ISyE6740 – Course Project

Traffic Collision Analysis

Project Final Report

April 27, 2024

Team Information (Team 60)

Geeyavudeen Musthafa (gmusthafa3@gatech.edu)

Geeyavudeen is an IT Development Manager, leading a few strategic initiatives for the Treasury Services of Bank of New York, Mellon. He has immense experience in leading IT Development projects, mostly for the financial services sector.

Roles and Responsibilities in Final Work include EDA, feature engineering, model development, model assessment, future work suggestions, and final report review.

Mohammed Al-Desouky (ma@gatech.edu)

A mechanical engineer with 20+ years of experience in structural analysis, engineering software development, IT management, contract management, and project management. Mohammed works currently as a senior project manager leading projects worth 500+ million USD. In the past, Mohammed worked on the development of an optimization model to maximize material utilization in a 12 billion USD project.

Roles and Responsibilities in Final Work include data engineering, EDA, feature engineering, model review, visualizations, and final report write-up.

Table of Contents

1 Objective/Problem

1.1 Abstract

Road traffic crashes are a leading cause of death in many countries, especially those with people living in high population density environments. According to the United States Center for Disease Control:

> *Road traffic crashes are a leading cause of death in the United States for people ages 1–54, and they are the leading cause of nonnatural death for U.S. citizens residing or traveling abroad. It is estimated that fatal and nonfatal crash injuries will cost the world economy approximately $1.8 trillion dollars (in 2010 USD) from 2015–2030.*

Clearly, there is great value in reducing traffic collisions, or in reducing the severity of those collisions that inevitably occur. With that in mind, our project seeks to identify those characteristics of a collision that make it more likely to result in a death or an injury (or multiple deaths/injuries) and to allow people and organizations to use this information to make choices that reduce their own associated health, financial and emotional burdens. In this project, we will be focusing on one major city in the United States: Chicago.

This study aims at analyzing traffic crash data for Chicago from March 2013 to April 2024. Chicago is one of the largest US cities. It has more than 6,400 km of roads distributed over an area of over 600 km^2 [1]. Chicago has some of the largest traffic congestions in US [2]. The dataset at hand has information for more than 800,000 crashes.

1.2 Problem Statement

Given that a crash has occurred what is the expected crash severity in terms of injury? Moreover, which of these factors have the highest impact on such expectation?

- Time
- Location
- Weather
- Road conditions
- Lighting conditions
- Posted speed limit

2 Datasets

2.1 Data Sources

We will utilize two separate data sets for the City of Chicago. These are:

- *Traffic Crashes*: The Motor Vehicle Collisions crash table contains details on the crash event. Each row represents a crash event. The Motor Vehicle Collisions data tables contain information from all police reported motor vehicle collisions in Chicago.
  https://catalog.data.gov/dataset/traffic-crashes-crashes

- *TIGER/Line Shapefiles*: This is a geospatial dataset that contains ZIP code boundaries in the United States as per 2023 census.
  https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.html

3 Exploratory Data Analysis

This dataset has exactly 817,841 crash records. Each record represents a crash event. Each crash event is defined by 49 features. These features can be grouped into the following groups:

Table 1: Crashes Dataset Feature Groups

Group	# Features	Example of Features within Group
Road	6	Road's physical condition, existence of defects, and number lanes.
Traffic	5	Speed limit, existence traffic control devices and their conditions.
Surroundings	5	Existence of nearby work zone, weather, lighting
Crash	13	Crash date/type/cause, whether it is a hit-and-run or not
Police	5	Police report type, photos, statements taken
Location	6	ZIP code, street name, coordinates
Outcome	9	Crash outcome in terms of damage, injuries, fatalities

Out of the 49 features, we have 8 features that have no entries at all. i.e., all null. The following histogram shows the null entry percentage for all features.

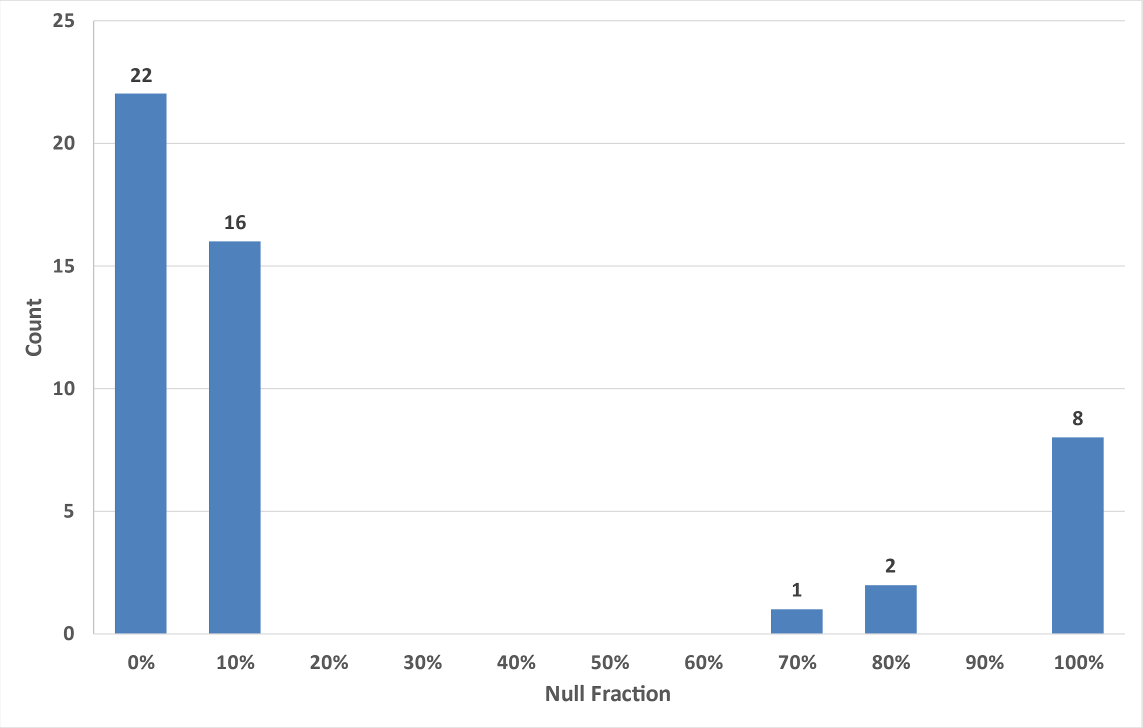

Figure 1: Null Entry Histogram

3.1 Dataset Features

In this section, we will try to explore what the dataset at hand is telling us without the use of analytical modeling. The dataset covers the time window from March 2013 till April 2024. The following sections are named after feature names.

3.1.1 ZIP Codes

For a starter, let's have a general look at injury distribution. The following figure shows Total Injuries distribution (both fatal and non-fatal), followed by Fatalities distribution, and finally Incapacitating Injuries distribution in the city of Chicago.

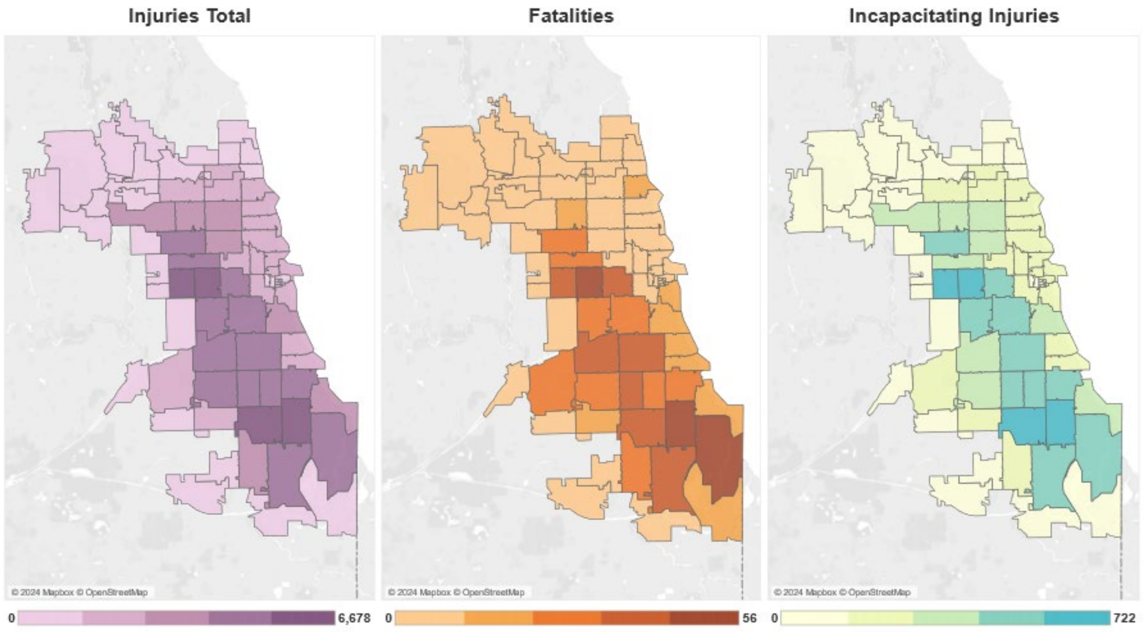
Injuries Total
Fatalities
Incapacitating Injuries
© 2024 Mapbox © OpenStreetMap
© 2024 Mapbox © OpenStreetMap
© 2024 Mapbox © OpenStreetMap
0
6,678
0
56
0
722

Figure 2: Chicago Crash Injuries Distribution by ZIP Code

3.1.2 *Streets*

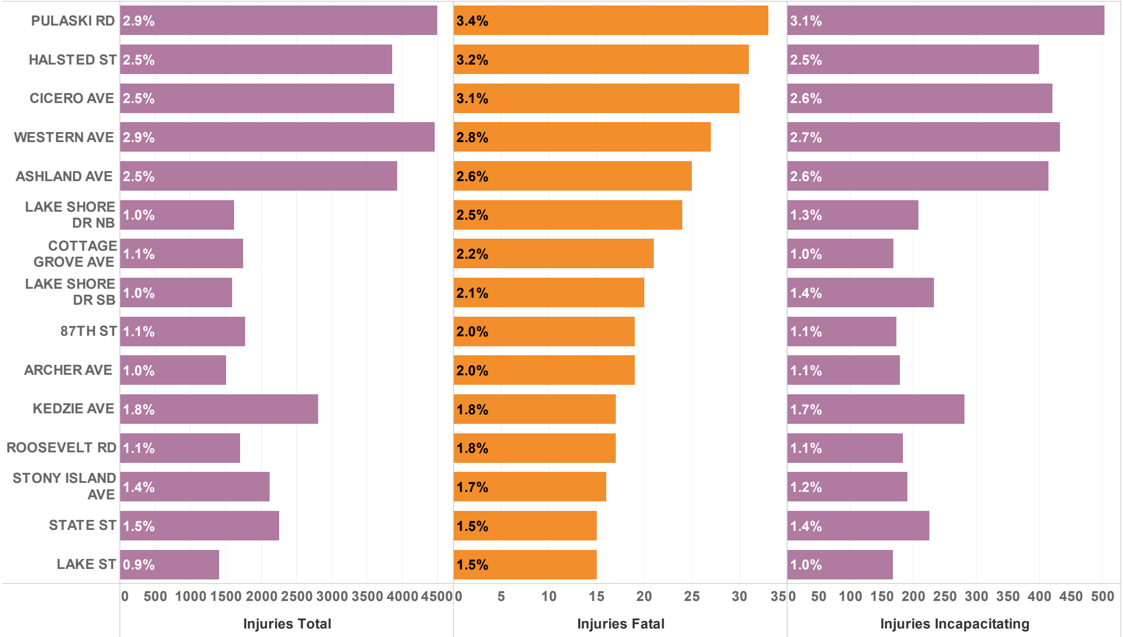

Figure 3: Crash Injuries Distribution by Street Name

As seen in Figure 3, *Halsted Street* is the street with the highest injury and fatality rate. It is also important to mention that the top 15 streets own 33.95% of fatalities, 25.82% of incapacitating injuries, and 25.49% of total injuries.

3.1.3 *First Crash Type*

This field holds the type of first collision in crash. This is the hit that started, in some cases, the chain reaction of crashes. We weren't planning on adding this, but the results were too interesting to leave out.

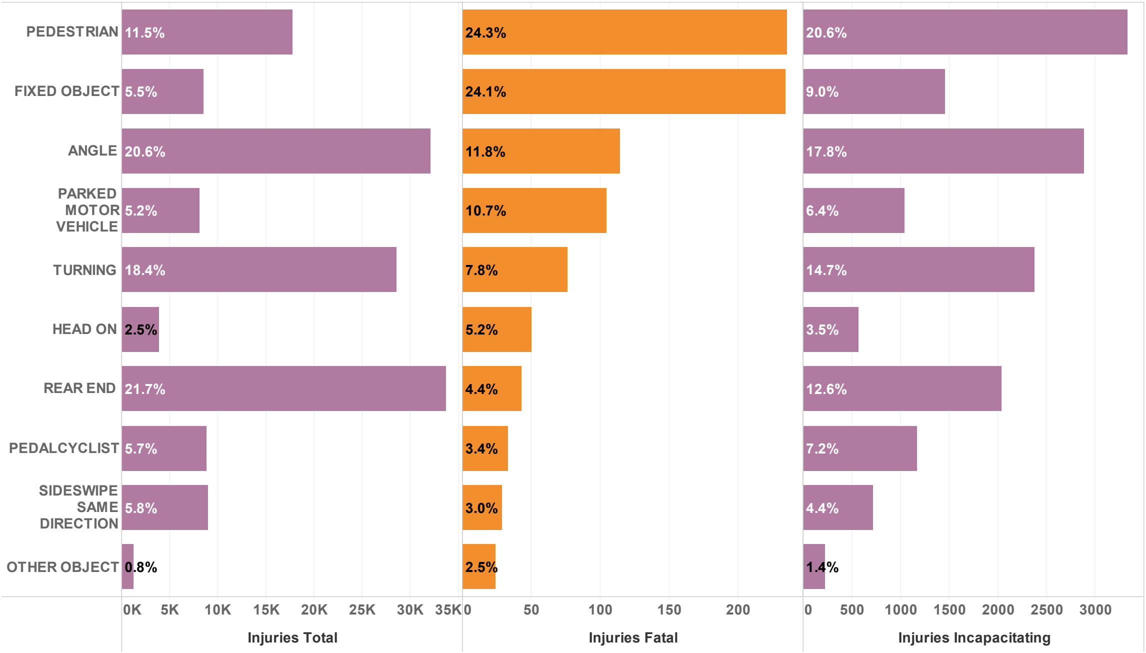

Figure 4: Injuries Distribution per First Crash Type

Crashing with a *Fixed Object* and *Pedestrians* own 48.40% of fatalities! It is difficult to ignore the use of mobile phones while driving -for drivers- or while walking -for pedestrians- as a root cause.

3.1.4 *Lighting Condition*

This field holds the light condition at time of crash, as determined by reporting officer. Surprisingly, 87.9% of the accidents happened under good lighting conditions!

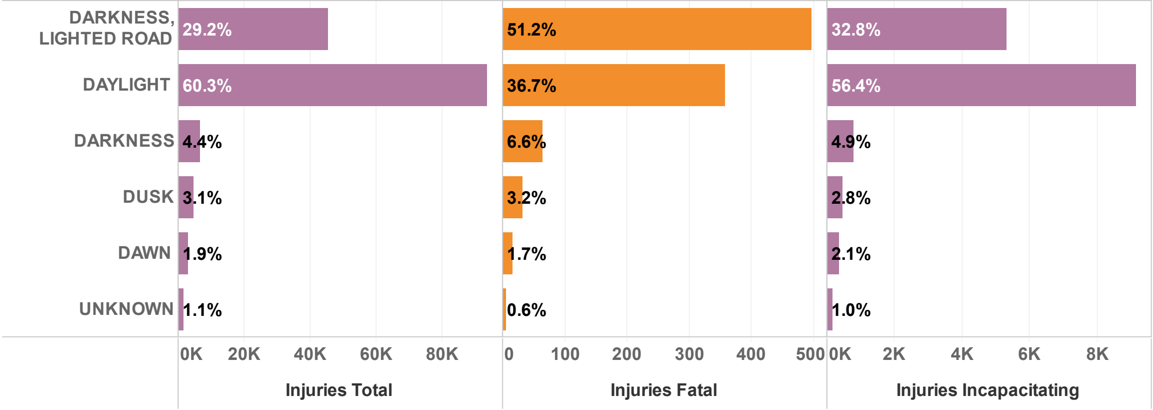

Figure 5: Lighting Condition Statistics

3.1.5 *Primary Contributing Cause*

This field holds the factor, which was most significant in causing the crash, as determined by officer judgment. The `UNABLE_TO_DETERMINE` entry was removed from the figure below for better visualization. Please note that this feature addresses causes related to driver. From Figure 6, we can conclude that most fatalities are attributable either driver's physical condition, not properly being able to control the vehicle, or ignoring the traffic signal.

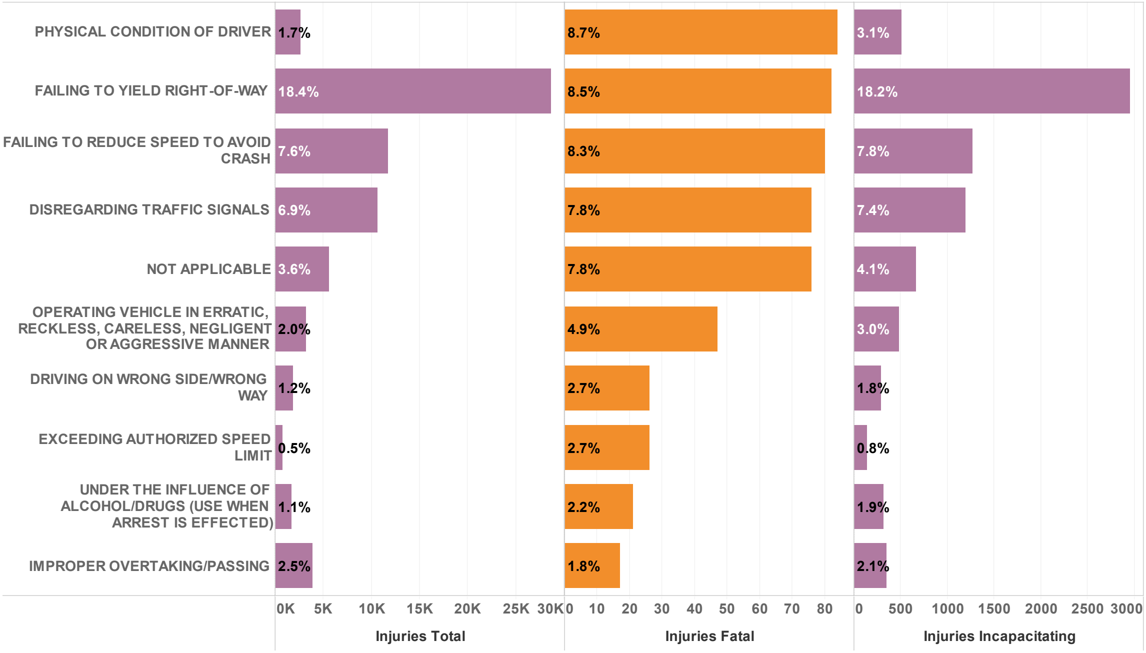

Figure 6: Primary Contributing Cause - Does NOT Include Undetermined Causes

3.1.6 Crash Hour

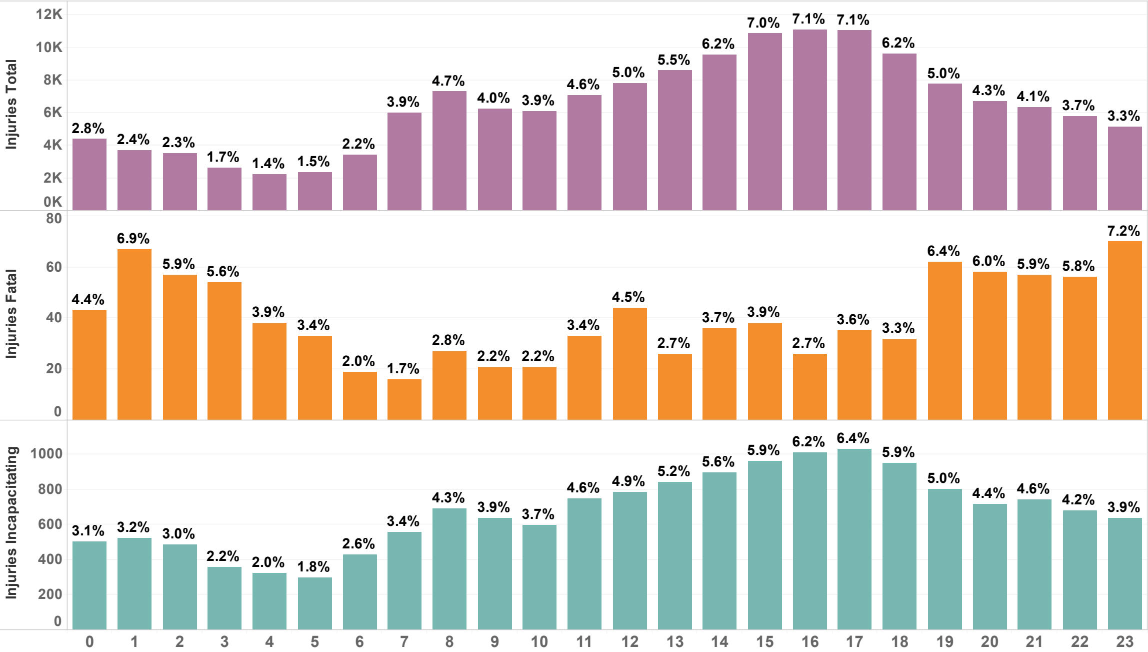

Figure 7: Crash Distribution over Hours of the Day

Crash Hour is a discrete numerical variable specifying at what time of the day the crash occurred. This is an important factor, sometimes during the day, the streets are more crowded than others.

3.1.7 Miscellaneous Features

Weather does not appear to be of significance. It seems that most crashes occur in clear weather. *Road Surface Condition* is on the same page as weather conditions. Most crash events occurred on dry roads.

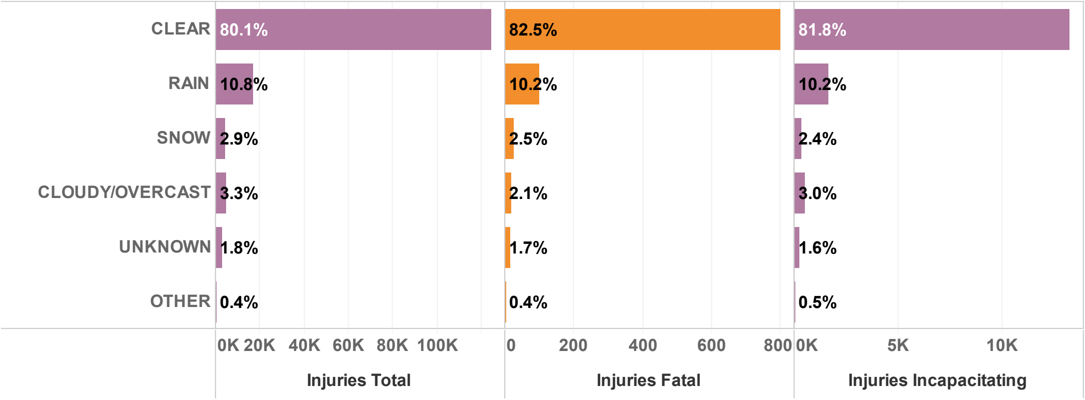

Figure 8: Weather/Injury Statistics

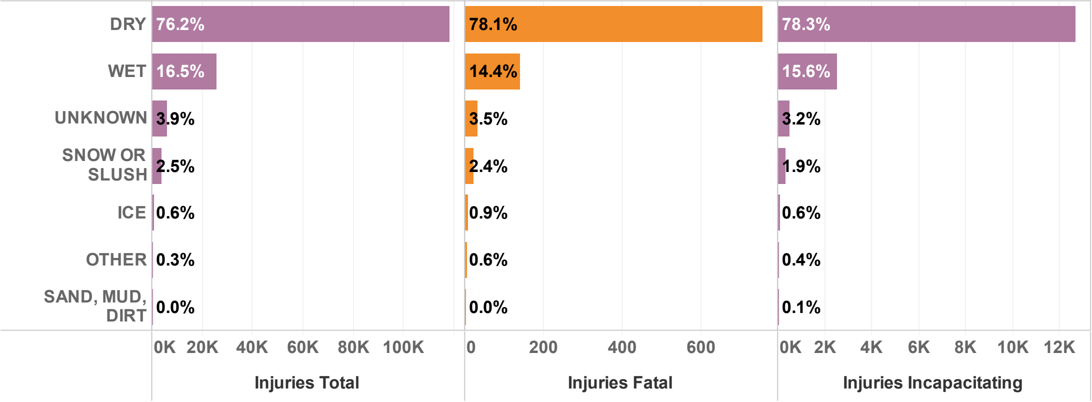

Figure 9: Road Surface Condition

3.2 Response Variable

Clearly, the number of injuries is the natural response. However, we have 7 injury-related features. We will deal with that in the Feature Engineering section.

3.3 Preliminary Feature Selection

Based on EDA, preliminary feature selection can take place to reduce the whole feature set into something suitable for modeling on our computers. Based on the above, the following features are selected:

- POSTED SPEED LIMIT
- WEATHER CONDITION
- LIGHTING CONDITION
- FIRST CRASH TYPE
- ROADWAT SURFACE CONDITION
- CRASH HOUR
- ZIP CODE

4 Feature Selection/Engineering

As seen previously, we have two major problems in our dataset: many response variables and many categories inside categorical features. We need to deal with both. We also need to reduce these features, because once they are one-hot-encoded, the model will explode.

4.1 Feature Engineering

4.1.1 ZIP Code Groups

ZIP code is a categorical variable that expresses the location zone of the crash event. We have 74 unique ZIP codes in our dataset, which will produce 74 binary variables when encoded. This is not good for modeling. Besides, we have a rough idea about the relation between ZIP codes and crash severity.

There are some areas which seem to have intense crashes that cause fatalities and/or incapacitating injuries. We can also notice that the same areas have higher injury rates in general. *In fact, 8 ZIP codes own 27.30% of total injuries, 36.64% of fatalities and 26.51% of incapacitating injuries*. ZIP codes will be broken into 4 groups: A, B, C, and D. Grouping is based on percentage of fatalities as the most severe type of injury:

- Group A contains top ZIP codes (9) that combined own around 40% of fatalities.
- Group B contains the following ZIP codes (11) that combined own the next 30% of fatalities.
- Group C contains the following ZIP codes (17) that combined own the next 20% of fatalities.
- Group D contains the remaining ZIP codes (38) which constitute the remaining 10%.

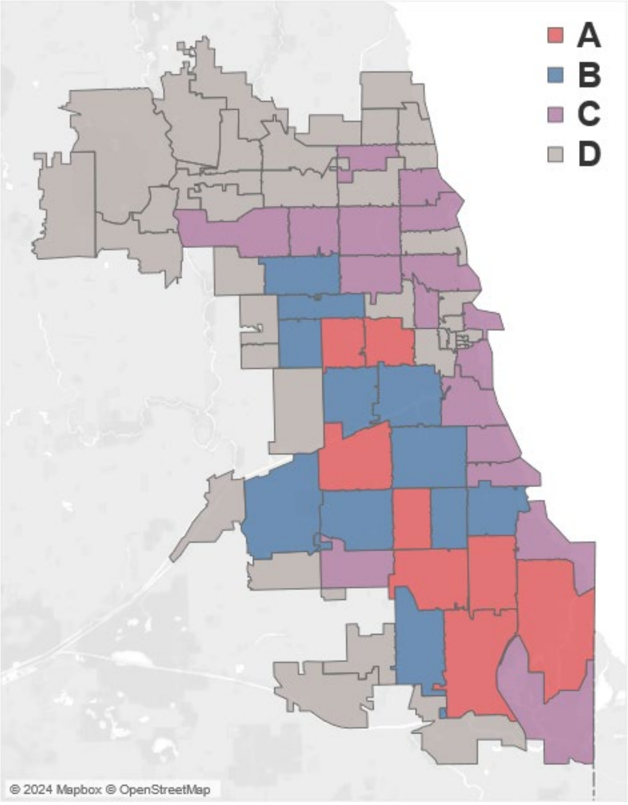

Figure 10: ZIP Code Groups

Conclusively, we decided to create a new feature, `ZIP_CODE_GROUP`, which will contain the above grouping for all crash events in our dataset. The following is the crash distribution per ZIP group.

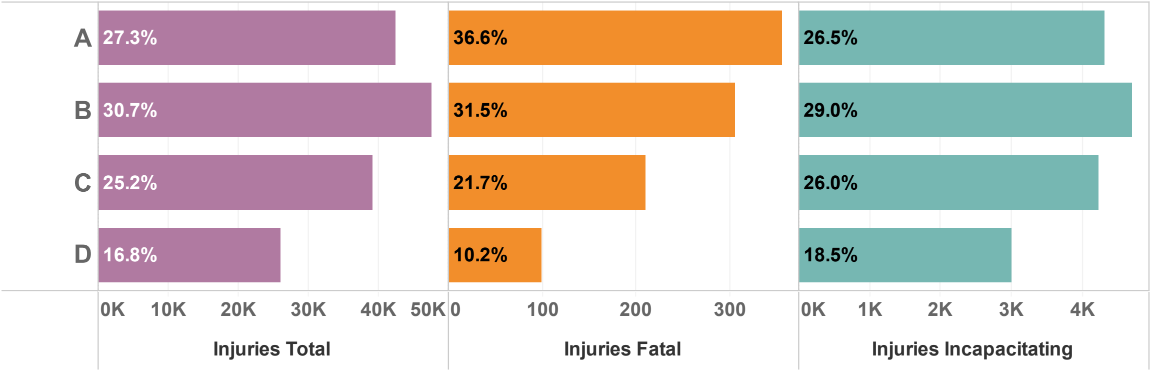

Figure 11: Crash Distribution per ZIP Code Group

4.1.2 Injury Class

This dataset has 7 injury-related features. This is a big number of response features to handle. To deal with this situation, we created a new feature called `INJURY_CLASS` that contains one of the following values for every crash event:

Table 2: Injury Classes

Class	Description
A	Fatal injuries
B	Incapacitating injuries
C	Non-incapacitating injuries
D	No injuries

4.1.3 Crash Hour

Crash hour -at the first sight- appears to be a numerical variable, but it is not. Increasing or decreasing crash hour -as a value- has no significance on response. In fact, it should be considered categorical. To reduce modeling burden, we created a new variable, `CRASH_HOUR_PERIOD`, that simply contains the period of the day, thus breaking the day into 4 periods instead of 24:

Table 3: Crash Hour Periods

Period	Hour Range
Morning	06:00 to 12:00
Afternoon	13:00 to 18:00
Evening	19:00 to 22:00
Night	23:00 to 05:00

Let's have a look at the result:

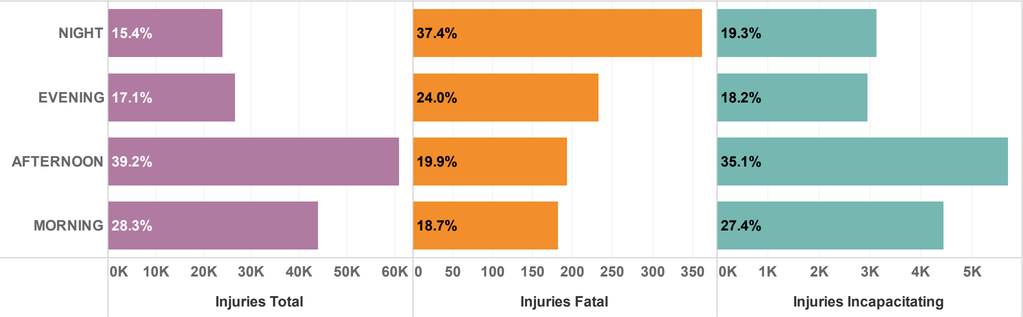

Figure 12: Crash Distribution per Period of the Day

4.2 Feature Selection

We have only categorical features. This -unfortunately- forced us to rule out dimensionality reduction (like PCA) and feature selection algorithms that depend on numerical features and/or response such as ANOVA. Please keep in mind that the response is now categorical (`INJURY_CLASS`).

We employed χ^2 based feature selection, which computes χ^2 statistic between each feature and class. The score can be used to select the best *n* features with the highest values for the χ^2 statistic from selected features, relative to the classes. Please note the χ^2 test measures dependence between stochastic variables, so using this function *weeds out* the features that are the most likely to be independent of class and therefore irrelevant for classification. With χ^2 feature reduction technique, the following are the best *k* for different models:

Table 4: Feature Selection Results

Model	**Best *k***
Multi-class classification using Decision Tree Classifier	80
Single-class classification using Logistic Regression	40
Single-class classification using Logistic Regression (with primary and secondary contributory clusters) More details in the Modeling Section	12

Other than Classification Models, we also build ensemble model Random Forest with 20 trees and maximum depth of 20 levels to get the important features. Random Forest classifier computes the importance of the features based on the impurity. The higher the importance score, the more important the feature is. The importance of a feature is computed as the (normalized) total reduction of the criterion brought by that feature. It is also known as the Gini importance. Based on this selection technique, Random Forest identified 41 features as important features which influenced the results with importance ≥ 0.005.

5 Methodology

We've built multiple models to evaluate and predict the injuries caused by Chicago crashes dataset. The details of each of the models can be found below.

5.1 Splitting Training/Testing Datasets

There are 817,841 records in the cleaned dataset. The following are the aggregate size of the dataset grouped by Injury classes.

Table 5: Training/Testing Split Sizes

Class	Description	Size	Percentage
A	Fatal injuries	896	0.11%
B	Incapacitating injuries	13,812	1.69%
C	Non-incapacitating injuries	98,027	11.99%
D	No injuries	705,106	86.21%
Total		**817,841**	

Since classes A and B constitute only 1.8% of the overall dataset, we can't use the regular split-up of 80-20% of training and test data. So, we have included all records of class A and class B in both training and test dataset and split only class C and D into 80% of training and 20% of test dataset. This modified selection ensures the model has enough info about serious injuries to build a balanced model.

5.2 Models

5.2.1 Decision Tree Classifier (Full Model)

The initial model was built with Decision Tree Classifier to perform multi-class classification to classify the data into 4 injury classes – A, B, C, D. We used `DecisionTreeClassfier` model from Scikit Learn package to train our model. We applied cross-validation with the following model parameters to select the best model but included all one-hot-encoded features. This is a complex model with 96 features included in it.

Table 6: Parameters of Decision Tree Classification Model

Parameters	Values	Selected by CV
Decision Tree Criterion	Gini, Entropy	Gini
Decision Tree Max Depth	4, 8, 12, 24, 48	4

Gini and Entropy are scoring metrics to determine the node splits in the Decision tree algorithm. Entropy measures the disorder or randomness in a dataset, whereas Gini quantifies the probability of misclassifying a randomly chosen element. We used `GridSearchCV` from sci-kit package to run cross-validation across different parameters to choose the best model parameters. With this approach, the best model is selected with Gini impurity method and maximum depth of 4.

5.2.2 Decision Tree Classifier (Reduced Model)

Since the initial model is a complex model with 96 features included in it, we've tried to simplify the model by including Feature selection technique to select the features based on Chi-squared metrics. We've used `SelectKBest` feature from Scikit Learn library to select the number of features based on best chi-squared score. The following are the model parameters for this reduced model.

Table 7: Parameters of Reduced Decision Tree Classification Model

Parameters	Values	Selected by CV
Decision Tree Criterion	Gini, Entropy	Gini
Decision Tree Max Depth	4, 8, 12, 24, 48	4
k-value of `SelectKBest`	12, 20, 40, 60, 80, 90	80

Though we used chi-squared based score to reduce the model, the cross-validation grid search has still selected 80 features which produced the best score. The Decision tree model parameters remain the same as Gini and max depth of 4 on the reduced model as well.

5.2.3 Logistic Regression (Full Model)

Then we've modified the response classes into a binary variable of serious injury or not. We've combined class A and B to indicate serious injury and C and D indicate minimal or non-injury. With this change, we've built a Logistic Regression model again using sci-kit library. The initial model was a full model with all 96 features included. The following are the model parameters for Grid Search Cross validation technique for logistic regression model.

Table 8: Parameters of Logistic Regression Model

Parameters	Values	Selected by CV
Logistic Regression - C	0.001, 0.01, 0.1, 1, 10, 100, 1000	0.1

We've used fixed parameters for number of iterations as 1000 to converge and tolerance for stopping the iterations as 0.1 and varied only regularization strength score C for Grid Search cross validation. With this approach, regularization strength is selected as 0.1.

5.2.4 Logistic Regression (Reduced Model)

We tried to simplify the Logistic Regression model by reducing the number of features, again by employing chi-squared based scoring metric. The following are the model parameters for the reduced model.

Table 9: Parameters of Reduced Logistic Regression Model

Parameters	Values	Selected by CV
Logistic Regression - C	0.001, 0.01, 0.1, 1, 10, 100, 1000	1
k-value of `SelectKBest`	12, 20, 40, 60, 80, 90	40

This model drastically reduced the number of features from 96 to 40 with regularization strength changed to 1, which produced the best chi squared score.

5.2.5 Logistic Regression (Feature Enhanced Model)

As part of our original feature selection based on exploratory data analysis, we've excluded two of the features `PRIM_CONTRIBUTORY_CAUSE` and `SEC_CONTRIBUTORY_CAUSE`, since 38.98% of dataset had `UNABLE_TO_DETERMINE` as the primary cause and 5.29% of the dataset as NOT APPLICABLE. These two causes don't provide a meaningful definition of the injuries, we've excluded these features altogether from our model. But the rest of the values from the primary contributory cause had a meaningful detail. So, we tried to enhance our Logistic Regression model to include this feature. But including this feature might skew the data, we filtered out 44% of data falling into `UNABLE_TO_DETERMINE` and `NOT_APPLICABLE` causes and included only the dataset with meaningful cause.

But there are around 40 different causes and including all these causes further complicate the model. So, we employed another model within the model to perform text analysis on these causes and cluster them into groups.

Table 10: Parameters of Reduced Logistic Regression Model

Contributory Cause	Cluster
IMPROPER BACKING	3
EVASIVE ACTION DUE TO ANIMAL, OBJECT, NONMOTORIST	0
DRIVING SKILLS/KNOWLEDGE/EXPERIENCE	0
FAILING TO YIELD RIGHT-OF-WAY	0
FOLLOWING TOO CLOSELY	0
IMPROPER LANE USAGE	3
UNDER THE INFLUENCE OF ALCOHOL/DRUGS (USE WHEN ARREST IS EFFECTED)	1
WEATHER	0
FAILING TO REDUCE SPEED TO AVOID CRASH	0
DISREGARDING TRAFFIC SIGNALS	0
TEXTING	0
IMPROPER OVERTAKING/PASSING	3
DISTRACTION - FROM INSIDE VEHICLE	0
OPERATING VEHICLE IN ERRATIC, RECKLESS, CARELESS, NEGLIGENT OR AGGRESSIVE MANNER	0
IMPROPER TURNING/NO SIGNAL	3
EQUIPMENT - VEHICLE CONDITION	0
EXCEEDING AUTHORIZED SPEED LIMIT	0
ROAD ENGINEERING/SURFACE/MARKING DEFECTS	0
DRIVING ON WRONG SIDE/WRONG WAY	0
DISREGARDING STOP SIGN	0
PHYSICAL CONDITION OF DRIVER	0
CELL PHONE USE OTHER THAN TEXTING	1
TURNING RIGHT ON RED	2
HAD BEEN DRINKING (USE WHEN ARREST IS NOT MADE)	1
VISION OBSCURED (SIGNS, TREE LIMBS, BUILDINGS, ETC.)	0
DISTRACTION - FROM OUTSIDE VEHICLE	0
ROAD CONSTRUCTION/MAINTENANCE	0
DISREGARDING ROAD MARKINGS	0
DISREGARDING YIELD SIGN	0
EXCEEDING SAFE SPEED FOR CONDITIONS	0
DISREGARDING OTHER TRAFFIC SIGNS	0
ANIMAL	0
RELATED TO BUS STOP	0
PASSING STOPPED SCHOOL BUS	0
MOTORCYCLE ADVANCING LEGALLY ON RED LIGHT	2
BICYCLE ADVANCING LEGALLY ON RED LIGHT	2
DISTRACTION - OTHER ELECTRONIC DEVICE (NAVIGATION DEVICE, DVD PLAYER, ETC.)	0
OBSTRUCTED CROSSWALKS	0

5.2.6 Text Clustering of Primary Contributory Causes

To analyze the primary contributory causes, we've built a text analytics clustering model using `KMeans` clustering algorithm. The following are the steps performed to build the text clustering.

1. Tokenize the words from the contributory causes.
2. Removed all stop-words which didn't provide a meaningful context and built a final bag of words.
3. Vectorize the tokens into floating point values suitable for modeling using the count of occurrences of the words.
4. Built K-Means model on the vectorized tokens to cluster the primary contributory causes.

We used `TfidfVectorizer` from sci-kit library to perform first 3 steps above and `KMeans` model for clustering. TF stands for Term Frequency and `TF-IDF` means term frequency times inverse document-frequency. These are techniques used in text analytics algorithms to vectorize the words for analytics algorithms. Since our scope of this project is to predict the injuries from crashes data and we use this technique only for Feature Engineering to include one of the text fields, we are not delving deep into this model. Based on this text clustering model, the following are the clusters for various contributory causes.

5.2.7 Logistic Regression Model

We built the logistic regression model after including the primary contributory cause clusters as one of the features. And below are the selected model parameters:

Table 11: Parameters of Logistic Regression Model

Parameters	Values	Selected by CV
Logistic Regression - C	0.001, 0.01, 0.1, 1, 10, 100, 1000	0.001
k-value of `SelectKBest`	12, 20, 40, 60, 80, 90	12

5.2.8 Random Forest Ensemble Model

Finally, we bult an ensemble model of Random Forest with 20 decision trees and max depth of 20. This model on the training dataset has generated the importance scores for the features based on the impurity scores. We've reduced the model by selecting top important scores with score ≥ 0.05. This selected around 41 features to reduce the model complexity.

6 Evaluation and Results

6.1 Evaluation

For all the models described previously, we've evaluated the models using the test dataset and compared the performances of each of the models. Here's the comparison of accuracy and F1 scores of each model.

Table 12: All Model Scores

Model	Accuracy Score	F1 Score
Decision Tree Classifier (Full Model)	85.72	85.72
Decision Tree Classifier (Reduced Model)	85.72	85.72
Logistic Regression (Full Model)	95.63	95.63
Logistic Regression (Reduced Model)	95.63	95.63
Logistic Regression (Feature Enhanced Model)	93.20	93.20
Random Forest (Reduced Model)	93.47	93.47

The following subsections display confusion matrices for all models.

6.1.1 Decision Tree Classifier

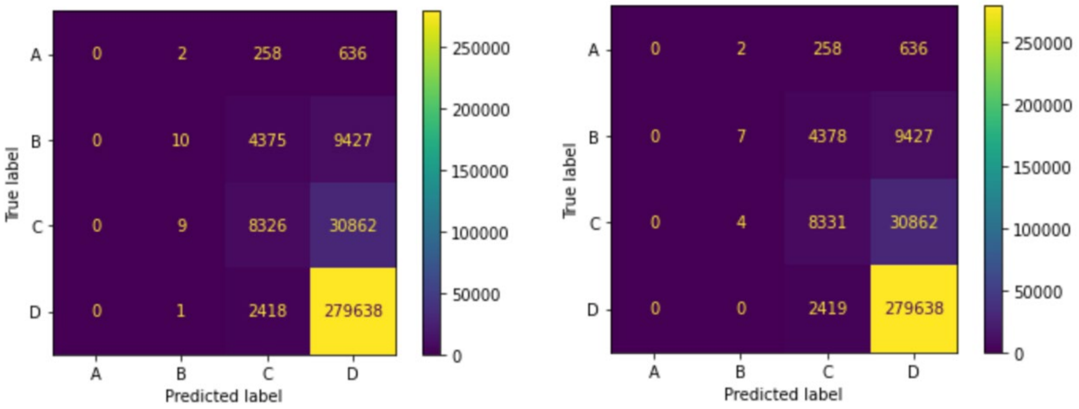

Figure 13: Confusion Matrices for Decision Tree Classifiers (Full, Reduced)

6.1.2 Logistic Regression

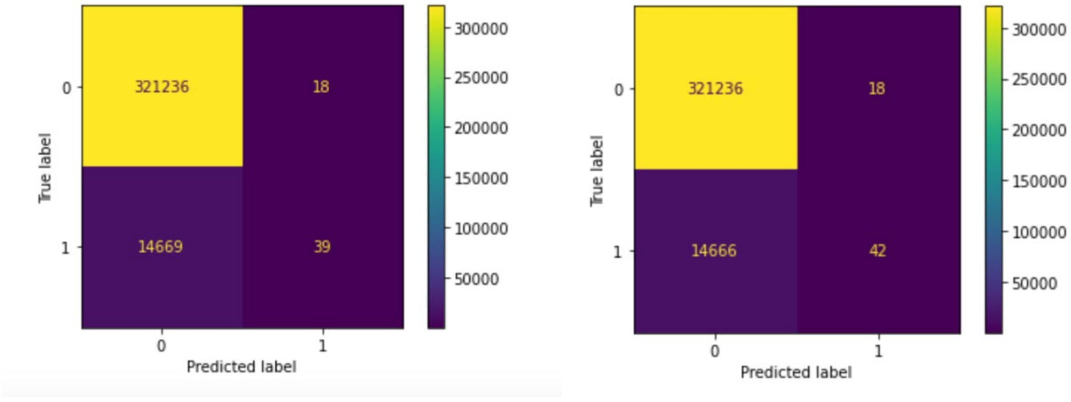

Figure 14: Confusion Matrices for Logistic Regression Classifiers (Full, Reduced)

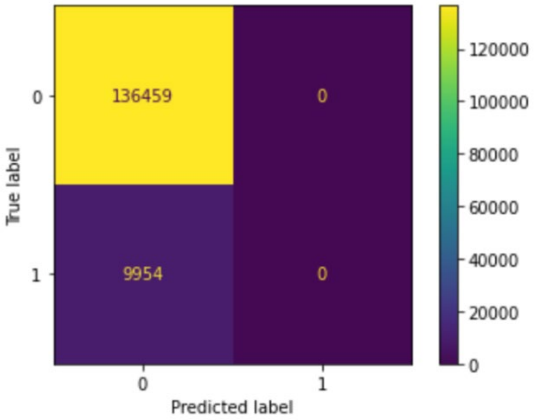

Figure 15: Confusion Matrices for Logistic Regression Classifiers (Enhanced)

6.1.1 Random Forest Model

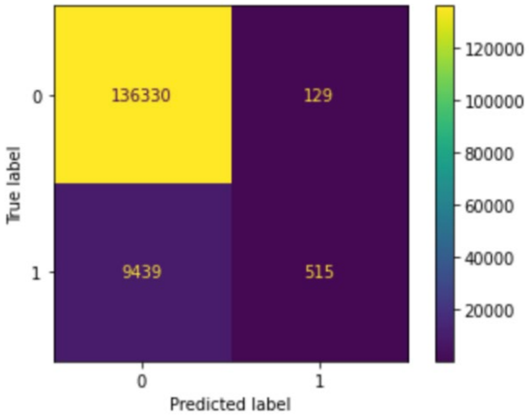

Figure 16: Confusion Matrices for Random Forest Model

6.2 Results

Though the accuracy scores of each of the models stand at 90%+, it's not reliable while looking into the confusion matrix of each of the models. Since our dataset is skewed with 98% negative results and only 2% positive results, the accuracy scores for predicting negative results are higher whereas the performance to predict the injuries stands low due to the lack of training datasets. Most of the models have classified only a minimal percentage of the injuries correctly. Random Forest ensemble model have predicted the injuries slightly better than other models.

Since the dataset is not balanced, the results are not very comforting. We tried to build a model on a smaller subset of the data which included all crashes resulted in fatal and incapacitating injuries but included only 20% of other crashes. And built a random forest model with 20 trees and max depth of 20 for a multi-class classification on the training dataset randomly chosen from the smaller subset. This model had an accuracy score of 93% and produced the below specified confusion matrix. We can see this model has better performance on predicting serious injuries than similar models on full dataset.

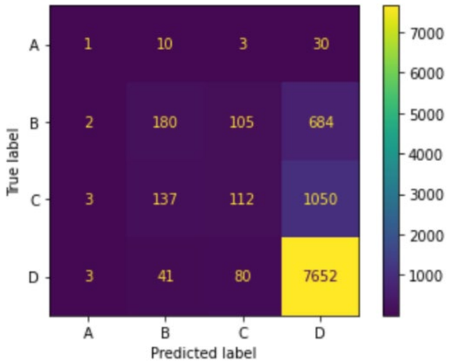

Figure 17: Balanced Model's CM

7 Future Work

- **Balancing the Dataset**: The dataset at hand is unbalanced. We suggest applying appropriate techniques to handle this imbalance. We believe that this will enhance model accuracy.
- **Perform Time-series Prediction**: From another point of view, this is a time series data that can be handled in a different way by using time-series models such as GARCH. We believe exploring this area would add more value to this work.
- **Collecting More Data:** It is true we have a lot of information on crash events, which are almost all categorical, but we believe that collecting information such as population density, traffic congestion rate, and average traffic speed will help build a better model. For example, the following figure has fatal injury density as an example. By visually comparing it to population density, we can easily estimate by naked eye how much population density is correlated with crash fatalities.

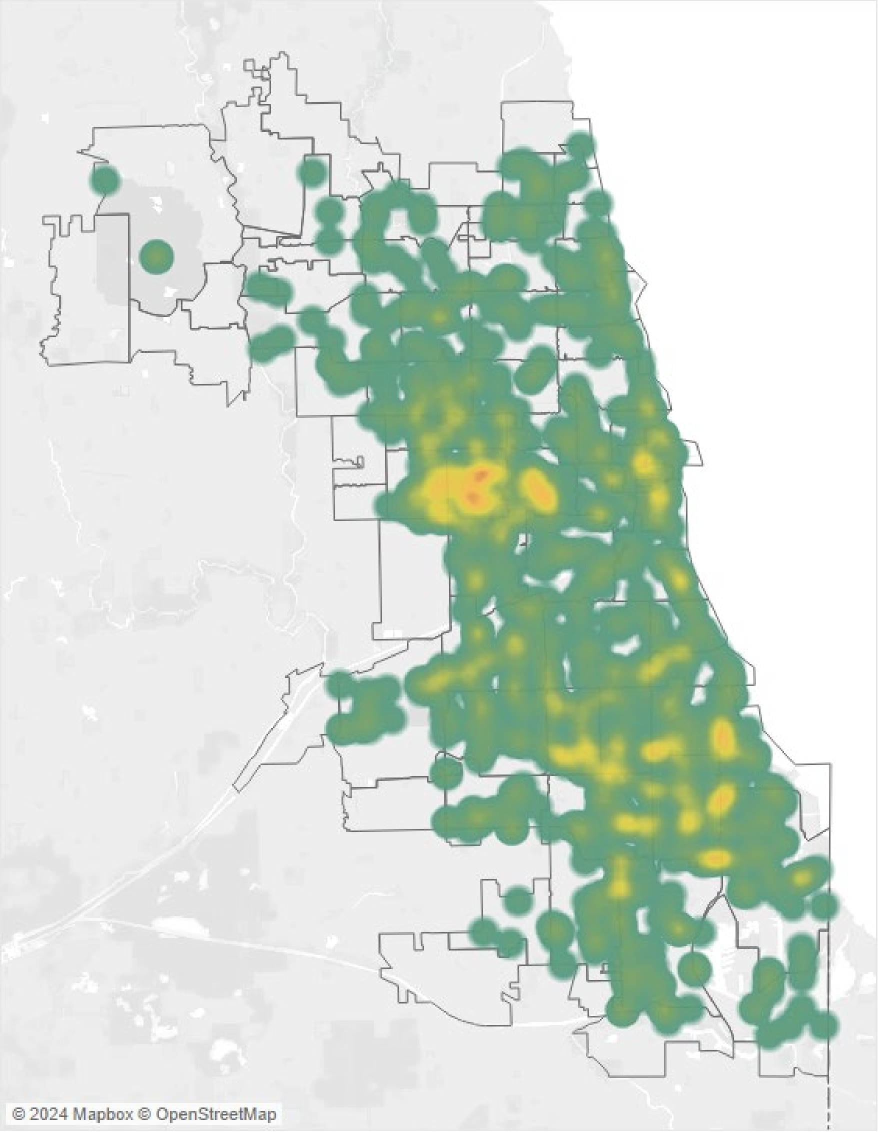

Figure 18: Fatalities Density Distribution over Chicago

8 References

[1] Herz, "Chicago Driving Guide," [Online]. Available: https://www.hertz.com/us/en/blog/planning-a-trip/chicago-driving-guide.

[2] "Chicago Leads Nation in 2022 Traffic Congestion," [Online]. Available: https://chicago.suntimes.com/2023/1/11/23550990/chicago-leads-nation-in-2022-traffic-congestion-report-says.

[3] "Chicago Data Portal - Traffic Crashes - Crashes," [Online]. Available: https://data.cityofchicago.org/Transportation/Traffic-Crashes-Crashes/85ca-t3if/about_data.