DSc Mid Sem Project

Predicting Accident Severity

Sanmay Sood (2021095)
Aayush Ranjan (2021003)
Siddharth Rajput (2021102)
Jyotirmaya Singh (2021055)

INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY
DELHI

Problem Statement

Traffic accidents cost U.S. citizens hundreds of billions annually, with severe accidents contributing significantly to these losses. Reducing such incidents is challenging but essential. A proactive traffic safety approach seeks to prevent unsafe conditions before accidents happen, making it crucial to identify key patterns and factors associated with severe accidents.

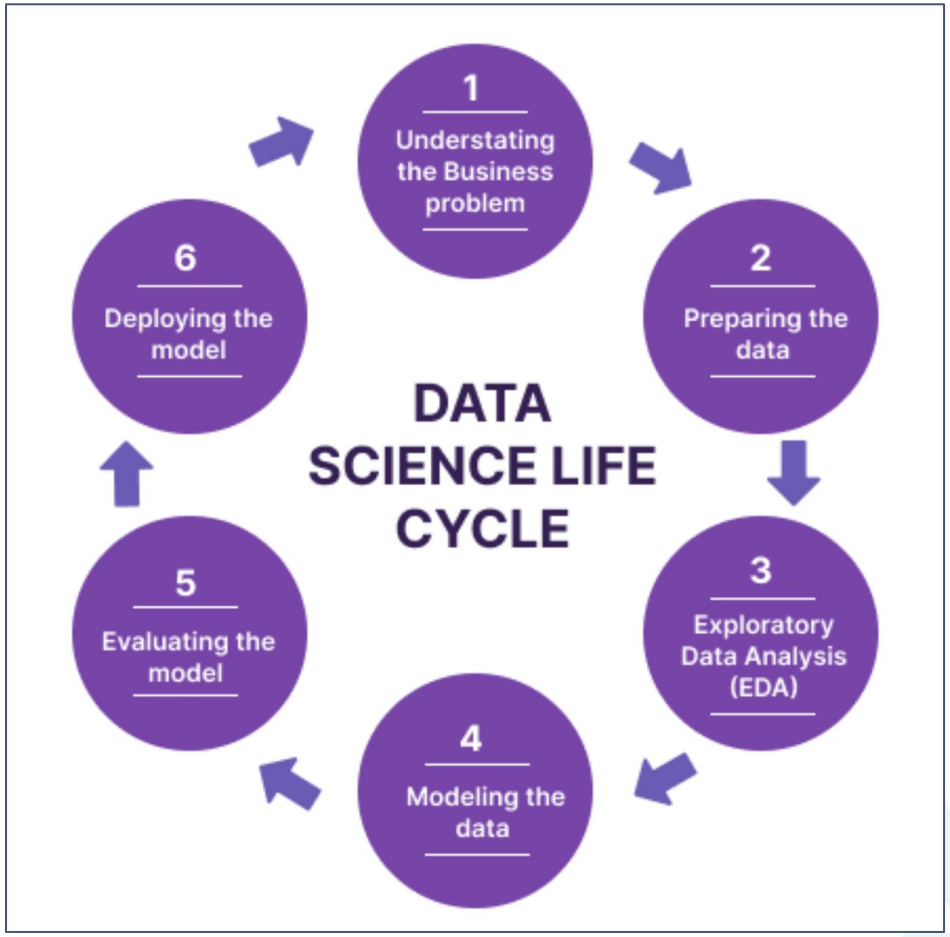
Project Goals:

- Identify key factors influencing accident severity.
- Develop a predictive model to forecast accident severity without needing specific accident details (e.g., no driver or vehicle information).

This project aims to integrate severity models with real-time traffic prediction systems to identify high-severity accidents as they occur. Predictive insights will support data-driven resource allocation, enabling proactive accident prevention and prioritizing high-risk areas.

Progress So Far

1. **WEEK 1-2**
   a. Preprocessing of data
2. **WEEK 3-4**
   a. In-depth Exploratory Data Analysis
3. **WEEK 5-7**
   a. Devising relevant hypotheses
   b. In-depth hypothesis testing
4. **WEEK 8**
   a. Validation Experiments



Dataset Overview

- Covers **1.1M+ accidents** in the U.S. (49 states) from **Feb 2019 to March 2023**.
- Data Source: **Reported by MapQuest** and **Bing** API.

Dataset Shape

- **1128394 rows**, **46 columns** (features)

Main Feature Categories

- **Traffic Attributes (12)**
  - ID, Severity (1-4), Start/End Time, Start/End GPS (Lat/Lng), Distance, Description.
- **Address Attributes (9)**
  - Street, City, County, State, Zip Code, Side (Right/Left), Country, Timezone.
- **Weather Attributes (11)**
  - Weather at closest airport (e.g., Temperature, Humidity, Visibility, Precipitation, Condition).
- **Point of Interest (POI) Attributes (13)**
  - Proximity indicators (e.g., Amenity, Crossing, Junction, Stop, Traffic Signal).
- **Period-of-Day (4)**
  - Day/Night status based on Sunrise, Civil, Nautical, Astronomical Twilight.

Target Feature (Label):

- **Severity**: Level of accident impact (1 - Least, 4 - Most).

Preprocessing the Data

- **Data Sources**:
  - **MapQuest**: Primary source for severe accident data.
  - **Bing**: Data discarded due to differing format in severity levels.
  - **Rows Remaining**: 673,653
- **Removed Features**:
  - **ID**: No predictive value.
  - **TMC, Distance(mi), End_Time, End_Lat, End_Lng**: Post-accident data.
  - **Description**: Contains extracted points of interest.
  - **Zipcode**, **Timezone**, **Wind_Chill(F)**, **Wind_Direction** and **Weather_Timestamp**
- **Fixing Datetime format:**
  - Retained **'Start_Time'** for further analysis and dropped **'Weather_Timestamp'**.
  - Derived new time-based features from **'Start_Time'**, including **Year, Month, Weekday, Day of the Year, Hour**, and **Minute** to enhance analysis and modeling capabilities.

Preprocessing the Data

Processing Categorical Features

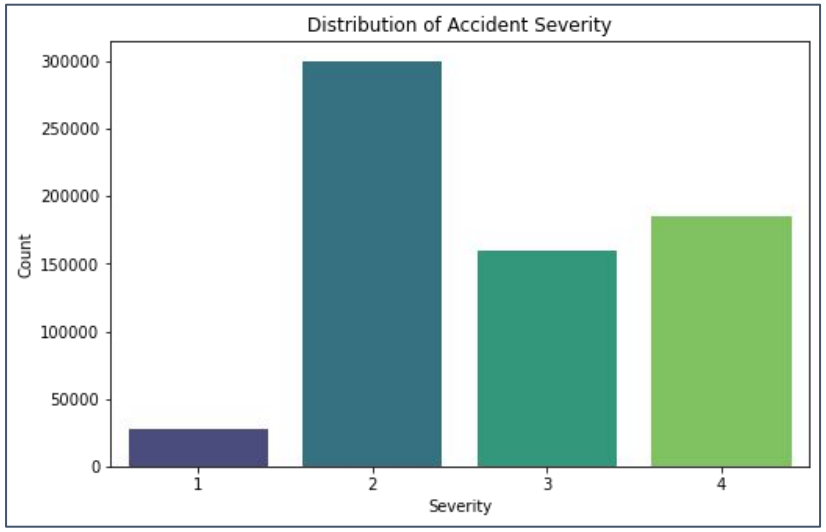
- **Dropped Categorical Features**:
  - **Country** and **Turning_Loop**: Both had only one unique value.
- **Weather Conditions**:
  - Created binary features for key conditions: **Clear, Cloud, Light Rain, Heavy Rain, Thunderstorm, Snow, Heavy Snow, Fog**, **Dust, Wind, Hail, Tornado, Smoke.**
  - Removed the original **'Weather_Condition'** column.
- **Imputation**:
  - The imputation approach groups data by **Airport_Code** and **Start_Month** to ensure geographical relevance and efficiency, filling missing values in continuous weather features with the median of each group.

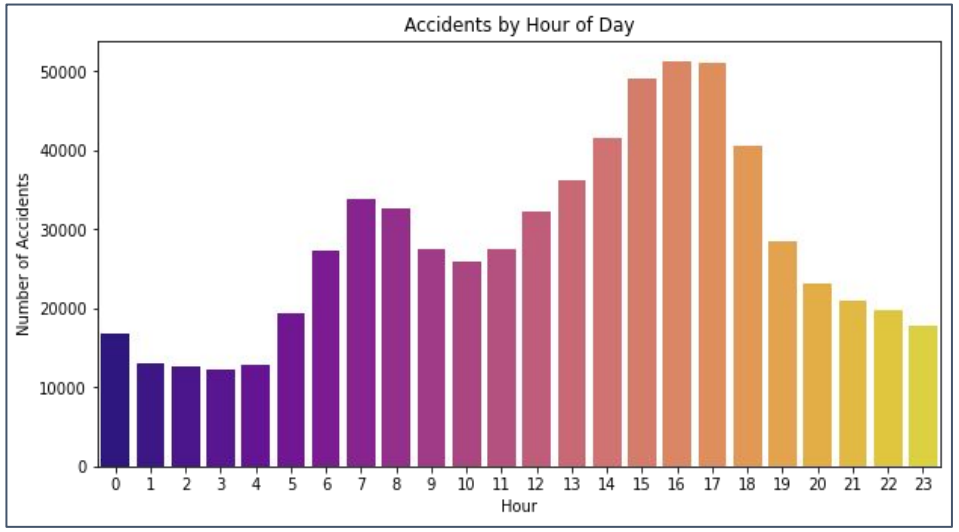
```
df_balanced.columns
✓ 0.0s
Index(['Severity', 'Start_Time', 'Start_Lat', 'Start_Lng', 'Street', 'City',
       'County', 'State', 'Airport_Code', 'Temperature(F)', 'Humidity(%)',
       'Pressure(in)', 'Visibility(mi)', 'Wind_Speed(mph)',
       'Precipitation(in)', 'Amenity', 'Bump', 'Crossing', 'Give_Way',
       'Junction', 'No_Exit', 'Railway', 'Roundabout', 'Station', 'Stop',
       'Traffic_Calming', 'Traffic_Signal', 'Sunrise_Sunset', 'Civil_Twilight',
       'Nautical_Twilight', 'Astronomical_Twilight', 'Clear', 'Cloud',
       'Light_Rain', 'Heavy_Rain', 'Thunderstorm', 'Snow', 'Heavy_Snow', 'Fog',
       'Dust', 'Wind', 'Hail', 'Tornado', 'Smoke', 'Year', 'Month', 'Weekday',
       'Day', 'Hour', 'Minute', 'Precipitation_NA'],
      dtype='object')
```

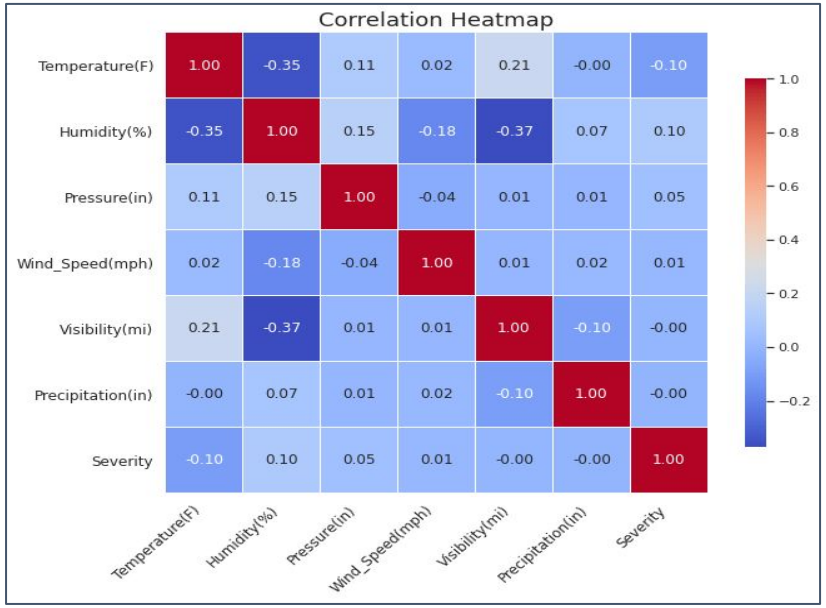
Handling Missing Data

- **Feature Modification**:
  - The **'Precipitation(in)'** feature was retained and a new binary feature, **'Precipitation_NA'**, was created to indicate missing values.
- **NaN Handling**:
  - Rows with missing values in less significant features (like **'City'**, **'Zipcode'**, and various twilight features) were dropped for convenience.
- **Value Imputation**:
  - Continuous weather features (**'Temperature(F)'**, **'Humidity(%)'**, **'Pressure(in)'**, **'Visibility(mi)'**, and **'Wind_Speed(mph)'**) with small missing values were filled using the median from groups defined by **'Airport_Code'** and **'Start_Month'** to maintain geographical and temporal relevance. Remaining missing values in these features were subsequently dropped for simplicity.

Exploratory Data Analysis



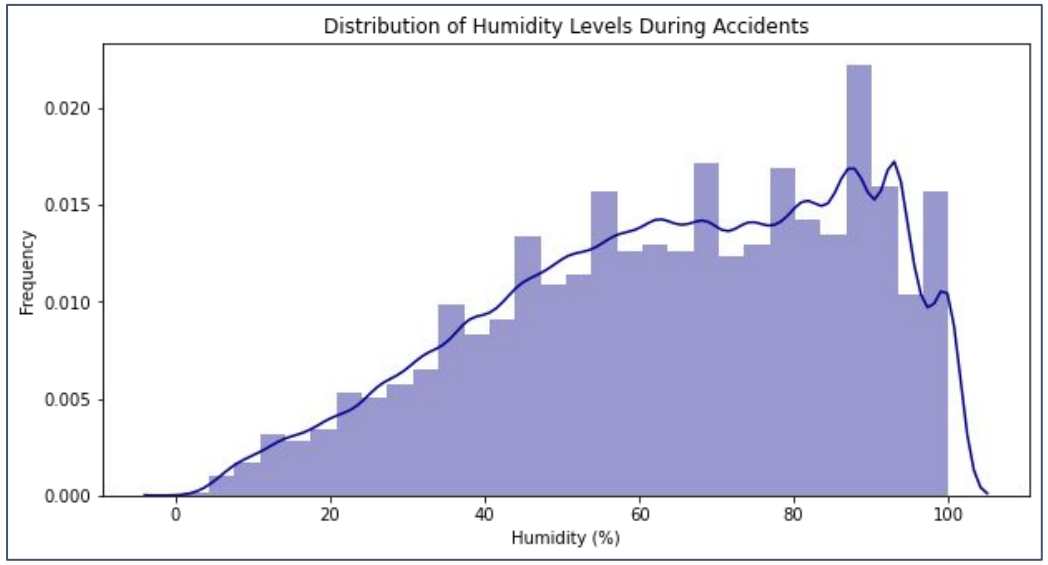


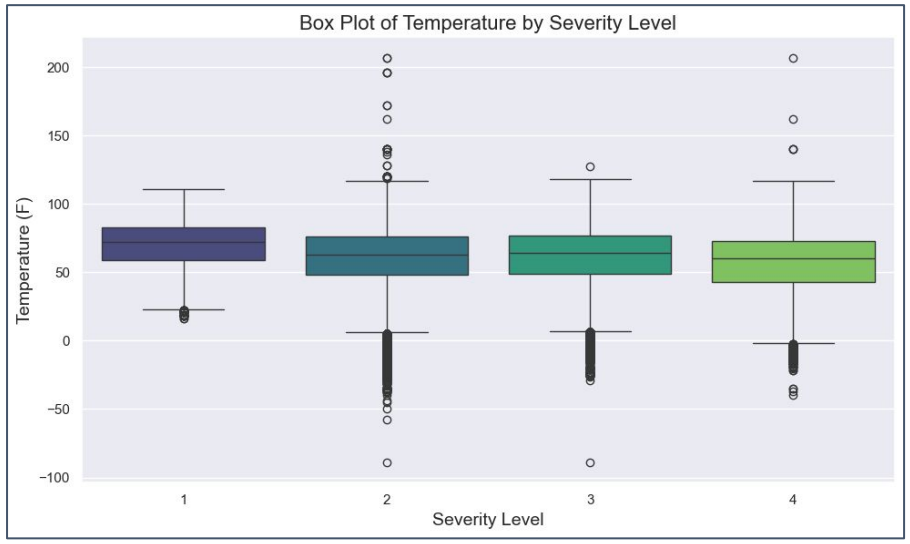


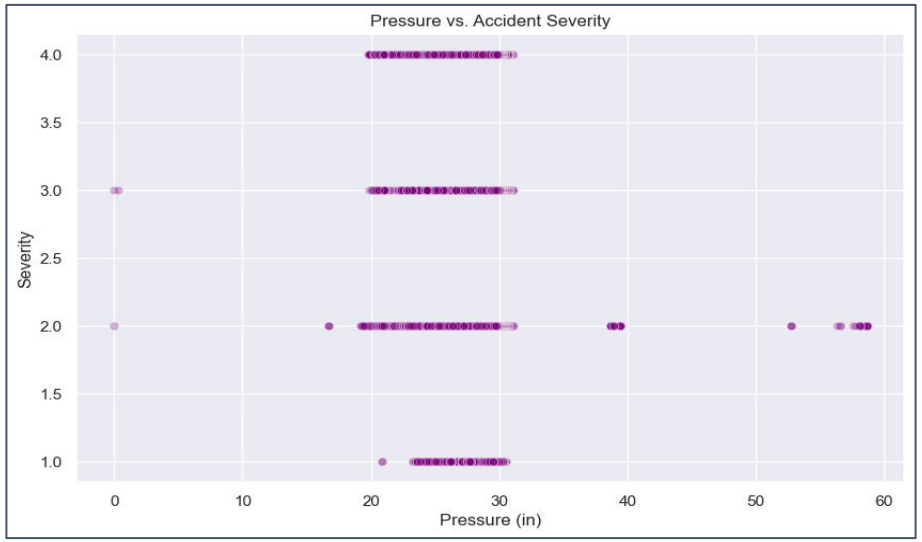
Skewness and Kurtosis of Continuous Features

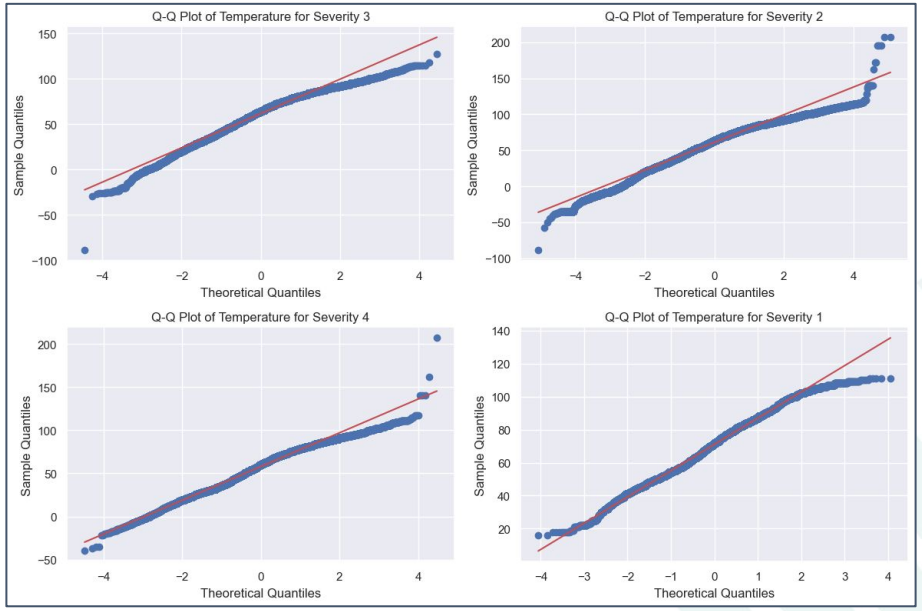
Feature	Skewness	Kurtosis	Interpretation
Temperature (°F)	-0.464	-0.162	Slightly left-skewed, low kurtosis
Humidity (%)	-0.389	-0.727	Slightly left-skewed, flat distribution
Pressure (in)	-3.360	15.931	Strongly left-skewed, heavy-tailed
Wind Speed (mph)	2.223	95.066	Strongly right-skewed, very peaked
Visibility (mi)	3.823	82.406	Highly right-skewed, very peaked
Precipitation (in)	90.437	9934.121	Extremely right-skewed, extremely peaked
Severity	0.185	-1.220	Almost symmetric, flat distribution

Exploratory Data Analysis









Hypothesis Testing : Chi-Square Test for Independence

a) Factor: Roundabout

Null Hypothesis (H_0): There is no association between *Severity* and *Roundabout*. (*Severity* is independent of *Roundabout*.)

Alternative Hypothesis (H_1): There is a significant association between *Severity* and *Roundabout*. (*Severity* depends on *Roundabout*.)

b) Factor: Heavy Rain

Null Hypothesis (H_0): There is no association between *Severity* and *Heavy Rain*. (*Severity* is independent of *Heavy Rain*.)

Alternative Hypothesis (H_1): There is a significant association between *Severity* and *Heavy Rain*. (*Severity* depends on *Heavy Rain*.)

c) Factor: Fog

Null Hypothesis (H_0): There is no association between *Severity* and *Fog*. (*Severity* is independent of *Fog*.)

Alternative Hypothesis (H_1): There is a significant association between *Severity* and *Fog*. (*Severity* depends on *Fog*.)

Hypothesis Testing : Chi-Square Test for Independence

Factor	Chi^2 Statistic	p-value	Conclusion
Roundabout	2.67	0.445	Severity is not significantly associated with the presence of a roundabout. (H_0 failed to reject)
Heavy Rain	311.34	3.49e-67	Severity has a significant association with heavy rain conditions. (H_0 rejected)
Fog	741.00	2.70e-160	Severity has a significant association with foggy conditions. (H_0 rejected)

Interpretation

- **$p < 0.05$**: Reject H_0 → *Severity* is associated with the factor.
- **$p \geq 0.05$**: Fail to reject H_0 → No association with *Severity*.

Hypothesis Testing

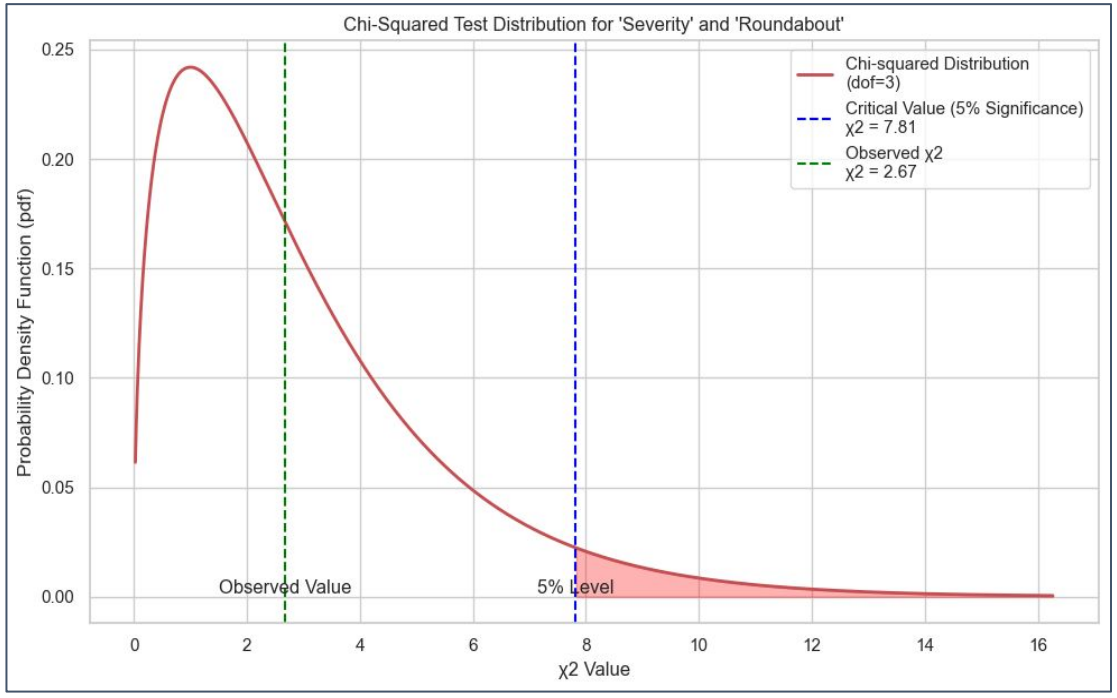


Fig. (a) Failing to reject the Null

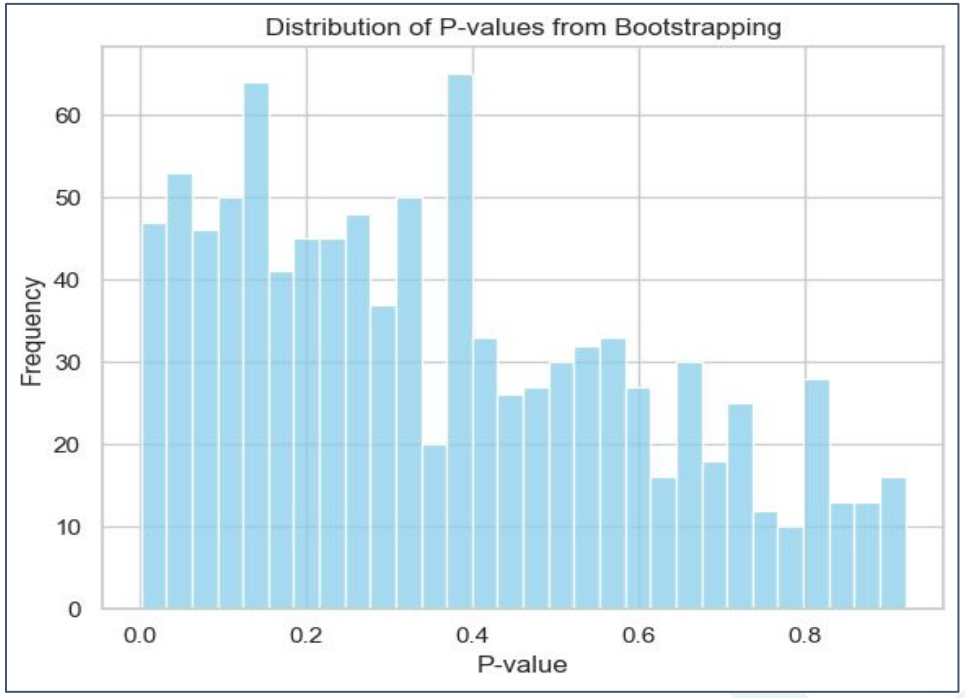


Fig. (b) Validation Experiment

Hypothesis Testing

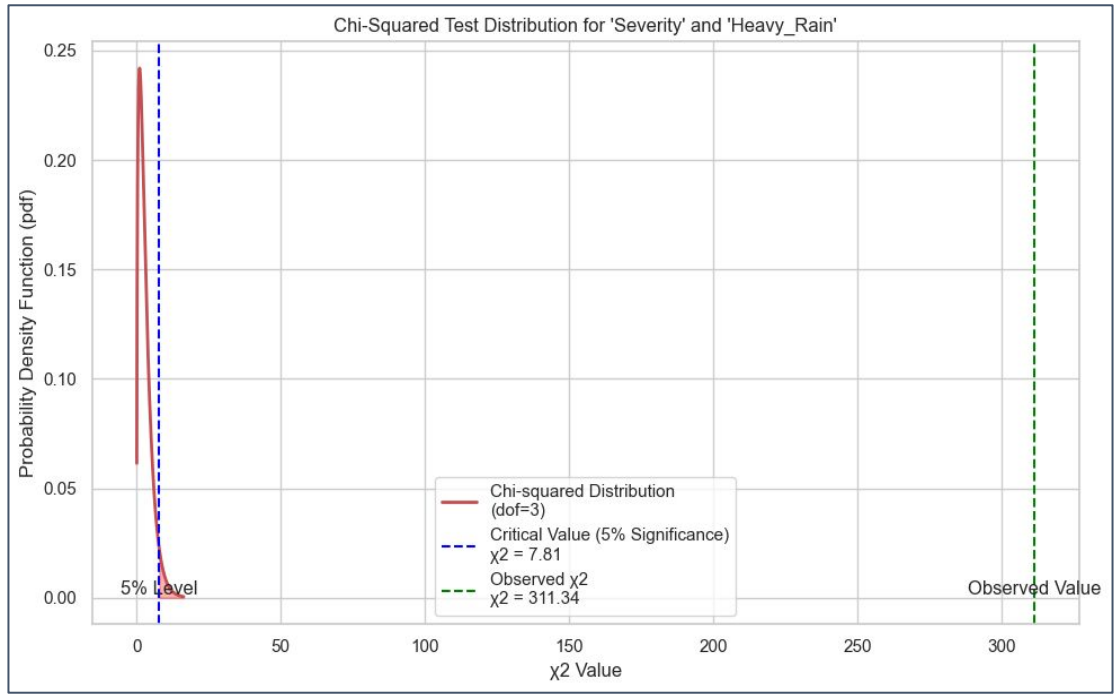


Fig. (a) Rejecting the Null

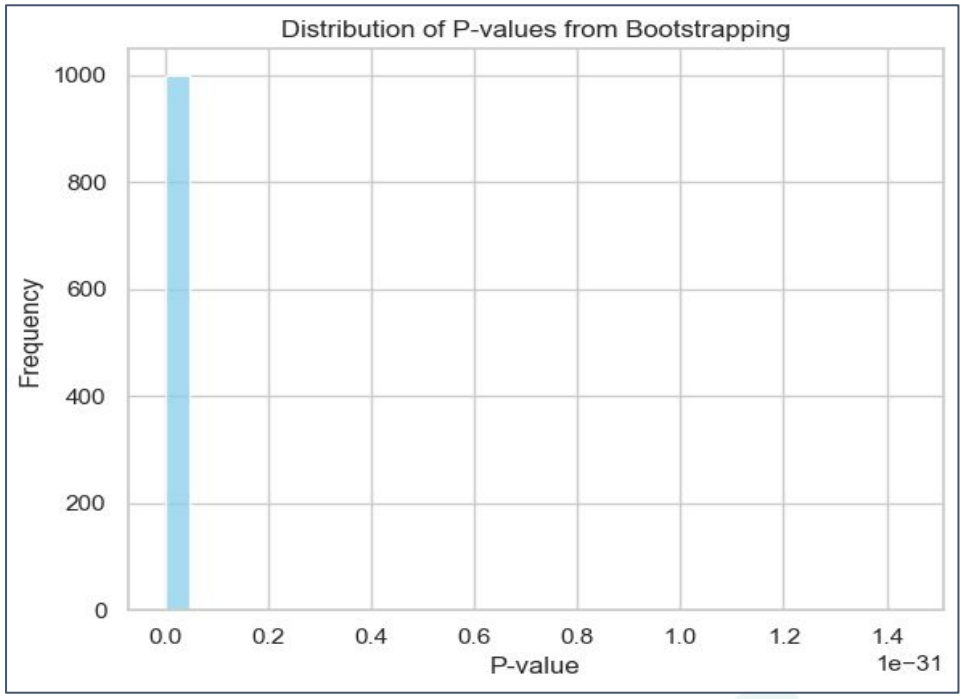


Fig. (b) Validation Experiment

Hypothesis Testing

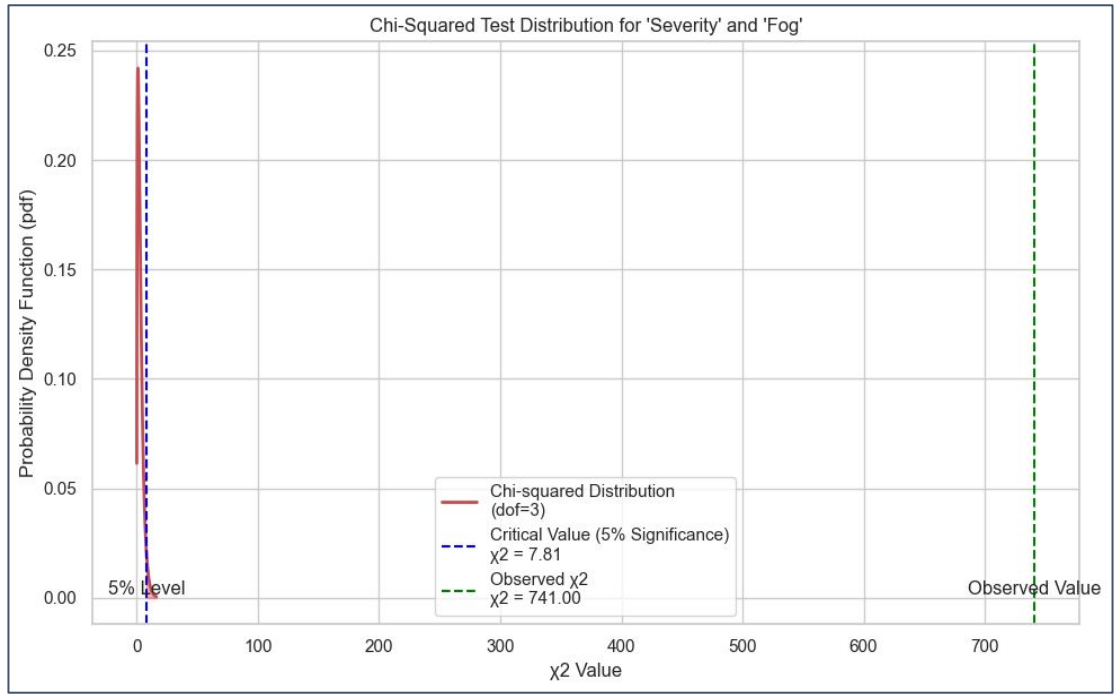


Fig. (a) Rejecting the Null

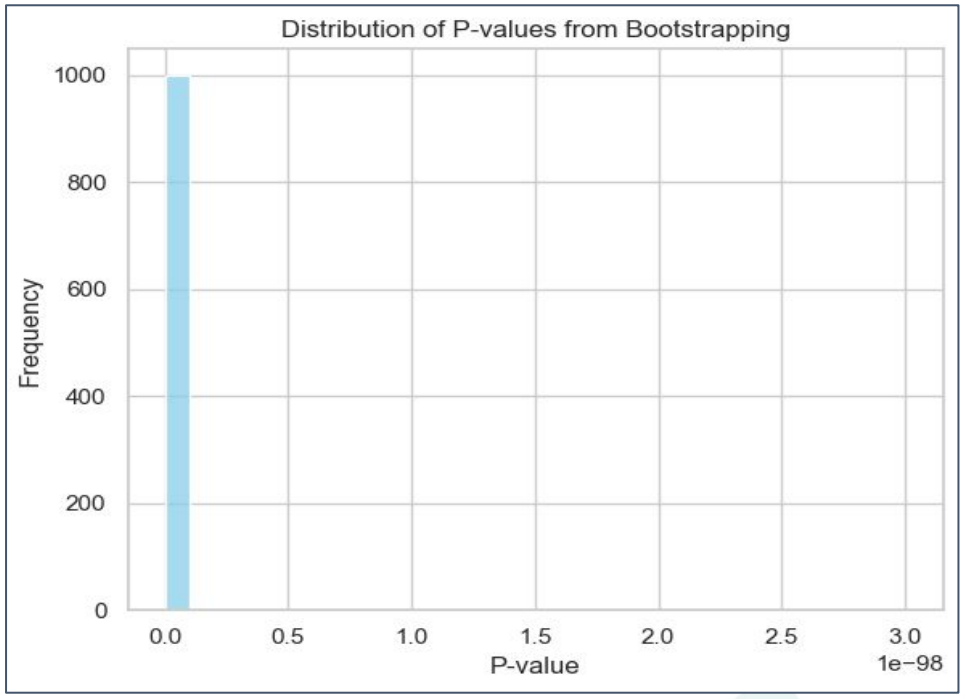


Fig. (b) Validation Experiment

Hypothesis Testing : z-test for Two Proportions

Null Hypothesis:

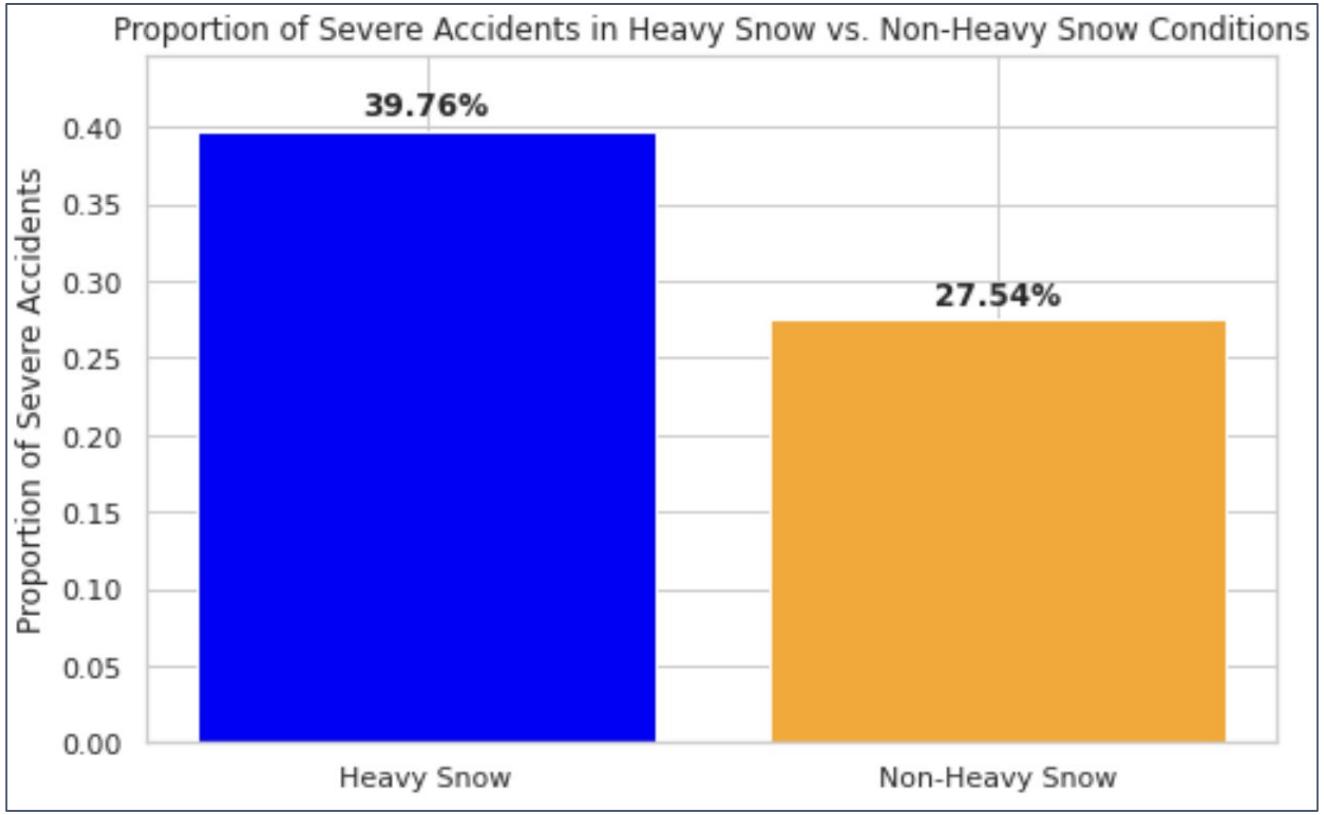
There is no significant difference in the proportion of severe accidents (Severity = 4) between Heavy Snow and non-Heavy Snow conditions. Mathematically, this implies $p_{\text{Heavy Snow}} = p_{\text{Non-Heavy Snow}}$ where p represents the proportion of severe accidents.

Alternate Hypothesis:

There is a significant difference in the proportion of severe accidents (Severity = 4) between Heavy Snow and non-Heavy Snow conditions. This implies $p_{\text{Heavy Snow}} \neq p_{\text{non-Heavy Snow}}$

Factor	z - Statistic	p-value	Conclusion
Visibility	7.497390801767131	6.510066792403325e-14	There is a significant difference in the proportions of severe accidents between Heavy Snow and non-Heavy Snow conditions.

Hypothesis Testing



The graph shows a higher proportion of severe accidents (39.76%) during **Heavy Snow** compared to **non-Heavy Snow** conditions (27.54%), suggesting Heavy Snow may increase accident severity

Hypothesis Testing : t-test for Correlation Coefficient

Null Hypothesis:

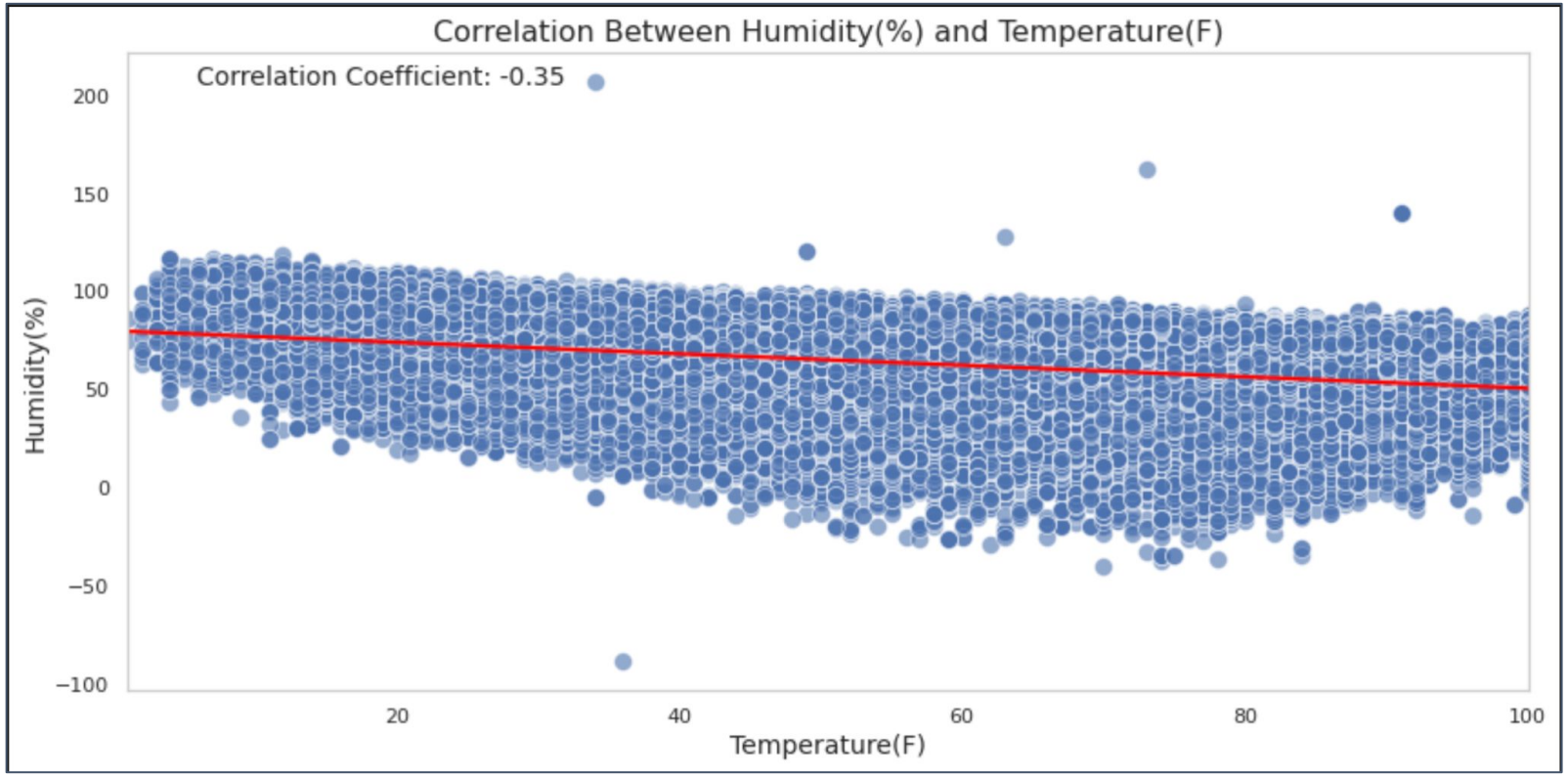
There is no significant linear correlation between Humidity(%) and Temperature(F). In other words, Humidity(%) and Temperature(F) are not linearly dependent on each other or the correlation coefficient r is equal to zero (r=0).

Alternate Hypothesis:

There is a significant linear correlation between Humidity(%) and Temperature(F). This implies that the correlation coefficient r is not equal to zero (r≠0).

Factor 1	Factor 2	Correlation Coefficient	T-statistic	Conclusion
Humidity(%)	Temperature(F)	-0.3486002963580502	-305.26697819618977	There is a significant correlation between Humidity(%) and Temperature(F).

Hypothesis Testing



This graph shows a **negative correlation** between **Temperature(F)** and **Humidity(%)**, with a correlation coefficient of **-0.35**. The scatter plot and red regression line indicate that as temperature increases, humidity tends to decrease. However, the correlation is moderate, not very strong.

Hypothesis Testing : z-test for One Population Proportion

Null Hypothesis:

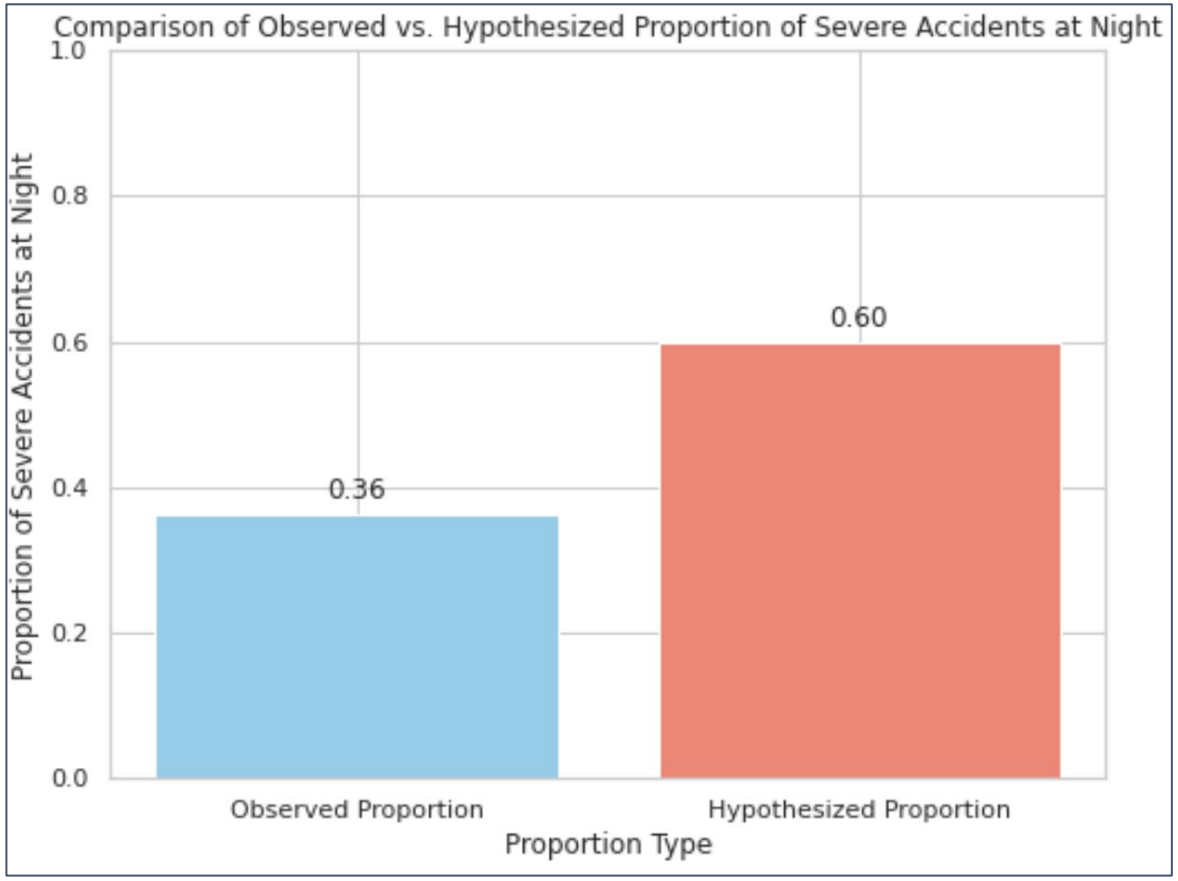
The proportion of nighttime severe accidents is 0.6 i.e the proportion of severe accidents occurring at night is equal to or less than 60% ($p \leq 0.6$).

Alternate Hypothesis:

The proportion of nighttime severe accidents is greater than 0.6 i.e the proportion of severe accidents occurring at night is greater than 60% ($p > 0.6$).

Factor	t - Statistic	p-value	Conclusion
Sunrise_Sunset	-288.1130152115506	1.0	Fail to reject the null hypothesis: Proportion of night-time severe accidents is not significantly greater than 0.6

Hypothesis Testing



This graph compares the **observed** proportion of severe accidents at night (**0.36**) with a **hypothesized** proportion (**0.60**). The observed proportion is significantly lower than the hypothesized one, suggesting that severe accidents may occur less frequently at night than hypothesized.

Hypothesis Testing : t-test for Two Population Mean

Null Hypothesis:

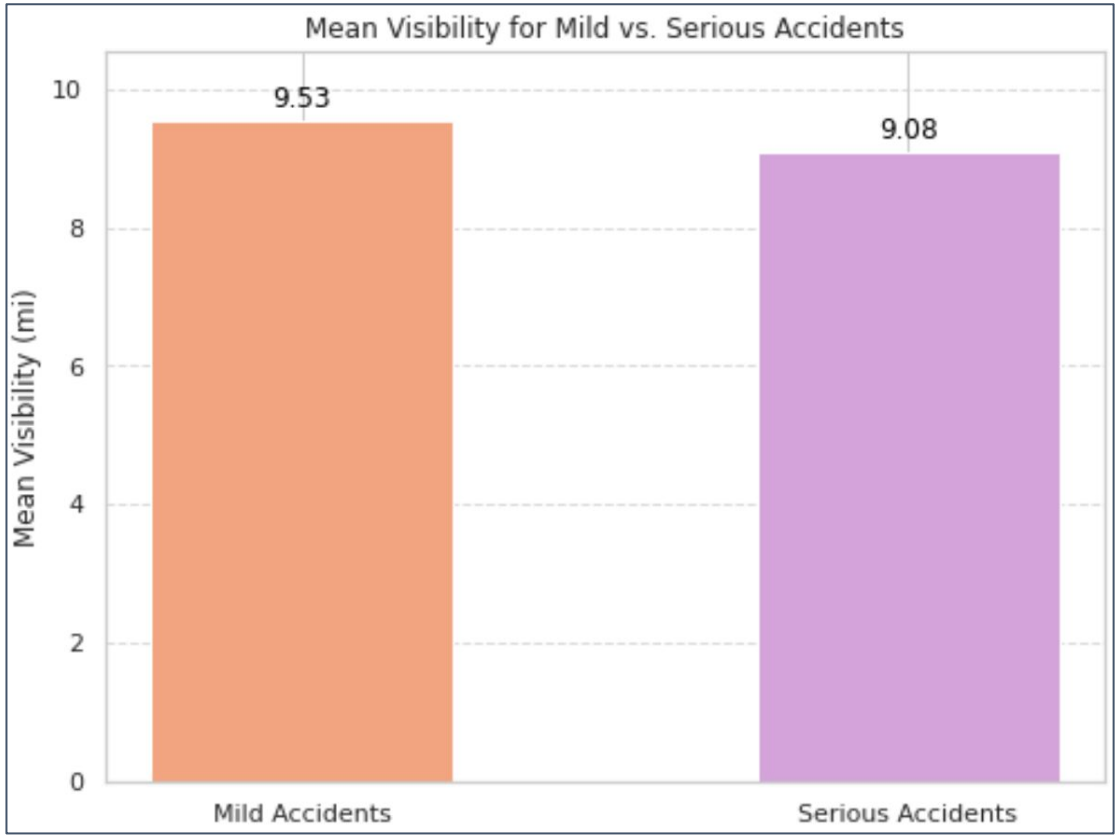
There is no significant difference in mean visibility between mild and serious accidents. This means that the mean visibility for Severity 1 accidents is equal to the mean visibility for Severity 4 accidents ($\mu_{mild} = \mu_{serious}$).

Alternate Hypothesis:

There is a significant difference in mean visibility between mild and serious accidents. This implies that the mean visibility for Severity 1 accidents is not equal to the mean visibility for Severity 4 accidents ($\mu_{mild} \neq \mu_{serious}$).

Factor	t - Statistic	p-value	Conclusion
Visibility	33.3872033	1.2812538e-241	There is a significant difference in visibility between mild and serious accidents

Hypothesis Testing



This graph compares the **mean visibility** for **mild accidents** (9.53 miles) and **serious accidents** (9.08 miles). Mild accidents tend to occur under slightly better visibility conditions than serious accidents, though the difference is small.

Future Work

1. **Feature Engineering**: Further refine feature engineering efforts, potentially exploring additional derived features like traffic congestion levels, seasonal trends, or more granular time-of-day analysis, which might correlate with accident severity.

2. **Dimensionality Reduction:** Reduce the number of features, or dimensionality, to potentially improve model performance and computational efficiency. In particular, reduction of computation is especially important for real-time applications. By focusing on the most relevant features, dimensionality reduction techniques will help streamline the model, making it faster and less prone to overfitting.

3. **Model Enhancements**: Explore machine learning or deep learning models, such as Gradient Boosting, Random Forest, or neural networks, to improve prediction accuracy and capture complex patterns in accident data.

Post Midsem

Models

Problem:

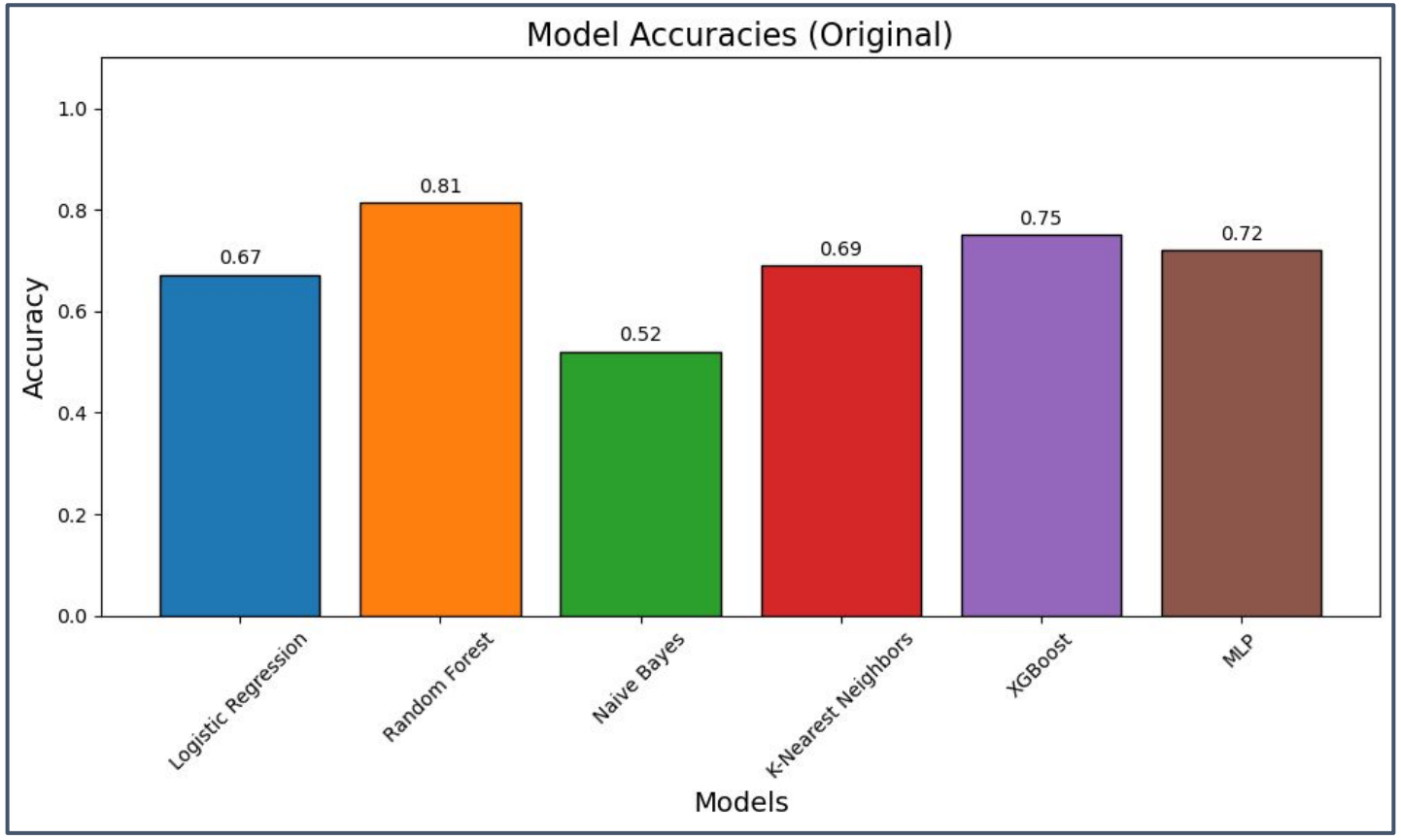
- The task was Severity prediction and the problem was modelled as a **classification** problem where the target variable was **'Severity'** and the possible classes were {1, 2, 3, 4}**.** Severity measures the level of accident impact (1 - Least, 4 - Most).

Models Used:

- **Logistic Regression:** Simple Linear Classification
- **Random Forest:** Ensemble of decision trees, robust to overfitting.
- **Naive Bayes:** Probabilistic, works well with high-dimensional data.
- **XGBoost:** Fast, efficient gradient boosting algorithm for improved accuracy and handling large datasets.
- **K-Nearest Neighbours:** Instance-based learning, works well with clear clusters.
- **Multi-Layered Perceptron:** Deep learning model, great for complex, non-linear relationships.

Results: Accuracy

Train Size: 404,191 | Test Size: 202,096

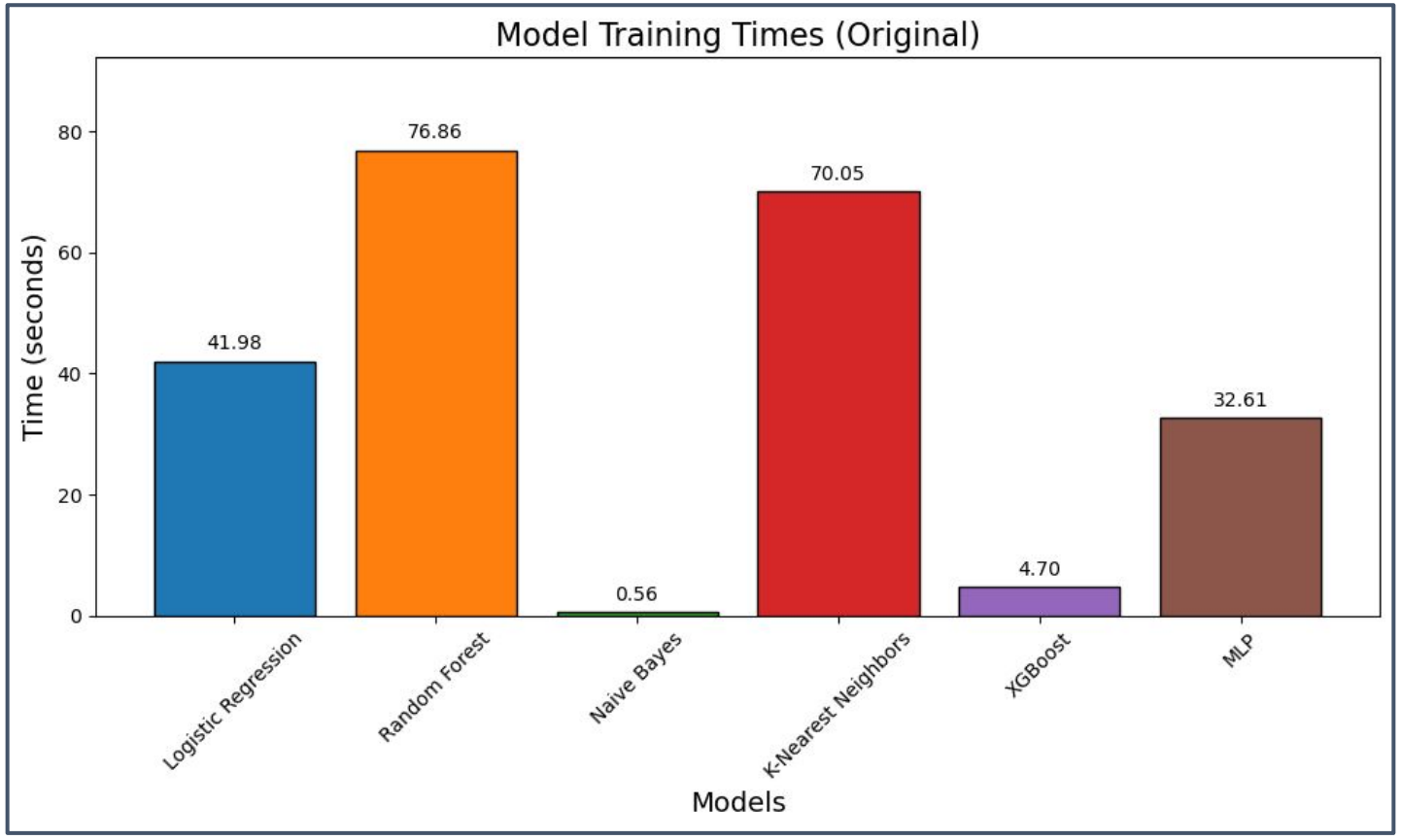


Insights:

- **Random Forest**: Best performer, excels at capturing complex patterns through ensemble learning.
- **Naive Bayes**: Poor performance, likely due to its assumption of feature independence, which doesn't hold here.
- **XGBoost and MLP**: Solid performance, showcasing the effectiveness of advanced models.
- **Logistic Regression**: Decent performance, could improve with feature selection or regularization.
- **KNN**: Performs moderately well, suitable for non-linear decision boundaries, but less effective than Random Forest or XGBoost.

Results: Time Taken

Train Size: 404,191 | Test Size: 202,096



Insights:

- **Naive Bayes**: Fastest training time, ideal for quick predictions but less accurate.
- **XGBoost**: Quick training with solid performance, balancing speed and accuracy.
- **Logistic Regression**: Moderate training time, suitable for linear relationships but slower than simpler models.
- **MLP**: Faster than Random Forest, and performs well with complex data.
- **K-Nearest Neighbors**: Moderate training time with good accuracy, but more time-consuming than XGBoost and MLP.
- **Random Forest**: Slowest training time but performs well with complex data, offering high accuracy.

Linear Reduction

Singular Value Decomposition:

- Reduces dimensionality by retaining the top singular values.
- Commonly used for matrix completion and noise reduction.
- Helps in extracting latent features and improving computational efficiency.

Principal Component Analysis:

- Transforms data into orthogonal components ordered by variance explained.
- Reduces dimensionality while preserving data variance.
- Uses eigenvalue decomposition to identify principal components.
- Widely used in feature reduction for exploratory data analysis and machine learning.

Non Linear Reduction

JL (Johnson-Lindenstrauss Lemma):

- **Purpose:** Reduces dimensionality while preserving distances between points.
- **How it works:** Projects high-dimensional data into a lower-dimensional space using a random matrix.
- **Key Concept:** Ensures distances are approximately preserved with high probability.
- **Use Case:** Efficient for large datasets (e.g., clustering, machine learning).
- **Benefits:**

  Fast and scalable.

  Minimal information loss, preserving data structure.

Feature Selection

Feature Selection

- **Purpose:** Used L1 regularization to shrink less important feature coefficients to zero, automatically selecting relevant features.

- **How it works:** Lasso adds a penalty term to the loss function, driving some coefficients to zero. The alpha parameter controls regularization strength. (**alpha used: 0.1**)

- **Benefits:**
  Automatically removes irrelevant features.
  Reduces overfitting and simplifies the model.
  Efficient feature selection for large datasets.

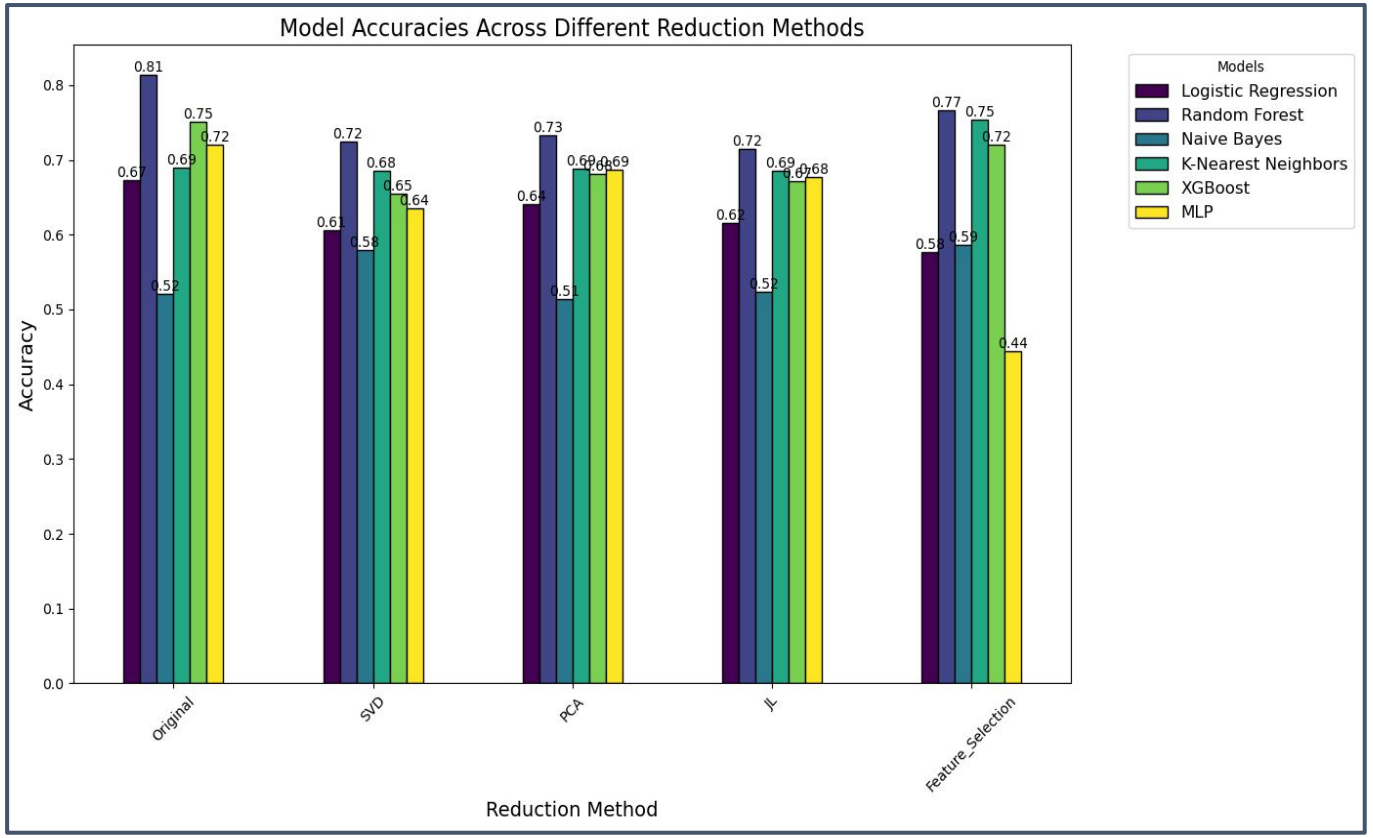
- **Use Case:** Ideal for high-dimensional datasets, especially in regression and classification tasks.

Results

	Logistic Regression		Random Forest		Naive Bayes		KNN		XGBoost		MLP	
	Accuracy (in %)	Time Taken (in s)	Accuracy (in %)	Time Taken (in s)	Accuracy (in %)	Time Taken (in s)	Accuracy (in %)	Time Taken (in s)	Accuracy (in %)	Time Taken (in s)	Accuracy (in %)	Time Taken (in s)
Original	0.6726	41.98	0.8131	76.86	0.5206	0.56	0.6893	70.05	0.7505	4.70	0.7200	32.61
SVD	0.6065	1.00	0.7247	19.49	0.5796	0.09	0.6848	15.43	0.6541	2.68	0.6354	32.07
PCA	0.6409	7.54	0.7333	41.32	0.5134	0.26	0.6877	46.77	0.6811	4.47	0.6868	33.91
JL	0.6162	5.49	0.7151	37.12	0.5231	0.23	0.6854	49.04	0.6715	4.80	0.6777	31.61
Feature Selection	0.5762	33.14	0.7661	7.26	0.5858	0.06	0.7538	3.89	0.7209	2.14	0.4445	32.88

Results: Accuracy

Train Size: 404,191 | Test Size: 202,096

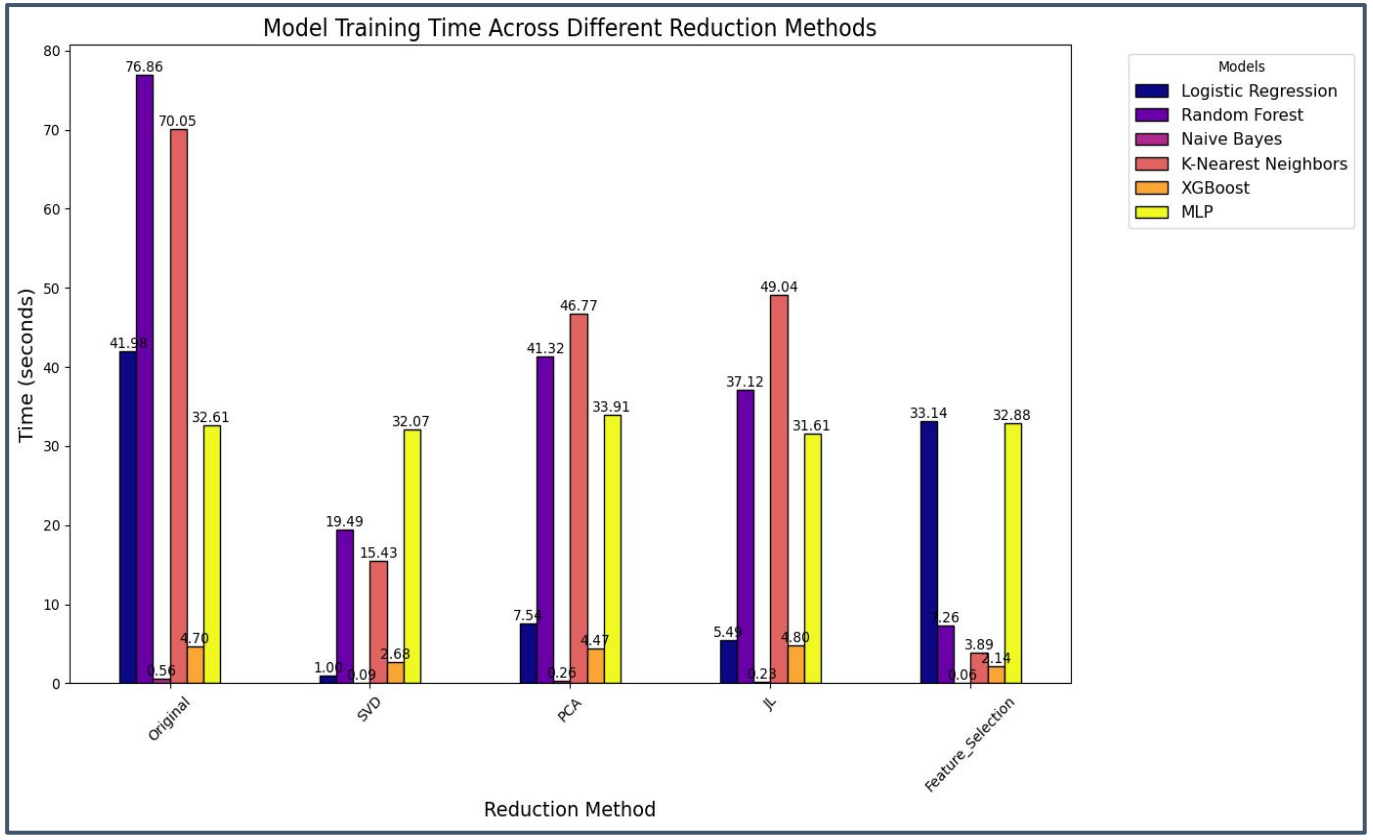


Insights:

- **Random Forest** consistently performs the best across all reduction methods, with the highest accuracy in **Original** (0.8131).
- **Logistic Regression** and **K-Nearest Neighbors** show decent performance in **Original**, but **KNN** performs better in **Feature Selection** (0.7538).
- **Naive Bayes** consistently underperforms, with the lowest accuracy in most methods, although it improves slightly in **Feature Selection**.
- **XGBoost** performs well in **Original** (0.7505), but its performance decreases across most reduction methods.
- **MLP** struggles with **Feature Selection**, showing the lowest accuracy in that method (0.4445).

Results: Time Taken

Train Size: 404,191 | Test Size: 202,096



Insights:

- **Naive Bayes** is the fastest across all methods, taking only **0.56 seconds** in **Original** and even less in other methods.
- **Logistic Regression** has moderate time consumption, with **41.98 seconds** in **Original**, but the time drastically reduces to **1.00 second** with **SVD**.
- **Random Forest** is the most time-consuming model, with **76.86 seconds** in **Original**, though its time drops in **Feature Selection** (7.26 seconds).
- **K-Nearest Neighbors** and **MLP** both have high training times, especially in **Original** (**70.05 seconds** for KNN and **32.61 seconds** for MLP).
- **XGBoost** is fast, taking only **4.70 seconds** in **Original** and remaining efficient across all reduction methods.

Conclusion & Future Work

Conclusion:

- **Random Forest** performed the best across all reduction techniques, achieving high accuracy and robustness.
- **XGBoost** and **KNN** showed strong performance, especially in the **Original** dataset, but accuracy varied with reduction techniques.
- **Logistic Regression** was decent in **Original**, but its performance dropped with dimensionality reduction.
- **Naive Bayes** consistently underperformed, likely due to its assumption of feature independence.
- **MLP** struggled with **Feature Selection**, showing the lowest accuracy in that method.

Future Work:

- **Feature Engineering**: Incorporate additional features like **traffic congestion**, **seasonal trends**, and **time-of-day analysis** to improve model predictions.
- **Model Enhancements**: Explore advanced **Deep Learning** models to capture complex patterns and improve accuracy.
- **Real-time Predictions**: Focus on optimizing models for **real-time prediction** while maintaining efficiency and high performance.

THANK YOU !