# The Causality Between Road Accidents and Weather Conditions

A Relational Database Approach for Enhanced Insights

**Data Managment**

Khajjou Yasmine
Sestito Alfredo
Tadesse Amen Sime

**CdLM Data Science**
**DISCO**

**February 2024**

# Contents

# List of Figures

# Introduction

Road accidents pose significant risks to public safety and have multifaceted causes, including environmental factors such as weather conditions. Weather conditions such as rain, snow, fog, and ice can profoundly impact road visibility, traction, and vehicle control, thereby increasing the likelihood of accidents. 799 people died due to sudden weather phenomena in central Europe from 2010 to 2020 [1]. Thus, understanding the correlation between weather conditions and accident occurrence is crucial for identifying weather-related risk factors and evaluating the effectiveness of current safety measures under different weather conditions. In response to this concern, our project seeks to enhance the analysis of car accident causality by integrating historical API weather data into our analysis framework. And examining the interplay between weather conditions and accident occurrence, we aim to gain deeper insights into the underlying factors contributing to road accidents and improve safety measures accordingly. To achieve our objectives, we used a comprehensive accident dataset sourced from Kaggle and using web API. This dataset includes essential attributes such as accident date and location, augmented with meteorological features that are useful for analysing the causality between weather conditions and road accidents.

# Data Acquisition

In the pursuit of assembling a comprehensive dataset for our analysis, we employed two distinct methods:

## 2.1 Accidents Dataset

Initially, we sourced Kaggle, a renowned platform for datasets, to procure datasets with various features of accidents [2] . These datasets encompassed crucial attributes:

1. Accident Date: Represents the date of each accident occurrence, providing temporal context crucial for analyzing trends and patterns.
2. Light Conditions: Indicates the lighting conditions at the time of the accident, influencing visibility and potentially contributing to accident severity.
3. Longitude and Latitude: Geographical coordinates pinpointing the location of each accident, essential for spatial analysis and hotspot identification.
4. Accident Fatality: Denotes whether the accident resulted in fatalities, a critical metric for assessing accident severity.
5. Number of Deceased Individuals: Provides the count of individuals who lost their lives in each accident, aiding in casualty assessment.
6. Number of Injured Individuals: Indicates the number of individuals injured in each accident, offering insights into the magnitude of physical harm.
7. District: Specifies the district or administrative region where each accident occurred, enabling regional analysis and policy targeting.

To streamline our analysis, we then merged these datasets into a unified CSV file, thereby consolidating all accident features under one repository.

However, to optimize data retrieval efficiency and focus our analysis on weather-related causalities, we opted to narrow down our dataset. Specifically, we confined our dataset geographically to Kent and East London, as these regions exhibit a higher incidence of accidents within the United Kingdom [2].

## 2.2 Weather API

Subsequently, leveraging the temporal information provided in the dataset, we used a weather API to retrieve weather data corresponding to the selected time frame and geographic area [3]. By harnessing the date, latitude, and longitude data from our accident dataset, we integrated weather data, thus enriching our dataset with crucial meteorological insights such as: Certainly! Here are the explanations of each feature retrieved from the weather API:

- maxtempC: The maximum temperature recorded during the day, in degrees Celsius.
- mintempC: The minimum temperature recorded during the day, in degrees Celsius.
- totalSnow: The total amount of snowfall measured in centimeters during the specified time period.
- sunHour: The total hours of sunlight received during the day.
- uvIndex: A measure of the strength of ultraviolet radiation from the sun, indicating the potential risk of harm to exposed skin.
- moon illumination: The percentage of the moon's surface illuminated by sunlight.
- moonrise: The time at which the moon rises above the horizon.
- moonset: The time at which the moon sets below the horizon.
- sunrise: The time at which the sun rises above the horizon.

- sunset: The time at which the sun sets below the horizon.
- DewPointC: The temperature at which air becomes saturated with moisture, measured in degrees Celsius.
- FeelsLikeC: The perceived temperature, which factors in humidity and wind conditions, expressed in degrees Celsius.
- HeatIndexC: A measure of how hot it feels when relative humidity is factored in with the actual air temperature, in degrees Celsius.
- WindChillC: The temperature it ”feels like” when the effects of wind speed are factored in with the actual air temperature, expressed in degrees Celsius.
- WindGustKmph: The maximum wind speed observed during the specified time period, measured in kilometers per hour.
- cloudcover: The percentage of the sky covered by clouds.
- humidity: The amount of moisture present in the air, expressed as a percentage.
- precipMM: The amount of precipitation (rain, snow, etc.) measured in millimeters.
- pressure: The atmospheric pressure at the location, measured in millibars.
- tempC: The current temperature at the location, measured in degrees Celsius.
- visibility: The distance at which objects can be clearly seen, measured in kilometers.
- winddirDegree: The direction from which the wind is blowing, measured in degrees.
- windspeedKmph: The speed of the wind, measured in kilometers per hour.

In the ensuing sections, we shall delve into the specifics of our data preprocessing techniques, including data cleaning, normalization, and feature engineering, to ensure the robustness and reliability of our dataset. Through meticulous curation and refinement, we aim to equip ourselves with a comprehensive dataset poised to unravel the intricate relationship between weather properties and accident features.

# Data Integration and Enrichment

At the culmination of the data acquisition process, our dataset configuration stands as follows: We possess the primary dataset encompassing information about the accidents that transpired in the Kent region during the year 2022. Additionally, we have procured 12 distinct datasets delineating the weather conditions for each month of 2022, meticulously collected using the geographical coordinates of the accident sites corresponding to the respective months.

## 3.1 Dataset Integration and Refinement

The initial step entailed amalgamating these 12 disparate weather datasets into a unified entity, encapsulating the comprehensive weather conditions for the entire year at the precise locations impacted by accidents. While the API furnished ancillary information, deemed irrelevant for our analytical objectives, we undertook dataset refinement by excising superfluous attributes, namely: ’sunHour’, ’uvIndex’, ’moonillumination’, ’moonrise’, ’moonset’, ’sunrise’, ’sunset’, ’pressure’, ’tempC’.

### 3.1.1 Addressing Type Heterogeneity Conflict

A ”type heterogeneity” conundrum arose with the ’location’ attribute provided by the API. The output string data, approximating latitude and longitude post a designated decimal figure, posed an inevitable mismatch quandary during joint operations with the Accidents dataset. Hence, a decision was made to reconstruct the ’location’ attribute from the ground up. In the Weather Condition Dataset, coordinates were extracted from the string to fashion a tuple with numerical values. Correspondingly, in the Accident Dataset, an identical tuple was formulated utilizing the Latitude and Longitude values already present.

### 3.1.2 Dataset Attribute Optimization

The Accidents dataset featured an attribute titled Weather Conditions, including 5.5% missing values and demonstrating insignificance for our analytical pursposes. Given that API data integration adeptly addressed this issue, the decision was made to exclude this attribute, thereby fortifying the dataset's effeciency and comprehensiveness. In conclusion, the meticulous integration and refinement of our dataset have resulted in a robust and comprehensive resource for analyzing the interplay between accident occurrences and weather conditions in the Kent region during the year 2022. The strategic optimization of dataset attributes and the design of a flexible database structure lay a solid foundation for conducting analyses and deriving meaningful insights from the data.

# Database Structure

## 4.1 Rationale for Relational Database

The selected database model for our data is relational, due to the nature of the historical data, which includes accidents and weather conditions records, partially represented in textual format. The rationale behind this choice is:
- Analytical Flexibility: we projected the need for a variety of analytical tasks, including quantitative analysis and data mining. Thanks to the model's built-in flexibility, complex queries, and analyses are made easier by integrating and manipulating data across multiple tables easily.
- Focus on Read Operations: since our project is predominantly read-intensive, we gave top priority to selecting a database model that is designed to make data retrieval and processing efficient.
- Structured Data: The relational model's structured approach aligns well with the structure and nature of our dataset's distinct entities.

## 4.2 Database Structure Design

We embarked on constructing a database housing three distinct entities: Accidents, Weather Conditions, and District Area. This strategic choice affords enhanced flexibility for end-users, enabling independent analyses on distinct tables without necessitating exhaustive searches across the entire dataset each time.

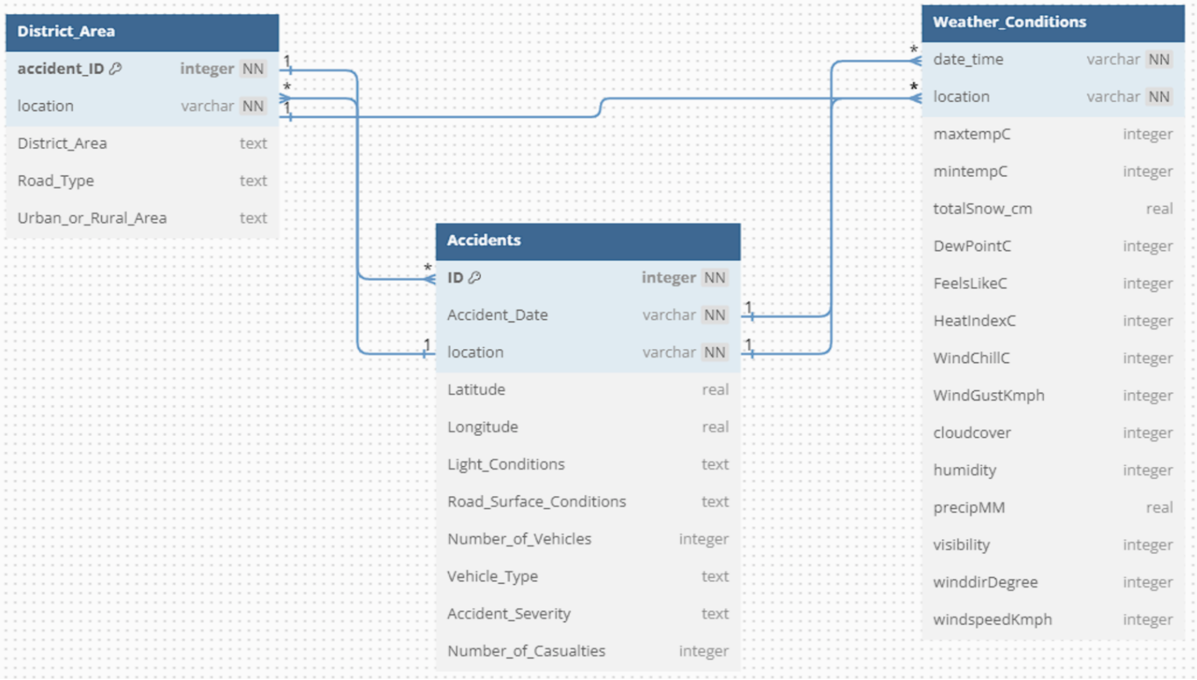
1. Accidents: each row represents a single accident; this entity also represents a connection point with the others, as it is made up of unique IDs, the date, the

position, and everything that exclusively concerns the dynamics of the accident, i.e. the number of vehicles involved, the number of casualties and the severity of the accident itself.

2. District_Area: this represents the geographical position where an accident occurred and is made up of the coordinates, but also of the district, the type of road, and the type of area, be it urban or rural. The District Area dataset was curated by segregating select attributes from the Accident dataset, including: 'ID' (previously devised for accident classification), 'location', 'District_Area', 'Road_Type', 'Urban_or_Rural_Area'.

3. Weather_Conditions: this entity exclusively represents the detailed weather conditions of each location on the date relating to the incident. It consists of information such as temperature (maximum, minimum, and perceived) but also the dew point, humidity, wind conditions, and precipitation, whether rain or snow.

This relational model is the OnetoMany paradigm, wherein the Accidents table acts as a bridging entity. By leveraging the relational model's strengths, we can facilitate deeper understanding and comprehensive insights for our dataset, ultimately contributing to understanding the dynamics within the dataset context.



| District_Area | | |
|---|---|---|
| accident_ID | integer | NN |
| location | varchar | NN |
| District_Area | text | |
| Road_Type | text | |
| Urban_or_Rural_Area | text | |

| Accidents | | |
|---|---|---|
| ID | integer | NN |
| Accident_Date | varchar | NN |
| location | varchar | NN |
| Latitude | real | |
| Longitude | real | |
| Light_Conditions | text | |
| Road_Surface_Conditions | text | |
| Number_of_Vehicles | integer | |
| Vehicle_Type | text | |
| Accident_Severity | text | |
| Number_of_Casualties | integer | |

| Weather_Conditions | | |
|---|---|---|
| date_time | varchar | NN |
| location | varchar | NN |
| maxtempC | integer | |
| mintempC | integer | |
| totalSnow_cm | real | |
| DewPointC | integer | |
| FeelsLikeC | integer | |
| HeatIndexC | integer | |
| WindChillC | integer | |
| WindGustKmph | integer | |
| cloudcover | integer | |
| humidity | integer | |
| precipMM | real | |
| visibility | integer | |
| winddirDegree | integer | |
| windspeedKmph | integer | |

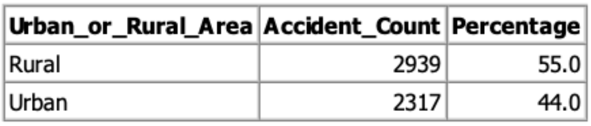
**Figure 4.1:** Database Schema

## 4.3 Possible Queries

An example of possible queries using Cypher for database analysis:

- **The Comparative Analysis of Urban and Rural Area Accidents: Percentage Distribution Relative to Total Accidents:**

```
SELECT Urban_or_Rural_Area, count(*) as Accident_Count, round((count(*)*100)/
(SELECT count(*) from District_Area), 2) as Percentage
FROM District_Area
GROUP by Urban_or_Rural_Area
;
```

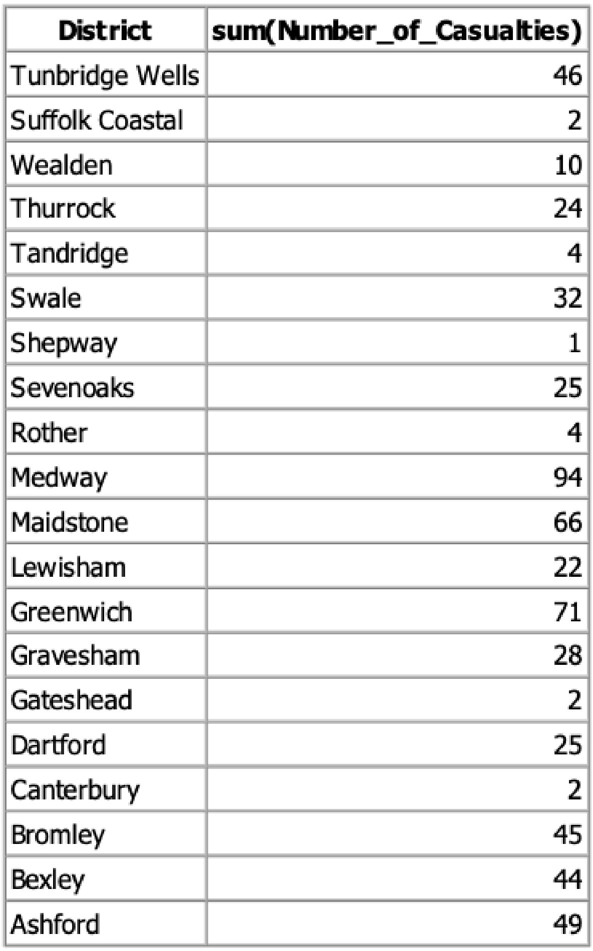


| Urban_or_Rural_Area | Accident_Count | Percentage |
|---|---|---|
| Rural | 2939 | 55.0 |
| Urban | 2317 | 44.0 |

**Figure 4.2:** Output - Table Display of The Query

- **District Areas and Causalities Count with High-Speed Wind and Low Temperature:**

```
SELECT District, sum(Number_of_Casualties)
FROM Accidents join District_Area on Accidents.location = District_Area.location
where Accidents.location in (SELECT location
FROM Weather_Conditions
WHERE FeelsLikeC <= 5 AND windspeedKmph >= 20)
GROUP by District
ORDER by Number_of_Casualties DESC
;
```



| District | sum(Number_of_Casualties) |
|---|---:|
| Tunbridge Wells | 46 |
| Suffolk Coastal | 2 |
| Wealden | 10 |
| Thurrock | 24 |
| Tandridge | 4 |
| Swale | 32 |
| Shepway | 1 |
| Sevenoaks | 25 |
| Rother | 4 |
| Medway | 94 |
| Maidstone | 66 |
| Lewisham | 22 |
| Greenwich | 71 |
| Gravesham | 28 |
| Gateshead | 2 |
| Dartford | 25 |
| Canterbury | 2 |
| Bromley | 45 |
| Bexley | 44 |
| Ashford | 49 |

**Figure 4.3:** Output - Table Display of The Query

# Data Quality

An analysis was carried out on the datasets before and after the integration which consists of a comparison between the datasets.

## 5.1 Completeness

1. In the initial state of the Kent dataset (KENT_2022_dataset) before integration, there were notable data quality issues. Specifically, there were 288 missing values for the 'Road_Type' attribute and 288 for the 'Weather_Conditions' attribute. With a completeness percentage of 99.2% before integration. As Metrics, we checked the number of missing values as well for any redundancies using a Python script.

2. The District Area Dataset exhibited a completeness rate of 94.5% both before and after integration, with 5.5% missing values. However, no specific actions were mentioned to address these missing values in the dataset.

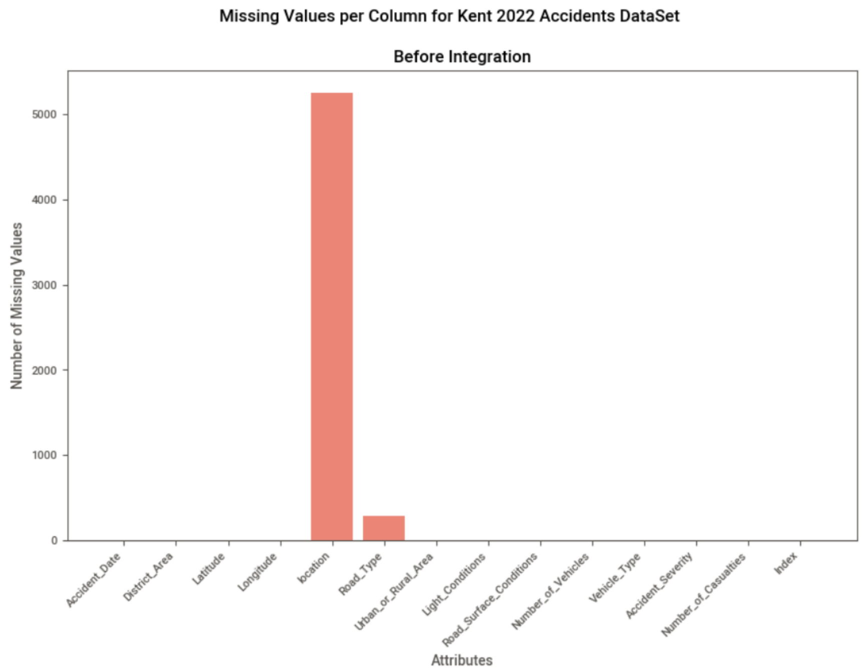


**Figure 5.1:** Missing Values per Column for Kent_2022_Accidents Dataset Before Integration.

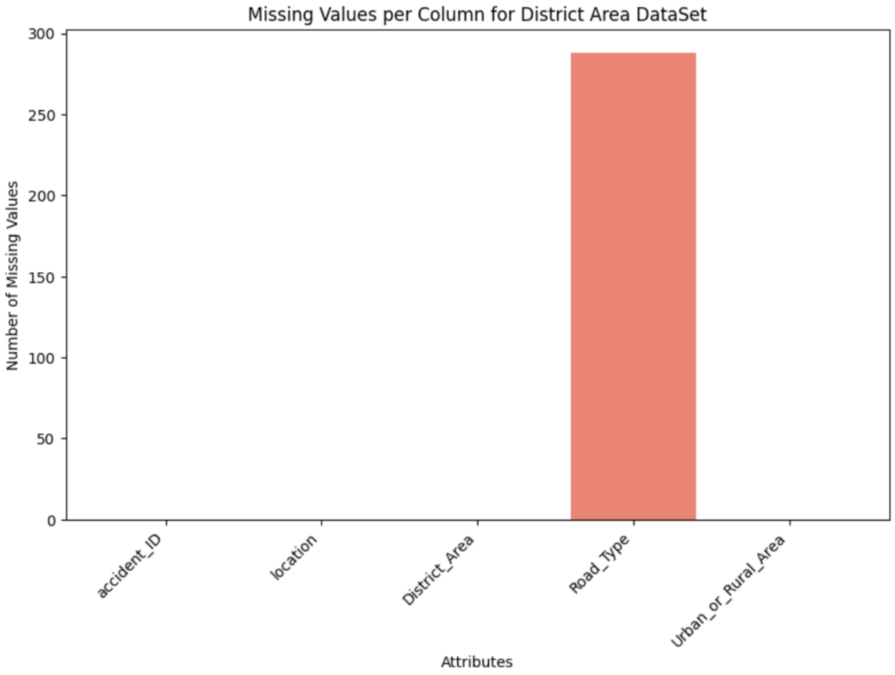


**Figure 5.2:** Missing Values per Column for District_Area Dataset.

3. On the other hand, the Weather Conditions Dataset demonstrated 100% completeness before and after integration, without any missing values reported.

## 5.2 Consistency

Additionally, all the datasets(Kent_2022_dataset, District Area, Weather_Condtions) also passed critical data checks, including ”Date Format Check” and ”Geographical Checks” (Latitude, and Longitude Range Check) to maintain the integrity of the geographic data specific to the KENT region before and after the integration.

## 5.3 Outlier Detection

We checked for outliers in the weather conditions dataset and found several outliers such as WindSpeed, Visibility, Precipitation, Humidity, Total_Snow, and Max Temperature before and after integration. As an improvement, we can use Data transformation, such as applying logarithmic transformations, which can be effective in reducing the influence of extreme values on statistical measures. Or setting a threshold beyond which values are considered outliers and limiting them to that threshold, preventing their disproportionate impact.

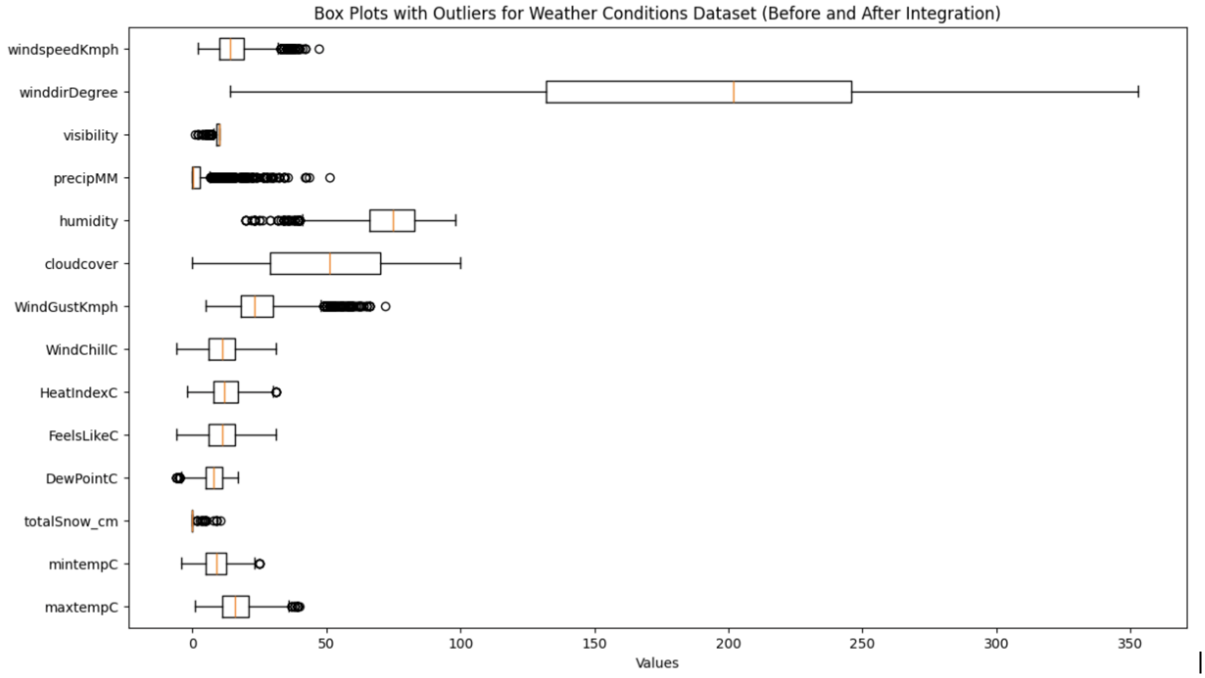


**Figure 5.3:** Box Plot with Outliers for Weather Conditions Dataset Before and After Data Integration.

# Conclusion and Future Work

In conclusion, we created a relational one-to-many database using a comprehensive accident dataset sourced from Kaggle encompassing crucial attributes like accident date, and location; enriched with meteorological insights obtained via a weather API such as max/min temperature, snowfall, sunlight hours, UV index. In the process of integrating and refining our dataset, we merged 12 weather datasets into a unified entity for the entire year 2022 in Kent. And optimized attributes, addressed type heterogeneity conflicts, and removed redundant features. All datasets passed critical data checks and outlier detection was identified, warranting data transformation for better statistical accuracy. The relational database model was chosen for its analytical flexibility, efficiency in read operations, and compatibility with structured data. Three distinct entities were constructed: Accidents, District Area, and Weather Conditions. Future work could involve temporal and spatial analysis, predictive modeling, integration of additional data sources, enhanced data visualization, real-time data integration, and collaborative analysis. By exploring these avenues, the project can advance understanding and contribute to improved road safety and accident prevention efforts.

# Bibliography

[1] Pilorz, W., et al. (2023). ”Fatalities Related to Sudden Meteorological Events Across Central Europe from 2010 to 2020.” *International Journal of Disaster Risk Reduction*, 88, 103622.

[2] Clarke, David D., et al. (2010). ”Killer crashes: fatal road traffic accidents in the UK.” *Accident Analysis  Prevention*, 42(2), 764-770.

[3] World Weather Online. (n.d.). World Weather Online API. Retrieved from https://www.worldweatheronline.com/weather-api/

[4] Kaggle. (n.d.). Road Accident Casualties. Retrieved from https://www.kaggle.com/datasets/willianoliveiragibin/road-accident-casualties?resource=download