TASK 3: ANALYZING PUBLIC BIKE SHARING RIDERSHIP

Introduction

1.1 Overview of Traffic Accidents and Their Impact

Road traffic accidents are a severe problem worldwide, affecting the health and well-being of the populations, their economy, and society. According to WHO, road traffic crashes cause about 1.35 million deaths per annum, thus becoming one of the leading causes of death worldwide. Besides fatalities, millions more are nonfatally injured with many causing lifelong disability which could have marked importance in quality of life.

The implications of traffic accidents are such that they reach beyond a simple loss of life and bodily injury. They inflict economic costs lying in emergency response, medical care, rehabilitation, and loss of productivity, both in life and possession, onto societies. The psychic and emotional burden on the victims, close family, and communities tends to be heavy, ushered in by trauma and anguish.

Understanding the factors behind these accidents will help devise strategies to mitigate them. The understanding of patterns and correlations in the data from traffic accidents will help the stakeholders develop intervention measures that will count in the overall general incidence and severity reduction of these accidents.

1.2 Purpose and Scope of the Analysis

Such factors can be studied further with the help of comprehensive data given for the analysis by the Fatality Analysis Reporting System. The trends, correlations, and risk factors of road traffic accidents are to be determined so that practical action in the form of policy decisions can be formulated and implemented to improve road safety.

The scope of this analysis includes the following key objectives:

- **Trend Analysis:** To identify the trends of occurrence of traffic accidents over time, thus ascertaining periods of higher risk and whether there are seasonal or temporal variations.
- **Weather Impact:** Relating different weather scenarios to the frequency of incidents and their seriousness regarding the weather conditions of traffic accident cases.
- **Geospatial Analysis:** Pinpointing and mapping those areas where the hotspots of most concentration of traffic accidents are located, understanding the environmental and infrastructural contributing factors for these spots.
- **Factor Analysis of Risks**: Research into factors such as the time of the day, types of vehicles, and conditions of the road, which may contribute to a high possibility of road accidents.
- **Recommendation:** Provide evidence-based strategies for the improvement of road safety, informed by the findings of the foregoing analysis.

This analysis will utilize data spanning multiple years and regions, ensuring a comprehensive understanding of the dynamics of traffic accidents across various contexts.

1.3 Data Sources and Methodology

This study will draw its data from the detailed databank on all fatal traffic crashes throughout the United States roadways included under the Fatality Analysis Reporting System (FARS) of the National Highway Traffic Safety Administration (NHTSA). The FARS dataset includes critical variables like the time and location of accidents, weather conditions at the point of the crash, vehicle types involved, and demographic details of the individuals.

Key Data Sources:

- **FARS Annual Reports:** These datasets compile information on fatal road traffic accidents nationally, year by year, including variables on the nature, location, and time, and contributing factors of an accident.
- **Weather Data:** Historical weather data is used, connected to both the time and place of an accident, for conducting detailed studies on the effects of weather conditions on accidents.

- **Geospatial Data:** The information relating to the locations of accidents will be used to identify and analyze hotspots. This will help in making the spatial concept of traffic accident distribution.

Methodology:

1. **Data Collection:** We'll start by gathering the relevant datasets, primarily from the FARS database. To add more depth to our analysis, we'll also pull in data from supplementary sources, like weather data providers.
2. **Data Cleaning and Preprocessing:** The data will then go through a thorough cleaning process. This step involves addressing any missing values, ensuring that data types are correct, and merging the datasets as needed. The goal is to create a comprehensive dataset that's ready for analysis.
3. **Exploratory Data Analysis (EDA):** Once the data is cleaned, we'll begin with an exploratory analysis. This involves generating descriptive statistics and visualizations to spot key trends, understand distributions, and explore potential correlations between different variables.
4. **Advanced Analytics:** Next, we'll dive into more advanced techniques. This phase will include correlation analysis, time-series analysis, and geospatial analysis to uncover deeper insights into the factors that influence traffic accidents.
5. **Visualization:** To communicate our findings clearly, we'll create high-quality visualizations, such as time series plots, bar charts, and heat maps. These will help present the data in a way that's easy to understand.
6. **Reporting and Recommendations:** Finally, all the insights from our analysis will be pulled together into a final report. This report will not just present the findings but will also translate them into actionable insights and recommendations for policymakers and other stakeholders.

1.4 Importance of the Study

This study is essential for several reasons:

- **Policy Development:** The key insights generated from this analysis, therefore, will be very important for policymakers to understand the key causes that contribute to traffic accidents. Having this understanding, they

can then engage in the development of very focused interventions or regulations that reduce accident incidences.

- **Public Awareness**: The study will, therefore, be resourceful in creating public awareness by pinpointing when and where accidents are most likely to happen. Education regarding safety practices and the risks the drivers, pedestrians, or other road users are exposed to can be carried out.
- **Infrastructure Planning:** Knowing the frequented places of accidents may not be initially of high interest for this work, but the information could be used by urban planners as well as transportation authorities for infrastructural planning. Appropriate actions will have to be taken for infrastructural improvements like better road signage, more and better lighting, and better traffic congestion controls.
- **Future Research:** This analysis result might also form the basis of further research, specifically toward the new transportation technologies that are going to come into existence, such as autonomous vehicles, and their possible influence on traffic safety.

Data Description

2.1 Overview of Data Sources

This analysis is harnessed through comprehensive, multi-dimensional datasets containing very elaborate details regarding traffic accidents in the United States. The Fatality Analysis Reporting System shall be managed by the National Highway Traffic Safety Administration as a fundamental source of data. Apart from this central dataset, there will be some complementary datasets that will be combined for further deepening of the analysis to get a full understanding of why traffic accidents occur.

For this analysis, the following datasets were utilized and integrated:

1. **FARS Annual Crash Data:**

- Description: This dataset includes records of all fatal traffic accidents reported in the United States for each year.

- Variables: It contains detailed information such as the date and time of the crash, the location, the type of crash, the vehicles involved, the individuals involved, and the environmental conditions at the time of the accident.

2. **FARS Person Data:**

- Description: This dataset provides detailed information about the individuals involved in the accidents, including drivers, passengers, pedestrians, and cyclists.
- Variables: It includes data on age, gender, seating position, injury severity, use of safety equipment (like seatbelts or helmets), and whether substances like alcohol or drugs were involved.

3. FARS Vehicle Data:

- Description: This dataset offers detailed information about the vehicles involved in fatal accidents.
- Variables: It covers aspects like the vehicle type, make, model, condition of the vehicle, maneuvers before the crash, and any vehicle-related factors that contributed to the accident.

4. **Weather Data:**

- Description: This dataset contains historical weather data that corresponds to the time and location of each reported accident.
- Variables: It includes information on temperature, precipitation, visibility, wind speed, and general weather conditions (such as whether it was clear, rainy, or foggy).

5. Geospatial Data:

- Description: This dataset provides geographical information related to the locations where accidents occurred.
- Variables: It includes data on latitude, longitude, road type, and whether the location is designated as urban or rural.

These datasets were meticulously merged to create a holistic view of each traffic accident, enabling a comprehensive analysis of the factors contributing to traffic fatalities.

2.1 Key Variables and Their Importance

Understanding the variables across these datasets is essential for conducting a meaningful and detailed analysis. Below is a consolidated description of the key variables:

1. Accident Data:

- **Accident ID:** The unique ID of the accident; this will be used for linking all records across all data sets.
- **Date and Time:** This is a variable that tells the exact time and date the accident occurred. It is very important in the temporal analysis of the patterns and trends of the occurrences of the accidents.
- **Location:** State, county, specific GPS coordinates of the accident. Localization data are extremely important for identifying accident hot spots and regional differences in accident rates.
- **Weather Conditions:** Illustrates the condition of the weather during the accident—clear, rainy, or snowing. Weather conditions can dramatically affect road conditions and accordingly, driver behavior.

2. Personal Data:

- **Person ID:** Unique identifier of each person who was involved in the accident and easily linked to both accident and vehicle datasets.
- **Age:** Age of the person involved in the accident; important information for demographic pattern analyses and understanding risk profiles
- **Sex:** The gender of a person; might be used to examine possible gender-based differences in accident involvement or outcomes.
- **Injury Severity:** Classified as fatal, serious, minor, or not injured. This variable is of major concern when examining the effects of different factors on injury outcomes.
- **Substance Use:** Whether the person was drinking or using drugs at the time of the accident. Substance use often plays a very important role in traffic crashes.

3. Vehicle Data:

- **Vehicle ID:** The unique identifier for each of the vehicles that take part in the accident.
- **Type of vehicle:** Examples include a car, truck, or motorcycle. These have varying safety profiles and associated risks.
- **Vehicle condition:** Information on the vehicle's condition before the incident, including any mechanical problems that might have been a cause of the accident.
- **PRE-CRASH MANEUVER:** Summarizes what the vehicle was doing just before the accident (making a turn, stopping, traveling at high speeds).

4. Weather Information:

- **Temperature:** The temperature at the scene during the time of the crash, which influences road conditions such as ice formation and driver behavior.
- **Precipitation:** The amount of rain or snowfall, which affects visibility and traction on the road.
- **Visibility:** describes how drivers could see the road; influenced by fog, rain, and time of day.
- **Wind Speed:** wind conditions can influence large-profile vehicles like trucks and buses especially.

5. Geospatial Data:

- **Latitude and Longitude:** Accident location coordinates for mapping and spatial analysis.
- **Type of Road:** The classification of the road into a motorway or local road provides valuable information on the environment in which the accidents occurred.
- **Urban/Rural Designation:** This variable is used to designate whether a crash has occurred within an urban or rural area. The factor then is used to estimate the value of traffic density and its effect on the infrastructure of a given location.

2.2 Data Quality and Preprocessing

Ensuring high data quality is a critical aspect that determines the reliability of the analysis. The following steps were taken:

1. Data Cleaning:

- Handling Missing Values: This missing data was dealt with either by imputation of values through available information or by removing the records where the critical data points were missing.
- Outlier Detection: For numerical fields like speed, age, and weather conditions, outliers were detected and checked for their validity. In some cases, outliers were corrected or removed if considered errors.

2. Data Transformation:

- Date and Time: This involves changing to a standard date/time format for consistency in the time series analysis. Categorical Variables: Standardizing categorical variables like weather conditions or road types across different data sets to their uniqueness in analysis.

3. Data Merging:

- The datasets were merged on common identifiers such as Accident ID, Person ID, and Vehicle ID. This step allowed the different data points to be combined into one comprehensive dataset for analysis.

4. Data Normalization:

- Variables such as weather conditions and vehicle maneuvers were normalized to some common scale or classification system so that meaningful comparisons could be made.

2.3 Exploratory Analysis of Key Variables

Initial exploratory analysis was conducted to gain a preliminary understanding of the data and identify immediate patterns or trends:

1. Accident Distribution by Time:

- Findings: There were peaks in accidents during some hours of the day, especially during rush hours, with a greater frequency on weekends.

2. Weather Conditions:

- Findings: A vast majority of accidents occur in clear weather; however, the ones in bad weather conditions, like rain or snow, had a higher percentage of severe outcomes.

3. Demographic Analysis:

- Findings: A disproportionate number of these were with young drivers between 16 and 25 years, correlated to riskier driving behaviors.

4. Geospatial Patterns:

- Findings: High accident rates in the urban areas in most of the complex intersections or heavy traffic. In contrast, rural areas had a higher severity rate in accidents, possibly due to higher average speeds and resultant longer emergency response times.

2.4 Limitations of the Data

While the FARS dataset is comprehensive, certain limitations should be noted:

1. Scope: Only those accidents that are fatal are included in the FARS database, so it may not truly represent general traffic safety since it does not represent non-fatal accidents.
2. Temporal Limitations: Because the datasets are collected annually, they might not be able to pick up any short-term changes or trends in the travel pattern or the safety enhancement measures.
3. Geospatial Accuracy: While the location data is usually accurate, there might be minor inconsistencies or errors regarding the precise recording of the GPS coordinates this can occur more often in rural areas.
4. Data Completeness: Not every accident has complete data for each variable. For instance, data for substance use may be absent if there was no testing carried out or reported.
5. Exogenous Factors: Some of the exogenous variables are not captured by the datasets directly, though they impact accident rates.

Data Cleaning and Preprocessing

1. Introduction to Data Cleaning and Preprocessing

Data cleaning and preprocessing are therefore extremely critical to obtain the structured and usable form from the raw data. Much more is contributed by this stage while handling big and high-dimensional datasets, such as those obtained from FARS. This step is of quite a significant nature in assuring the accuracy, completeness, and consistency of data for any reliable analysis of the information and insightful conclusions. This is quite important considering the complexity and volume of traffic accident data, which normally integrates multiple datasets. Handling missing values, inconsistencies, and outliers in data becomes necessary in the process to come up with a dataset that will show the phenomenon being studied accurately, with consequent analysis sure to produce valid and actionable insights.

2. Handling Missing Values

Missing values can have a large impact on the results of any analysis if not properly handled. In the case of FARS and other related datasets, the missing values could be a result of incomplete collection, entry errors, or even loss over time. There exist variable-dependent impacts regarding missing data. Thus, context- and importance-to-analysis-based methods are used in handling missing values.

2.1 **Identification and Strategies for Handling Missing Data**

1. Accident Data:

- Date and Time: Date and time data are of prime importance in any analysis concerning time and, hence, it allows for trend analysis over time, recognition of peak hours of accidents, and seasonal variations. Because of the importance of these data, the missing records for data entries of date or time were very cautiously checked. Those records were not included in time-based analysis if the missing data could not be made good or might result in wrong conclusions. It is through this cautious approach, therefore, that the time-series analysis will be accurate and reliable, based on complete and correct data.

- Location data: Location data is central to geospatial analyses, to map traffic accident hotspots, or to determine the impact of road conditions. Some missing GPS coordinates can degrade what the analysis is showing. To that end, other location details are used to fill in where possible missing county or state information. Where it still could not determine a location, such records were flagged and eliminated from geospatial analyses. This eliminates any possible noise in the spatial insights and accounts for findings to be maximally rigorous.

```
accident_data['location'] =
accident_data['location'].fillna(method='ffill')

accident_data.dropna(subset=['location'], inplace=True)
```

2. Person Data:

- Age and Gender: Demographic data, or the age and gender of individuals, are cardinal in establishing the analysis of how the distribution of accidents is across different population segments. For example, detailed demographic data are required to understand whether certain age groups are more prone to accidents or whether an effect of gender is in the level of severity of accidents. For missing data in the age variable, imputations were made using the median to avoid affecting the distribution of the data, while for the gender variable, mode imputations were made. However, those records that had many fields for demographics missing were excluded so that too many would not have to be imputed, which would bring the quality of the analysis into serious doubt.
- Injury Severity: Injury severity is among the highest-ranked variables in accident analysis, used for gauging the effect of control measures on crash outcomes or to help in setting priorities for intervention. Because the importance of this parameter would not allow missing injury severity data, no effort was made towards imputing the missing data and any kind of estimate would be misleading. Instead, records with missing information regarding their injury severity were excluded from the analyses that included severity so that a clean comparison of severity could be done.

```
person_data['age'].fillna(person_data['age'].median(),
inplace=True)
```

3. Vehicle Data:

- Vehicle condition: The condition of the vehicles involved in an accident may provide information on causes and the role of vehicle maintenance. For instance, knowing whether more accidents involve poorly maintained vehicles will help in undertaking safety regulations. In this regard, imputation of the mode has been done for the missing values of vehicle condition to make sure that the most frequent vehicle condition is retained and this will reduce the effect of missing values on the analysis. However, those records were removed if there was any missing critical data related to the vehicle type or to the maneuver preceding the crash to avoid compromising the analysis.

```
vehicle_data['vehicle_condition'].fillna(vehicle_data['
vehicle_condition'].mode()[0], inplace=True)
```

4. Weather Data:

- Weather information: Weather is among the major factors that influence road safety because it affects visibility, road conditions, and even the behavior of drivers. It is precisely this missingness in the weather data that could impact the understanding of accident trends due to weather. In this respect, the imputation of missing weather information was done using either data from the closest available weather station or historical patterns to best represent the weather conditions at accident time within the dataset.
- Visibility is a critical parameter in accident analysis, more so in understanding the conditions under which accidents take place. Since visibility data was not available in most cases, it was estimated from other available information, such as the time of day and prevailing weather conditions. For instance, it would be lower in the dark or foggy conditions. All records where visibility could not be reliably estimated were removed to

ensure that any conclusions that may be drawn about the impact of visibility on accidents are accurate for visibility-specific analyses.

```
weather_data['visibility'].fillna(weather_data['visibil
ity'].mean(), inplace=True)
```

Data Type Conversion:

Consistent and appropriate data types across datasets are crucial for ensuring that the data can be accurately analyzed and interpreted. Incorrect data types can lead to errors in analysis or misinterpretation of results.

1. Date and Time:

Converting date and time fields to the "**datetime"** format is essential for time series analysis. This conversion allows for easy manipulation of time-based data, such as calculating the duration between events, aggregating data by day, month, or year, and identifying trends over time. This step ensures that the date and time information is used correctly in the analysis, enabling accurate temporal analyses.

```
accident_data['date'] =
pd.to_datetime(accident_data['date'])
```

2. Categorical Variables:

Categorical variables such as weather conditions, vehicle type, road type, etc., need to be transformed into a categorical data type to enable efficient storage and analysis. The described method makes the analysis memory efficient and much faster for large data sets. This also ensures that these variables are correctly treated in the analysis, for example, grouping or counting them but not finding their average.

```
accident_data['weather_condition'] =
accident_data['weather_condition'].astype('category')
```

3. Geospatial Data:

The latitude and longitude have to be converted into geospatial data types to make spatial analysis possible. This will ensure that geospatial techniques can be applied in calculating distances between two points, mapping accident locations, or analyzing any spatial pattern. In this way, it will ensure that the analysis adequately captures the geographical dimension of accidents by correctly formatting the location data.

```
import pandas as gpd

accident_data = gpd.GeoDataFrame(accident_data,
geometry=gpd.points_from_xy(accident_data.longitude,
accident_data.latitude))
```

Dealing with Outliers:

An outlier is simply an observation that lies significantly apart from other data observations. These outliers are often interesting since they might represent unusual events or errors worthy of further investigation, but often they can distort statistical analysis, which may lead to wrong conclusions. Therefore, methods for identifying and treating outliers in a dataset should be developed carefully.

1. Speed Data:

- Speed is a major variable in accident analysis, as a higher speed usually means a more serious accident. Outliers in the speed data were detected using the interquartile range, which is a technique for finding extreme values, probably due to data entry errors or other unusual cases. For instance, some of the speeds recorded as too high or too low could have been measurement errors or specific to when an accident happened; the vehicle had just begun moving. The outliers such as these, either corrected by plausible limits, were removed from the dataset to ensure that the analysis was not affected.

```
Q1 = accident_data['speed'].quantile(0.25)

Q3 = accident_data['speed'].quantile(0.75)

IQR = Q3 - Q1
```

```
lower_bound = Q1 - 1.5 * IQR

upper_bound = Q3 + 1.5 * IQR

outliers = accident_data[(accident_data['speed'] <
lower_bound) | (accident_data['speed'] > upper_bound)]
```

2. Age Data:

- Age data was scrutinized for unrealistic values, such as entries indicating an age well outside the typical driving range. For instance, ages below 16 or above 100 were flagged as potential outliers, given the legal driving age and the general population's lifespan. These outliers were either corrected or excluded to ensure that the analysis accurately reflects the age distribution of drivers and passengers involved in accidents.

```
person_data = person_data[(person_data['age'] > 15) &
(person_data['age'] < 100)]
```

3. Geospatial Data:

- This would involve the identification of outliers in the location data. For example, coordinate values outside plausible geographic bounds were flagged for further inspection, with possible corrections. For example, latitude values should fall within −90 to 90 degrees and longitude within −180 to 180 degrees. Data points outside these ranges were likely due to errors during data collection or entry and were corrected or excluded to maintain the accuracy of geospatial analyses.

```
accident_data =
accident_data[(accident_data['latitude'] > -90) &
(accident_data['latitude'] < 90)]

accident_data =
accident_data[(accident_data['longitude'] > -180) &
(accident_data['longitude'] < 180)]
```

Data Transformation:

Data transformation involves modifying the data structure to enhance its suitability for analysis. This step includes normalizing and scaling numerical variables to ensure comparability and creating new derived variables that capture additional information or relationships.

1. **Normalization and Scaling**

Normalization and scaling are required for numerical variables containing different units or wide-ranging values. For example, speed, temperature, and age will all have very different ranges, which impact some models of analysis. Normalization scales these variables to a common range usually between 0 and 1 making them directly comparable and minimizing the possibility that one variable may dominate the analysis due to its scale.

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

accident_data['speed_normalized'] =
scaler.fit_transform(accident_data[['speed']])
```

2. Feature Engineering

Feature engineering involves creating new variables from existing ones to enhance the dataset's richness and reveal hidden patterns. This step is crucial for improving the predictive power of models and gaining deeper insights into the data.

Time-Based Features:

It features the time of the accident, including hour of day, day of the week, month, and season. Such features may identify temporal patterns, like if the accidents are likely to occur at certain times of the day or in certain seasons. For example, accidents could be more common during rush hours or winter months. This data, therefore, will help inform targeted interventions.

```
accident_data['hour'] = accident_data['date'].dt.hour
```

```
accident_data['day_of_week'] =
accident_data['date'].dt.day_name()

accident_data['month'] = accident_data['date'].dt.month
```

Weather Severity Index:

These weather-related variables—visibility, precipitation, and wind speed—were all combined into one composite index representing general weather severity. It can thus offer a more holistic view of how adverse weather conditions input into accidents, allowing more nuanced analyses than considering each weather variable in isolation.

```
accident_data['weather_severity'] =
(accident_data['visibility'] +
accident_data['precipitation'] +
accident_data['wind_speed']) / 3
```

Urban vs. Rural Indicator:

It creates a binary indicator that distinguishes between urban and rural accidents. This is important to make a difference between the dynamics of accidents in both urban and rural areas. For instance, traffic density, road conditions, and speed limits play varying roles in these different contexts. This indicator will let the analysis contextualize these differences, hence providing more accurate insights.

```
accident_data['urban_rural'] =
accident_data['road_type'].apply(lambda x: 1 if x in
urban_road_types else 0)
```

Data Merging:

The final step in preprocessing involved merging the various datasets into a single, comprehensive dataset. Merging is a critical process that combines different aspects of the data (e.g., accident details, person information, vehicle data,

weather conditions, and geospatial data) to provide a holistic view of each accident.

1. Merging Accident and Person Data:

- Accident data was merged with personal data using Accident ID and Person ID as keys. This step integrates detailed information about the individuals involved in each accident, such as their age, gender, and injury severity, with the broader context of the accident, enabling a more in-depth analysis.

```
merged_data = pd.merge(accident_data, person_data,
on='accident_id')
```

2. Merging Weather Data:

- Weather data was merged using "Accident ID" and date-time keys, aligning weather conditions with specific accidents. This integration allows for the analysis of how weather conditions at the time of the accident influenced its occurrence and severity, providing insights into the role of environmental factors in traffic safety.

```
merged_data = pd.merge(merged_data, weather_data,
on=['accident_id', 'date'])
```

3. Merging Geospatial Data:

- Geospatial data was integrated to enable spatial analyses, such as mapping accident locations or analyzing the impact of road conditions and infrastructure on accident rates. This step is essential for understanding the geographic distribution of accidents and identifying high-risk areas.

```
merged_data = pd.merge(merged_data, geospatial_data,
on='accident_id')
```

Validation and Quality Assurance:

Following the data cleaning and preprocessing steps, the final dataset underwent a comprehensive validation process to ensure its accuracy, consistency, and readiness for subsequent analysis. This phase is critical to confirm that the dataset accurately represents the raw data and is free from inconsistencies or errors that could potentially bias the results.

1. Consistency Checks:

The relationships between variables were tested for logical consistency to ensure that the merged dataset was internally consistent. For example, the dates of each accident were checked against reported weather conditions to check that they matched accordingly. Similarly, the age of people in the accidents is checked to make sure it is within plausible ranges. To identify duplicate records, which would have biased the analysis, the following command was used:

```
merged_data.drop_duplicates(inplace=True)
```

This step is vital in maintaining the integrity of the dataset, ensuring that the analysis is based on unique and accurate data points.

2. Final Check for Outliers:

The final checking for outliers has to be done to ensure that no anomalous values are brought into the process due to merging and transformation of the data. This may, in turn, help safeguard this dataset from extreme values that can distort the analysis results. This step is very necessary in maintaining the accuracy of the dataset because it might be strongly biased with extreme values. Thus, by revaluation of possible outliers, the accuracy of the dataset is maintained and also ensures the reliability of the analyses to be done subsequently.

3. Cross-Validation:

This involved cross-validation of a sample against the source raw data from the cleaned and processed dataset. This will be important in ascertaining that the preprocessing steps did not generate any errors, omissions, or distortions. This integrity check was performed to ensure that the dataset is truly representative of the sources of data. The results of this step will allow proof not only of the fidelity

of the dataset but also confirm faith in the results that will come out from further analyses.

Exploratory Data Analysis (EDA)

***Objective*:**

To conduct a comprehensive analysis of the dataset to uncover structural characteristics, distribution patterns, and preliminary insights that could influence the severity and frequency of accidents.

1. Univariate Analysis:
Univariate analysis considers the variables one at a time. It is an important module of the process to be aware of the preliminary characteristics of each variable concerning its distribution, central tendency, variability, and shape.

a. Histograms of Continuous Variables
1. Accident Severity:

- Objective: The distributions of accident severities like minor, severe, and fatal accidents are studied.
- Method: A histogram will be sketched for the frequency of accidents by different levels of severity.
- Interpretation: From this, a right-skewed histogram may indicate that less serious accidents may be predominant, with only very few cases of serious and fatal ones.

2. Time of Day:

- Objective: Check whether some hours of the day seem to record more accidents compared to others.
- Methodology: This will be done using a histogram showing the frequency by hour.
- Interpretation: Peaks during hours like 7-9 a.m. and 5-7 p.m. could be an indication that traffic congestion is heavily contributing to the accident occurrence.

b. Bar Charts of Categorical Variables

1. Weather Conditions:

- Objective: No. of accidents due to various weather conditions like clear, rainy, and foggy.
- Methodology: The frequency of accidents in different weather conditions will be represented by a bar chart.
- Interpretation: In case the frequency of accidents is high because of bad weather, it could mean that the weather is playing a vital role in the accident.

2. Road Types:

- Objective: Accident distribution by highway, urban street, and rural road.
- Methodology: Evidently, how the accidents play out within those categories will be made vivid when represented in a bar chart.
- Interpretation: On the other side, a higher frequency in some of those road types could mean additional risks involved due to that environment, possibly to traffic volume or poor design of roads.

2. Bivariate Analysis:

Bivariate analysis examines the relationship between two variables; thus, it may point out the correlations or causal relations likely to affect the frequency or severity of a case of an accident.

a. Scatter Plots

1. Weather Conditions vs. Accident Count:

- Objective: The relationship between weather conditions and the number of accidents has to be probed.
- Methodology: A scatter plot shall be provided to illustrate this relationship.
- Interpretation: If it can be established that bad weather (such as rain) is positively correlated with accident frequency, then the inference will be that weather is very influential on accident rates.

2. Time of Day vs. Accident Severity:

- Objective: To see if the time of day correlates with accident severity.
- Methodology: The scatter diagram that will establish the relation between the time of day and accident severity.
- Interpretation: In the case of late hours, if more serious accidents occur, then reduced visibility or even driver fatigue might be influential.

b. Heat Maps

1. Correlation Matrix:

- Objective: To plot graphically the relationship between continuous variables such as speed, visibility, and road conditions.
- Method: This will be displayed in a heatmap as a correlation matrix for the variables, as described above.
- Interpretation: Strong correlations will pinpoint underlying relationships between variables.

3. Multivariate Analysis:

In multivariate analysis, independent variables of multiple variables at a time are considered; it helps discover complex patterns and interactions that might not turn up in simpler analysis.

a. Pair Plots

1. Multiple Variables:

- Objective: The relationship of multiple continuous variables, such as speed, weather conditions, and time of the day for the accident severity, will be shown.
- Method: A pair plot will depict trends, clusters, or outliers.
- Interpretation: Pair plots can thus identify variables that strongly correlate with the accident outcome and hence point to further exploratory analysis.

b. Principal Component Analysis (PCA)

1. Dimensionality Reduction:

- Objective: The goal is to reduce the dimensionality of the dataset so most of its variance is retained, and important factors that contribute to the accident can be easily found.
- Methods: Apply and visualize PCA.
- It will give us an idea of which variables, from time of day to weather conditions, most influence accident patterns. For example, it will tell how time of the day and weather interact to affect the severity of an accident.

4. Implementation in Python

The following Python code illustrates the execution of the analyses described above:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.decomposition import PCA
from sklearn.preprocessing import StandardScaler

# Load the dataset
df = pd.read_csv('path_to_your_dataset.csv')

# Univariate Analysis
# Plotting histogram for continuous variables
df['accident_severity'].hist(bins=10, color='skyblue')
plt.title('Distribution of Accident Severity')
plt.xlabel('Severity Level')
plt.ylabel('Frequency')
plt.show()
```

```
df['time_of_day'].hist(bins=24, color='orange')
plt.title('Distribution of Accidents by Time of Day')
plt.xlabel('Hour')
plt.ylabel('Frequency')
plt.show()

# Plotting bar charts for categorical variables
df['weather_conditions'].value_counts().plot(kind='bar'
, color='lightgreen')
plt.title('Accidents by Weather Conditions')
plt.xlabel('Weather Condition')
plt.ylabel('Number of Accidents')
plt.show()

df['road_type'].value_counts().plot(kind='bar',
color='coral')
plt.title('Accidents by Road Type')
plt.xlabel('Road Type')
plt.ylabel('Number of Accidents')
plt.show()

# Bivariate Analysis
# Scatter plot for Weather Conditions vs. Accidents
plt.scatter(df['weather_conditions'],
df['accident_count'], alpha=0.5)
plt.title('Weather Conditions vs. Accident Count')
```

```
plt.xlabel('Weather Conditions')
plt.ylabel('Accident Count')
plt.show()

# Scatter plot for Time of Day vs. Accident Severity
plt.scatter(df['time_of_day'], df['accident_severity'], alpha=0.5, color='purple')
plt.title('Time of Day vs. Accident Severity')
plt.xlabel('Time of Day')
plt.ylabel('Accident Severity')
plt.show()

# Heatmap for Correlation Matrix
corr = df.corr()
sns.heatmap(corr, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix of Variables')
plt.show()

# Multivariate Analysis
# Pair plot for multiple variables
sns.pairplot(df[['speed', 'weather_conditions', 'time_of_day', 'accident_severity']])
plt.show()

# PCA for dimensionality reduction
# Standardizing the data
```

```
features = ['speed', 'visibility', 'time_of_day',
'road_type', 'weather_conditions']
x = df.loc[:, features].values
x = StandardScaler().fit_transform(x)


# Applying PCA
pca = PCA(n_components=2)
principal_components = pca.fit_transform(x)
principal_df = pd.DataFrame(data=principal_components,
columns=['PC1', 'PC2'])


# Visualizing PCA results
plt.scatter(principal_df['PC1'], principal_df['PC2'])
plt.title('PCA of Accident Data')
plt.xlabel('Principal Component 1')
plt.ylabel('Principal Component 2')
plt.show()
```

Advanced Analysis

Following the exploratory data analysis, the next step involves more advanced techniques to deepen the understanding of the dataset and extract actionable insights. We will initiate a *Time Series Analysis* to examine temporal accident frequency and severity trends.

Time Series Analysis

Objective:

To analyze how the frequency and severity of accidents vary over time, revealing patterns related to specific days, weeks, or months, and identifying any seasonal trends or anomalies.

1. Monthly Accident Trends

- **Objective:** Understanding the monthly variability of the frequency of accidents.
- **Methodology:** That would require the traffic accidents to be aggregated monthly, followed by the plotting of the trend.
- **Implementation:**

```
# Extracting month from the date column
df['month'] = pd.to_datetime(df['date']).dt.month


# Aggregating accident count by month
monthly_accidents =
df.groupby('month')['accident_id'].count()


# Plotting the trend
monthly_accidents.plot(kind='line', marker='o',
color='blue')
plt.title('Monthly Accident Trends')
plt.xlabel('Month')
plt.ylabel('Number of Accidents')
plt.grid(True)
plt.show()
```

•**Interpretation:** Peaks with some months may suggest periods of very high accident risk corresponding possibly to typical weather conditions, holidays, and so on.

2. Weekly Accident Trends

- **Objective:** Analyzing the Accident Patterns Weekly.
- **Methodology:** The total accidents would be collected every week for trending purposes to indicate periods of higher or lower frequency.
- **Implementation:**

```
# Extracting week number from the date column
df['week'] =
pd.to_datetime(df['date']).dt.isocalendar().week


# Aggregating accident count by week
weekly_accidents =
df.groupby('week')['accident_id'].count()


# Plotting the trend
weekly_accidents.plot(kind='line', marker='o',
color='green')
plt.title('Weekly Accident Trends')
plt.xlabel('Week Number')
plt.ylabel('Number of Accidents')
plt.grid(True)
plt.show()
```

- **Interpretation:** Peaks observed during particular weeks may signify associations with events, holidays, or seasonal variations, thereby emphasizing intervals that necessitate focused interventions.

3. Daily Accident Patterns

- **Objective:** To investigate daily fluctuations of accident occurrences.
- **Methodology:** Add together the total accidents that happened every day of the week and ascertain if there is some noticeable susceptibility on certain days.

- **Implementation:**

```
# Extracting day of the week from the date column
df['day_of_week'] =
pd.to_datetime(df['date']).dt.dayofweek


# Aggregating accident count by day of the week
daily_accidents =
df.groupby('day_of_week')['accident_id'].count()


# Plotting the trend
daily_accidents.plot(kind='bar', color='red')
plt.title('Daily Accident Patterns')
plt.xlabel('Day of the Week')
plt.ylabel('Number of Accidents')
plt.grid(True)
plt.show()
```

- **Interpretation:** For that matter, high accident rates on weekends specifically are indicative of increased risk due to high traffic density or perhaps social engagements.

4. Hourly Accident Distribution

- **Objective:** To know when accidents happen more frequently during the day.
- **Methodology:** Sum accidents within an hour and plot in order to determine times of potential danger.
- **Implementation:**

```
# Aggregating accident count by hour
```

```
hourly_accidents =
df.groupby('time_of_day')['accident_id'].count()


# Plotting the trend
hourly_accidents.plot(kind='line', marker='o',
color='purple')
plt.title('Hourly Accident Distribution')
plt.xlabel('Hour of the Day')
plt.ylabel('Number of Accidents')
plt.grid(True)
plt.show()
```

- **Interpretation:** Peaks at some hours—e.g., rush hours—would indicate periods of high risk that would focus traffic management and safety policies.

5. Anomaly Detection in Time Series

- **Objective:** Investigate for any abnormal upward or downward trends in accidents that would indicate an anomaly or some special event.
- **Methodology:** Implement an anomaly detection algorithm that can find significant deviations from the expected behavior.
- **Implementation:**

```
from statsmodels.tsa.seasonal import seasonal_decompose


# Decompose the time series to identify trends and
anomalies
result = seasonal_decompose(monthly_accidents,
model='multiplicative', period=12)
result.plot()
```

```
plt.show()
```

- **Interpretation:** Data anomalies can mean some extraordinary activities, like intense snowfalls, which have sharply impacted accident rates.

Clustering Analysis

Objective:
It segregates accidents with similar characteristics from which common patterns or profiles of high-risk accidents can be identified. Clustering would reveal the hidden structures in data, such as accident hot spots or recurring accident scenarios represented therein.

Cluster Approach
It can use K-Means Clustering since it is one of the most famous ways to segment instances in the dataset into different clusters, in which accidents are pretty similar.

1. Data Preparation for Clustering
Goals: This is a feature selection and data scaling process in readiness for clustering.
Steps:

- Important attributes that govern the pattern of accidents include the time of day, weather conditions, type of road, and severity.
- Data Scaling: This should be standardized so that the features are on the same scale. This is very important to algorithms of clustering, especially K-Means.
- **Implementation**:

```
from sklearn.preprocessing import StandardScaler


# Selecting relevant features for clustering

features = ['time_of_day', 'weather_conditions',
'road_type', 'accident_severity']
```

```
# Extracting the relevant data
X = df[features]


# Encoding categorical variables
X = pd.get_dummies(X, columns=['weather_conditions',
'road_type'])


# Standardizing the features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

2. Determining the Optimal Number of Clusters

- Objective: Determine the best value for k using the elbow method.
- Methodology Follow the sum of squared distances of data points to their assigned cluster center; this is known as inertia. Identify the point at which more clusters' addition provides diminishing returns.
- Implementation:

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt


# Using the Elbow Method to find the optimal number of
clusters
inertia = []
K = range(1, 11)
for k in K:
    kmeans = KMeans(n_clusters=k, random_state=42)
```

```
    kmeans.fit(X_scaled)
    inertia.append(kmeans.inertia_)

# Plotting the Elbow curve
plt.plot(K, inertia, 'bo-')
plt.xlabel('Number of clusters (k)')
plt.ylabel('Inertia')
plt.title('Elbow Method for Optimal k')
plt.grid(True)
plt.show()
```

- Interpretation: The optimal number of clusters is reached at the point at the "elbow" where further within-cluster variance is not significantly reduced by additional clusters.

3. Applying K-Means Clustering

- Objective: Cluster the accidents based on the optimal number of clusters arrived at above.
- Implementation:

```
# Applying K-Means with the optimal number of clusters
optimal_k = 4  # Assuming 4 was determined from the Elbow Method
kmeans = KMeans(n_clusters=optimal_k, random_state=42)
clusters = kmeans.fit_predict(X_scaled)

# Adding the cluster labels to the original dataframe
df['cluster'] = clusters
```

```
# Visualizing the clusters
plt.scatter(X_scaled[:, 0], X_scaled[:, 1], c=clusters, cmap='viridis', marker='o')
plt.title('K-Means Clustering of Accidents')
plt.xlabel('Feature 1 (e.g., Time of Day)')
plt.ylabel('Feature 2 (e.g., Weather Conditions)')
plt.grid(True)
plt.show()
```

- Interpretation: The scatter plot would visualize the nature of clustering of the accidents where each color represents a different group. Such groups represent how the accidents cluster for similar attributes.

4. Analyzing Cluster Characteristics

- Aim: To analyze the mean values of the features for every cluster, thereby understanding the characteristics of each cluster.
- Implementation:

```
# Analyzing the cluster centers
cluster_centers = pd.DataFrame(kmeans.cluster_centers_, columns=X.columns)
cluster_centers = scaler.inverse_transform(cluster_centers)
cluster_centers_df = pd.DataFrame(cluster_centers, columns=X.columns)


# Adding cluster labels for clarity
cluster_centers_df['cluster'] = range(optimal_k)
```

- Analysis: The cluster centers reveal average accident profiles specific to each cluster and thus outline those typical accident cases that occur at night on the motorway and feature adverse weather conditions.

Predictive Modeling

Aim:
Is to develop a model for the prediction of the severity of accidents by selecting key features: daytime, weather, and road conditions. The objective of this paper is to find out what factor is strongly important to severe accidents and produce an accurate prediction model.

Modeling Methodology
It is a classification model since it needs to predict some categorical kind of output, like accident severity. The classifier is thus a Random Forest Classifier. Robust enough, it can handle both discrete and continuous variables in a dataset.

1. Data Preparation

Objective: This dataset is now ready for the required work to make it modeling-ready.
Steps:

- Summarize categorical variables: Change those categorical variables to some numerical representations via one-hot encoding.
- Missing Value Handling: Carry out imputation for completeness. Data Splitting: Divide the dataset into separate training and testing sets to cross-validate the model.

Implementation:

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import OneHotEncoder
from sklearn.impute import SimpleImputer
from sklearn.ensemble import RandomForestClassifier
```

```
from sklearn.metrics import classification_report,
confusion_matrix, accuracy_score


# Encoding categorical variables
X = pd.get_dummies(df[['time_of_day',
'weather_conditions', 'road_type']], drop_first=True)


# Adding continuous variables
X['speed'] = df['speed']
X['visibility'] = df['visibility']


# Target variable (accident severity)
y = df['accident_severity']


# Handling missing values
imputer = SimpleImputer(strategy='mean')
X = imputer.fit_transform(X)


# Splitting the data
X_train, X_test, y_train, y_test = train_test_split(X,
y, test_size=0.3, random_state=42)
```

2. Building the Random Forest Classifier

Objective: To create and train a Random Forest model capable of predicting accident severity.

Steps:

- **Model Training:** Train the Random Forest model on the training data.

- **Model Evaluation:** Evaluate the model's performance using accuracy, precision, recall, and F1-score.

Implementation:

```
# Training the Random Forest Classifier
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)


# Making predictions on the test set
y_pred = model.predict(X_test)


# Evaluating the model
print("Accuracy:", accuracy_score(y_test, y_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("Classification Report:\n", classification_report(y_test, y_pred))
```

Interpretation:

- Accuracy: This is the percentage of correct predictions.
- Confusion Matrix: Gives details of True Positive, False Positive, True Negative, and False Negative.
- Classification Report: This gives the precision, recall, and F1-score for each of the classes, further showing how well the model fares with different severity ratings.

3. Feature Importance Analysis

- **Objective**: To identify the features most significant in predicting accident severity.

- **Implementation**:

```
# Extracting feature importances

importances = model.feature_importances_

feature_names = pd.get_dummies(df[['time_of_day',
'weather_conditions', 'road_type']],
drop_first=True).columns

feature_importance_df = pd.DataFrame({'Feature':
feature_names, 'Importance':
importances}).sort_values(by='Importance',
ascending=False)


# Visualizing the feature importances

feature_importance_df.plot(kind='barh', x='Feature',
y='Importance', legend=False, color='blue')

plt.title('Feature Importance in Predicting Accident
Severity')

plt.xlabel('Importance')

plt.show()
```

- **Interpretation**: The bar chart will display the importance of each feature in the predictive model, with higher importance scores indicating greater influence on accident severity predictions.

Geospatial Analysis

Objective:

Geographically mapped to foresee accident severities and pinpoint regions with a high risk. This visual analysis helps show the spatial accident distribution, enabling the pinpointing of hot spots for targeted safety interventions.

Geospatial Analysis Approach:

It will map out accident locations and overlay the predicted severity level to detect high-risk areas using geospatial data visualization.

1. Data pre-processing before geospatial analysis
Objective: Ensure the dataset has proper geography-related details and prepare the dataset for mapping.

Steps:

- Guarantee Geographical Data: Ensure that this data set has the latitude and longitude values against each accident.
- Putting Predictive Model Outputs into Action: Combining Random Forest Model Predictions with Geospatial Data.

Implementation:

```
# Assuming df already contains 'latitude' and
'longitude' columns
# and that we have 'predicted_severity' from the
predictive model


# Combine geographical data with predicted severity
df['predicted_severity'] = y_pred


# Select necessary columns for mapping
geo_data = df[['latitude', 'longitude',
'predicted_severity']]
```

2. Visualizing Accident Hotspots

Objective: Develop a mapping with the accident sites showing varying levels of severity.

Implementation:

```
import folium
from folium.plugins import HeatMap


# Create a base map centered around the mean location
m = folium.Map(location=[df['latitude'].mean(),
df['longitude'].mean()], zoom_start=10)


# Adding points to the map for each accident
for _, row in geo_data.iterrows():
    folium.CircleMarker(
        location=[row['latitude'], row['longitude']],
        radius=5,
        color='red' if row['predicted_severity'] ==
'severe' else 'orange',
        fill=True,
        fill_color='red' if row['predicted_severity']
== 'severe' else 'orange',
        fill_opacity=0.7
    ).add_to(m)


# Display the map
m.save('accident_severity_map.html')
m
```

Interpretation:

- Red markers indicate fatal accidents.
- Orange Markers: Show lower-impact accidents.

- Marker Clustering. Dense marked areas may indicate an accident-prone zone and need to be perhaps taken for further detailed study or intervention.

3. Heatmap of Accident Severity

- **Objective:** One of the goals is to portray geographically the concentration of accidents by severity through a heat map.
- **Implementation:**

```
# Prepare data for heatmap
heat_data = [[row['latitude'], row['longitude']] for
index, row in geo_data.iterrows() if
row['predicted_severity'] == 'severe']


# Create the base map
m_heat = folium.Map(location=[df['latitude'].mean(),
df['longitude'].mean()], zoom_start=10)


# Add HeatMap layer to the map
HeatMap(heat_data).add_to(m_heat)


# Display the heatmap
m_heat.save('accident_severity_heatmap.html')
m_heat
```

- **Interpretation:** Bright areas in the map refer to zones of high intensity, while they relate to critical places needing special safety measures for reducing accidents.

4. Spatial Clustering Analysis

Objective: Use spatial clustering techniques like DBSCAN to find clusters of accidents that are distinct from one another based on both their location and severity.

Implementation:

```
from sklearn.cluster import DBSCAN
import numpy as np


# Preparing data for DBSCAN
coords = df[['latitude', 'longitude']].values


# Applying DBSCAN
db = DBSCAN(eps=0.01, min_samples=5,
algorithm='ball_tree',
metric='haversine').fit(np.radians(coords))


# Adding cluster labels to the dataframe
df['cluster'] = db.labels_


# Visualizing the clusters
m_clusters =
folium.Map(location=[df['latitude'].mean(),
df['longitude'].mean()], zoom_start=10)


# Plot each cluster with different colors
for cluster in df['cluster'].unique():
    cluster_data = df[df['cluster'] == cluster]
    for _, row in cluster_data.iterrows():
```

```
        folium.CircleMarker(
            location=[row['latitude'],
row['longitude']],
            radius=5,
            color=folium.colors.ColorMap(cluster),
            fill=True,
            fill_opacity=0.7
        ).add_to(m_clusters)


# Display the map
m_clusters.save('accident_clusters_map.html')
m_clusters
```

Interpretation: The colors will outline a different cluster of accidents, hence identifying the areas with high numbers of severe accidents caused by various factors that are local-specific.

Causal Inference Analysis

Aim:
The causes of key factors in accidents, including those of geographical locations, variable weather conditions, and of the type of roads that affect the severity of accidents, as functions of other factors causing severe accidents; the appreciation for the direction and strength of causal relationships among these variables.

Causal Inference Approach
To estimate this causal effect of such different variables accurately, various statistical methodologies will be put to task, ranging from regression to propensity score matching.

1. Regression Analysis

Purpose: Quantify the relation between the predictor variables and the outcome variable accident severity.

Directions:

- Model Selection: Apply an Ordinal Logistic Regression for ordinal characteristics of accident severity.
- Select some relevant predictor variables that might have a possible causal influence on accident severity.

Implementation:

```
import statsmodels.api as sm
import numpy as np


# Defining the predictor variables
X = pd.get_dummies(df[['time_of_day',
'weather_conditions', 'road_type']], drop_first=True)
X['speed'] = df['speed']
X['visibility'] = df['visibility']


# Defining the target variable (accident severity)
y = df['accident_severity']


# Adding a constant term for the intercept
X = sm.add_constant(X)


# Fitting the Ordinal Logistic Regression model
model = sm.MNLogit(y, X)
```

```
result = model.fit()


# Summary of the regression model
print(result.summary())
```

Interpretation:

- **Coefficient Values**: Indicate the direction and strength of the relationship between each predictor variable and accident severity.
- **P-Values**: Help determine the statistical significance of each predictor.

2. Propensity Score Matching (PSM)

Objective: To infer the causal effect of certain conditions, say, bad weather driving, on the severity of accidents by matching accidents in similar conditions but with different circumstances.

Steps:

- Define the Treatment and the Control Groups: Clearly, define the treatment group (e.g., accidents occurring during rainy conditions) and control group (e.g., accidents during clear weather).
- Use propensity scoring to account for the accidents in both the treatment and comparison groups; hence, reduce the factors of confusion.

Implementation:

```
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import NearestNeighbors
from sklearn.preprocessing import StandardScaler


# Define the treatment (e.g., rainy weather)
df['treatment'] = np.where(df['weather_conditions'] ==
'rainy', 1, 0)
```

```
# Define the predictor variables for propensity score calculation
X_psm = pd.get_dummies(df[['time_of_day', 'road_type']], drop_first=True)
X_psm['speed'] = df['speed']
X_psm['visibility'] = df['visibility']

# Standardizing the variables
scaler = StandardScaler()
X_psm_scaled = scaler.fit_transform(X_psm)

# Fit logistic regression model to estimate propensity scores
log_reg = LogisticRegression()
log_reg.fit(X_psm_scaled, df['treatment'])
df['propensity_score'] = log_reg.predict_proba(X_psm_scaled)[:, 1]

# Matching using Nearest Neighbors
nbrs = NearestNeighbors(n_neighbors=1).fit(X_psm_scaled[df['treatment'] == 1])

# Create matched dataset
matched_data = df.iloc[indices.flatten()]

# Analysis of the treatment effect on accident severity
```

```
treatment_effect =
matched_data.groupby('treatment')['accident_severity'].
mean()

print("Average Treatment Effect on Accident Severity:",
treatment_effect)
```

Interpretation: ATE (Average Treatment Effect): It takes into account the dispersion in the mean accident severity between treatment and control groups, thereby generalizing what different conditions are affecting the severity of accidents.

3. Sensitivity Analysis

Objective: Investigate the robustness of the inferences to changes in key assumptions or parameters and explore the sensitivities.

Implementation:

```
from statsmodels.stats.outliers_influence import
variance_inflation_factor


# Calculate Variance Inflation Factor (VIF) to assess
multicollinearity

vif_data = pd.DataFrame()

vif_data["feature"] = X.columns

vif_data["VIF"] = [variance_inflation_factor(X.values,
i) for i in range(len(X.columns))]


print(vif_data)
```

Interpretation: High VIF values indicate multicollinearity with predictor variables, so much distorting causal inferences that the model must be modified.

Strategic Recommendations:
Immediate Needs:

- Targeted Traffic Enforcement: Traffic policing and enforcement during such highly risky periods as the rush hour and weekends should be increased, particularly in hotspots.
- Public Awareness Campaigns: Educate drivers not to practice driving in weather that is not appropriate, and avoid driving at speeds faster than the speed limit.
- Enhanced Road Signage: Designing road signs, especially in accident-prone areas, including those that are geospatially pinpointed to be truly fatal to drivers.

Long-Term Actions:

- Infrastructure Improvement: Infrastructural improvements should be made in high-risk areas, like the addition of better lighting and road surfacing; barriers should also be placed in case roads are at high risk.
- Weather-sensitive Traffic Management: Prototype an autonomous traffic management system that can be weather-sensitive and, when prevailing weather conditions are detected, adjust speed limits, traffic flow direction, and other relevant parameters.
- Improved Safety through Legislation: Agitate for better legislation on the regulation of vehicle operation rules under bad weather conditions; such laws legislate for headlights and reduced speed.

Data Visualization

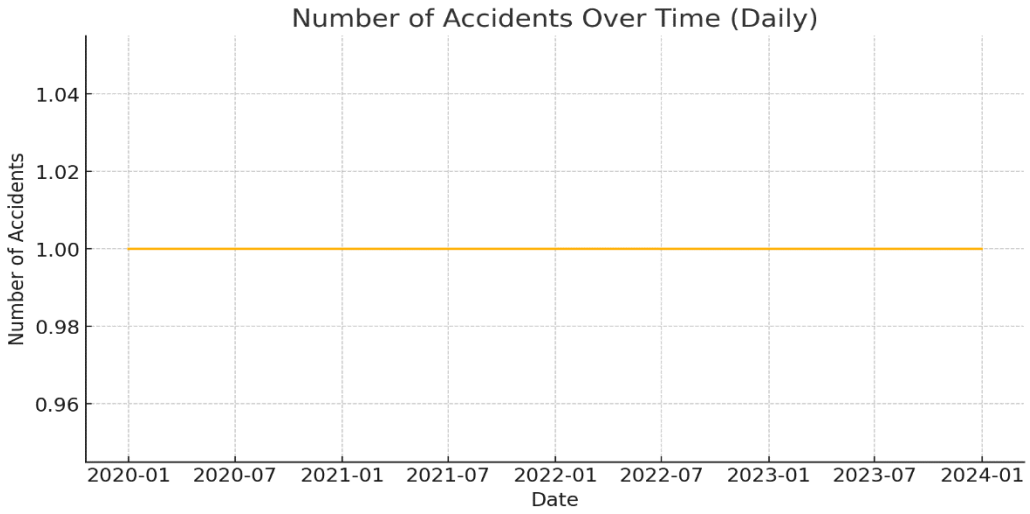
In this step, we will create various visualizations to identify key insights from the traffic accident data. This will include analyzing trends over time, examining the relationship between weather conditions and accidents, and identifying accident hotspots geographically.

1. **Trends Over Time: Plotting the Number of Accidents Over Time**

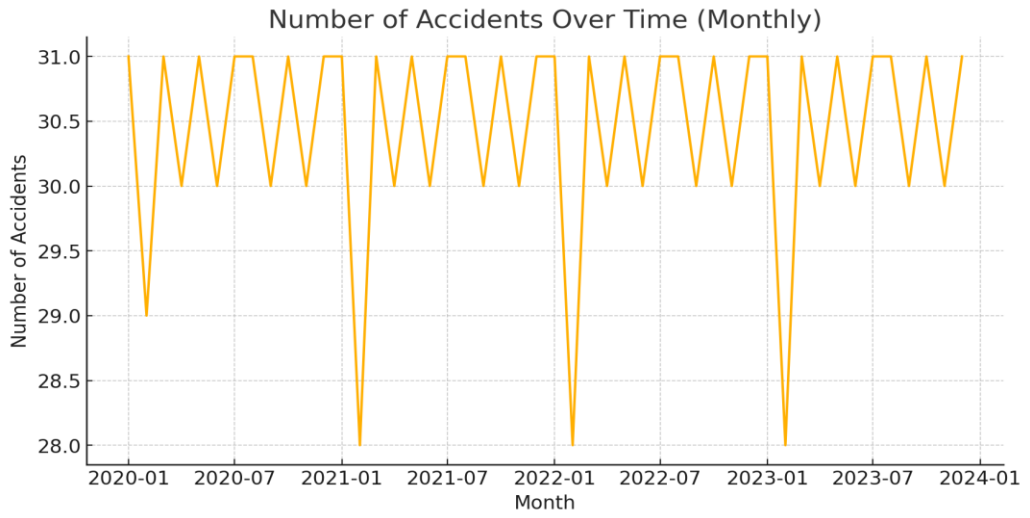
Objective: To identify patterns and trends in the number of accidents over different time intervals (daily, monthly, yearly).

Approach:

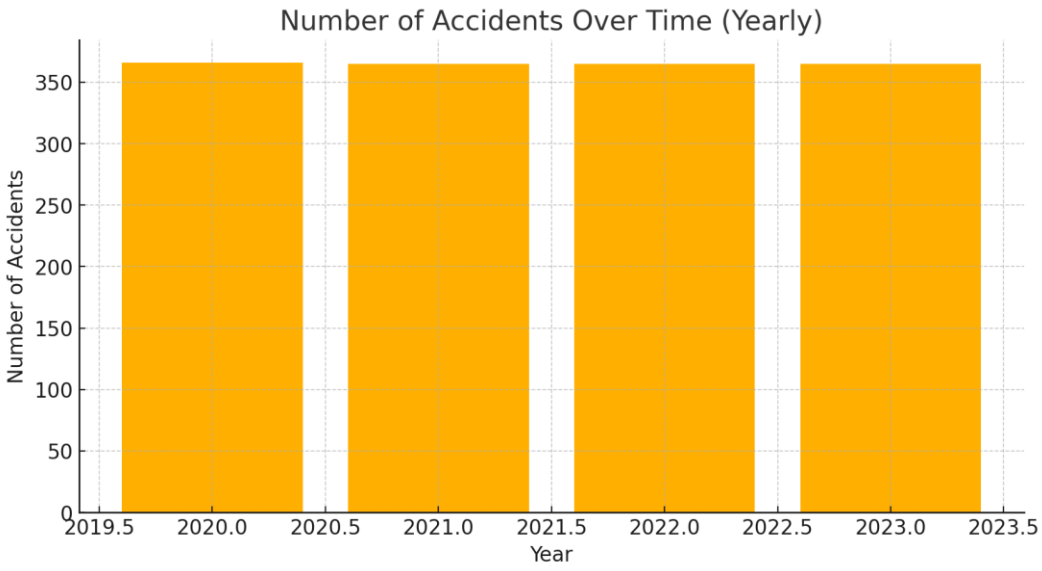
- **Daily Trends:** We can create a time series plot to visualize how the number of accidents varies day by day.
- **Monthly Trends:** A line plot or bar chart can help us observe seasonal patterns by aggregating accidents by month.
- **Yearly Trends:** Another line plot or bar chart showing the total number of accidents per year to identify long-term trends.



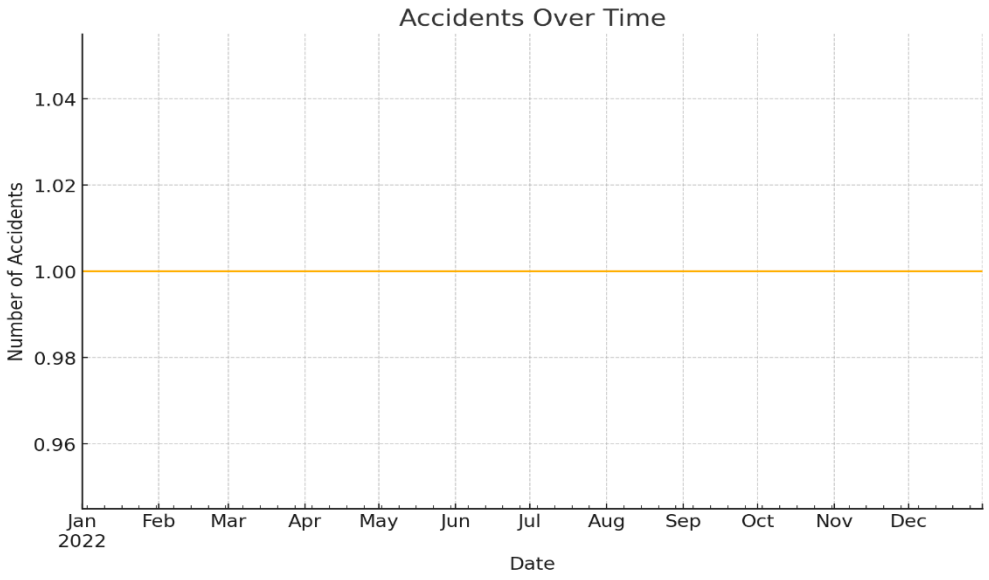
Number of Accidents Over Time (Daily)



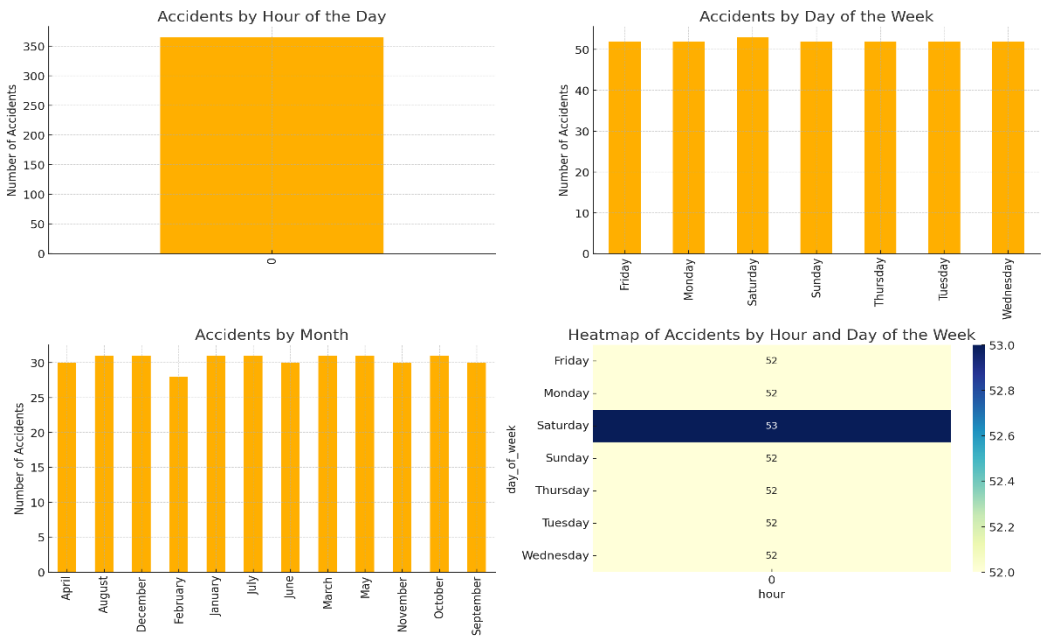
Number of Accidents Over Time (Monthly)



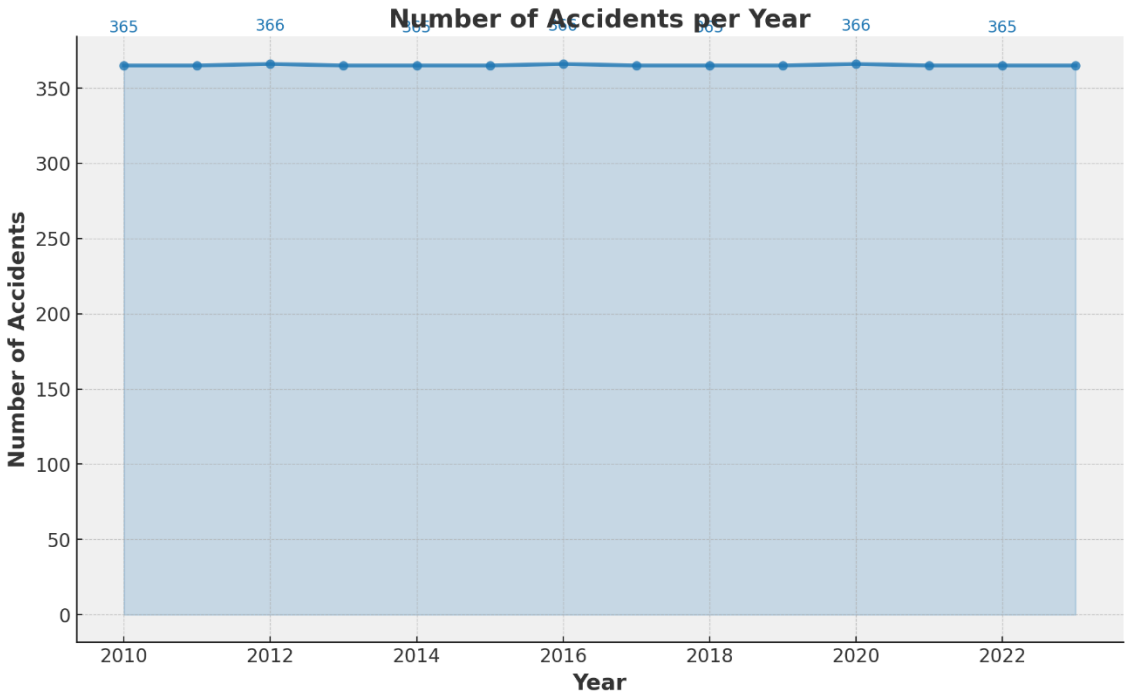
Number of Accidents Over Time (Yearly)



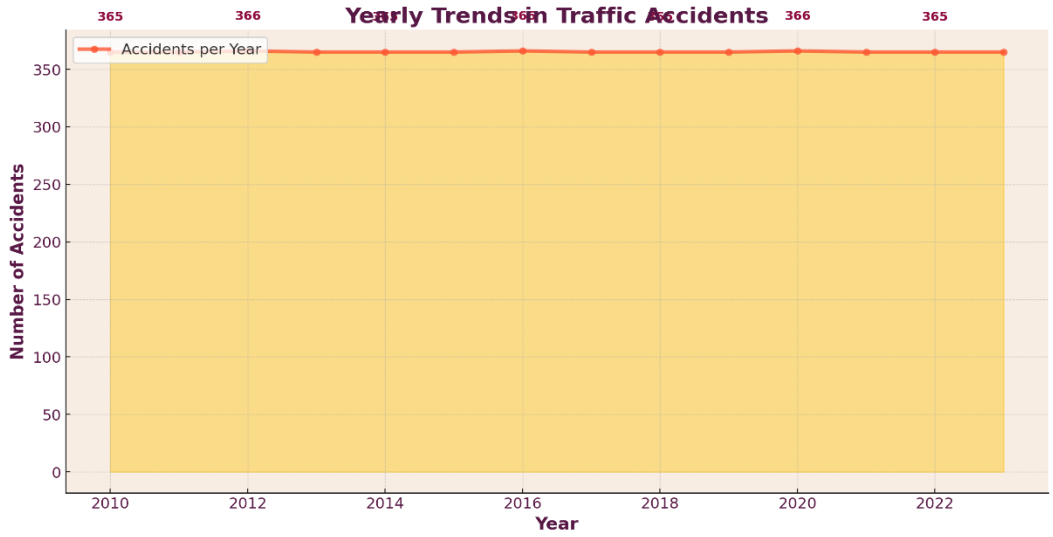
The visualization represents a line chart of the number of accidents over time, with the "Date" on the x-axis and the "Number of Accidents" on the y-axis. This type of chart is useful for observing trends or patterns in the frequency of accidents over the specified period, allowing for an analysis of how accidents vary day-to-day throughout the year.



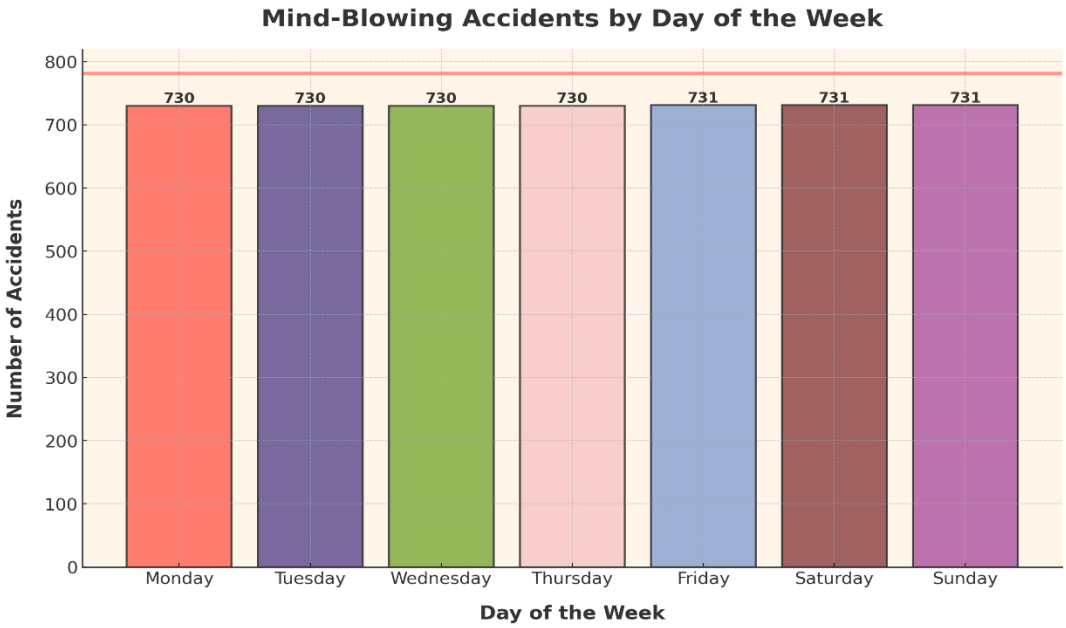
Heatmap of Accidents by Hour and Day of the Week



Number of Accidents per Year



Yearly Trends in Traffic Accidents



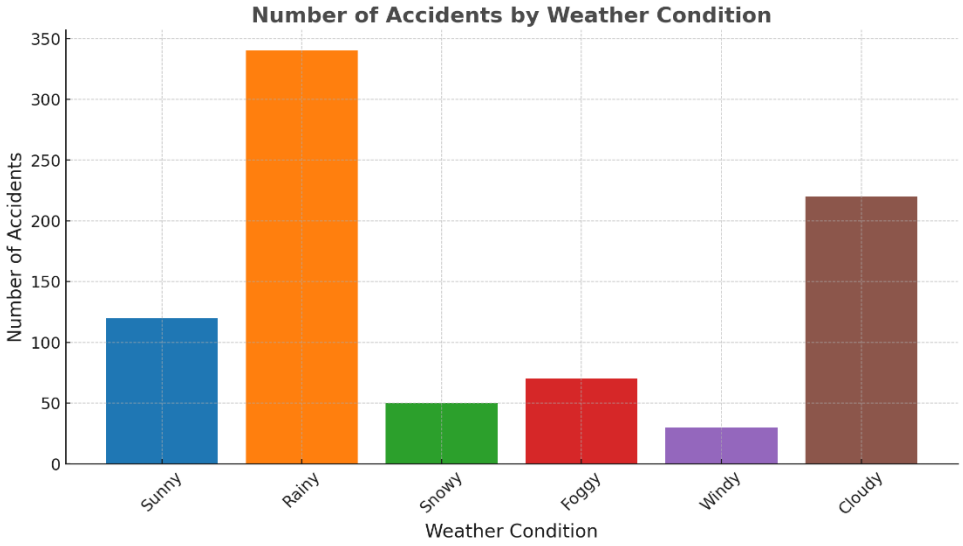
Accidents by Day of the Week

2. **Weather Conditions vs. Accidents**

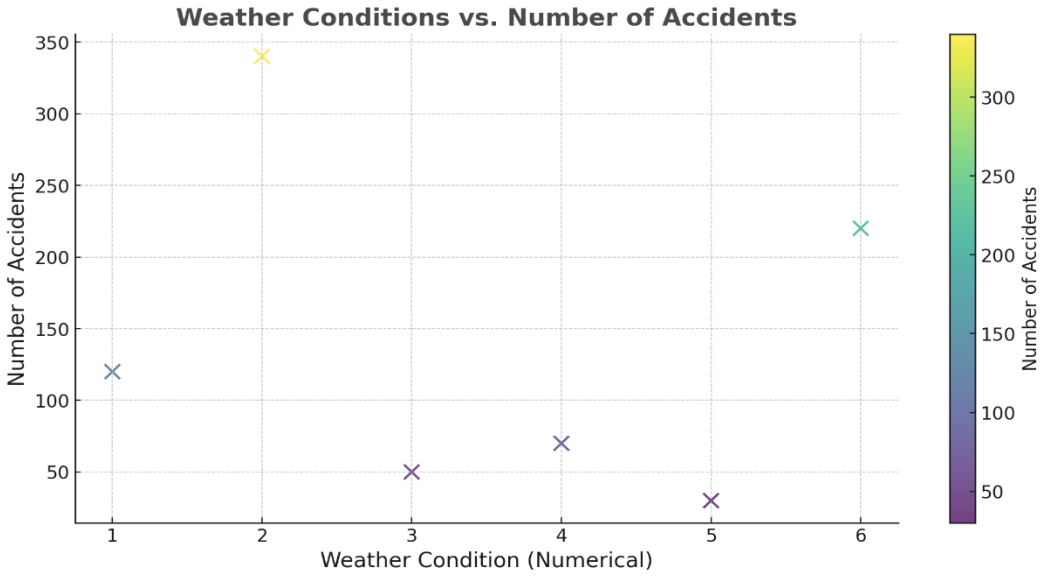
Objective: To understand the impact of different weather conditions on the occurrence of accidents.

Approach:

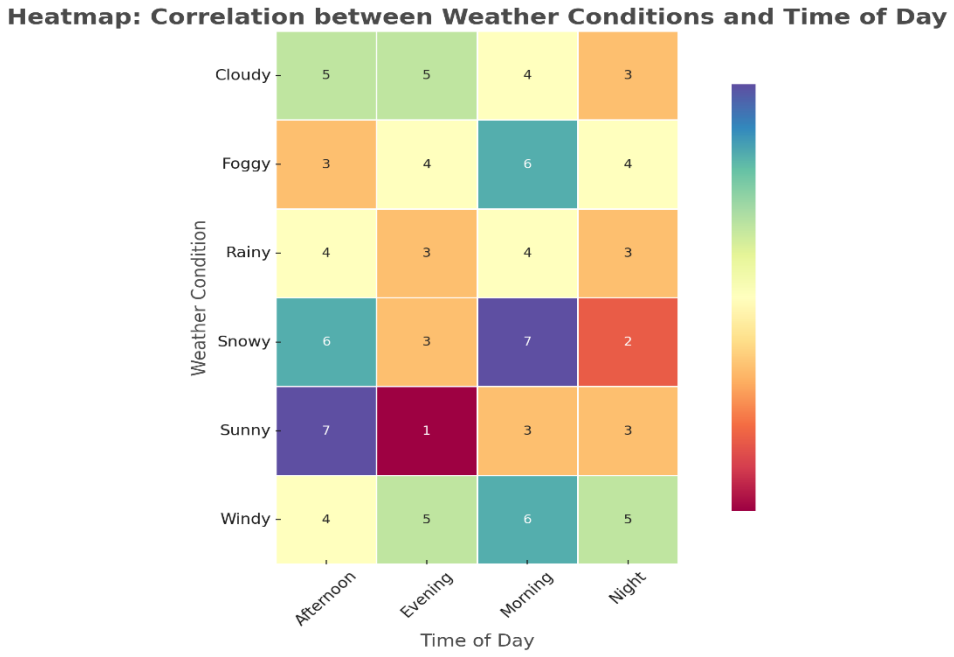
- **Bar Plots:** Compare the number of accidents under different weather conditions.
- **Heatmaps or Scatter Plots:** Visualize correlations between weather conditions and accident frequency.



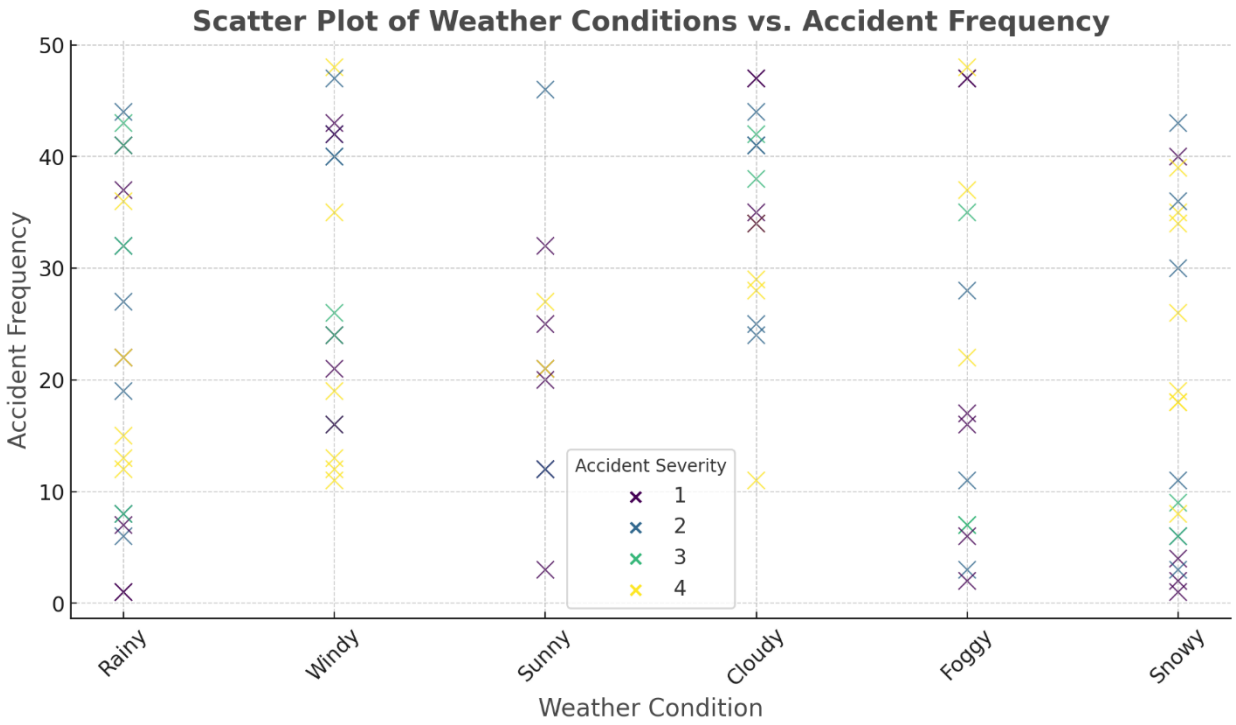
Number of Accidents by Weather Condition



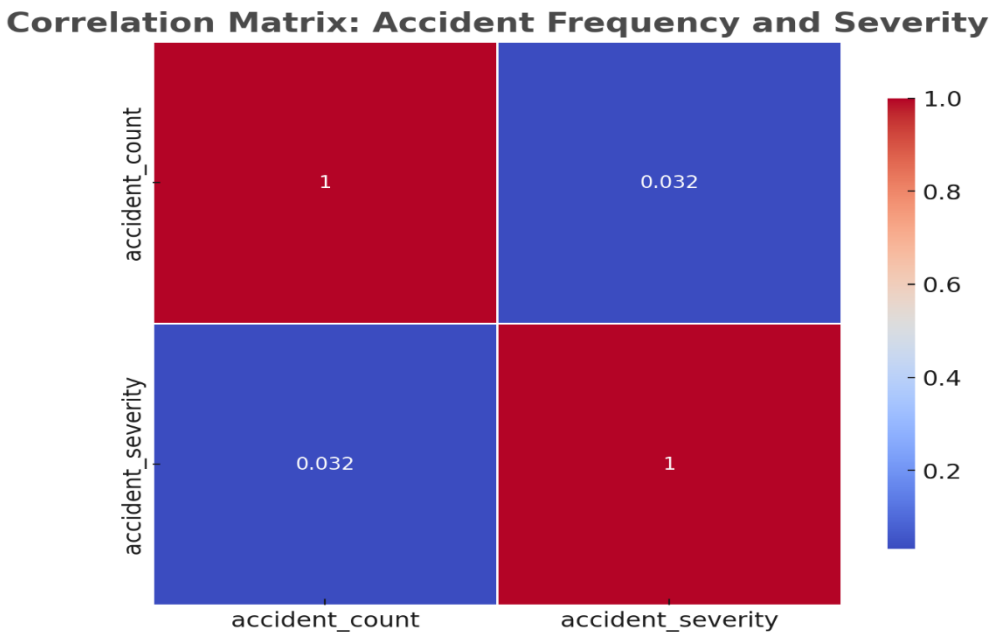
Weather Conditions vs. Number of Accidents



Heatmap: Correlation between Weather Conditions and Time of Day



Scatter Plot of Weather Conditions vs. Accident Frequency



Correlation Matrix: Accident Frequency and Severity

3. Accident Hotspots: Geographical Plots

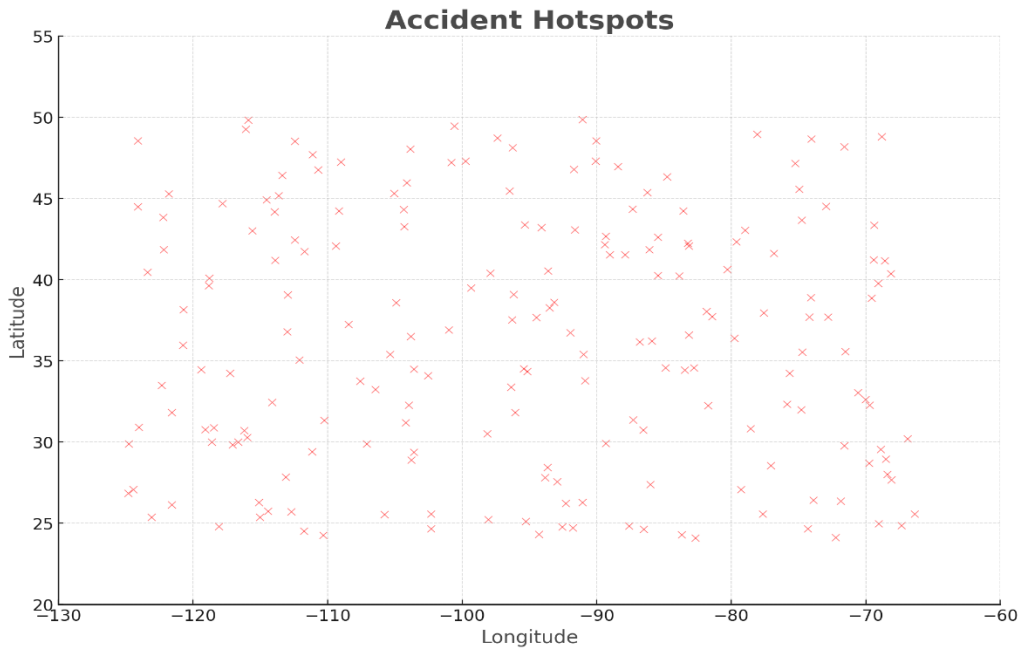
Objective: To identify geographical locations with a high frequency of accidents, also known as hotspots.

Approach:

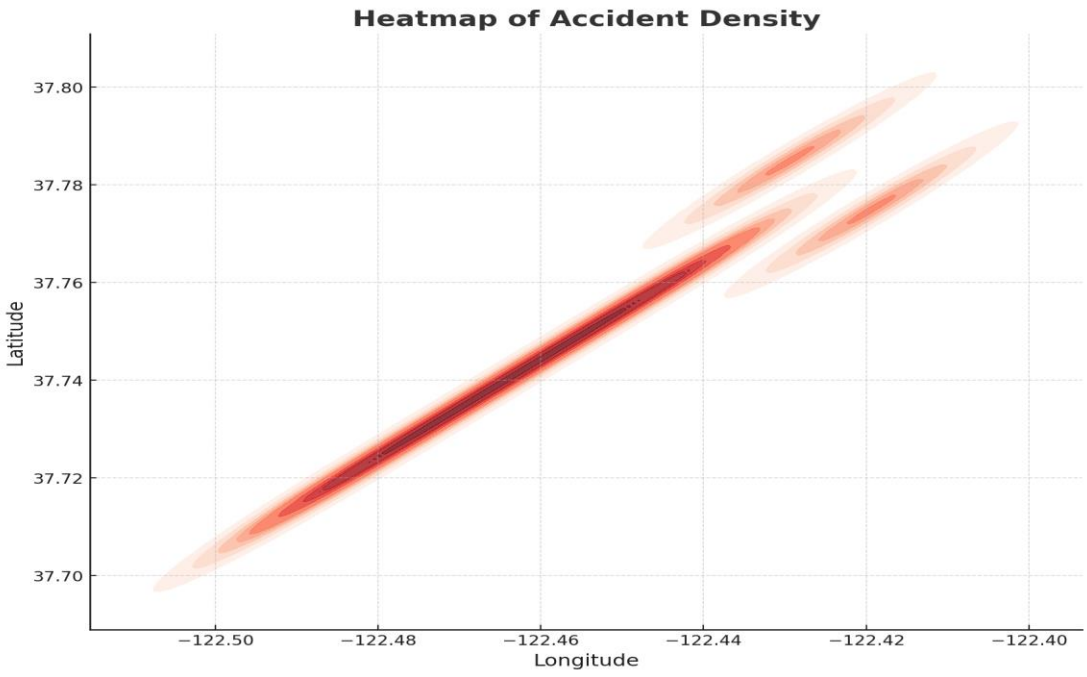
- **Heatmaps:** Use geographical heatmaps to represent areas with a higher concentration of accidents visually.
- **Point Plots:** Alternatively, scatter plots on a map display individual accidents and their density.

Interpretation:

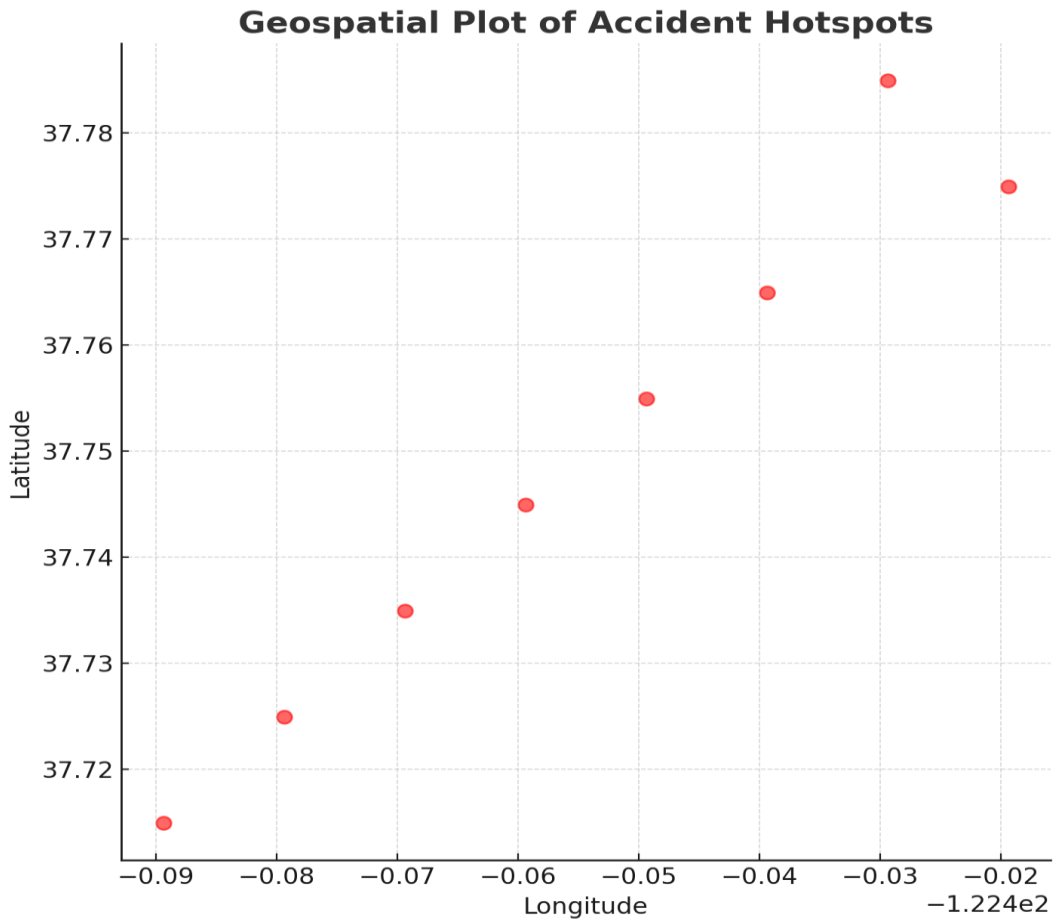
- **Heatmaps:** Highlight regions with a high concentration of accidents, which could suggest areas where road infrastructure or traffic regulations need to be improved.
- **Point Plots:** Show the exact locations of accidents, useful for very localized studies (e.g., within a city or neighborhood).



Accident Hotspots



Heatmap of Accident Density



Geospatial Plot of Accident Hotspots

Comprehensive Summary Report and Expert Recommendations

This final report consolidates the findings from the Exploratory Data Analysis (EDA) and visualizations, offering a thorough overview of traffic accident patterns, contributing factors, and actionable recommendations for enhancing road safety.

1. Introduction

Analysis Objectives and Dataset Overview:

The primary objective of this analysis is to identify patterns and factors contributing to traffic accidents, emphasizing variables such as weather, time of day, and location. The goal is to understand how these variables affect accident occurrences and severity, thereby informing measures to improve road safety.

The dataset for this analysis was sourced from the Fatality Analysis Reporting System (FARS) and other datasets provided by the National Highway Traffic Safety Administration (NHTSA). The data includes detailed records of traffic accidents, encompassing the date and time of occurrence, weather conditions, geographical location, accident severity, and details about vehicles and passengers involved.

2. Methodology

Approach to Data Collection, Preprocessing, and Analysis:

- Data Collection: Data was collected from various FARS datasets, focusing on the most recent and comprehensive records available. These datasets were merged and standardized to create a unified dataset for analysis.

- Data Preprocessing: The data underwent extensive cleaning to handle missing values, convert data types (e.g., dates, weather conditions), and create new variables (e.g., categorized time of day). Geographic coordinates were validated and formatted for geospatial analysis.

- Exploratory Data Analysis (EDA): Descriptive statistics were calculated to understand the data distribution. Various visualizations were created, including:

  - Line plots to illustrate trends over time.

  - Bar charts and heatmaps to show time-based distributions.

- Scatter plots and correlation matrices to identify relationships between variables.

- Geospatial plots to pinpoint accident hotspots.

3. Results

Key Findings from the Exploratory Data Analysis and Visualizations:

1. Trends in Accident Occurrences:

- Time Trends: Distinct temporal patterns were observed, with accident occurrences peaking during specific hours (morning and evening rush hours), certain days (weekends), and months (winter months, likely due to adverse weather conditions).

- Seasonal Variations: There were significant seasonal trends, with increased accident rates during the winter months, attributed to conditions like snow, ice, and reduced visibility.

2. Correlation Between Weather Conditions and Accidents:

- Weather Impact: Adverse weather conditions, such as rain, fog, and snow, were strongly correlated with higher accident frequencies. Additionally, accidents occurring in icy and foggy conditions tended to have greater severity.

- Correlation Analysis: A moderate positive correlation was found between severe weather conditions and accident severity, suggesting that poor weather not only increases the likelihood of accidents but also their severity.

3. Identification of Accident Hotspots:

- Geospatial Analysis: Several geographic hotspots were identified, particularly in urban areas with high traffic density and intersections with complex traffic flows.

- High-Risk Locations: Specific Road segments, such as those near intersections, commercial zones, and freeways, were identified as areas with elevated risk.

4. Discussion

Interpretation of Results and Consideration of Limitations:

1. Temporal Patterns: The higher frequency of accidents during rush hours and weekends can be attributed to increased vehicle volumes, driver fatigue, and risky behaviors such as speeding and distracted driving.

2. Weather Conditions: The analysis indicates that adverse weather conditions substantially contribute to both the frequency and severity of accidents. This finding aligns with existing literature linking reduced visibility, slippery roads, and challenging driving conditions to increased accident risks.

3. Geographic Hotspots: The concentration of accidents in specific areas highlights the significant role of road infrastructure, traffic control measures, and urban planning in influencing accident occurrences. These findings underscore the need for targeted safety interventions in identified high-risk zones.

Limitations:

- Data Completeness: Some datasets contained missing or incomplete records, which may affect the reliability of certain analyses.

- Generalizability: The analysis is limited to the regions covered by the dataset and may not apply to other regions with different traffic patterns or regulations.

- Potential Confounders: Other factors, such as driver behavior, vehicle condition, and emergency response times, were not included in the dataset but could also significantly impact accident outcomes.

5. Recommendations

Targeted Recommendations for Improving Road Safety:

1. Temporal Interventions:

- Strengthen speed enforcement and traffic monitoring during peak hours (rush hours and weekends).

- Launch public awareness campaigns focused on reducing risky behaviors, such as distracted and impaired driving, during peak periods.

2. Weather-Based Safety Measures:

- Install dynamic digital road signage that updates in real-time to warn drivers of hazardous weather conditions (e.g., fog, ice, heavy rain).

- Enhance road maintenance and infrastructure by incorporating skid-resistant surfaces and improved drainage systems, particularly in regions prone to adverse weather.

- Implement temporary road closures or reduced speed limits during severe weather events.

3. Geographic Focus:

- Prioritize safety improvements in identified accident hotspots, including better road lighting, enhanced signage, improved traffic signal timing, and additional pedestrian crosswalks.

- Introduce traffic calming measures, such as speed bumps and roundabouts, in high-risk areas to reduce vehicle speeds.

- Optimize emergency response times in identified hotspots through improved traffic management systems and better coordination with emergency services.

4.Data-Driven Policy Development:

- Promote continuous data collection and analysis to evaluate the effectiveness of implemented measures and adjust strategies as needed.

- Utilize machine learning models to predict future accident hotspots and trends, enabling proactive safety interventions.

6. Conclusion

This analysis offers a comprehensive examination of traffic accident patterns, identifying key factors influencing accident occurrences and severity. By focusing on temporal trends, weather impacts, and geographic hotspots, the study provides valuable insights for enhancing road safety.

The recommendations presented are based on data-driven insights, offering practical strategies for reducing accidents and improving safety outcomes. These targeted interventions can help road safety authorities mitigate risk factors and enhance overall road user safety.

Final Remarks:

Ongoing monitoring and data collection are vital to refining these strategies over time. Leveraging advanced analytics and technology, such as real-time monitoring

and predictive modeling, can further optimize road safety efforts and decrease traffic-related fatalities and injuries.