Risk Prediction and Assessment in the Construction Industry

Zhixing Wang Department of Civil and Environmental Engineering University of Illinois Urbana–Champaign Urbana, IL, USA
zw88@illinois.edu

Zain Sitabkhan Department of Civil and Environmental Engineering University of Illinois Urbana–Champaign Urbana, IL, USA
zsita@illinois.edu

Deago Sirenden Department of Civil and Environmental Engineering University of Illinois Urbana–Champaign Urbana, IL, USA
deagofs2@illinois.edu

Zhihui Da Department of Civil and Environmental Engineering University of Illinois Urbana–Champaign Urbana, IL, USA
zhihuid2@illinois.edu

Abstract

This project focuses on risk prediction and assessment in the construction industry using incident and accident data from New York City. By applying regression-based models, the objective is to predict fatality and injury outcomes, as well as generate a weighted index to evaluate the severity of such events. The study contributes to understanding which attributes most strongly influence construction-related incidents and provides insights that may improve safety measures in the industry.

keywords:“Construction Safety”, “Risk Prediction”, “Accident Reports”, “Regression Analysis”

Contents

1. Introduction

1.1 Background & Motivation

Although construction safety has been studied for more than two decades, research that integrates real incident reports with feature extraction and machine–learning–based prediction is still relatively new. This line of work remains in an early stage.[1]In New York City, construction makes up an unusually high share of workplace deaths—over 20% of all fatalities. Many of these cases are tied to safety violations and gaps in oversight or enforcement [2].

In our preliminary regression model, we selected temperature, humidity, and HVI (heat vulnerability index) as our new parameters. However, in the neural network model, we include direct indicators of safety noncompliance and supervision gaps, while the statistical models use meteorological variables to capture environmental effects on incidents.

Construction safety has long been a central concern in civil engineering, with most prior work focusing either on structural health monitoring or on post-incident analyses based on official reports. However, studies that combine text-based incident reports, systematic feature extraction, and machine-learning-based predictive modeling remain relatively scarce.Although New York City provides a large amount of construction-related data, incidents are distributed very unevenly across boroughs. Manhattan reports far more cases because of its dense concentration of projects and workers, while several other boroughs have relatively few. This imbalance adds another layer of difficulty for both modeling and interpretation [3].

This imbalance also leads to many zero observations. When zeros dominate the data, standard Poisson models often break down because their core assumption—mean equals variance—no longer holds. Heavy zero inflation also makes coefficient estimates in standard regression models unstable and less trustworthy [4].In addition, the small data size, limited feature availability, and weak model interpretability can significantly affect machine learning models. These constraints narrow the kinds of conclusions models can draw and make it harder to develop reliable predictive tools [5].

Therefore, it's vital to use machine learning methods to test our model on New York City data with varying parameter settings.The purpose of the model is to help avoid incidents and accidents at New York City by identifying the dominant attributes influencing outcomes, thereby guiding proactive protection measures in construction management.

1.2 Objectives

Our main objective is to analyze different types of construction incidents at New York City that happened within 1 or 2 years from now. For this project, we would mainly be examining the nature of construction related incidents and accidents as well as performing correlations with the data by examining the prevalence of each incident and accident at each of the five boroughs of New York City. We would want to see where each type of incident has the highest probability of occurring, and where specifically measures should be implemented to prevent these types of incidents. Finally, keeping track of when these incidents occurred will also be critical as the data could also be used to calculate the frequency of accidents over time.

1.3 Overview of Analytical Plan

In Section 2, we begin by preprocessing the data and, through a correlation-mapping analysis, identify and introduce additional relevant parameters. Section 3 conducts preliminary regression modeling to diagnose the limitations of classical regression approaches and to determine that subsequent modeling efforts should primarily adopt a classification framework. Sections 4 and 5 present the clustering and tree-based baselines, specifically k-means clustering and decision tree models. Sections 6 and 7 develop the neural-network–based classification and regression models; although our main focus remains on classification, we still explore regression models for the purpose of methodological completeness and comparative analysis.

2. Exploratory Data Analysis

This section will mainly focus on introductory data analysis with some preliminary tables and plots describing critical aspects of the data. Visible patterns will be discussed with the data that has been formulated and the coming sections below will describe how we plan on applying and modelling this data.

2.1. Basic Information

The dataset consists of construction-related incidents and accidents at New York City in each of the five boroughs. It provides a large-scale CSV file suitable for predictive analysis.[6] The dataset includes approximately 958 rows, each representing an accident or incident record, and 20 columns containing attribute fields of these records.

2.2 Attributes

Table 1. Attribute Definitions and Descriptions

Attribute Name	Unit/Type	Description
BIN	Integer	Building Identification Number (unique ID for each building)
Accident Report ID	Integer	Unique identifier of each accident report
Incident Date	Date	Date of the incident or accident
Record Type Description	Category (Text)	Record type, distinguishing Incident from Accident
Check2 Description	Category (Text)	Detailed category of the incident, e.g., Construction Related, Mechanical Equipment, Worker Fall
Fatality	Integer	Number of fatalities
Injury	Integer	Number of injuries
House Number	Text/Number	House number of the incident location
Street Name	Text	Street name of the incident location

Borough	Category	Administrative borough (e.g., Manhattan, Bronx, Brooklyn)
Block	Integer	Geographic block number
Lot	Integer	Lot number within the block
Postcode	Integer	Postal code of the location
Latitude	Float	Latitude coordinate of the incident location
Longitude	Float	Longitude coordinate of the incident location
Community Board	Integer	Community board identifier
Council District	Integer	City council district identifier
BBL	Integer	Borough-Block-Lot unique cadastral identifier
Census Tract (2020)	Integer	Census tract number from the 2020 census
Neighborhood Tabulation Area (NTA) (2020)	Text	Neighborhood Tabulation Area (NTA) code from 2020

Proposal for attribute usage will be made, focusing on those with predictive relevance.

2.3 Correlation Mapping

In this data integration step, we enriched the dataset by incorporating external environmental and vulnerability factors. The Heat Vulnerability Index (HVI) was integrated by joining it with the dataset using 'postcode' as the linking key. We then performed a data cleaning step to ensure data quality by removing records with missing values. Following this, climate variables, specifically 'AvgTemp' (Average Temperature) and 'AvgPrecip' (Average Precipitation), were merged into the dataset. This process of joining HVI, merging climate data, and handling missing values resulted in a final, refined dataset containing 415 valid observations, which was then used for the subsequent correlation and regression analyses

Table 3. Integrated Dataset with Climate and HVI Variables

Borough	Postcode	YearMonth	IncidentCount	Fatality	Injury	AvgTemp	AvgPrecip	HVI
Bronx	10451	Jun-24	3	1	2	71.7	4.4	5

Preliminary inspection indicates that higher-HVI areas (typically in the Bronx and parts of Brooklyn) correspond to marginally elevated injury counts, hinting at interactions between heat exposure and worker safety.

Some other parameter will be added such as Noncomplaint Count and IssueNumber (in section 6) in order to solve the regression model problems.

2.3.1 Weighted HVI

Weighted averaging is used when different observations contribute unequally to an aggregate measure.In another word it will directly contain the information about the borough.

2.3.2 Global Correlation

A global correlation analysis was conducted among key variables: TotalIncidents, Fatality, Injury, AvgTemp, AvgPrecip, and HVI.

Table 5. Correlation Matrix of Incident, Climate, and Vulnerability Variables

	TotalIncidents	Fatality	Injury	AvgTemp	AvgPrecip
TotalIncidents	1.000	0.120	0.958	0.025	0.023
Fatality	0.120	1.000	0.075	−0.007	−0.153

We conducted a global correlation analysis to understand the initial linear relationships between the primary variables. The results, partially shown in the table1, reveal several key patterns.Most notably, there is a very strong positive correlation between TotalIncidents and Injury (Pearson $r = 0.958$), which is expected as most incidents involve injuries.In contrast, Fatality demonstrates a weak correlation with the other variables in the table. The correlation with TotalIncidents is low ($r = 0.120$), and it is even weaker with Injury ($r = 0.075$). The environmental variables, AvgTemp ($r = -0.007$) and AvgPrecip ($r = -0.153$), also show negligible linear relationships with fatalities.Furthermore, analysis of the Heat Vulnerability Index (HVI) indicated a moderate negative correlation with both incidents and injuries, with correlation coefficients (r) observed in the range of approximately −0.57 to −0.606. This suggests that areas with higher vulnerability scores may, counterintuitively, be associated with fewer reported incidents in this dataset, prompting the need for further, more nuanced analysis.

2.3.3 Log-scaled Correlation

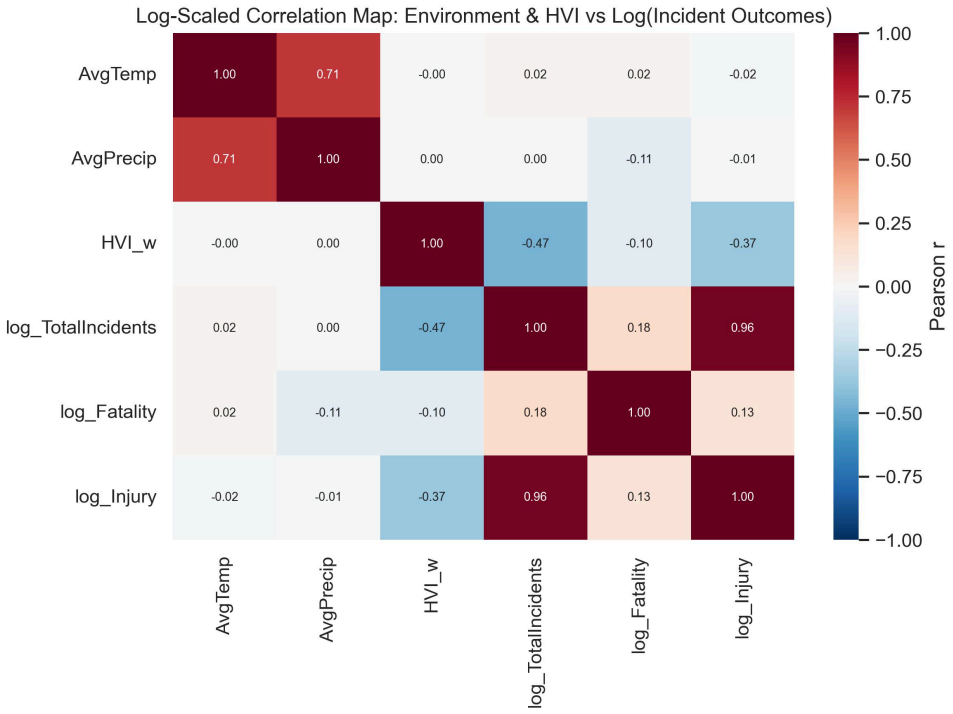

Figure 1: Correlation heatmap after log scaling

Results show a strong positive correlation between TotalIncidents and Injury (r ≈ 0.96) and a negative correlation between HVI and Fatality (r ≈ −0.57). Although counterintuitive at first glance, this may reflect underreporting or mitigation interventions in high-vulnerability areas. These relationships were visualized using a log-scaled correlation heatmap, emphasizing nonlinear

dependencies that justify the use of both Poisson and Negative Binomial regression models in the next section.

2.4 Preprocessing

2.4.1 Data Integration and Cohort

For the data integration and cohort definition, we first filtered the dataset. We then applied groupby operations to extract and aggregate key information. This aggregation was performed by 'borough (Area)', month, and 'postcode'. The 'postcode' attribute serves as a critical key, as it is directly used to link and integrate the Heat Vulnerability Index data [7].

2.4.2 Borough × Month Aggregation

To explore trends over time, the data were aggregated by borough and month. The aggregation reveals that incidents tend to cluster during the spring and summer months, aligning with increased construction activity.

Table 2. Monthly Aggregation of Incidents by Borough and Postcode

Borough	Postcode	YearMonth	IncidentCount	Fatality	Injury
Bronx	10451	Feb-24	1	0	1
Bronx	10451	Mar-24	1	0	1
Bronx	10451	Apr-24	1	0	1
Bronx	10451	Jun-24	3	1	2

Excerpt shown above; full panel saved as `monthly_borough.csv`.

2.5 Results for Preprocessing

Four of the preliminary plots below count incidents such as fatalities and injuries that happened at each borough and district. The last four show the total injuries and fatalities that happened in New York City during each month, and then the accumulation of injuries and fatalities over time from January 2024 to October 2025. With this data, we can go even further with borough or district specific statistics regarding these construction incidents.

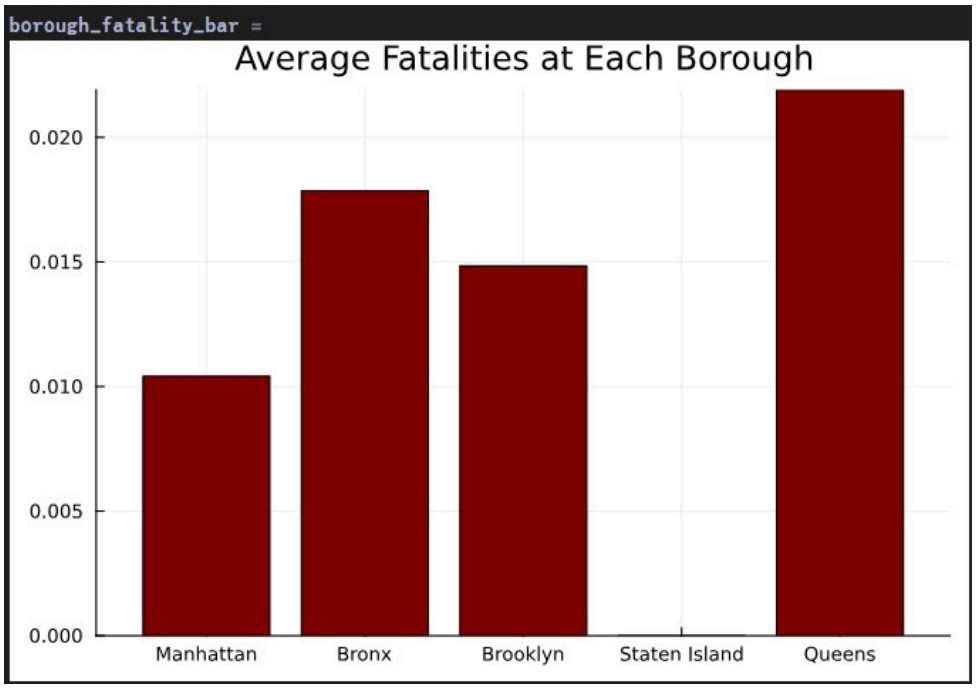

Figure 2: Average Fatalities at Each Borough

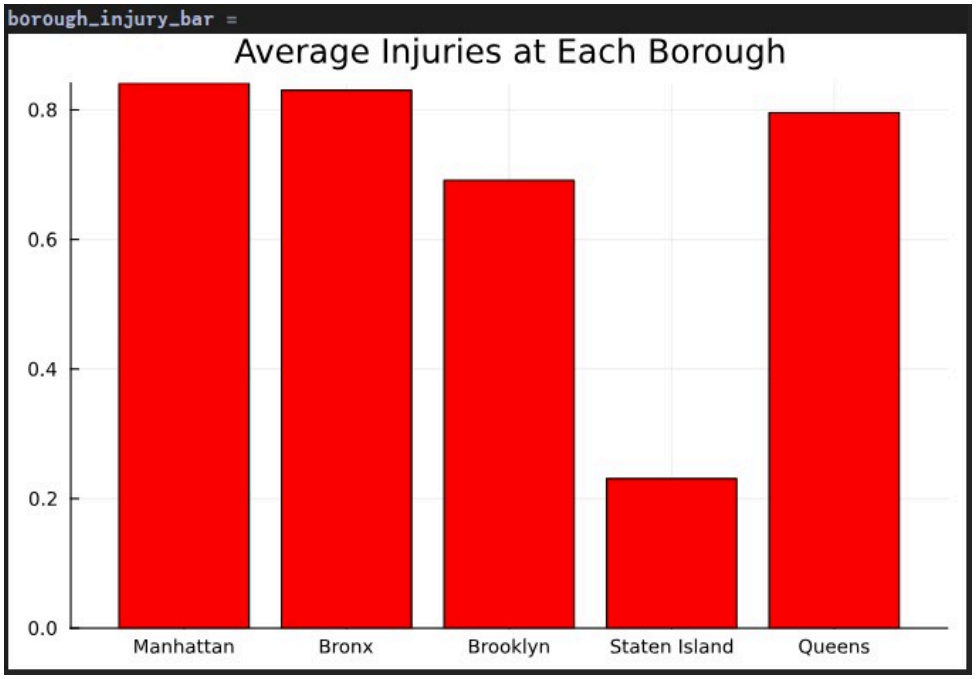

Figure 3: Average Injuries at Each Borough

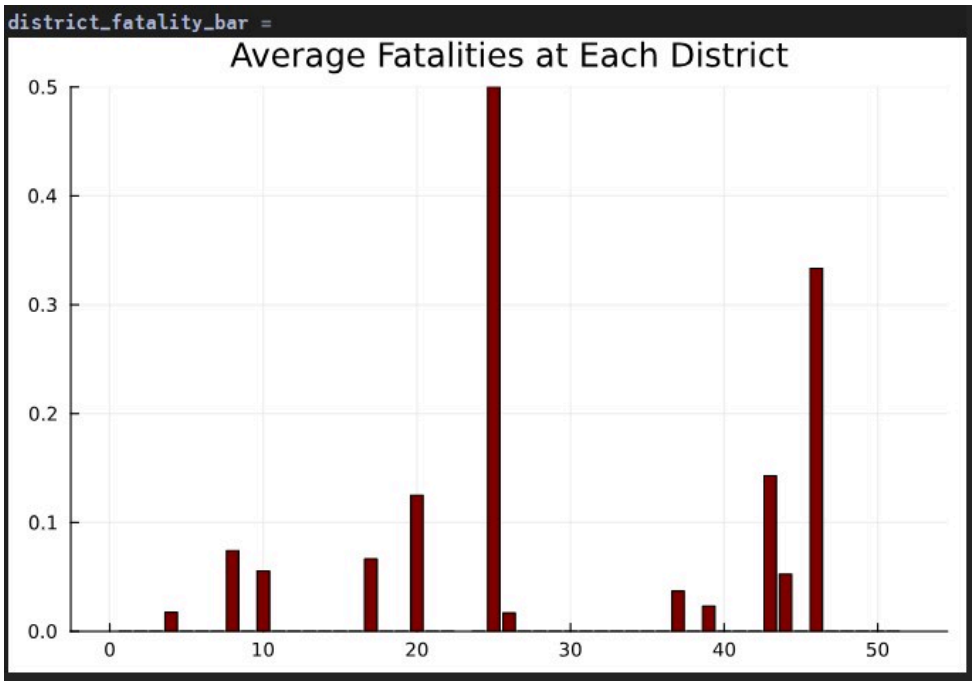

Figure 4: Average Fatalities at Each District

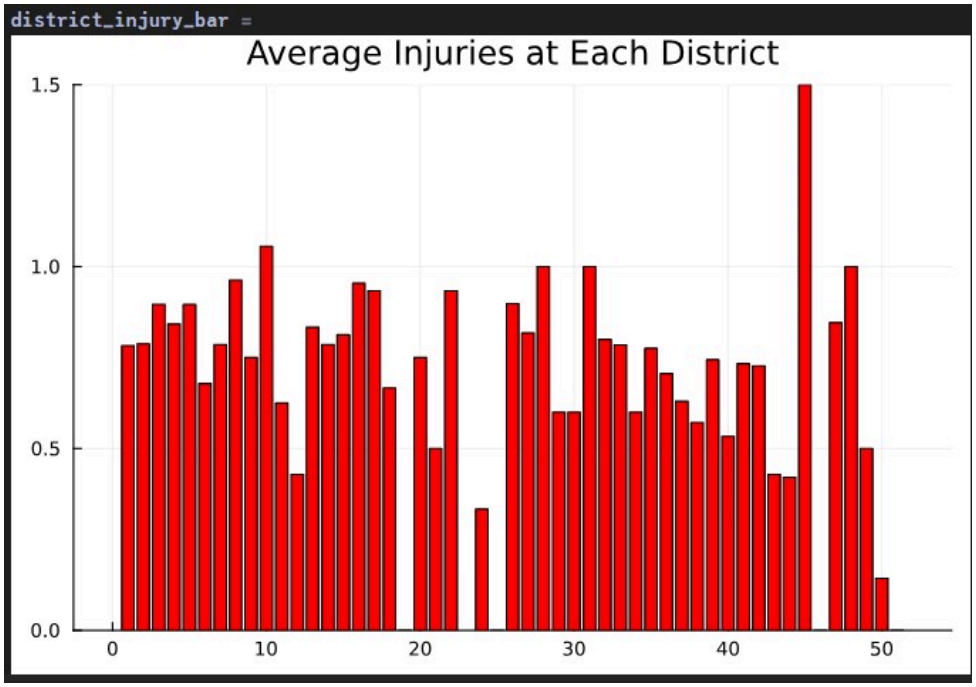

Figure 5: Average Injuries at Each District

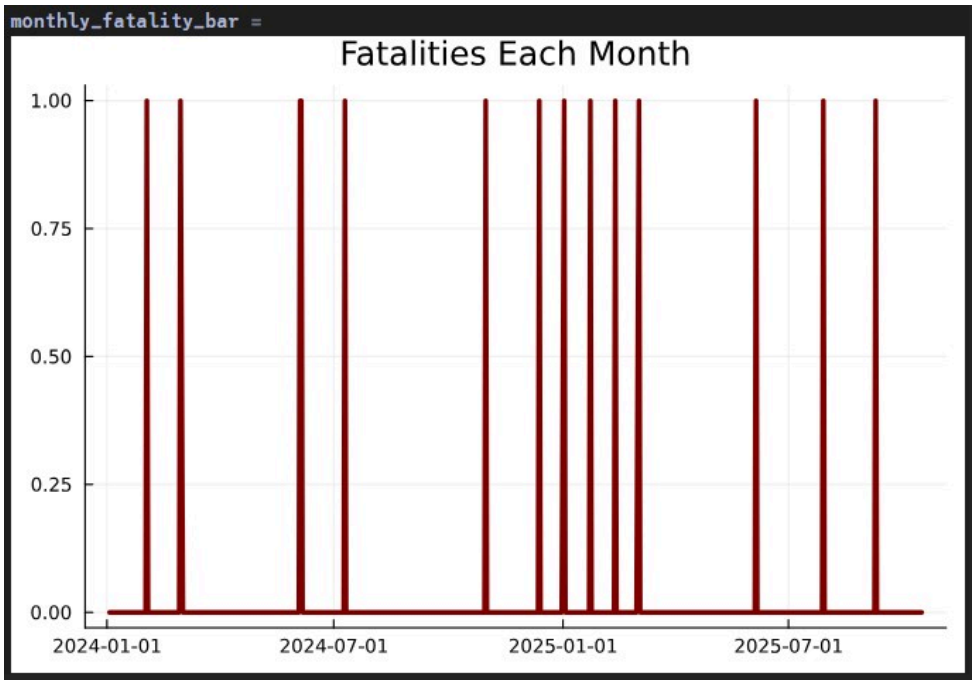

Figure 6: Cumulative Fatalities Overtime

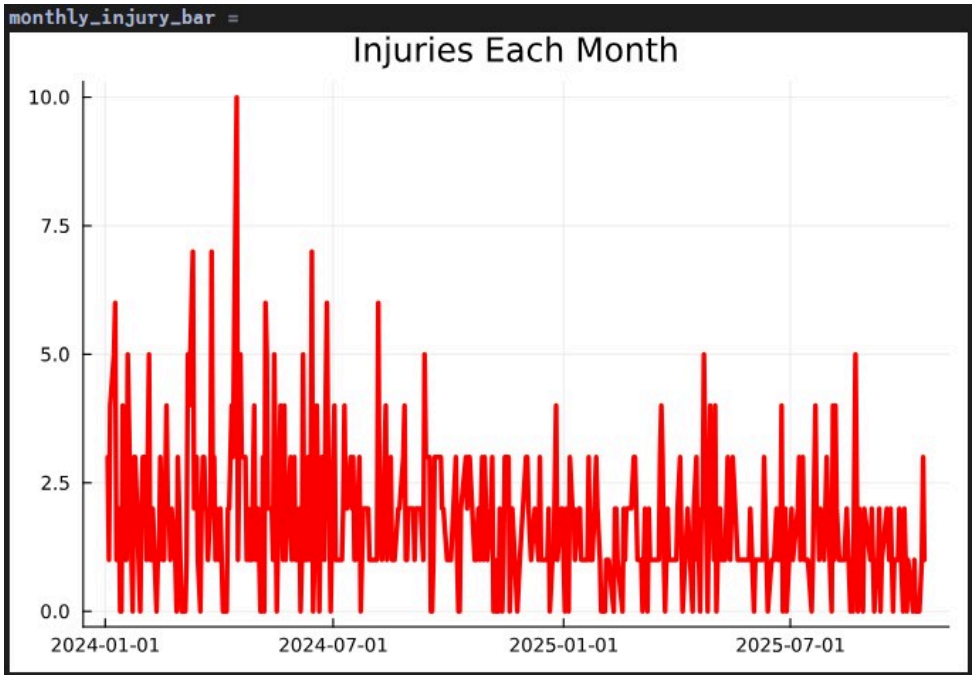

Figure 7: Cumulative Injuries Overtime

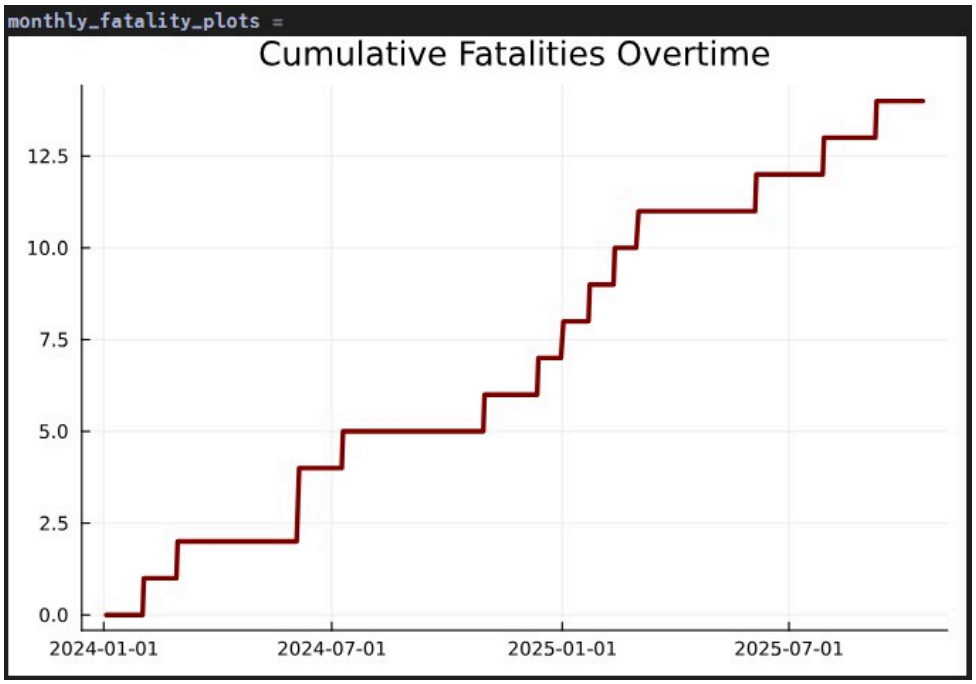

Figure 8: Fatalities Each Month

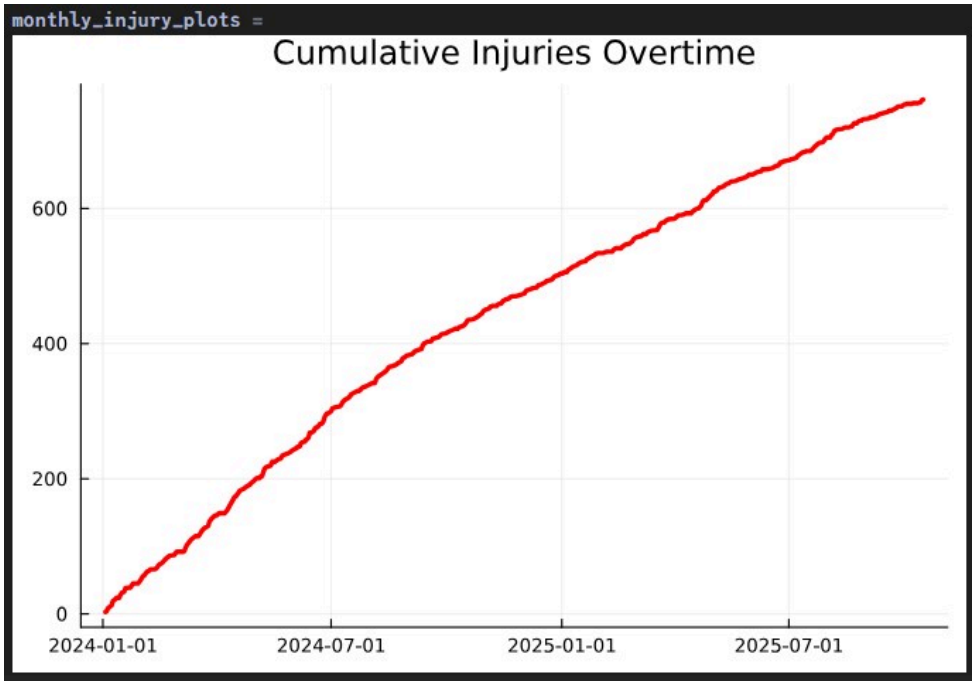

Figure 9: Fatalities Each Month

2.5.1 Averaging the Data

Table 4. Average Incident and Injury Rates by Borough

Borough	AvgFatality	AvgInjury	AvgIncident	FatalityRate%	InjuryRate%
Bronx	0.019	1.10	1.31	1.47	83.82

On average, 83.8% of incidents resulted in at least one reported injury, whereas fatalities were rare (around 1.5% of all cases). The Bronx recorded the highest injury rate, followed closely by Brooklyn.

2.6 Discussion

This section synthesizes the primary findings from our exploratory data analysis. We aggregated the data to compute and examine key descriptive statistics. Specifically, we calculated the average number of fatalities, average number of injuries, and average incident counts for each of the five boroughs. From this, we also derived the fatality and injury rates (as percentages) per borough to better understand the proportional risk. Furthermore, our summary includes an analysis of temporal patterns. We investigated monthly trends by charting the frequency and cumulative totals of both fatalities and injuries over the study period. These initial summaries provide a foundational understanding of which areas are most affected and how incident severity fluctuates over time.

3.Methodology

3.1 Preliminary Predictive Modeling & Model Limitation

3.1.1 Result figures and Explaination

At the begining we try to use several traditional prediction model in order to figure out limitation about the data of the model.

3.1.2 Poisson Model (Injury)

Table 6. Poisson Regression Model Results for Injury Counts

Variable	coef	std err	z	P>|z|	0.025	0.975
Intercept	0.1547	1.059	0.146	0.884	−1.921	2.230

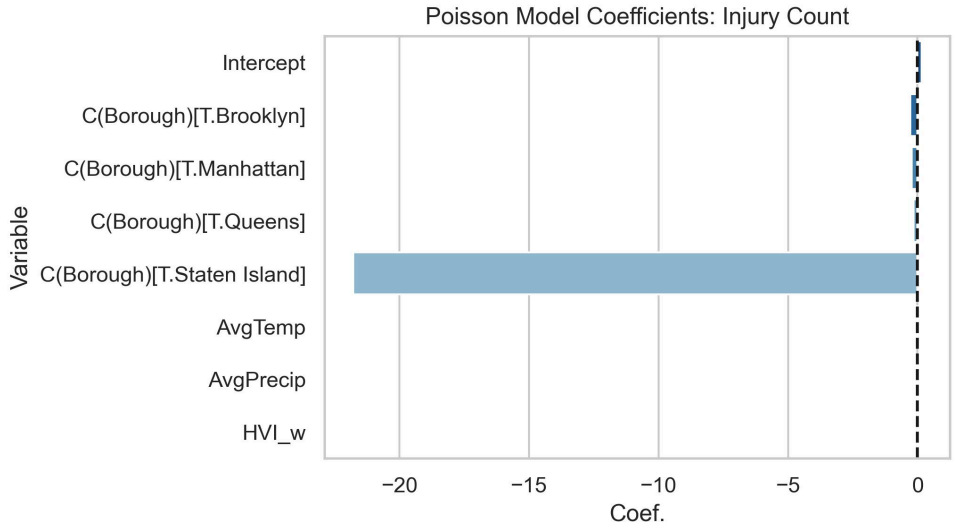

Figure 10: Poisson injury model coefficients

3.1.3 Negative Binomial Model (Fatality)

Table 7. Negative Binomial Regression Model Results for Fatalities

Variable	coef	std err	z	P>|z|	0.025	0.975
Intercept	−9.6771	10.237	−0.945	0.345	−29.742	10.387

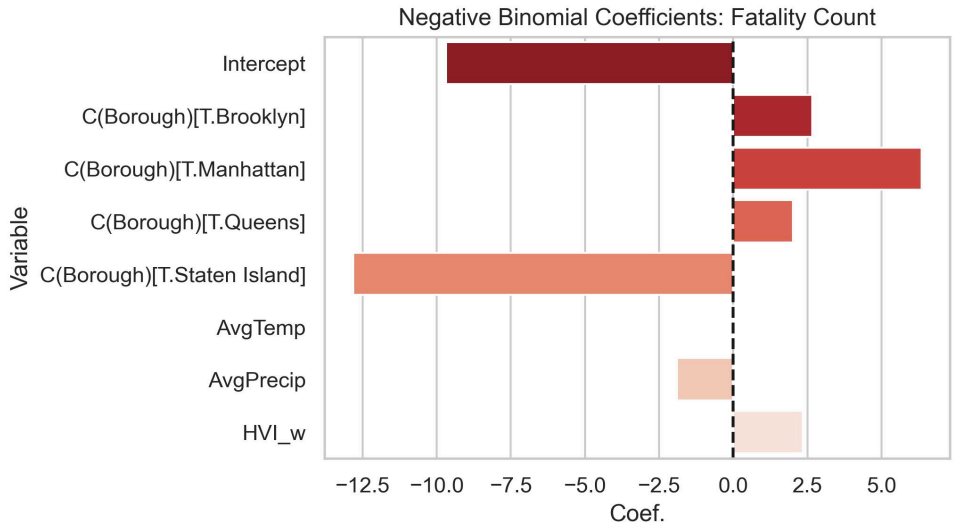

Figure 11: Negative binomial fatality model coefficients

3.1.4 Logistic Model[10]

Table 8. Logistic Regression Results for Binary Fatality Events

Variable	coef	std err	z	P>|z|	0.025	0.975
Intercept	−8.1713	8.01e+06	−1e-06	1.000	−1.57e+07	1.57e+07

3.1.5 Visualization

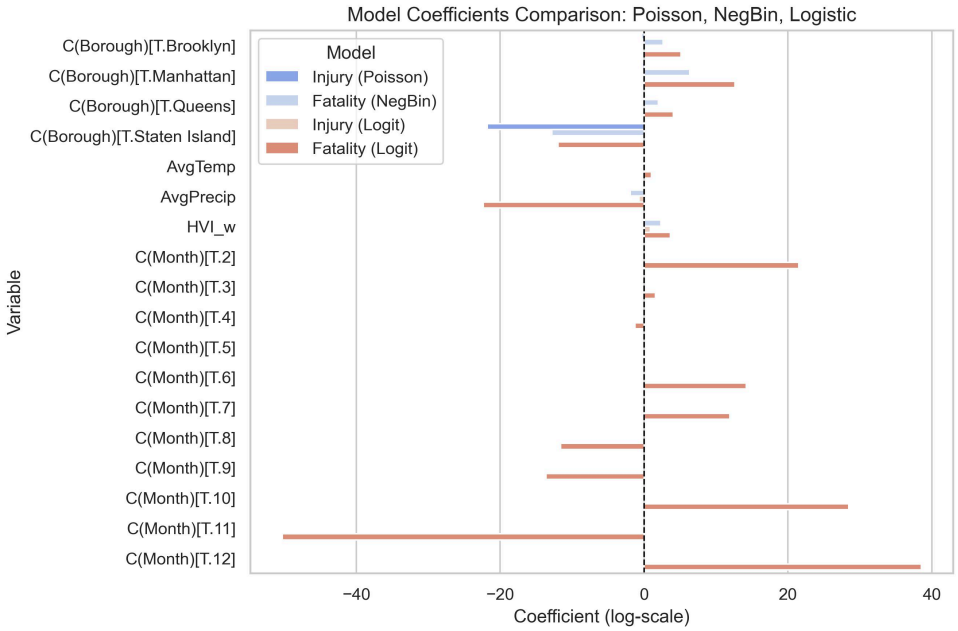

Figure 12: Coefficient comparison

3.1.6 Disccusion the Limitation of Data for Preliminary Regression Model

In this study, we selected Poisson, Negative Binomial, and Logit models based on the following considerations. First, the dataset contains a substantial number of zeros, resulting in pronounced sparsity. Second, although the injury variable takes non-zero values, the fatality variable appears only as 0 or 1 throughout the dataset. Given these characteristics, we employ a Poisson regression to

describe the relationship between injury counts and the relevant covariates. To address the rarity of fatality events—which may induce over-dispersion—we further introduce a Negative Binomial model[8], thereby relaxing the restrictive assumption that the variance must equal the mean. [9]In addition, because fatality is inherently a binary outcome, it naturally aligns with a logit regression, allowing us to model the occurrence of a fatal event directly as a 0/1 response. The purpose of this set of preliminary regressions is to diagnose the sparsity and information content of the data, which in turn guides the selection of our downstream modeling tasks. Based on these diagnostic results, our subsequent experiments place greater emphasis on classification, as it is better suited to the underlying data structure.

Poisson Model

In Fig.10, we observed that the significance levels of several parameters—such as the borough dummy variables, meteorological indicators, and the HVI—are generally weak. Some coefficients even exhibit numerical irregularities, for example, in Staten Island, accompanied by notably wide confidence intervals. Although correlation mapping suggests strong pairwise correlations, these signals must be interpreted with caution due to several limitations. First, the correlation map is inherently dominated by linear relationships. Second, the model does not capture how the variables depend on one another. For example, the HVI has a strong correlation with the outcome, but it is also very tightly linked to the borough variable. This suggests that the apparent effect of the HVI is mostly reflecting borough-specific patterns rather than the index itself. Because of this, it is not surprising that many predictors fail to reach statistical significance in the Poisson or other count-based models. The limited number of observations from Staten Island also adds to the overdispersion problem, which in turn makes the coefficient estimates less stable. As a result, when adopting Poisson or other count-distribution models, it is not surprising that many predictors are not statistically significant. In addition, the sparsity of observations from Staten Island directly contributes to overdispersion, further affecting coefficient stability.

Negative Binomial Model

In Fig.11, although the Negative Binomial model is, in principle, more suitable for handling sparsity, the results still appear unreasonable. The intercept is much larger than expected, and both the z-values and p-values show little statistical significance. Staten Island, as noted earlier, remains the main source of sparsity-related distortion. Even with the relaxed variance assumption, the model does not gain meaningful explanatory power or demonstrate real learning capability. In short, the combination of extremely rare events and an imbalanced, irregular data structure prevents the Negative Binomial model from fitting the outcome in any convincing way.

Three Model Comparison

In Fig.12, for both the Poisson and Negative Binomial models, most coefficients cluster tightly around zero with very limited variation. In contrast, the Logit model produces several abnormal coefficients, especially for borough indicators with sparse observations and for certain months. The magnitudes are far beyond those in the other two models (particularly when compared to the Negative Binomial), implying that fatality behaves even less predictably under the logit specification.

This kind of “coefficient blow-up” in the Logit model usually signals quasi-complete separation: within specific borough–month combinations, fatal events either almost never occur or never occur

at all. In our case, it is mostly the “almost never” situation. When that happens, logistic regression tends to push the associated coefficients toward ±∞ in an attempt to fit those extreme patterns. This indicates that fatality as a 0/1 outcome suffers from even stronger sparsity and imbalance than when treated as a count, reinforcing the idea that fatal events resemble a rare-event classification problem rather than a conventional regression target.

Summary

Taken together, the sparsity and irregular structure of our data make standard regression models unsuitable unless additional, more informative predictors are introduced—such as the new parameters incorporated in Section 7.

3.2 K-Means Classification Models Methodology

3.2.1 Data Preparation and Cleaning

To create this model, four features from the dataset were implemented (longitude, latitude, number of injuries, and boroughs). This dataset will analyze four out of the five boroughs which are Manhattan, Brooklyn, Queens, & Bronx. The fifth borough, Staten Island, will be excluded from the data set due to its lack of available data.

For easier visualization and to prevent data overlap, longitude and latitude were rounded to three significant figures. Since the injury values often return as 0 or 1 within the dataset, they were grouped by the rounded longitude and latitude location and then summed for further data cleaning before plotting. The data was then grouped into boroughs and then plotted all together in one plot.

The dataset would then be implemented into Julia and then these variables would be separated out to create the plots.

3.2.2 Model Architecture

The architecture for this model is simple with the features being the four independent variables along with the specific city locations, which is an output of this dataset. Note that the names of the cities were not a part of the dataset but can mainly be derived from approximating or interpolating the locations from the longitude and latitude.

Input Selections: Longitude, latitude, injuries, and boroughs

Output Selection: Cities/counties with the highest concentration

3.2.3 Weighted K-Means Function

To characterize the data better, a weighted k-means function was implemented to create weighted centroids for each borough. First, four additional functions had to be created before doing the k-means function (init_centroids, calc_distances, calc_groups, and update_centroids!). After these four functions resulted in a working weighted k_means function, rows with missing data values had to be dropped to further clean the data. After variables were created for the point locations and the weights (in terms of injury count), the centroids were plotted and implemented into the plot.

3.2.4 Model Evaluation

The plots are in the form of a scatter map showing the four cluster groups representing four of the boroughs in New York City. These cluster groups also closely represent how the actual boroughs are oriented within New York City, which allows for accurate data visualization. The second plot shows the additional weighted centroids in the form of stars that came from the weighted k-means function.

3.3 Decision Tree Classification Models Methodology

3.3.1 Data Preparation

For this model, three features from the data set where implemented (borough, check description, injuries). Similarly to the k-means methodology, Staten Island will also be excluded from this dataset. The check descriptions list the different reasons of these construction incidents that occurred. This model's check descriptions will contain "Worker Fell", "Mechanical Construction Equipment", "Material Failure", "Scaffold/Shoring Installations", "Excavation/Soil Work". There was an additional check description in the dataset called, "Other Construction", but that would be excluded because of the ambiguous nature of this description.

3.3.2 Model Architecture

The model will contain two independent variables and one dependent variable.

Input Selections: Boroughs and check descriptions

Output Selection: Injuries (number and percentage)

There will be five decision trees in total, with four of them displaying the results from each borough and the last one displaying the overall results across the four boroughs from the dataset. The three components of a typical decision tree are the root nodes, branches, internal nodes, and leaf nodes.

Root Node: Borough

Internal Nodes: Check Description

Leaf Nodes: Injuries

There will be three leaf nodes per each internal node, characterizing where injuries would be more than one, equal to one, or less than one (zero) for each incident. The injuries' numbers and percentages would then be displayed for each of these leaf nodes.

3.3.3 Classification & Grouping of Data

The data would be grouped by borough and analyzed using Microsoft Excel. Excel's functions would be used to calculate the sums and percentages for the injuries. The actual models would then be created using Microsoft PowerPoint.

3.3.4 Model Evaluation

The model will show the structure of the model as described in the Model Architecture section, with the nodes appearing as rectangles and the branches appearing as arrows.

3.4 Neural Network Classification Models Methodology

3.4.1 Data Preparation

To enhance model performance, two additional parameters were integrated:

NoncompliantCount, representing the frequency of non-compliant behaviors , [16]

IssueNumber, representing the volume of active construction projects in a specific area and month, allowing for temporal lags[17]

Five input features were selected from the final dataset **(df_final)**: Average Temperature **(AvgTemp)**, Average Precipitation **(AvgPrecip)**, Weighted Heat Vulnerability Index **(HVI_w)**, NoncompliantCount, and IssueNumber. The target variable, Injury, was binarized: samples with an injury count greater than zero were labeled as "Injury Occurred" (1), while others were labeled as 0. To address dimensional discrepancies, a **StandardScaler** was applied to standardize all input features. The dataset was randomly partitioned into a training set (80%) and a validation set (20%).

3.4.2 Model Architecture

A three-layer Feedforward Neural Network (FNN) was adopted as the predictive model. The architecture is defined as follows:

Input Layer: Corresponds to the five input features.

First Hidden Layer: 16 neurons utilizing the ReLU activation function.

Dropout Layer: Applied with a rate of 0.1 to mitigate overfitting.

Second Hidden Layer: 8 neurons utilizing the ReLU activation function.

Output Layer: A single neuron outputting unnormalized logit values, which are transformed into injury probabilities via a Sigmoid function.

This structure provides the necessary nonlinear expressive capacity[12] to capture complex interactions among the multivariate inputs.

3.4.3 Loss Function and Optimization

Given the significant class imbalance (where "No Injury" cases far exceed "Injury" cases), the model utilizes a Weighted Binary Cross-Entropy with Logits Loss function. A positive weight, calculated as $\text{pos}_{\text{weight}} = \frac{N_{\text{neg}}}{N_{\text{pos}}}$, is automatically applied to balance the classes. The Adam optimizer[14], with a learning rate of $1 * 10^{-4}$, is employed to update network parameters and minimize the loss function during each iteration.

3.4.4 Training and Validation

The model was trained for 300 epochs. Loss and Accuracy for both training and validation sets were calculated in each epoch to monitor convergence trends. Dropout was active during training to enhance generalization. Key metrics were logged every 50 epochs. Finally, training/validation loss curves and validation accuracy curves were plotted to assess the stability of model convergence.

3.4.5 Model Evaluation

ROC Curve & AUC: The Receiver Operating Characteristic (ROC) curve was plotted[15] using validation results, and the Area Under the Curve (AUC) was calculated to quantify overall

classification performance. Youden's J statistic (TPR — FPR) was utilized to determine the optimal classification threshold.

Confusion Matrix: Matrices were generated for both the default threshold (0.5) and the optimal threshold to visualize classification accuracy, false positive rates, and false negative rates.

Precision-Recall-F1 Analysis: Precision, Recall, and F1 scores were calculated across a threshold range of [0.1, 0.9] with a step size of 0.05. Curves were plotted to evaluate trade-offs under different judgment criteria, identifying the threshold that maximizes the F1 score.

3.5 Neural Network Regression Models Methodology

3.5.1 Data Preparation and Cleaning

This study employs an improved Neural Network Regression model to predict the count of construction-related injuries. To enhance prediction stability and robustness, the model incorporates mechanisms for data denoising, standardization, and nonlinear feature extraction. **Missing Values**: Missing values in the Injury column were filled with zero and converted to floating-point format.

Feature Engineering: The Month variable was extracted from YearMonth, and Borough was processed using one-hot encoding.

Denoising: Extreme values (outside the 1st and 99th percentiles) were removed for Temperature, Precipitation, HVI, Noncompliant Count, Issue Number, and Injury counts. Samples exhibiting concurrent extreme heat and precipitation were excluded, as were records with negligible construction activity (low IssueNumber). HVI values were capped within a reasonable upper limit.

Log Smoothing: Logarithmic smoothing was applied to high-variance features (Noncompliant Count, Issue Number, Precipitation) to prevent dominance by single variables.

Due to the inherent sparsity of the data, additional regularization terms were omitted to avoid further underfitting or gradient convergence issues.

3.5.2 Feature Standardization

Input features comprised Average Temperature, Average Precipitation, Heat Vulnerability Index, Noncompliant Count, Issue Number, Month, and borough encoding columns. All input variables were standardized. To ensure numerical stability and facilitate gradient convergence, the target variable **(Injury)** was normalized using its mean and standard deviation. The dataset was subsequently partitioned into an 80% training set and a 20% validation set.

3.5.3 Model Architecture

A Neural Network model named InjuryRegressor was defined with a multi-layer nonlinear structure:

Input Layer: Corresponds to all processed input features.

First Hidden Layer: 32 neurons using the LeakyReLU activation function.

Dropout Layer: Rate of 0.1, used to prevent overfitting.

Second Hidden Layer: 16 neurons using the LeakyReLU activation function.

Third Hidden Layer: 8 neurons using the LeakyReLU activation function.

Output Layer: Single neuron outputting the predicted injury count.

LeakyReLU was selected because it maintains non-zero gradients in the negative interval, avoiding the vanishing gradient problem, which is particularly suitable for regression tasks involving sparse data.

In this process, we did not adopt feature-function regularization or similar structures, essentially because the model was not even capable of overfitting — it could not fully learn the patterns in the first place.

3.5.4 Loss Function and Optimizer

The model uses Mean Squared Error (MSE) as the loss function. The Adam optimizer was selected with a learning rate of $3 * 10^{-4}$, balancing convergence speed and stability through automatic learning rate adjustment. Training and validation losses were recorded to monitor convergence trends and generalization performance. Note: Traditional nonlinear count models (e.g., Poisson and Negative Binomial) were tested but excluded due to convergence failures during training.

3.5.5 Training and Validation

The following metrics were calculated using validation predictions:

R^2 (Coefficient of Determination): Measures the proportion of variance explained by the model. An $R^2 < 0$ indicates the model failed to learn effectively.

RMSE (Root Mean Square Error): Reflects the average magnitude of prediction error.

MAE (Mean Absolute Error): Measures the average deviation between predicted and actual values.

Additionally, scatter plots of predicted vs. actual values were generated to assess fit; a point cloud clustering near the diagonal indicates good predictive performance.

3.5.6 Model Improvements

Approach 1: Hybrid Lag and Group Bias Linear Model

Concept: Incorporates time-lag features and borough-specific biases into linear regression to capture temporal inertia and regional disparities.

Feature Processing: Retained only samples with construction records; applied log smoothing to non-compliant counts, permits, and precipitation; generated one-period lag features by borough and month.

Structure: Includes global linear weights and regional bias terms to reflect baseline risks across boroughs.

Results: While the model demonstrated some capacity to explain regional differences, overall R^2, RMSE, and MAE metrics remained poor, indicating limited fit.

Approach 2: Two-Stage Hybrid Model (No Lag, Strict Denoising)

Concept: Adopts a "Classify-then-Regress" structure to improve stability under sparse data conditions.

Stage 1 (Classification): Uses a neural network to determine the probability of an injury occurring (Injury > 0).

Stage 2 (Regression): For confirmed injury samples, estimates the actual count using a linear model with borough biases.

Results: The classification stage achieved high accuracy (0.8–0.9), effectively identifying high-risk months. However, while the regression stage showed a slight improvement in R^2 over single-stage models, overall predictive capability remained unsatisfactory due to the limited volume of data.

4 Results and Discussion

4.1 Introduction

In the previous section, the methodology of the four kinds of models was explained. This section will display and discuss the models pertaining to each respective section. The effectiveness of these models as well as their performance will be described, along with a critical analysis of their features.

4.2 K-Means Classification Models

4.2.1 Models

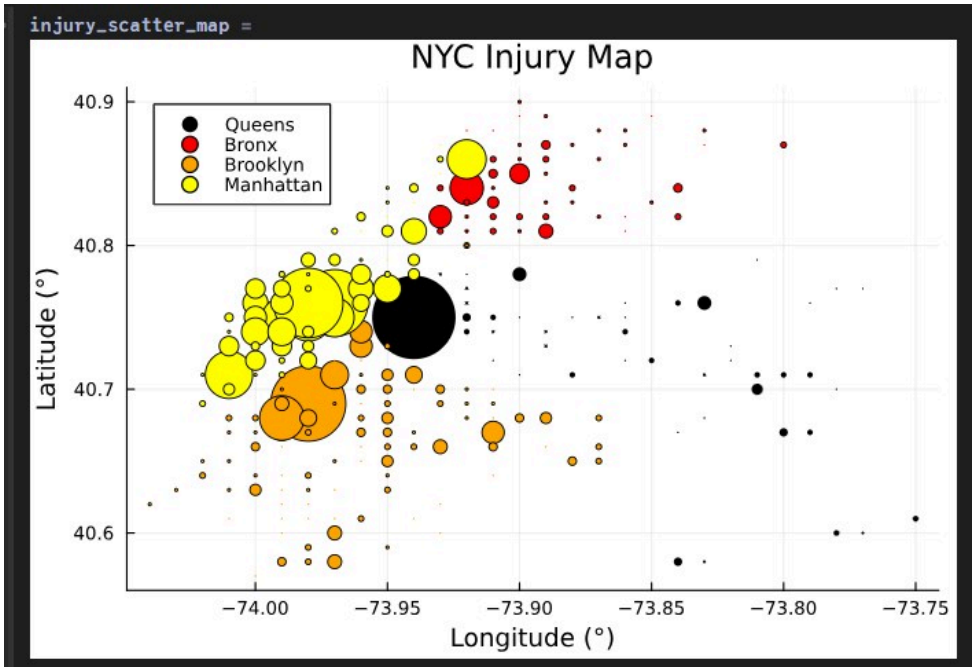

Figure 13: Spatial Distribution of Injuries from Each Borough

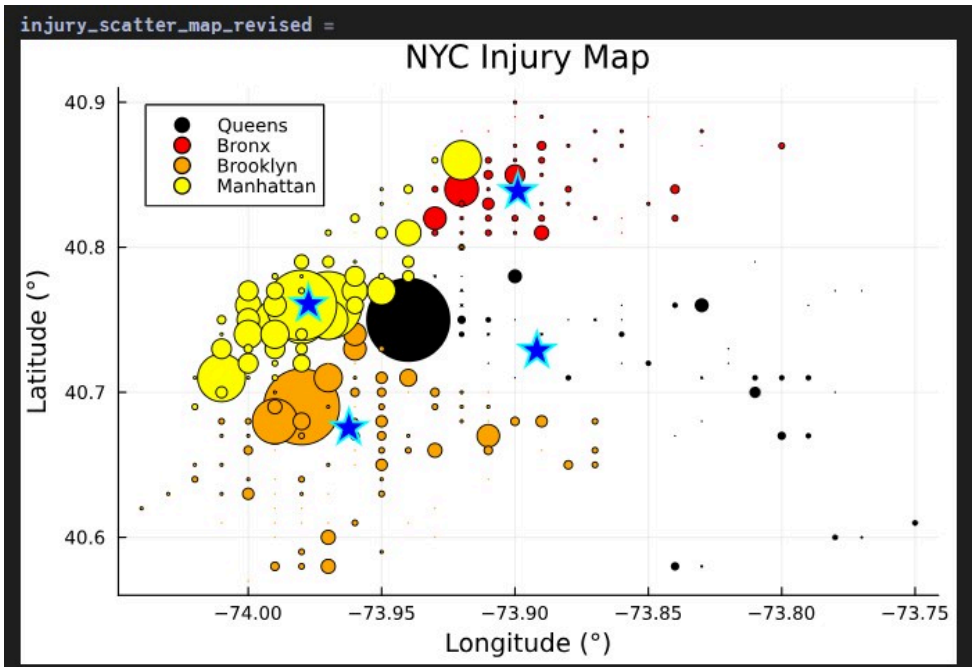

Figure 14: K-Means Model w/ Centroids

4.2.2 Discussion

These spatial distributions show a different concentration of injuries scattered across the four boroughs of New York City. There is a lot of information that can be derived from these two graphs.

The large concentration of injuries grouped together in Manhattan shows where Time Square is. In this densely populated area, it makes sense that there are a lot of construction projects that take place due to the high property value of this area. The high concentration of injuries located in Brooklyn and Queens are also close to Times Square as well, which also demonstrates that these incidents must have taken place within the most populated areas of the city.

From this data, it can be derived that there should be stricter Occupational Safety and Health Administration (OSHA) regulations when it comes to potentially hazardous construction projects within these areas of New York City. Specifically, these areas might also be subdivided into specific districts and these districts can also come up with their own regulations to counteract the possibility of construction fatalities.

Finally, the main role of these stars, which show the weighted k-means centroids of the data is to better characterize and understand the data, especially with the incidents still piling on each other even with the preliminary amount of data cleaning. For example, the star that corresponds to Manhattan shows that the concentration of injuries mainly happens around Times Square. The plots above show a relatively linear relationship of injuries overtime, which shows that this accurately predicts that more construction projects in this area will likely lead to more injuries. The same applies to Brooklyn, where the star is located in the area where it is more densely populated (closer to Times Square).

However, the centroids in Queens and Bronx do not give a lot of information as the injuries were more spatially distributed throughout the boroughs rather than being concentrated in one location. Some boroughs are bigger than others, so this demonstrates the primary shortcomings of this model. What can also be derived is the bias in terms of the size of each borough. The main reason Manhattan has the most incidents is most likely because it is the most populated area of New York

City. This would also demonstrate that there would be more construction projects in an area like that, and that additional data regarding the number of construction projects at each borough over time would also be needed to make accurate predictions of this data.

4.3 Decision Tree Classification Models

4.3.1 Models

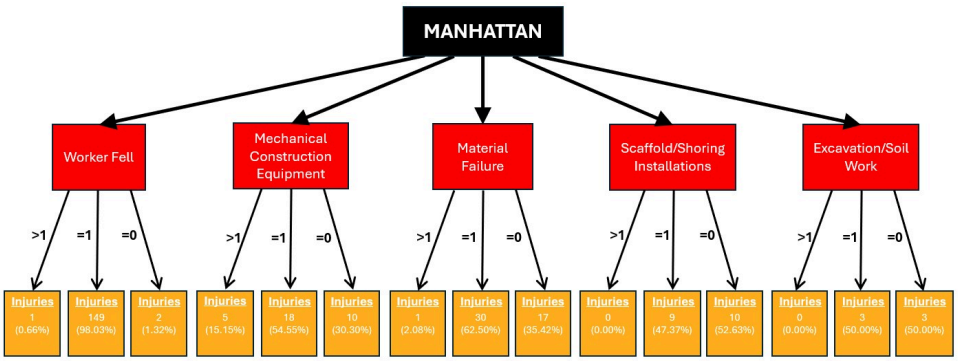

Figure 15: Manhattan Injury Classification Tree

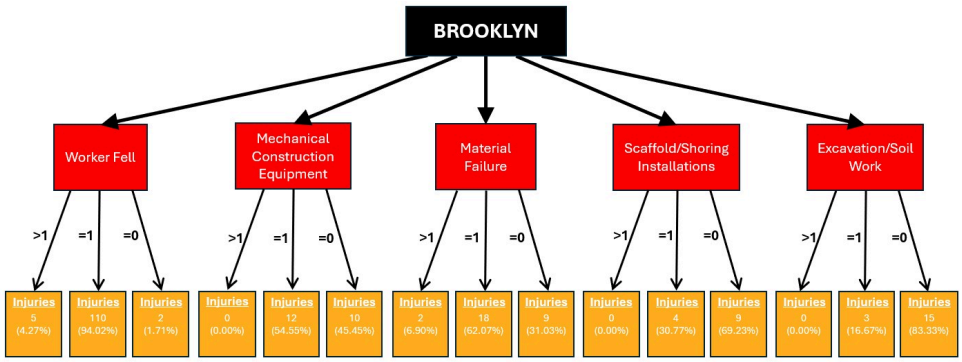

Figure 16: Brooklyn Injury Classification Tree

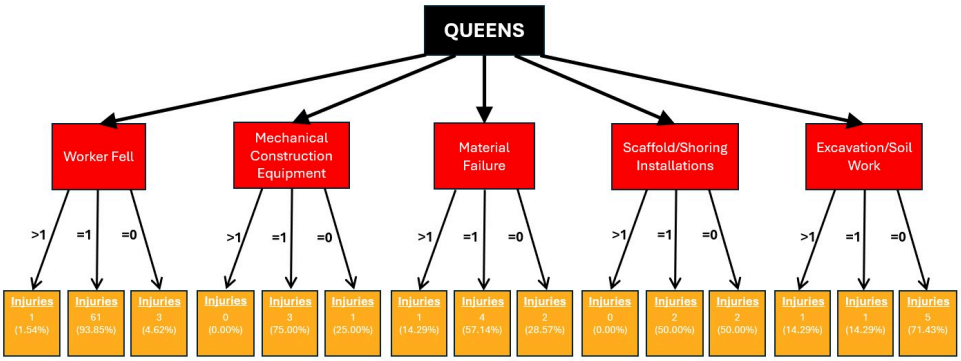

Figure 17: Queens Injury Classification Tree

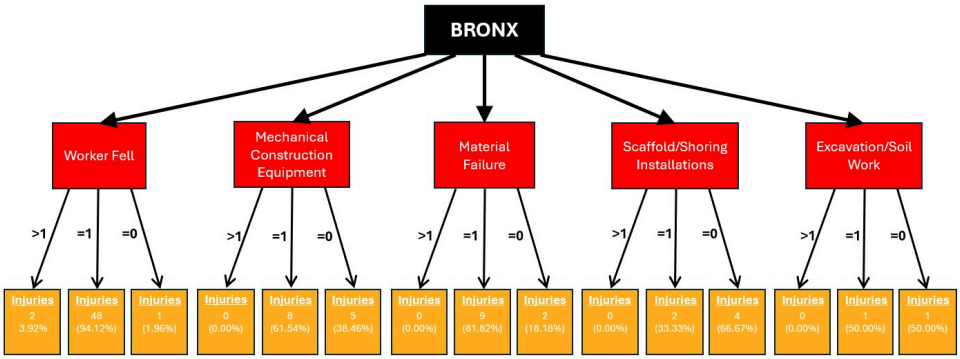

Figure 18: Bronx Injury Classification Tree

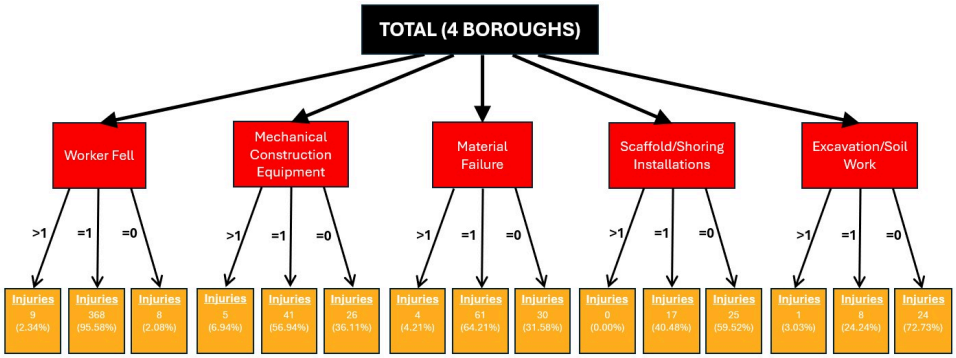

Figure 19: Four Boroughs Total Injury Classification Tree

4.3.2 Discussion

Model Interpretability and Structure: Unlike the Neural Network models discussed in Section 4.4, which provide probabilistic outputs, the Decision Tree Classification models offer a transparent, white-box visualization of risk factors. As shown in Figures 15 through 19, the models successfully hierarchized the data, utilizing Incident Type as the primary splitting criterion at the internal nodes. This structural arrangement confirms that while the borough determines the baseline environment, the specific nature of the incident—such as is the most critical determinant of injury severity.

Borough-Specific Risk Pathways: The topology of the decision trees reveals distinct risk profiles across the boroughs. The Manhattan tree (Figure 15) and Brooklyn tree (Figure 16) exhibit more complex branching structures compared to Queens and the Bronx. This complexity likely reflects the higher density and variety of construction projects in these core boroughs, where risks are multifaceted. Specifically, in Manhattan, branches related to **Material Failure** and **Mechanical Construction Equipment** are prominent, suggesting that infrastructure-heavy projects in dense urban environments carry specific equipment-related risks. In contrast, the trees for Queens and the Bronx (Figures 17-18) show a more streamlined structure, where **Worker Fall** remains the dominant predictor of injury outcomes.

Risk Hierarchy and Classification Logic: A critical insight from the leaf nodes is the identification of high-probability injury pathways. Across the aggregate model (Figure 19), incidents categorized as **Worker Fall** and **Scaffold/Shoring Installations** consistently lead to leaf nodes associated with injury occurrences. This aligns with the finding in Section 4.2 that certain incident types are inherently more dangerous regardless of location. The decision trees effectively function as a rule-based classifier: if an accident involves a fall or excavation, the probability of it being an injury-causing event is statistically maximized.

Operational Implications: While the Neural Network provided higher predictive accuracy through non-linear feature combination, the Decision Trees provide actionable safety rules. The **If-Then** logic derived from these trees supports the development of targeted regulatory checklists. This complements the **Conservative Prediction Strategy** discussed in Section 4.4.2 by providing clear interpretability for on-site safety officers, allowing them to identify high-risk scenarios before they escalate into actual injuries.

4.4 Neural Network Classification Models

4.4.1 Models

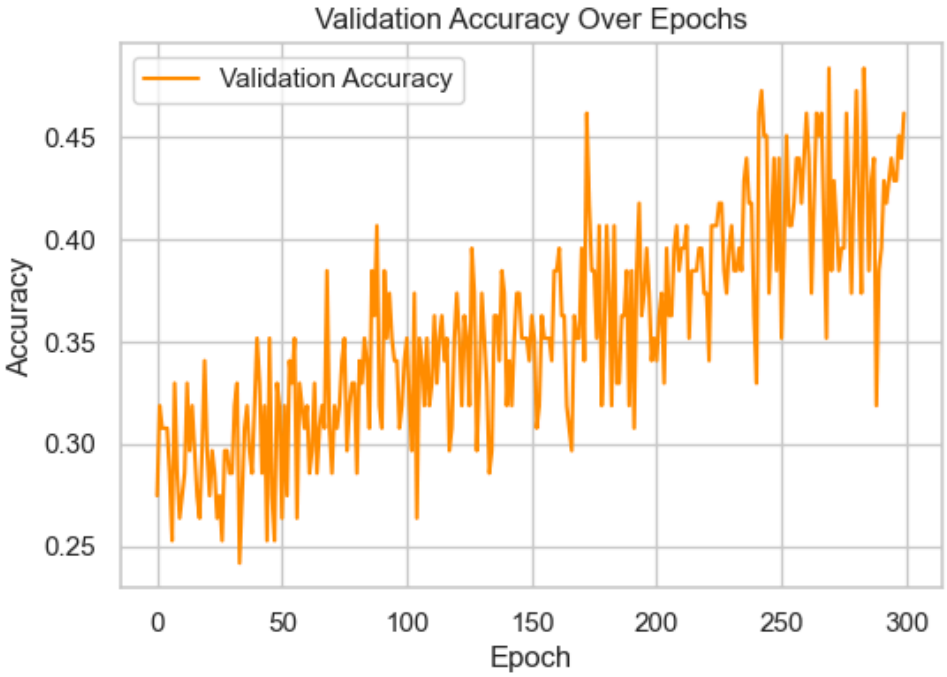

Figure 20: Validation Accuracy Over Training Epochs

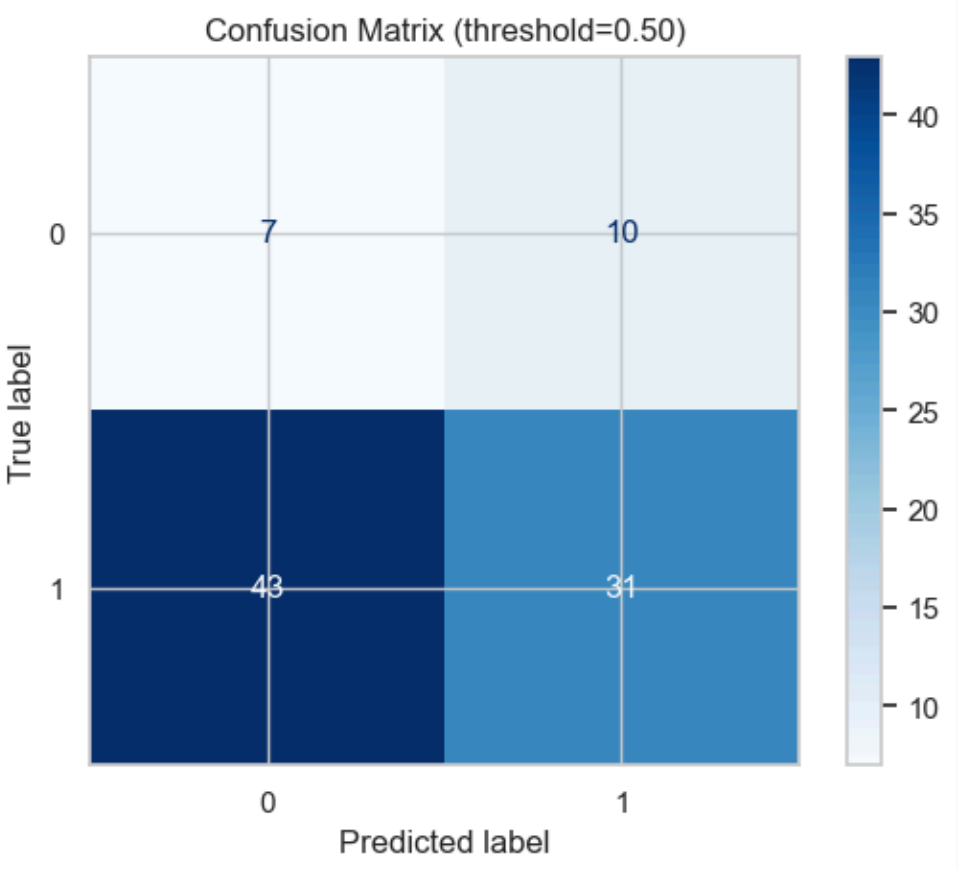

Figure 21: Confusion matrix of the neural network classifier (threshold = 0.50)

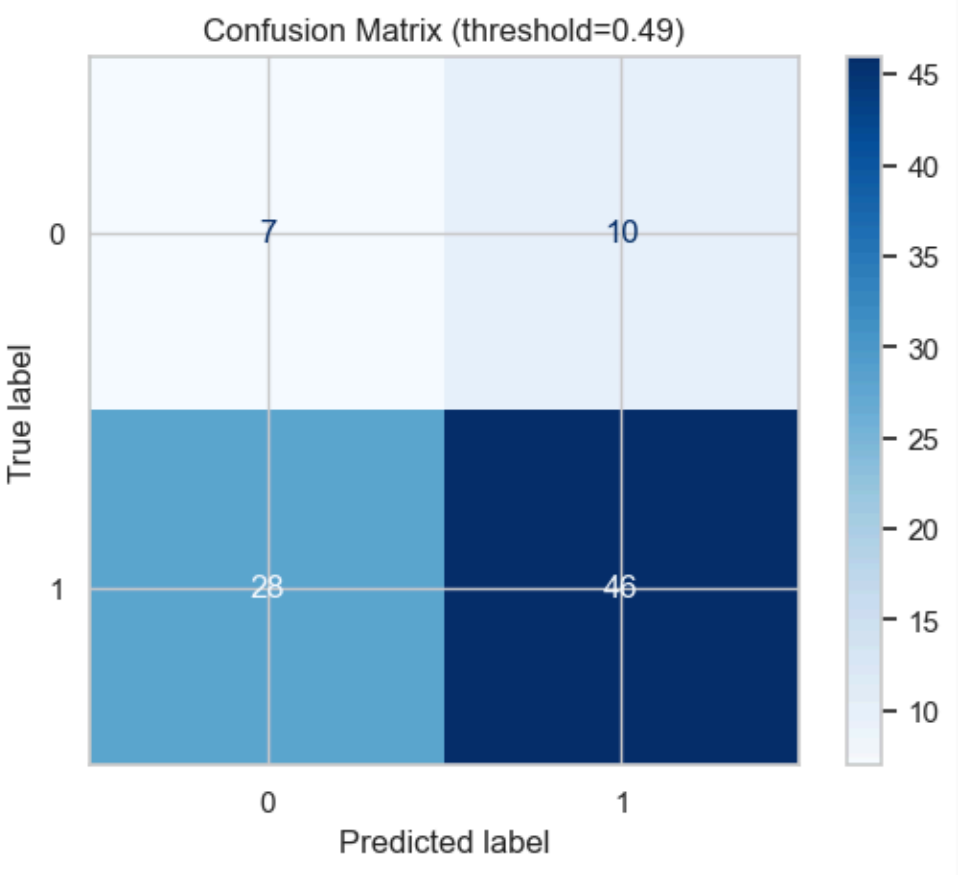

Figure 22: Confusion matrix of the neural network classifier (threshold = 0.49)

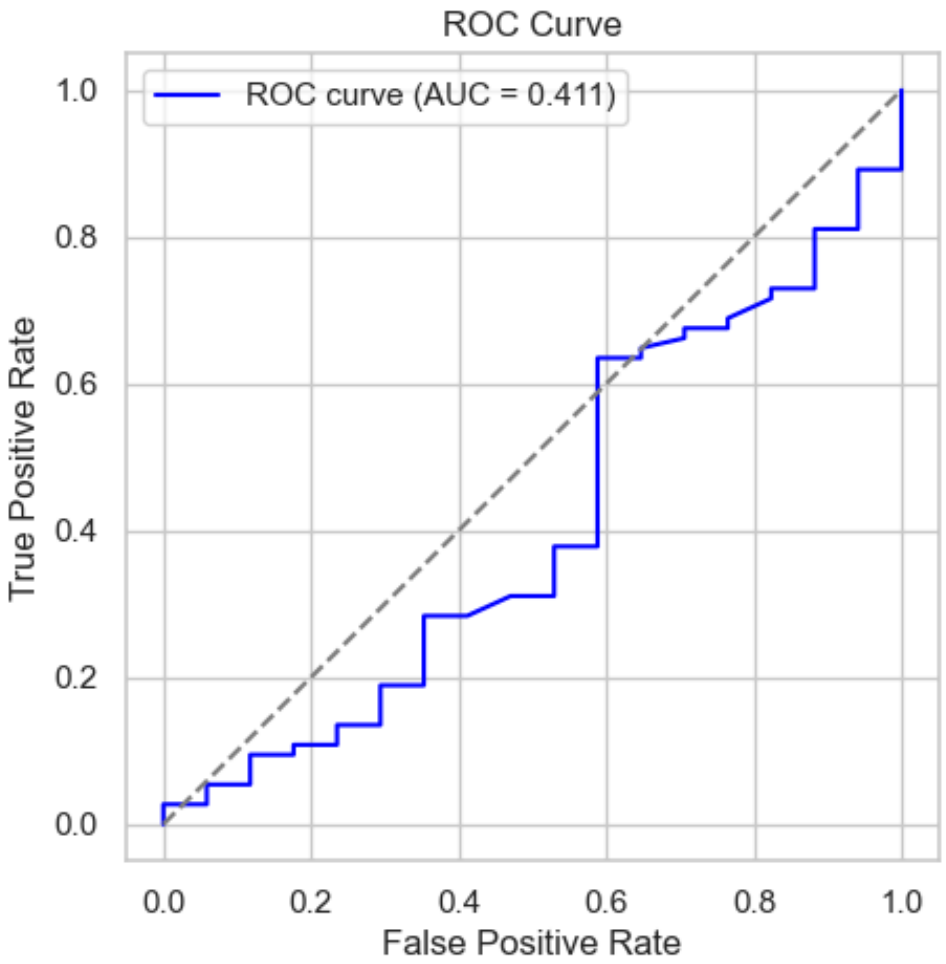

Figure 23: ROC curve with AUC = 0.411

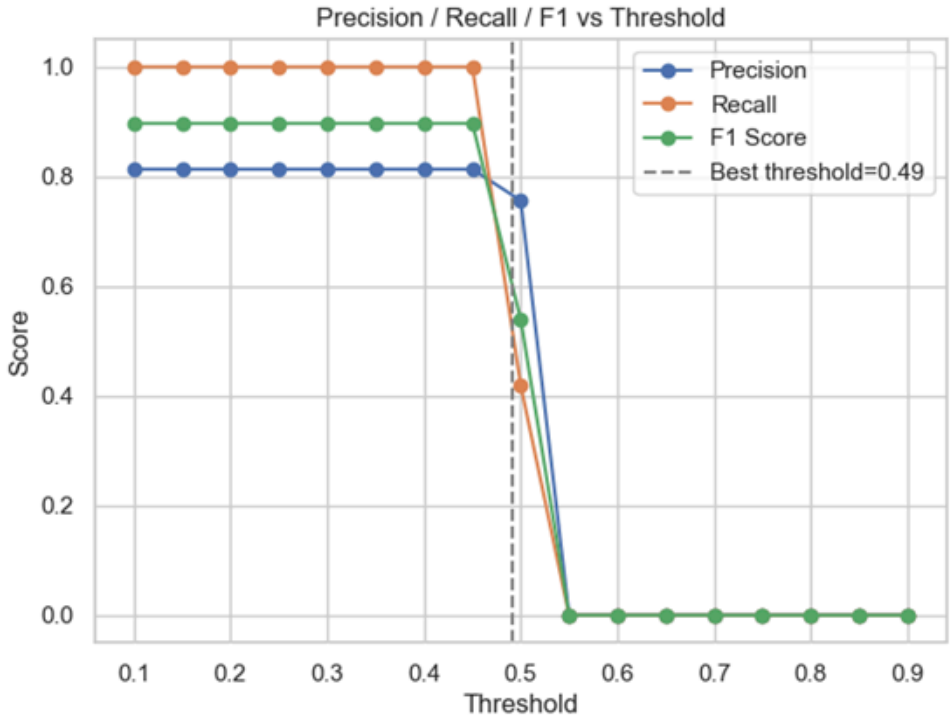

Figure 24: Precision, recall, and F1 score across decision thresholds

4.4.2 Discussion

As illustrated in the results, the validation accuracy exhibited significant fluctuation during the initial training phase but demonstrated a steady upward trend overall, rising from approximately 0.28 to nearly 0.45. This indicates that the model progressively learned effective relationships between features, leading to improved validation performance. While there is room for further accuracy improvement, the absence of significant overfitting suggests that the network architecture and regularization settings (Dropout=0.1) are reasonable and provide good generalization capability.

At the default threshold of 0.50, the model's identification of "Injury" (positive class) showed high recall but slightly lower precision. The confusion matrix results are as follows:

True Positives (TP) = 31 (Correctly identified injuries)

False Positives (FP) = 10 (Non-injuries incorrectly predicted as injuries)

True Negatives (TN) = 7

False Negatives (FN) = 43

These results suggest a conservative prediction strategy (preferring false alarms over missed detections). In the context of accident analysis, this bias is acceptable, as false negatives (missed injury predictions) typically carry a higher safety cost than false positives.

Applying the optimal threshold of 0.49, determined by Youden's J statistic, significantly improved the model's recognition capability:

TP increased to 46, and **FN** decreased to 28.

TN remained at 7, with a slight increase in **FP** to 10.

This adjustment achieved a better balance, enhancing overall classification accuracy while maintaining high recall. The significant reduction in missed detections (FN) compared to the default threshold highlights that threshold optimization is a critical step in tasks involving imbalanced datasets. Analysis of the metrics is defined as follows:

Precision: The proportion of true injuries among predicted injuries. High precision implies high confidence in positive predictions (few false alarms).

Recall: The proportion of actual injuries correctly identified. High recall implies comprehensive coverage of safety risks (few missed incidents).

F1 Score: The harmonic mean of Precision and Recall, providing a balanced metric for imbalanced datasets.

The plotted curves show the relationship between these metrics and the threshold. In the 0.1–0.49 range, all three metrics remain high: Recall stays near 1.0, Precision stabilizes around 0.8, and the F1 score approaches 0.9. However, beyond the 0.5 threshold, all metrics decline rapidly, indicating that an excessively high threshold makes the model overly conservative, resulting in missed positive samples. Consequently, 0.49 was selected as the optimal threshold, achieving an ideal balance between Recall and Precision and maximizing the F1 score.

4.5 Neural Network Regression Models

4.5.1 Models

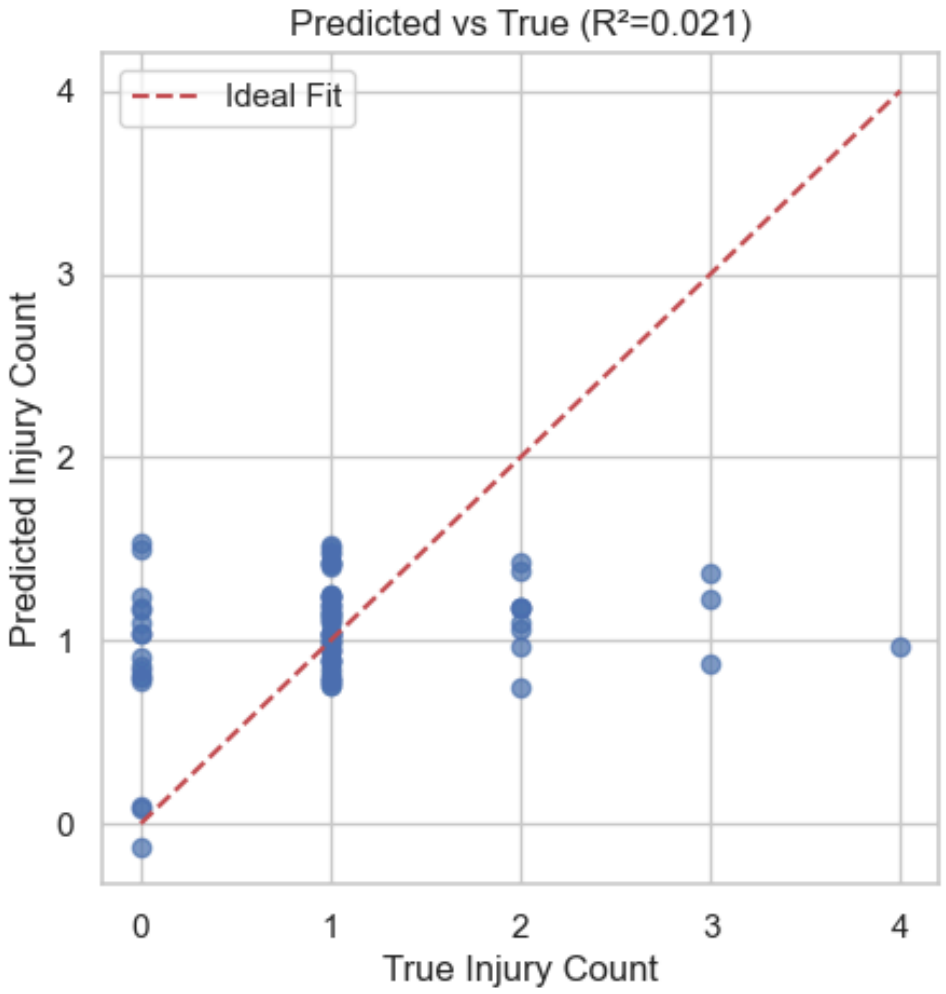

Figure 25: Baseline model performance—predicted vs. true injury counts

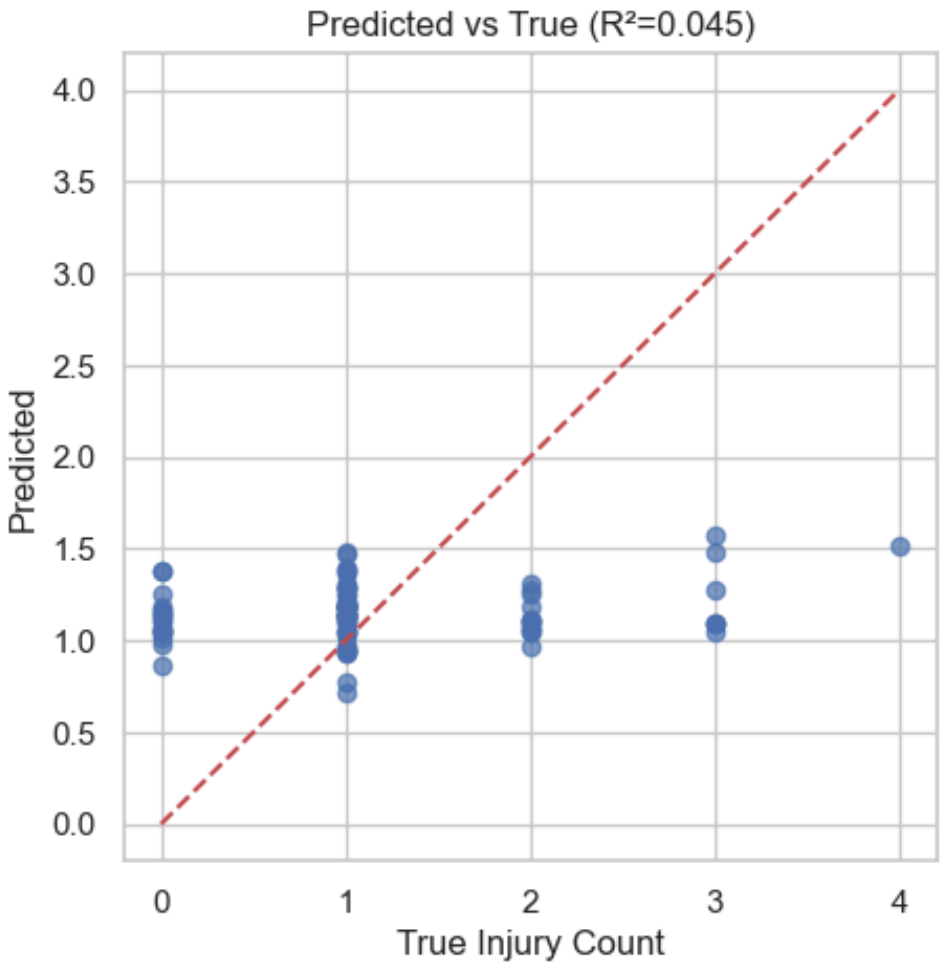

Figure 26: Hybrid Lag and Group Bias Linear Model—predicted vs. true injury counts

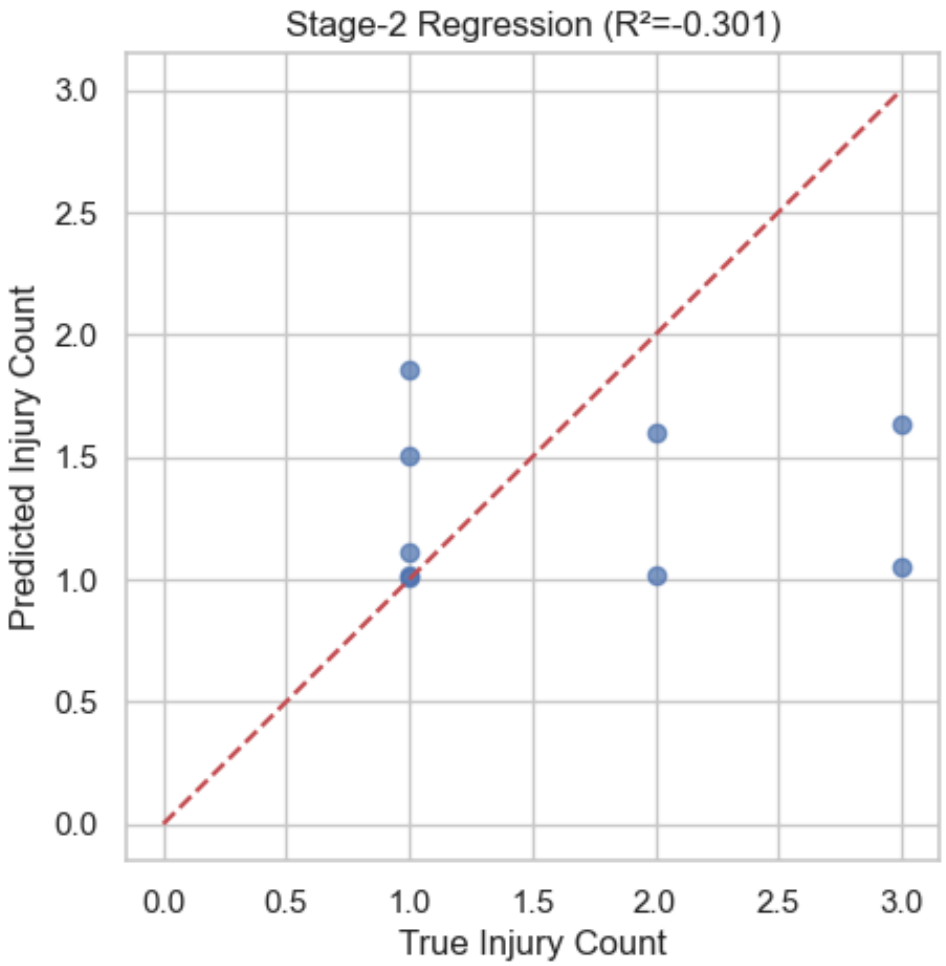

Figure 27: Two-Stage Hybrid Model—predicted vs. true injury counts

4.5.2 Discussion

As illustrated in the regression results, the predictive performance of the count-based models stood in stark contrast to the classification models. While the classification approach successfully identified high-risk scenarios, the regression models struggled to quantify the exact number of injuries, yielding suboptimal R^2 values across all three experimental setups.

Baseline and Hybrid Models Performance: The Baseline Neural Network Regressor (Figure 25) achieved an R^2 of only 0.021. A visual inspection of the scatter plot reveals a distinct regression to the mean phenomenon: while the true injury counts range from 0 to 4, the model's predictions cluster tightly in the narrow range of 0.5 to 1.5. This indicates that the model, unable to find strong signal patterns in the sparse data, defaulted to predicting the average injury rate to minimize the global loss function.

The introduction of temporal features in the Hybrid Lag and Group Bias Linear Model (Figure 26) resulted in a marginal improvement, raising the R^2 to 0.045. Although the scatter points show a slightly wider distribution compared to the baseline, the improvement is negligible. This suggests that construction injury events in this dataset lack significant temporal autocorrelation; the occurrence of an injury in the previous month does not strongly predict the magnitude of injuries in the subsequent month.

Two-Stage Model Limitations: Most notably, the Two-Stage Hybrid Model (Figure 27), which was designed to mitigate sparsity by first classifying injury occurrence and then regressing the count,

resulted in a negative R^2 of −0.301. While the classification stage (Stage 1) showed promise in filtering out zero-event months, the subsequent regression stage (Stage 2) suffered critically from data scarcity. After filtering for only positive-injury samples and applying strict denoising, the effective sample size became too small for the regressor to generalize. The negative R^2 implies that the model's predictions were worse than simply using the horizontal mean of the test data, highlighting the dangers of aggressive data segmentation on small datasets.

Summary of Regression Challenges: The poor performance across these models reinforces the findings from the preliminary methodology section: predicting the exact magnitude of construction accidents is significantly harder than predicting their probability. The high sparsity (zero-inflation) and the stochastic nature of accidents mean that the signal-to-noise ratio is too low for standard regression loss functions to converge effectively. Future improvements would likely require a significantly larger dataset to support complex architectures or the use of specialized loss functions like Zero-Inflated Poisson loss, provided that convergence issues can be overcome.

5. References

[1]Tixier, A. J.-P., Hallowell, M. R., Rajagopalan, B., & Bowman, D. (2016). Application of machine learning to construction injury prediction. Automation in Construction, 69, 102-114. https://doi.org/10.1016/j.autcon.2016.05.016
[2]NYCOSH. (2022, February 10). Deadly Skyline: An annual report on construction fatalities in New York State (2022 ed.). https://nycosh.org/wp-content/uploads/2022/02/NYCOSH_Deadly-Skyline-Report_2022.pdf
[3]Office of the New York State Comptroller. (2025, July). The construction sector in New York City: Post-pandemic trends (Report No. 8-2026). https://www.osc.ny.gov/files/reports/pdf/report-8-2026.pdf
[4]Carrivick, P. J. W., Lee, A. H., & Yau, K. K. W. (2003). Zero-inflated Poisson modeling to evaluate occupational safety interventions. Safety Science, 41(1), 53–63. https://doi.org/10.1016/S0925-7535(01)00057-1
[5]Junjia, Y., Alias, A. H., Haron, N. A., & Abu Bakar, N. (2024). Machine learning algorithms for safer construction sites: Critical review. Building Engineering, 2(1), 544. https://doi.org/10.59400/be.v2i1.544
[6] New York City Department of Buildings. (n.d.). *Incident Database* [Data set].
[7] Nayak, S. G., Shrestha, S., Kinney, P. L., Ross, Z., Sheridan, S. C., Pantea, C. I., Hsu, W. H., Muscatiello, N., & Hwang, S. A. (2018). *Development of a heat vulnerability index for New York State.* Public Health, 161, 127–137.
[8] Hilbe, J. M. (2011). *Negative binomial regression* (2nd ed.). Cambridge University Press.
[9] Cameron, A. C., & Trivedi, P. K. (2013). *Regression analysis of count data* (2nd ed.). Cambridge University Press.
[10] Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression* (3rd ed.). Wiley.
[11] Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
[12] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
[13] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830.
[14] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.
[15] Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters, 27*(8), 861–874.
[16] City of New York. (2025). *Official website of the City of New York*. Retrieved November 11, 2025, from https://www.nyc.gov/main
[17] City of New York. (n.d.). *DOB Job Application Filings* [Data set]. NYC Open Data. Retrieved November 11, 2025, from https://data.cityofnewyork.us/Housing-Development/DOB-Job-Application-Filings/ic3t-wcy2 [1NYC Maps. Maps of World. https://www.mapsofworld.com/usa/new-york-city-map.html#google_vignette