminishing returns).

Allows information to flow back across the network from the cost to compute the gradient. As a result, compute the derivative of the final node output with respect to each edge's node tail by looping over the nodes in reverse topological order, starting at the final node. This will allow us to see who is causing the most errors and adjust the parameters accordingly.

6.6 Model Explanation

The model created has 256 layers, along with dropouts. Dropouts are used when there is a need for stabilizing the overfitting of the model. To overcome that overfit problem, at every layer, some inputs will be dropped out and that happens with a random function in place. This ensures a better loss and accuracy.

In the below plots, the trend between loss and accuracy can be clearly observed.

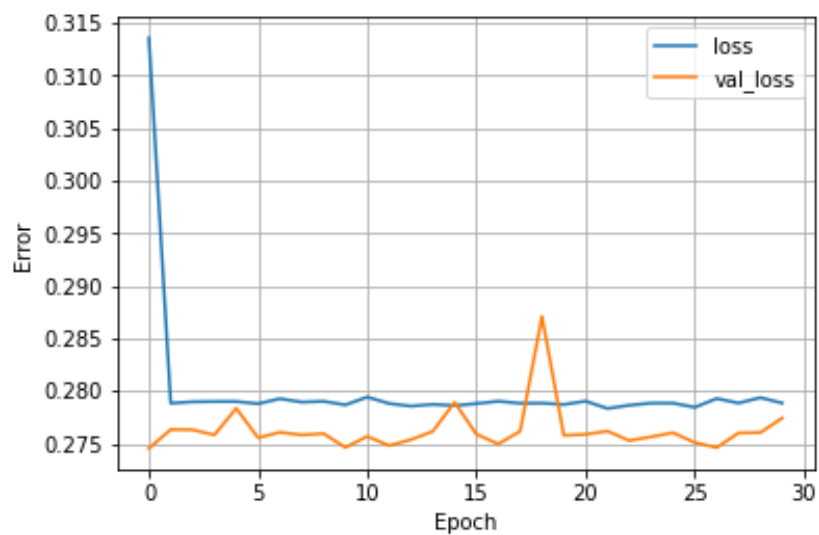


Figure 6.5: Plot of loss vs Validation loss

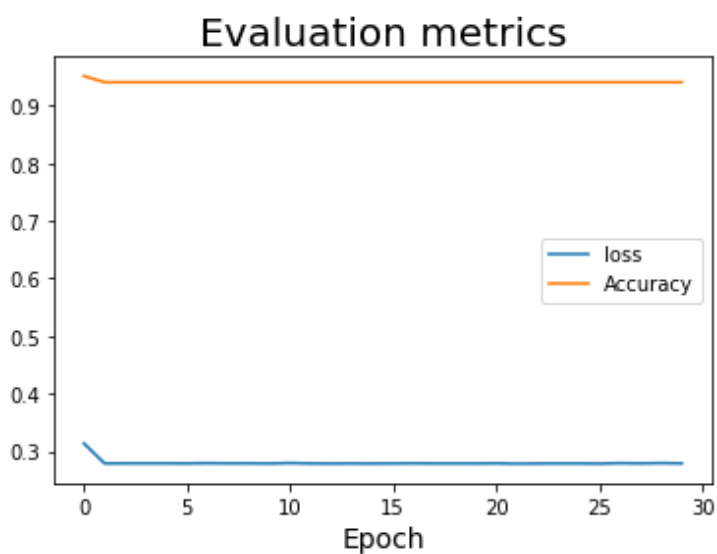


Figure 6.6: Plot of loss vs Accuracy

The model is set for 30 epochs. And the pattern of loss and validation loss from fig. 6.5 shows that after the first two to three epochs the loss and validation loss fluctuation is very insignificant. By the end of the last epoch, the loss and the validation loss is between 0.275 and 0.280.

And the plot of loss vs accuracy in the fig 6.6 gives an idea about how accuracy is unchanging similar to the loss after finishing nearly 4 to 5 epochs. The accuracy stands at 94.05 in the 5th epoch and remains the same till the 30th epoch is done.

The model is performing well given the parameters, as the accuracy is best and the predictions are well.

This brings close to the Neural Networks part of the study as well the end to the comparison between various techniques used in this study.

Chapter 7: Future Work and Conclusion

The study has a future scope as it involves modeling, training, testing and evaluation. Over time, data turns obsolete, the incoming new data can be fed to the models created making the models up to date. And fine tuning of the models can be done, using various feature extraction techniques.

7.1 Future Work

Apart from modeling, Mainly in this study, when dealing with the regression part. In case of Quasi-Poisson Regression, No implementation of Generalized Linear Models is found in any python package. To be able to use the same in Pyspark, no implementation of it is found even in the Pyspark module.

In order to use the same, the study can be extended in this regard to implement the Quasi Poisson GLM in python as well as in pyspark. There are a lot of dependencies for the Quasi Poisson GLM implementation in Pyspark. A look at Generalized Linear Models implementation in python gives an idea about how numpy and scipy libraries come handy in such cases.

7.2 Conclusion

The main focus throughout the study has been about modeling the number of claims using various techniques which includes Generalized Linear Models, Logistic Regression, SMOTE and Artificial Neural Networks. After looking at various models implemented, It can be observed that all the techniques arrive at a common conclusion that the predictions are similar as all techniques produce more zeros in their predictions compared to all other classes in the response variable. SMOTE technique helped in removal of imbalance in the data.

References

- [1] Introduction to Linear Regression Analysis by Douglas Montgomery, Elizabeth A. Peck, and G. Geoffrey Vining, 5th Edition
- [2]scribd.com
- [3]https://en.wikipedia.org/wiki/Poisson_distribution
- [4]https://www.researchgate.net/publication/288022415_AN_APPLICATION_OF_CLAIM_FREQUENCY_DATA_USING_ZERO_INFLATED_AND_HURDLE_MODELS_IN_GENERAL_INSURANCE
- [5]https://www.researchgate.net/publication/228856559_Spatial_modelling_of_claim_frequency_and_claim_size_in_non-life_insurance
- [6]<https://www.statology.org/poisson-distribution-assumptions/>
- [7]https://en.wikipedia.org/wiki/Logistic_regression
- [8]<https://www.ibm.com/cloud/learn/unsupervised-learning>
- [9]<https://www.ibm.com/in-en/cloud/learn/neural-networks>
- [10]<https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>
- [11]<https://databricks.com/glossary/pyspark>

Appendix

With reference to the data description about the variables in the CAS dataset is being described here.

Attribute Name	Description
advisorySpeed	At the time of the collision, the advisory (adv) speed (spd) at the crash site.
animals1	Derived variable to indicate how many times an 'Animal(s)' was struck in the crash.
areaUnitID	An area unit's unique identifier.
bicycle	The number of bicycles involved in the crash was derived as a variable.
bridge	The number of times a bridge, tunnel, abutments, and handrails were struck in the collision is indicated by this derived variable.
bus	The number of buses involved in the crash (excluding school buses, which are listed in the SCHOOL BUS field) is a derived variable.
carStationWagon	The number of automobiles or station waggons involved in the collision was derived as a variable.

cliffBank	The number of times a 'cliff' or 'bank' was impacted in the crash is indicated by this derived variable. Retaining walls are one example.
crashDirectionDescription	The crash's direction (dirn) in relation to the reference point. 'North', 'East', 'South', or 'West' are all potential values.
crashDistance	The distance (dist) between the crash and the crash reference location. The intersection of 'crash road' and 'side road' (refer to the 'cr rd sd rd' variable) is frequently used as a reference point.
crashFinancialYear	If known, the financial (fin) year in which a crash happened. This is shown in the form of a string field. 2004/2005.
crashLocation1	The first part of the 'crash location'. It could be a road name, a route position (RP), a landmark, or something else entirely, such as 'Ninety Mile Beach.' In reports and other documents, it's used to describe where something is located.
crashLocation2	The second part of the 'crash location'. It could be the name of a side street, a landmark, or something else. In reports and other documents, it's used to describe where something is located.
crashSeverity	The seriousness of a collision. The letters 'F' (fatal), 'S' (severe), 'M' (minor), and 'N' (not fatal) are all possible possibilities (non-injury). The worst injury experienced in the crash at the time of entry is used to determine this.

crashSHDescription1	Indicates whether a crash occurred on a State Highway (SH) marked with a '1' or on another road type marked with a '2'.
crashYear	If known, the year in which a crash occurred.
debris	The number of times rubble, stones, or anything dropped or flung from a vehicle(s) was struck in the crash is a derived variable.
directionRoleDescription	The main vehicle involved in the collision's direction (dirn). North, south, east, and west are all possibilities.
ditch	The number of times a 'ditch' or 'waterable drainage channel' was struck in a crash is indicated by this derived variable.
easting	The easting coordinate of an object (typically a crash) given in NZMG was accurate to 1 metre in the WGS84 datum. Please keep in mind that not all crashes may be linked to GPS coordinates.
fatalCount	This is a tally of the number of people who died as a result of the crash.
fence	The number of times a 'fence' was hit in the accident was derived into a variable. This includes letterboxes, hoardings, privately owned roadside furniture, hedges, and sight rails, among other things.
flatHill	Whether the road is flat or sloped. Possible values include 'Flat' or 'Hill'.

guardRail	The number of times a guard or guard rail was struck in the crash was derived as a variable. This includes barricades marked 'New Jersey,' 'ARMCO,' sand-filled barriers, wire catch fences, and so on.
holiday	If a collision occurred during the holiday periods of 'Christmas/New Year,' 'Easter,' 'Queens Birthday,' or 'Labour Weekend,' the value will be 'None.'
houseOrBuilding	The number of times a house, garage, shed, or other building(Bldg) was struck in the crash was derived as a variable.
intersectionMidblock	A derived variable that shows if a collision occurred at a crosswalk (intsn). The 'intsn midblock' variable is calculated using the 'intersection' and 'junction type' variables.
junctionType 1	The type of intersection where the accident occurred. 'Driveway,' 'Roundabout,' 'Crossroads,' 'T Junction,' 'Y Junction,' or 'Multileg' are all examples of possible road intersections.
kerb	The number of times a kerb was impacted in the incident, and how many times it contributed directly to the crash..
light	The light that was present at the time and location of the crash. 'Bright Sun,' 'Overcast,' 'Twilight,' 'Dark,' or 'Unknown' are examples of possible values.
meshblockId	The unique identifier of a meshblock.

minorInjuryCount	A count of the number of minor injuries (inj) associated with this crash.
moped	The number of times a kerb was impacted in the incident, and how many times it contributed directly to the crash.
motorcycle	The number of motorcycles involved in the incident was derived as a variable.
northing	The northing coordinate of an object (typically a crash) given in NZMG was accurate to 1 metre in the WGS84 datum. Please keep in mind that not all crashes may be linked to GPS coordinates.
NumberOfLanes	On the accident route, the number(num) of lanes.
objectThrownOrDropped	Derived variable to indicate how many times objects were thrown at or dropped on vehicles in the crash.
outdatedLocationDescription1	Indicates whether the location for this accident is an old location ('TRUE') or a new place ('FALSE').
otherObject	The number of times an object was struck in a crash and the object struck was not pre-defined and was derived into a variable.
otherVehicleType	The number of other vehicles involved in the crash (not included in any other category) is a derived variable.
overBank	The number of times an embankment was struck or driven over during a crash is a derived variable. Other vertical variables are included in this variable.

parkedVehicle	The number of times a parked or unattended car was struck in the crash was derived as a variable. Trailers can be included in this variable.
phoneBoxEtc	The number of times a telephone kiosk, traffic signal controllers, bus shelters, or other public furniture was struck in the crash was derived as a variable.
pedestrian	The number of pedestrians involved in the accident was derived as a variable. Pedestrians on skateboards, scooters, and wheelchairs are included.
postOrPole	The number of times a post or pole was impacted in the crash was derived as a variable..
region	The local government (LG) region is identified. The boundaries correspond to those of the territorial local authority (TLA).
roadCharacter1	The nature of the road in general. 'Bridge' and 'Moto' are examples of possible values.
roadCurvature1	The road's curve has been simplified. 'Curved' and 'Straight' are two options.
roadLane	The road's lane arrangement. '1' (one way), '2' (two way), 'M' (where a median exists), 'O' (for off-road lane configurations), and" (for unknown or invalid configurations).
roadMarkings	The crash site's road markers. 'Ped Crossing' (for pedestrian crossings), 'Raised Island," Painted Island,' No

	Passing Lanes,' Centre Line,' No Marks,' or 'Unknown' are all possible options.
roadSurface	At the crash scene, the road surface description is in effect. 'Sealed' or 'Unsealed' are two options.
roadworks	The number of times an object linked with 'roadworks' was struck during the crash (including signs, cones, drums, and barriers, but not roadwork vehicles).
schoolBus	The number of school buses involved in the collision was derived as a variable.
seriousInjuryCount	The number of significant injuries (inj) sustained as a result of this collision.
slipOrFlood	Derived variable to indicate how many times landslips, washouts or floods (excluding rivers) were objects struck in the crash
speedLimit	At the time of the collision, the speed (spd) limit (lim) in effect at the crash site. Maybe a number, or 'LSZ' for a limited speed zone.
strayAnimal	The number of times a stray animal(s) was struck in the crash was derived as a variable.
streetLight	At the time of the collision, the street lighting was dim. 'On', 'Off', 'None', or 'Unknown' are possible values.
suv	The number of SUVs involved in the collision was derived as a variable.

taxi	The number of cabs involved in the accident was derived as a variable.
tlald	The unique identifier for a territorial local authority (TLA). Each crash is assigned a TLA based on where the crash occurred.
tlaName	The name of the territorial local authority (TLA) to which the crash was attributed has been revealed.
temporarySpeedLimit	The temporary (temp) speed (spd) limit (lim) at the crash site if one exists (e.g. for road works).
trafficControl	The crash site's traffic control (ctrl) signals. 'Traffic Signals,' 'Stop Sign,' 'Give Way Sign,' 'Pointsman,' 'School Patrol,' 'Nil,' or 'N/A' are all possible values.
trafficIsland	The number of times a traffic island, medians, or barriers were struck in the crash was derived as a variable.
trafficSign	Derived variable to indicate how many times 'traffic signage' (including traffic signals, their poles, bollards or roadside delineators) was struck in the crash.
train	The number of times a train, rolling stock, or jiggers were struck in the crash, whether stationary or moving, is indicated by this derived variable.
tree	The number of times trees or other growing items were struck during the collision is indicated by this derived variable.
truck	The number of trucks involved in the crash was derived as a variable.

unknownVehicleType	The number of automobiles involved in the collision was derived as a variable (where the vehicle type is unknown)
urban	The 'spd lim' variable was used to create a derived variable. 'Urban' (urban, spd lim 80) or 'Open Road' (open road, spd lim >=80 or 'LSZ') are two possible values.
vanOrUtility	The number of vans or utes involved in the collision was derived as a variable.
vehicle	The number of times a stationary attended car was impacted in the crash was derived as a variable.
waterRiver	The number of times a stationary attended car was hit in the incident was derived into a variable.
weatherA	Weather at the time/location of the crash. See for more information. 'Fine,' 'Mist,' and 'Dark' are examples of possible values.
weatherB	The weather at the time and location of the crash. Values 'Frost,' 'Strong Wind,' or 'Unknown' can be found in weather a.