CITS 5504 - Data Warehousing

Project 1

23914381 Aparna Sasi Nair

24066497 Cavin Cajetan D'Souza

Problem Statement:

The overall objective of this project is to build a data warehouse using real-world datasets which in our case is Fatal_crashes_December_2024 and Fatalities_December_2024 from the Australian Roads Deaths Database (ARDD) and to carry out a data mining activity which in this case is Association Rule Mining.

ETL Process: Data Cleaning, Preprocessing, and Integration

We performed a comprehensive **ETL (Extract, Transform, Load)** process on the Australian BITRE fatal crash datasets using **Python in a Jupyter Notebook environment**. We followed best practices based on the **Kimball dimensional modeling** approach, leveraging powerful Python libraries including **pandas**, **numpy**, and **datetime** for data manipulation and cleansing.

1. Extract Phase

We began by extracting data from two Excel files:

- *BITRE_Fatality*
- *BITRE_Fatal_Crash*

We skipped unnecessary header rows and loaded the sheets using pandas.read_excel() function. Later A *left join* on Crash ID was used to merge fatality-level data with its corresponding crash-level attributes. The reason we used a *left join* on Crash ID during the ETL process because our primary dataset was the *fatality-level table*, where each row represents an individual fatality. The *BITRE_Fatal_Crash* table, on the other hand, contains one row per crash incident and not per person.

2. Transform Phase

Removal of Redundancy is done by removing duplicate columns resulting from the merge and stripped suffixes (such as _crash and _fatality) to unify column naming. For cleaning we have done the following on various columns:

- **Involvement Flags**: We standardized flags (Yes, No, Unknown) and corrected inconsistent or multiple involvements in single-vehicle crashes.

```
# Column Groups
involvement_cols = ['Bus Involvement','Heavy Rigid Truck Involvement','Articulated Truck Involvement']
categorical_cols = ['National Remoteness Areas', 'SA4 Name 2021', 'National LGA Name 2021','National Road Type']
speed_col = 'Speed Limit'

# Clean Speed Limit
merged_df[speed_col] = merged_df[speed_col].replace('<40', 40)
# First convert -9 to NaN
merged_df[speed_col] = merged_df[speed_col].replace(-9, np.nan)
# Then ensure the column is of nullable integer type
merged_df[speed_col] = pd.to_numeric(merged_df[speed_col], errors='coerce').astype('Int64')

# Clean Involvement Flags
for col in involvement_cols:
    merged_df[col] = merged_df[col].astype(str).apply(
        lambda x: 'Unknown' if str(x).strip().lower() in ['', 'unknown', '-9', 'nan'] else x
    )

# Clean All Categorical Columns
for col in categorical_cols:
    merged_df[col] = merged_df[col].astype(str).apply(
        lambda x: 'Unknown' if str(x).strip().lower() in ['', 'unknown', '-9', 'nan'] else x
    )

# Special: Crash Type Rule
def fix_single_crash_vehicle_flags(row):
    if str(row['Crash Type']).strip().lower() == 'single':
        yes_count = sum([row[col] == 'Yes' for col in involvement_cols])
        if yes_count > 1:
            for col in involvement_cols:
                row[col] = 'Unknown'
    return row

merged_df = merged_df.apply(fix_single_crash_vehicle_flags, axis=1)
```

- **Speed Limit**: Replaced <40 with 40, removed -9, and converted to Int64 for nullable precision.

```
# Clean Speed Limit
merged_df[speed_col] = merged_df[speed_col].replace('<40', 40)
# First convert -9 to NaN
merged_df[speed_col] = merged_df[speed_col].replace(-9, np.nan)
# Then ensure the column is of nullable integer type
merged_df[speed_col] = pd.to_numeric(merged_df[speed_col], errors='coerce').astype('Int64')
```

- **Vehicle Involvement**: Created a derived column named *Vehicle Involvement* that categorizes crashes into combinations of involved heavy vehicles for enhanced interpretability.

```
# Define the logic for heavy vehicles involvement
# Define function for exact mapping
def get_involvement_category(row):
    bus = row['Bus Involvement']
    rigid = row['Heavy Rigid Truck Involvement']
    artic = row['Articulated Truck Involvement']

    # Check for fully unknown
    if bus == 'Unknown' and rigid == 'Unknown' and artic == 'Unknown':
        return 'Unknown'

    # Check all three are involved
    if bus == 'Yes' and rigid == 'Yes' and artic == 'Yes':
        return 'Bus, Articulated Truck and Heavy Rigid Truck Involved'

    # Pairwise checks
    if bus == 'Yes' and artic == 'Yes' and rigid != 'Yes':
        return 'Bus and Articulated Truck Involved as Known'
    if bus == 'Yes' and rigid == 'Yes' and artic != 'Yes':
        return 'Bus and Heavy Rigid Truck Involved as Known'
    if artic == 'Yes' and rigid == 'Yes' and bus != 'Yes':
        return 'Articulated Truck and Heavy Rigid Truck Involved as Known'

    # Single vehicle checks
    if bus == 'Yes' and rigid != 'Yes' and artic != 'Yes':
        return 'Only Bus Involved as Known'
    if artic == 'Yes' and bus != 'Yes' and rigid != 'Yes':
        return 'Only Articulated Truck Involved as Known'
    if rigid == 'Yes' and bus != 'Yes' and artic != 'Yes':
        return 'Only Heavy Rigid Truck Involved as Known'

    # All no
    if bus == 'No' and rigid == 'No' and artic == 'No':
        return 'No Heavy Vehicles Involved'

    # Default
    return 'Unknown'
```

- **Age and Gender**: Handled -9, 0, and missing values. Imputed missing age using the mode within 1_to_16 group.

```
# Age: -9 and 0 → NaN
merged_df['Age'] = pd.to_numeric(merged_df['Age'], errors='coerce')
merged_df['Age'] = merged_df['Age'].replace({-9: np.nan, 0: np.nan})

# Gender: -9 → 'Unknown'
merged_df['Gender'] = merged_df['Gender'].replace(-9, 'Unknown')
merged_df['Gender'] = merged_df['Gender'].apply(lambda x: 'Unknown' if str(x).strip().lower() == 'unknown' else x)
```

```
# Fix Age Group for rows where Age = 19 and Age Group is Unknown
merged_df.loc[
    (merged_df['Age'] == 19) & (merged_df['Age Group'].str.lower() == 'unknown'),
    'Age Group'
] = '17_to_25'

# Fix Age where the Age Group is known (1_to_16)
missing_before = merged_df[
    (merged_df['Age Group'] == '1_to_16') & (merged_df['Age'].isna())
].shape[0]

print(f"Missing ages before imputation: {missing_before}")

# Make sure Age is numeric
merged_df['Age'] = pd.to_numeric(merged_df['Age'], errors='coerce')

# Filter for rows where Age Group is 1_to_16 and Age is NOT null
ages_in_group = merged_df.loc[
    (merged_df['Age Group'] == '1_to_16') & (merged_df['Age'].notna()),
    'Age'
]

# Calculate mode
age_mode_1_to_16 = ages_in_group.mode().iloc[0]  # Take first mode if multiple
print(f"Mode of Age in '1_to_16' group: {age_mode_1_to_16}")

# Fill missing Age values in this group with the mode
merged_df.loc[
    (merged_df['Age Group'] == '1_to_16') & (merged_df['Age'].isna()),
    'Age'
] = age_mode_1_to_16

missing_after = merged_df[
    (merged_df['Age Group'] == '1_to_16') & (merged_df['Age'].isna())
].shape[0]
```

- **Time of Day**: Derived from crash time using logical bins (Day, Night).

```
# Only convert strings to time, keep time objects as-is
merged_df['Time'] = merged_df['Time'].apply(
    lambda x: datetime.strptime(x, '%H:%M:%S').time() if isinstance(x, str) else x
)

# Define time-of-day logic
def get_time_of_day(t):
    if pd.isna(t):
        return 'Unknown'
    elif t.hour >= 6 and t.hour < 18:
        return 'Day'
    else:
        return 'Night'

# Apply the function
merged_df['Time of day'] = merged_df['Time'].apply(get_time_of_day)
```

- **Holiday Indicator**: Created a new column to label Christmas, Easter, or Non-Holiday based on original columns.

```
# Apply the function
merged_df['Time of day'] = merged_df['Time'].apply(get_time_of_day)

# Create a Holiday Indicator column.
def assign_holiday(row):
    if row['Christmas Period'] == 'Yes':
        return 'Christmas'
    elif row['Easter Period'] == 'Yes':
        return 'Easter'
    else:
        return 'Non-Holiday'

merged_df['Holiday Indicator'] = merged_df.apply(assign_holiday, axis=1)
```

- **Categorical Columns**: Replaced -9, empty, and 'unknown' with Unknown.

Furthermore, we identified and removed rows where all 5 key geographic columns (National Remoteness Areas, SA4, LGA, Road Type, Speed Limit) were unknown or null This step preserved data quality and minimized distortions during analysis of geographic patterns to understand the relationships between location-based data.

```
columns_to_check = [
    'Speed Limit',
    'National Remoteness Areas',
    'SA4 Name 2021',
    'National LGA Name 2021',
    'National Road Type'
]

# Define helper function to check "unknown" status
def is_unknown_or_blank(val):
    return pd.isna(val) or val == '' or str(val).strip().lower() == 'unknown' or str(val).strip().lower() == 'undetermined'

# Apply the function across all specified columns
filtered_rows = merged_df[
    merged_df[columns_to_check].applymap(is_unknown_or_blank).all(axis=1)
]
```

3. Load Phase

Proceeding further, we have build the ***dimension*** and ***fact*** tables based on the cleaned dataset.

- ***dim_date****:* Captures Year, Month, Day of week, and holiday indicators.
- ***dim_location****:* Stores State, SA4, LGA name, and remoteness area. It references dim_lga for dwelling data .
- ***dim_person****:* Contains Gender, Age, and Age Group.
- ***dim_road_user****:* Represents Road user roles (e.g., Driver, Pedestrian).
- ***dim_vehicle_involvement****:* Stores flags and derived descriptions.
- ***dim_crash_type*** *and* ***dim_national_road_type***: Classify crash and road types.

Each dimension table was saved to a CSV file, using surrogate keys (e.g., RU1, CT2, NR3) for efficient joins and referential integrity.

The ***fact_crash*** table uses one row per person per crash, supporting both person-level and aggregated crash-level analyses. It includes foreign keys to all dimensions and fields like:

crash_id
date_id
location_id
person_id
road_user_id
speed_limit
number_fatalities
time, time_of_day

```
fact_crash_df.head()
```

crash_id	date_id	location_id	road_user_id	person_id	crash_type_id	vehicle_involvement_id	national_road_type_id	time	time_of_day	speed_limit	number_fatalities
20241115	D1	L1	RU1	P1	CT1	VI1	NR1	04:00:00	Night	100.0	1
20241125	D2	L2	RU1	P2	CT1	VI1	NR2	06:15:00	Day	80.0	1
20246013	D1	L3	RU1	P3	CT2	VI1	NR2	09:43:00	Day	50.0	1
20241002	D2	L4	RU1	P4	CT2	VI1	NR3	10:35:00	Day	100.0	1
20243185	D2	L5	RU2	P5	CT2	VI1	NR3	13:00:00	Day	100.0	1

While we mainly used a **star schema** for our design, we have used a **slightly snowflake-like structure** for the location data. Instead of putting everything in one dimension, we split it:
The **dim_lga** table holds information like **dwelling count and LGA names**
The **dim_location** table uses this and connects to the **fact table**
This helps **avoid repeating data** and makes the design **cleaner and easier to manage**, without affecting performance much.

Dimensional Modelling using Kimball's Four Steps

1. **Process Being Modelled**
   The core business process modelled in this project is the *occurrence* and *analysis* of fatal road crashes across Australia. It also involved analysis of crash events across various dimensions like time, location, vehicle involvement, road user and few more.

2. **Determination of Grain of the Fact Table**
   The grain chosen of the fact table is **one row per person involved in a fatal crash as** it allows us to analyse both crash-level statistics (using DISTINCT crash_id) and person-level analysis (e.g., most vulnerable groups or demographics of victims).
   Each row in the fact table corresponds to:
   A **unique combination of crash event + person involved**, which includes:
   - The time and location of the crash
   - Vehicle involvement and crash type
   - The person's demographics (age, gender)
   - The number of fatalities (same for all people in the same crash)

3. **Choosing the Dimensions**

   In designing the dimensional model for our road safety data warehouse, we carefully selected dimensions that would support the key business questions and analytical goals of our project. We included the ***dim_date*** table to enable time-based analysis at multiple levels—year, month, day of the week, and holiday indicators. This allowed us to detect trends in crash occurrences during both regular and holiday periods such as Christmas and Easter.
   To enable spatial analysis, we developed both ***dim_location*** and ***dim_lga*** dimensions. These allowed us to explore geographic patterns of crashes across states, SA4 regions, and Local Government Areas (LGAs). We enriched the ***dim_lga*** table with dwelling count data to normalize crash counts and calculate crash density per 1,000 dwellings which is an important measure for comparing regions of varying population sizes.

To analyse the individuals involved in crashes, we created the ***dim_person*** table, which includes gender, age, and age group. This helps us understand which demographics are most affected by fatal crashes. The ***dim_road_user*** dimension captures the role played by individuals in the crash (e.g., driver, passenger, pedestrian), enabling more detailed risk analysis. Additionally, we added a ***dim_vehicle_involvement*** dimension to capture whether heavy vehicles, buses, or articulated trucks were involved, since such vehicles are often associated with more severe outcomes.

To distinguish between different types of crashes, we included the ***dim_crash_type*** dimension (e.g., single vs. multiple vehicle crashes), which is crucial for understanding crash scenarios and severity. Lastly, the ***dim_national_road_type*** dimension was added to identify the classification of the road where the crash occurred—such as arterial roads, local roads, or highways.

We chose these dimensions to ensure that our model allows for flexible slicing and dicing of the data. This design supports drill-down and roll-up analysis across different hierarchies of time, location, and road user demographics. Our design assumes that each row in the fact table represents a unique person involved in a crash. By linking this central fact to a set of well-designed dimensions, we are able to answer complex business queries that are relevant for policymakers and safety officials aiming to improve road safety outcomes across Australia.

We created the following dimension tables:

Dimension Table	Description
dim_date	Contains year, month, day of week, holiday indicator, etc.
dim_location	Represents crash location using LGA, SA4, remoteness area, state
dim_lga	Enriched location data with dwelling count to normalize crash rates
dim_person	Includes age, gender, and age group of individuals involved
dim_road_user	Role in the crash (e.g., driver, pedestrian, passenger)
dim_vehicle_involvement	Information on whether heavy vehicles or buses were involved
dim_crash_type	Type of crash (e.g., single, multiple)
dim_national_road_type	Classification of roads (e.g., arterial, local, highway)

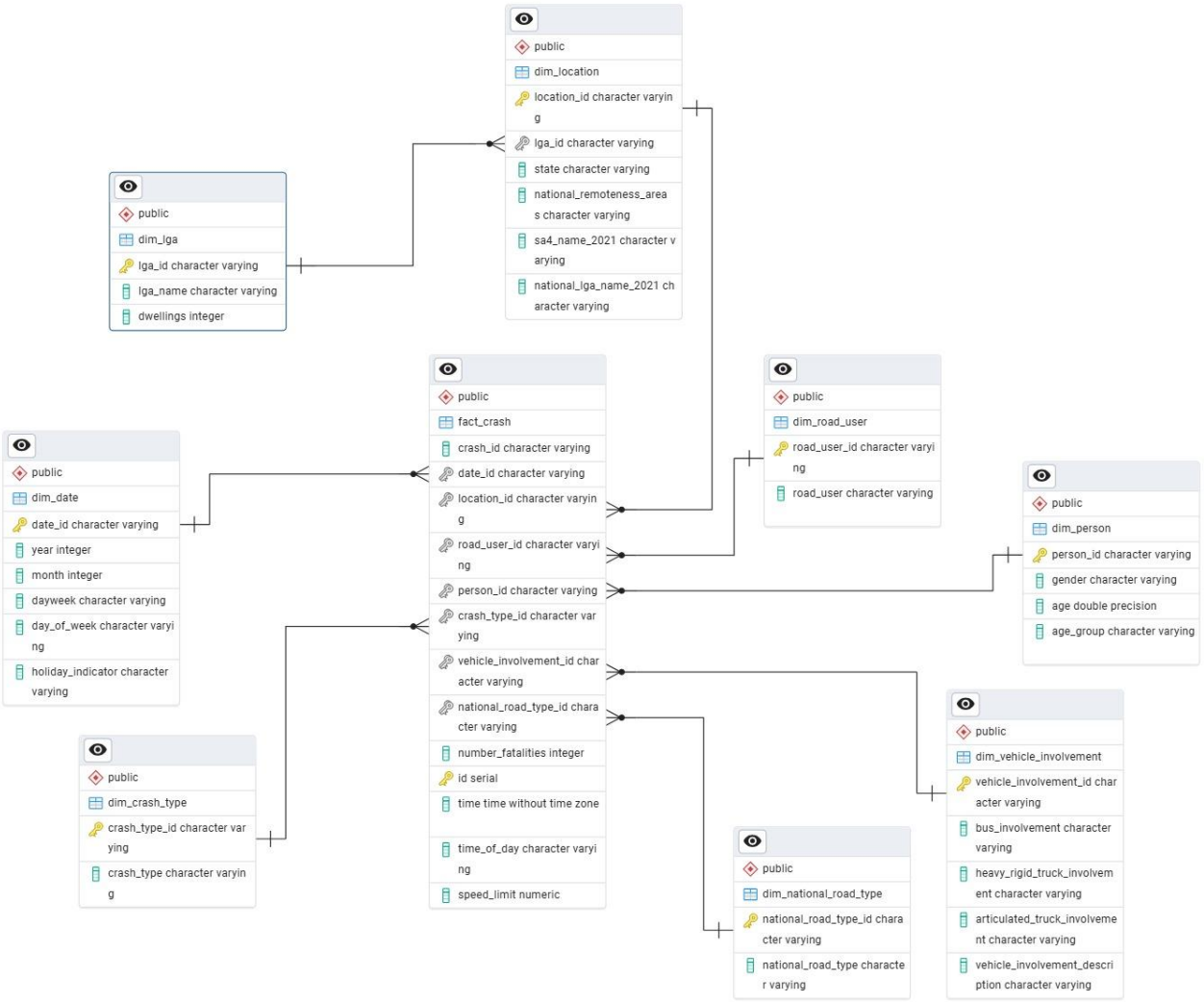
4. Identifyngi the Numeric Measures for the Fact Table:

Fact Column	Description
number_fatalities	Total number of fatalities in the crash (duplicated per person but handled in queries)
speed_limit	Speed limit at the crash site
time	Exact time of the crash
time_of_day	Day or night categorization

Our modelling Approach enables us to:

- Analyse crash frequency and severity across **time, space, and demographics**
- Identify **high-risk zones and groups**
- Evaluate **effectiveness of road policies and vehicle regulations**

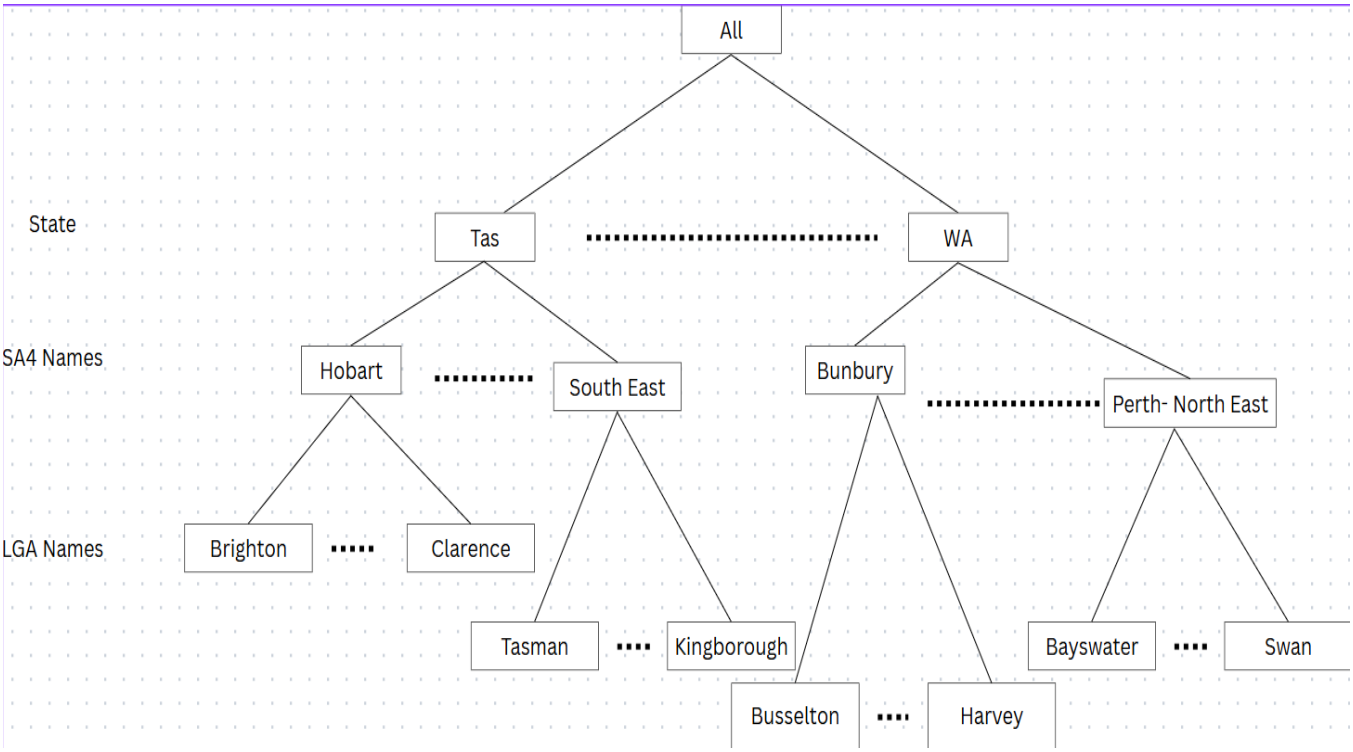
Schema Design:



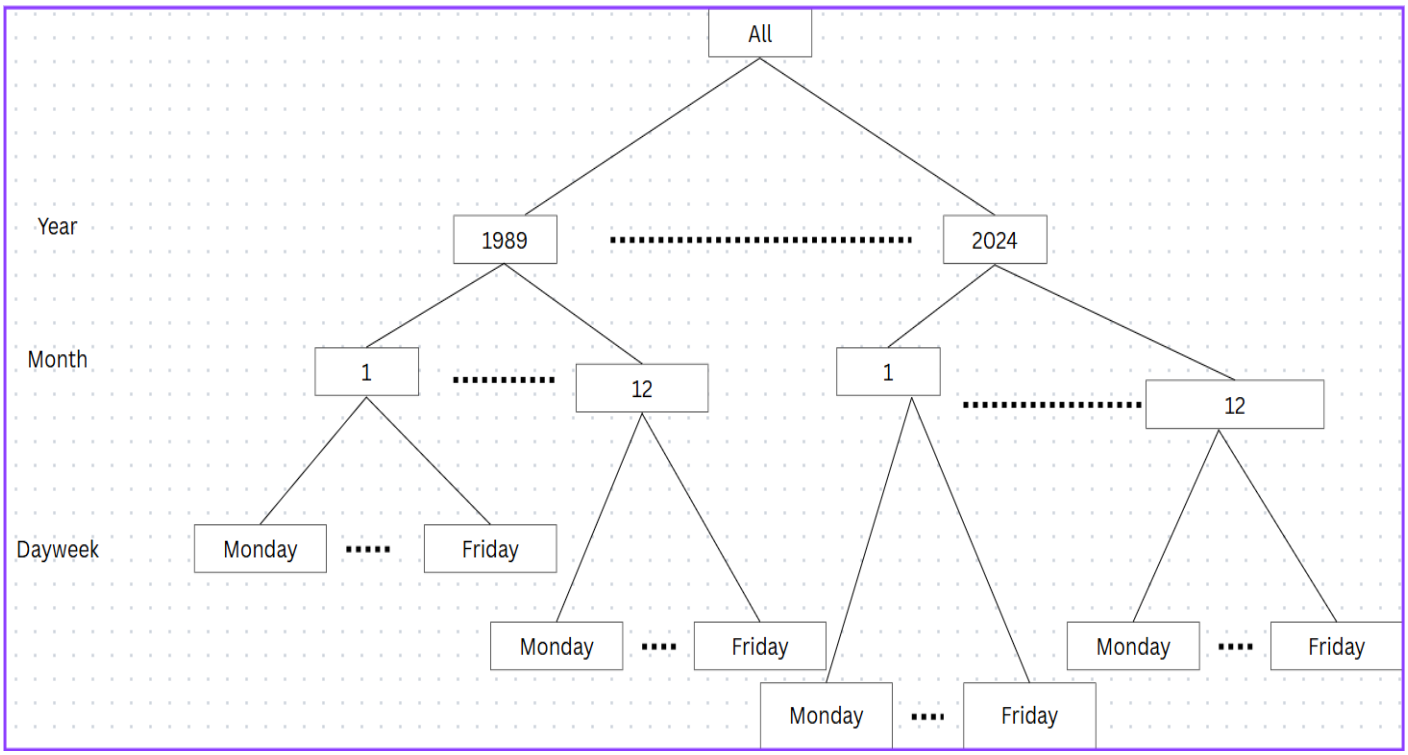
The above is our schema design after the creation of dimension tables and fact table using PostGRE SQL.

Concept Hierarchies:

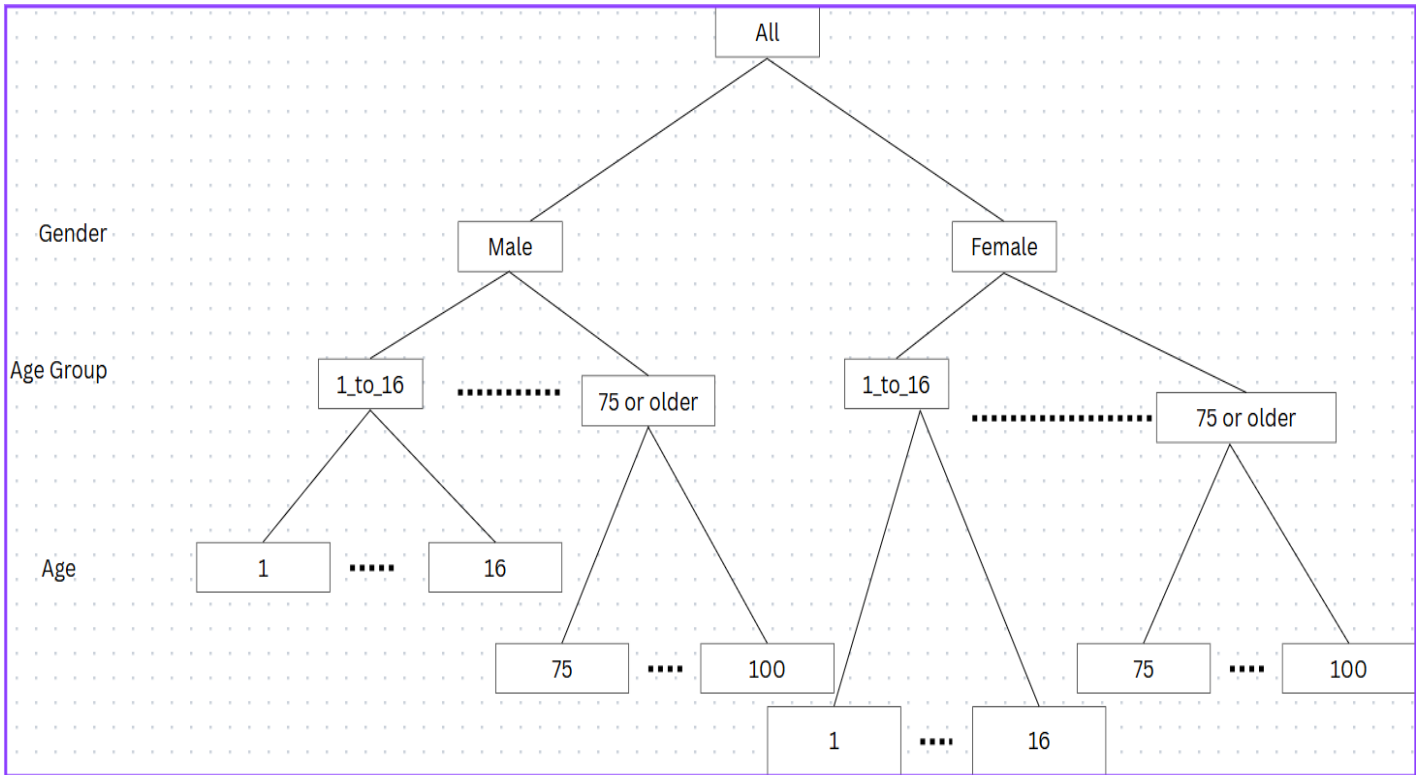
1. **Dim_location**



2. **Dim_date**

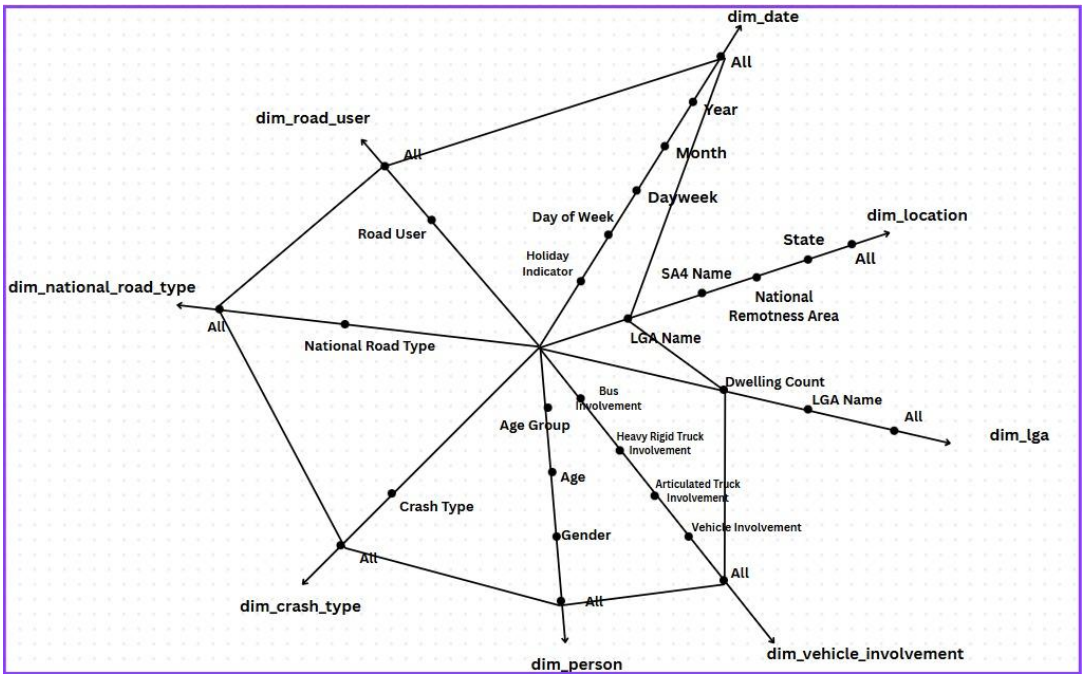


3. Dim_person

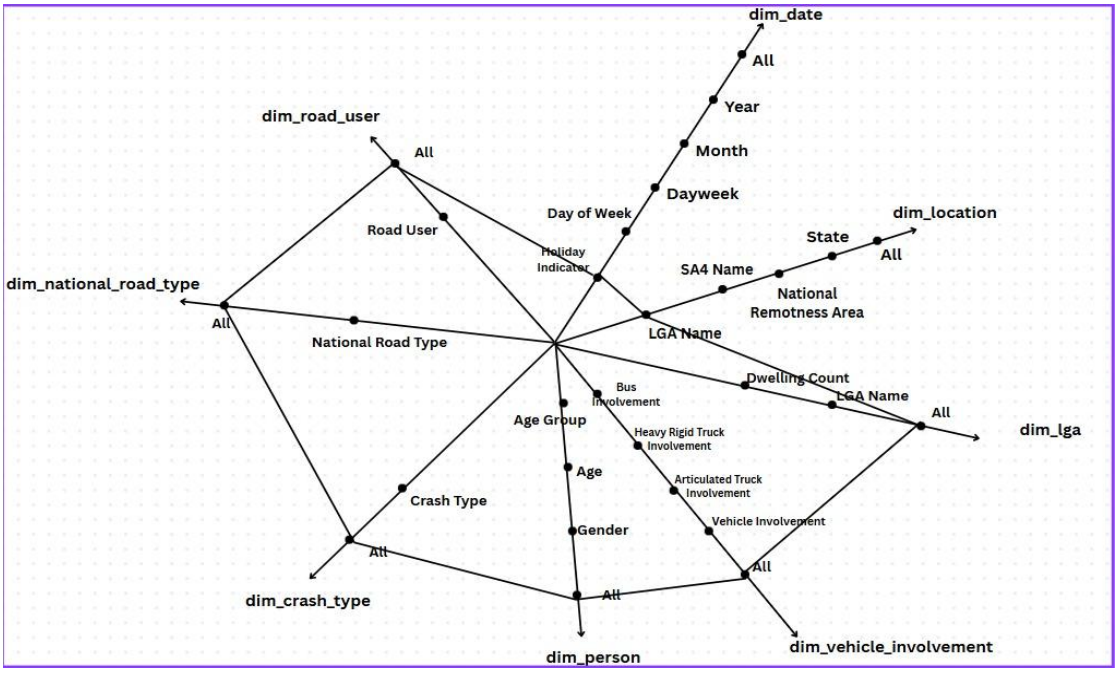

All
Gender
Male
Female
Age Group
1_to_16
75 or older
1_to_16
75 or older
Age
1
16
75
100
1
16
75
100

Business Questions:

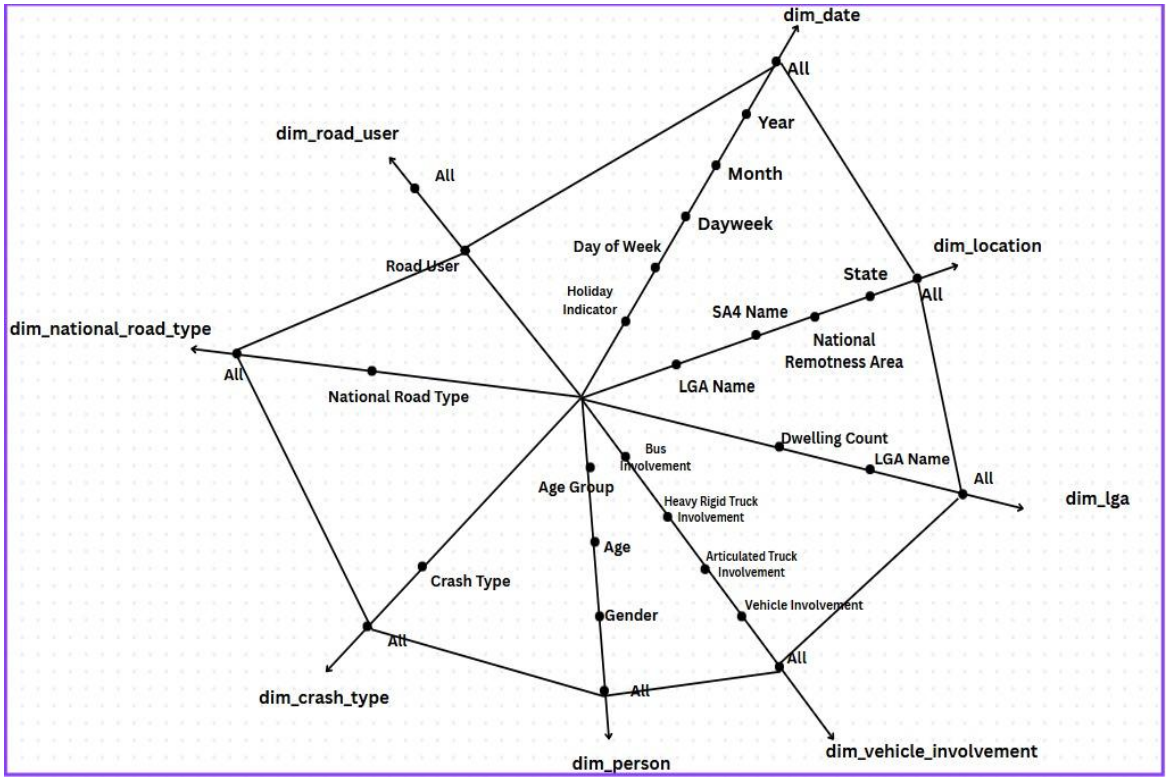
4. **Which Local Government Areas (LGAs) experience the highest crash density when normalized by dwelling count?**



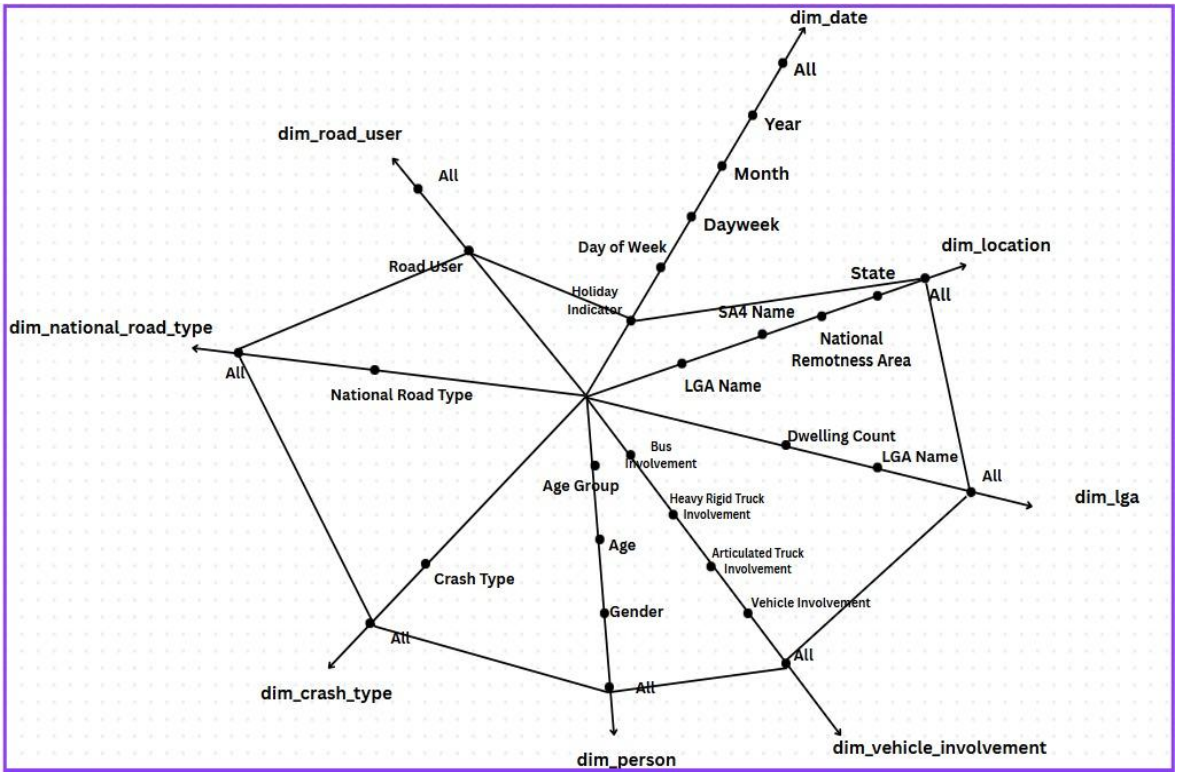
5. **Which LGAs emerge as fatal crash hotspots during holiday periods such as Christmas and Easter?**



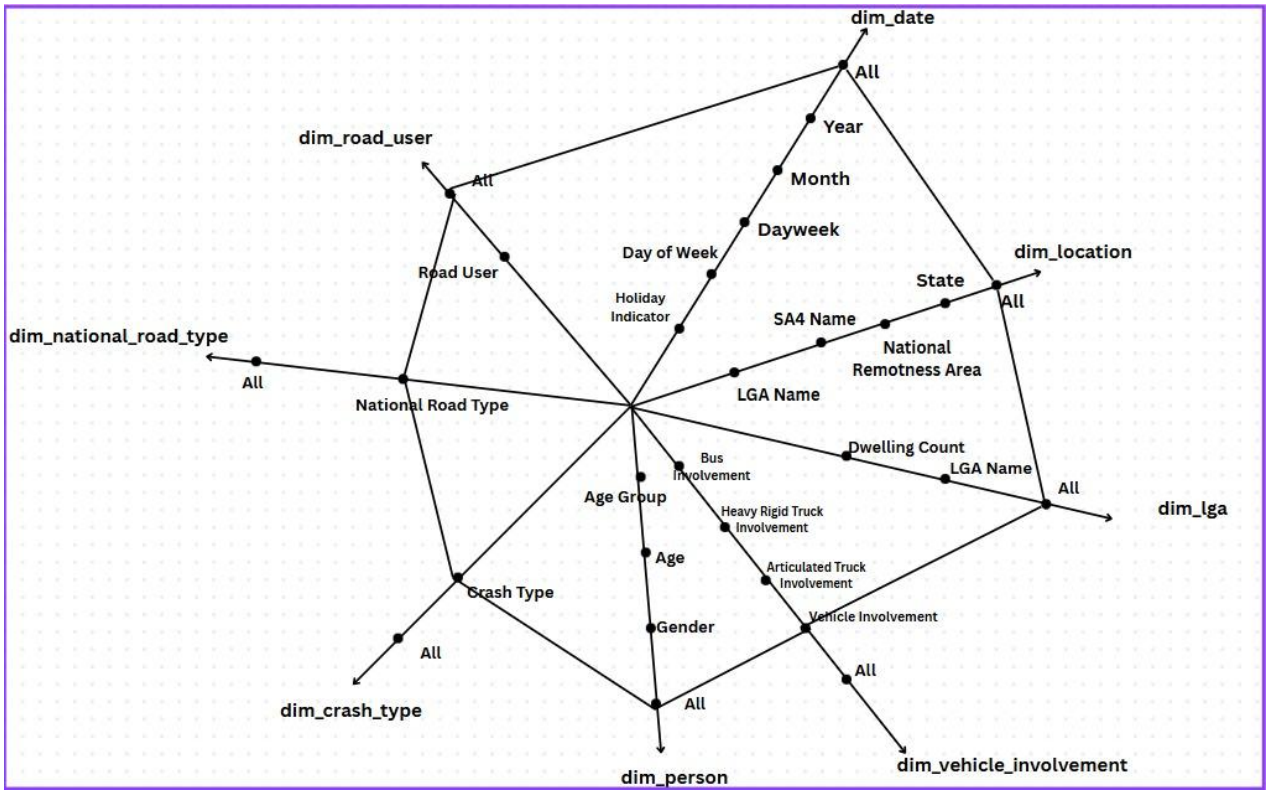
6. **What types of road users are most frequently involved in fatal crashes?**



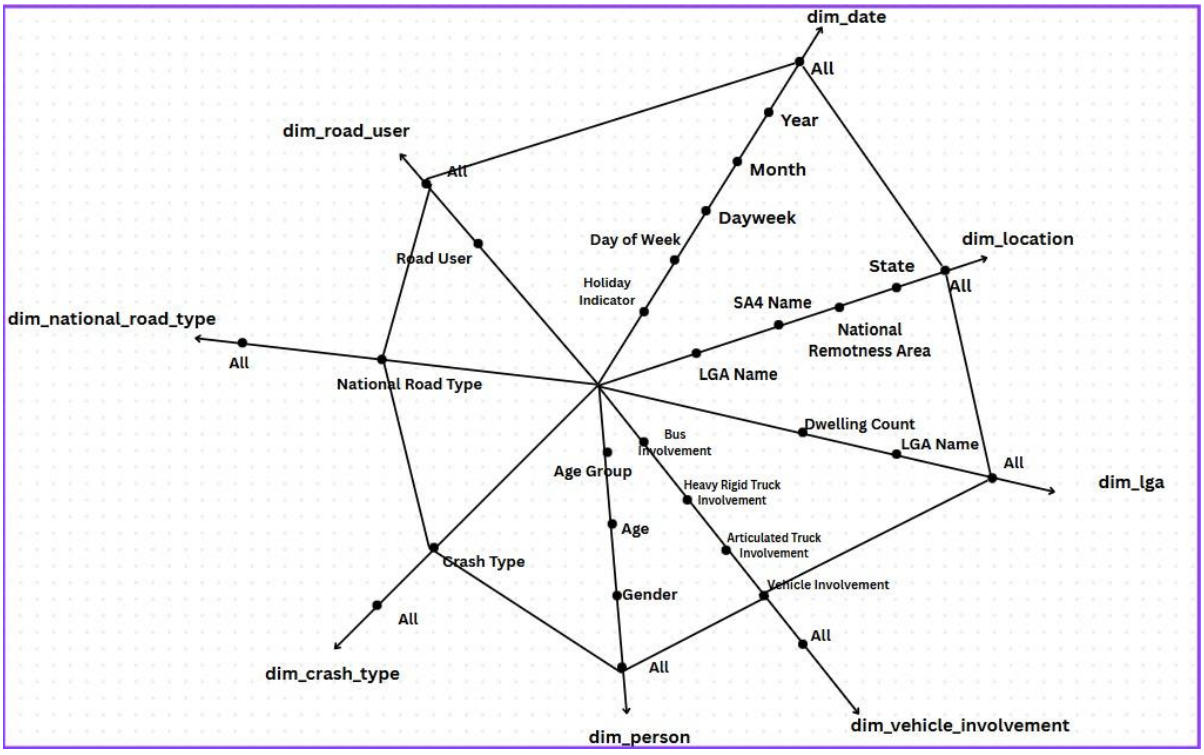
7. **Is there any pattern between road user types and holiday periods in the occurrence of fatal crashes?**



8. **Which vehicle involvement types are associated with the highest number of fatal crashes and fatalities?**

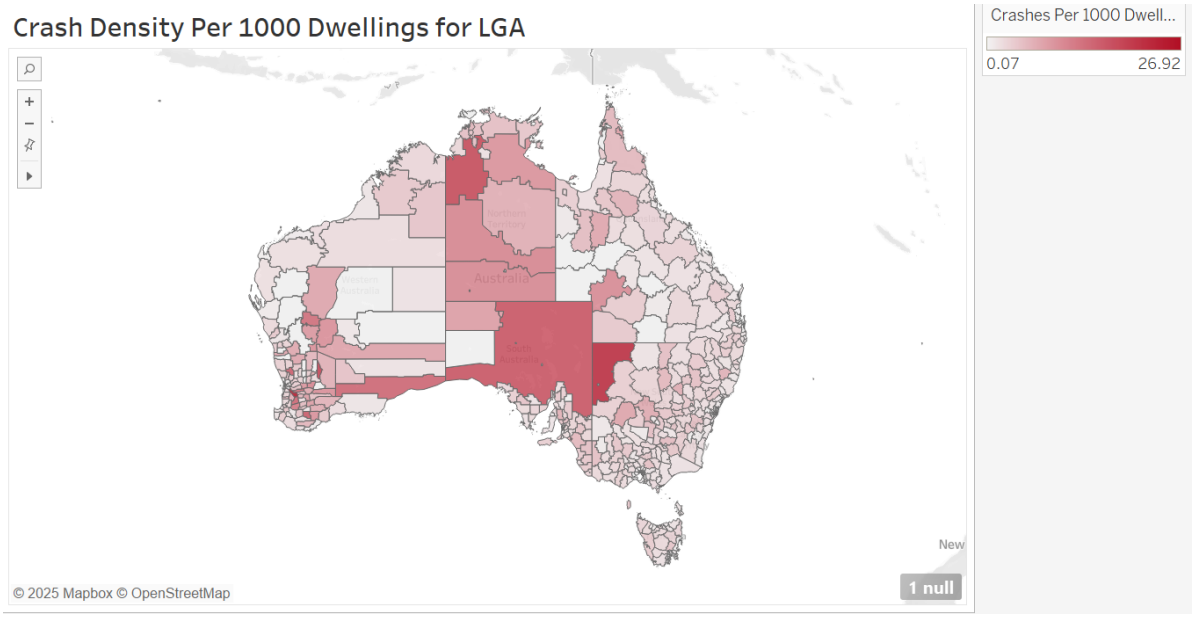


9. **Which combinations of crash types and road types contribute most to total fatalities?**



Visualisations and Insights for Business Queries:

1. Which Local Government Areas (LGAs) experience the highest crash density when normalized by dwelling count



This map visualizes ***crash density normalized by the number of dwellings*** across Australia's ***Local Government Areas (LGAs)***, using a choropleth map created in **Tableau.** Each LGA is shaded based on the **number of fatal crashes per 1,000 dwellings**, calculated using the

Formula: ***Crash Density = (Total Crashes / Dwellings) * 100***

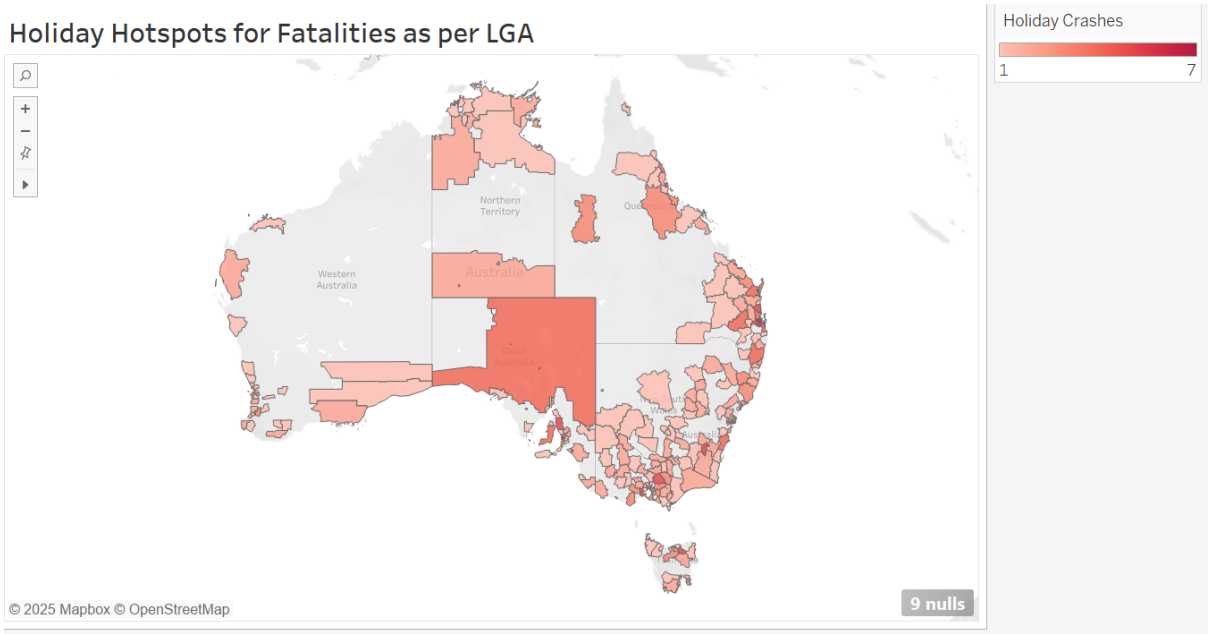
- ***Darker shades*** (deep red) indicate ***higher crash density*** relative to how many dwellings exist in the area.
- ***Lighter shades*** indicate ***lower crash density***.

This normalization is crucial because it adjusts for population/dwelling size, providing a **fair comparison** between densely and sparsely populated LGAs.

Insights:

- ***Wandering***, ***Unincorporated NSW***, and ***Westonia*** are among the LGAs with the ***highest crash density***, exceeding ***20 crashes per 1000 dwellings***.
- Several ***remote and low-population LGAs*** show very ***high crash density***, despite fewer total crashes — because their ***denominator (dwellings)*** is quite low.
- More populated metro LGAs like ***Brisbane***, ***Melbourne***, and ***Sydney*** tend to show ***lower normalized rates*** due to high dwelling counts, even if their total crashes are high.

2. Which LGAs emerge as fatal crash hotspots during holiday periods such as Christmas and Easter?



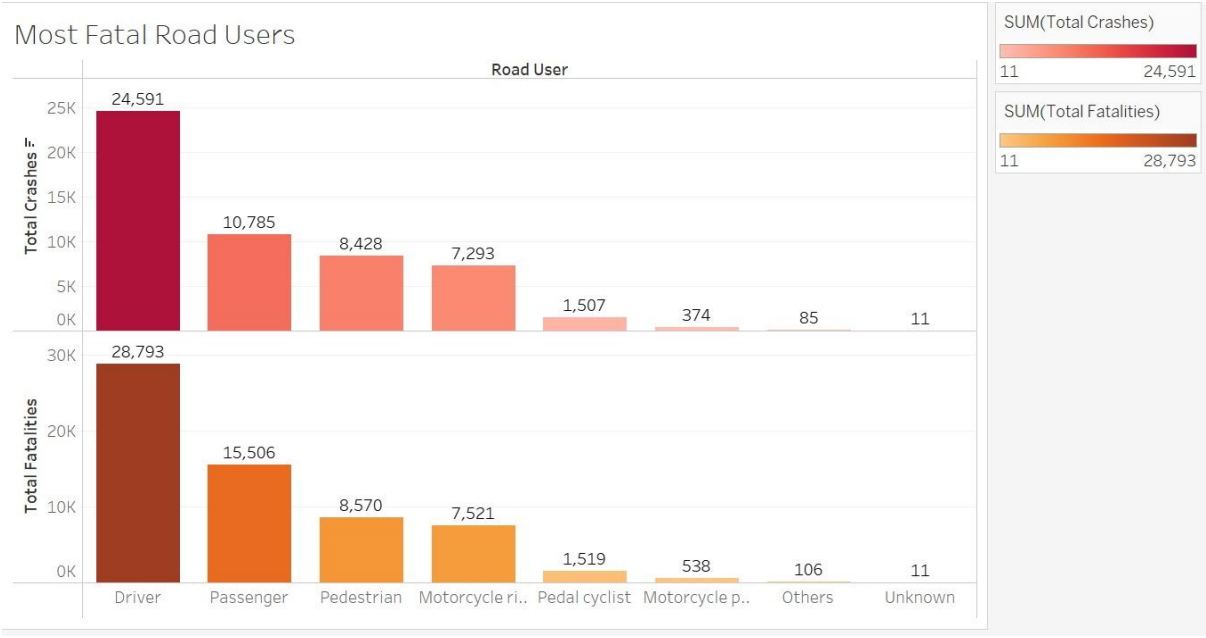
This Tableau map visualizes the number of **fatal crashes that occurred during major holidays** (Christmas and Easter) across **Australian Local Government Areas (LGAs)**.

- LGAs are color-coded from **light pink to dark red** based on the **number of fatal crashes** during the holiday periods.
- **Darker shades** represent LGAs with **more crashes** during holidays.
- LGAs with no data or 0 crashes appear **grey or blank**.
- The **right-hand legend** shows the color scale (from 1 to 7 crashes).

Insights:

- The LGA with the **most holiday crashes** is marked **"Unknown"** (1361 crashes). This suggests there are records without valid location mapping — possibly due to data quality issues.
- Among well-mapped LGAs:
    - **Brisbane (7 crashes)** leads the list, followed by:
        - **Logan**, **Canterbury-Bankstown** (6 each)
        - **Murrindindi**, **Sunshine Coast**, **Wakefield**, **Launceston**, **Ipswich**, and others with 5 crashes each.
- Many LGAs across **Queensland**, **New South Wales**, and **Victoria** show moderate counts (3–4 crashes).
- Large regions in **Western Australia** and **Northern Territory** show **sparse or missing data** — either due to:
    - Fewer crashes
    - Incomplete location mapping
    - Low population density

3. What types of road users are most frequently involved in fatal crashes?



This dual-bar chart created in Tableau presents a comparative analysis of fatal crashes in Australia based on ***road user types**. It highlights both the ****number of distinct crashes*** (top chart) and the ***total fatalities*** (bottom chart) by each category of road user.

Insights:

1. Drivers

- Crashes: 24,591
- Fatalities: 28,793
- Interpretation: Drivers are the most involved and the most fatally affected group, accounting for the highest fatality burden. This is expected given their control of vehicles, and it reinforces the need for driver safety education, especially around high-risk behaviors like speed and fatigue.

2. Passengers

- Crashes: 10,785
- Fatalities: 15,506
- Passengers face significant risk, particularly in high-speed or rural crashes. It also implies that crash severity for passengers is high.

3. Pedestrians

- Crashes: 8,428
- Fatalities: 8,570
- With nearly a 1:1 ratio of crashes to fatalities, pedestrians are extremely vulnerable. Crashes involving them are highly lethal, emphasizing the importance of urban planning, visibility, and pedestrian crossings.

4. Motorcycle Riders

- Crashes: 7,293
- Fatalities: 7,521
- Similar to pedestrians, motorcycle riders face high lethality. This underlines the need for protective gear enforcement, road hazard reduction, and training for motorcyclists.

5. Pedal Cyclists

- Crashes: 1,507
- Fatalities: 1,519
- Like motorcyclists, they are vulnerable due to limited protection. There's a need for safe cycling infrastructure and awareness campaigns for both riders and motorists.

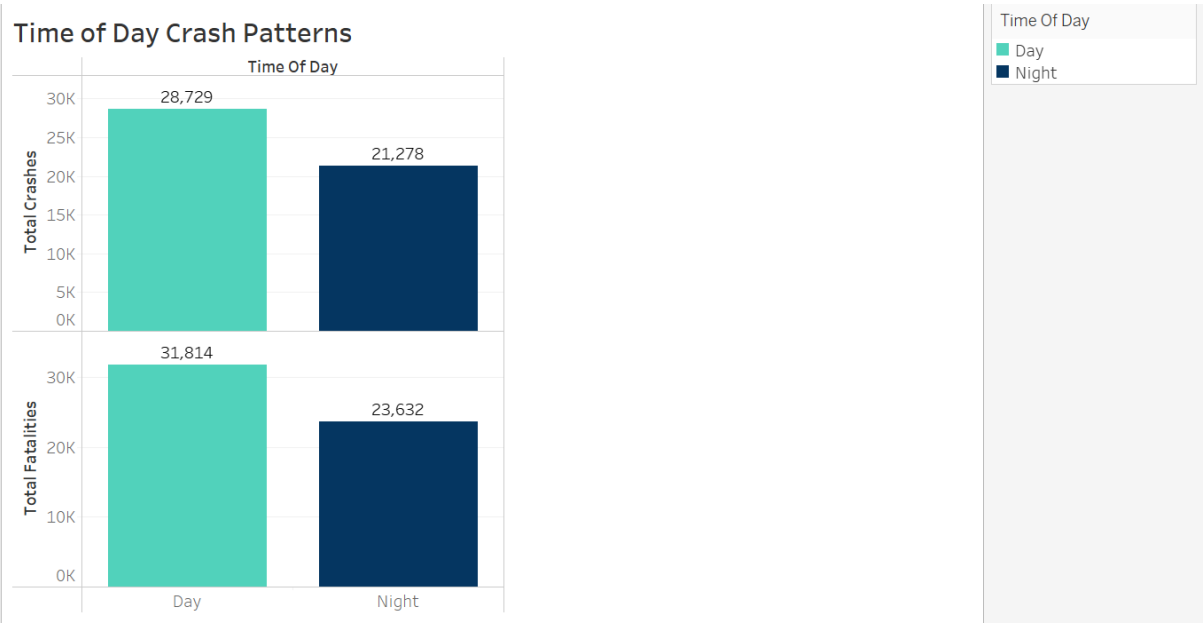
6. Motorcycle Pillion Passengers

- Crashes: 374
- Fatalities: 538
- Though involved in fewer crashes, their fatality-to-crash ratio is high, indicating high severity when they are involved.

7. Others & Unknown

- Low counts, suggesting proper categorization for most records. Still, this highlights data quality issues for a small portion of entries.

4. Is there any pattern between road user types and holiday periods in the occurrence of fatal crashes?



This dual bar chart visualization illustrates **the distribution of fatal crashes** and **associated fatalities** across **daytime and nighttime periods**.

Insights:

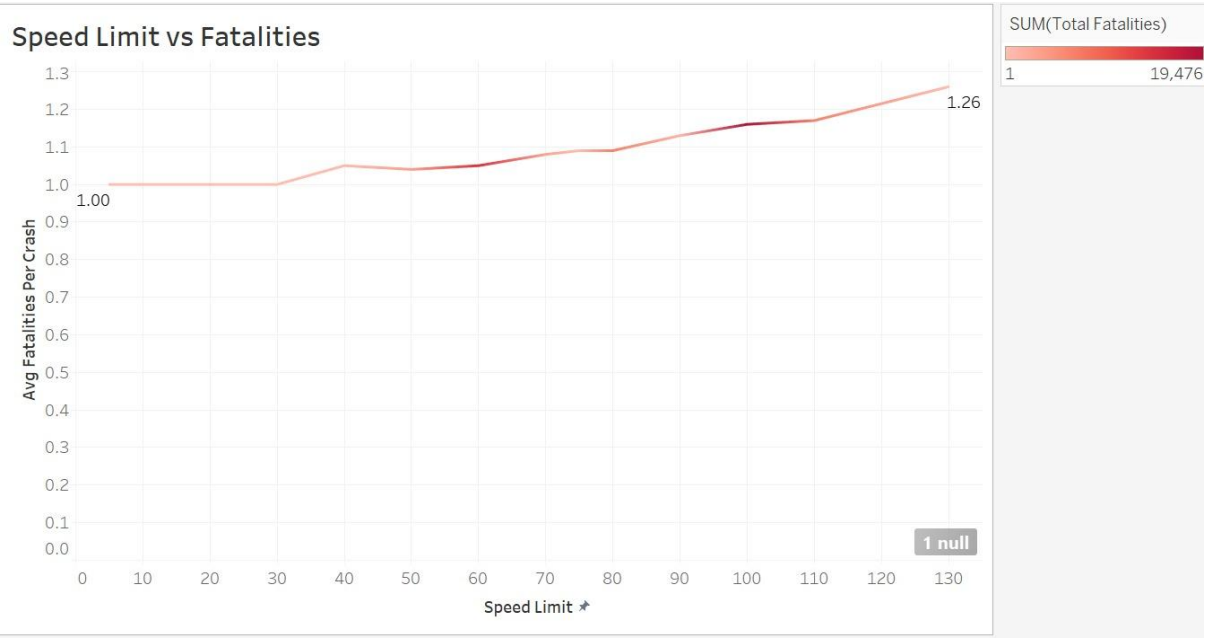
Daytime

- Highest volume of crashes (28,729), resulting in 31,814 fatalities.
- Crashes are more frequent during the day, likely due to higher levels of road activity — commuting, school runs, business deliveries, etc.
- Though daytime typically offers better visibility, factors like driver distraction, traffic congestion, and pedestrian exposure may elevate risks.

Nighttime

- Fewer crashes (21,278), but still high fatalities (23,632).
- This indicates a higher fatality-per-crash ratio, suggesting that night crashes are more severe.
- Likely contributing factors:
    - Poor visibility
    - Driver fatigue
    - Higher speeds due to lower traffic volumes
    - Alcohol or drug impairment

5. What is the relationship between Posted speed limit and number of fatalities?



This line graph visualizes the relationship between **posted speed limits** and the **average number of fatalities per crash**. Each point represents a speed limit zone and shows how severe crashes tend to be in that zone.

Insights:

 Flat Risk Zone (≤30 km/h):

- Crashes at very low speed limits (5–30 km/h) consistently have an average fatality rate of 1.00 per crash. These zones may involve pedestrians or urban intersections where crashes are rarer but still deadly due to vulnerable users.

 Moderate-Speed Zones (40–60 km/h):

- Fatality rates slightly increase, reflecting a balance between road activity and impact severity.
- These are typical urban and suburban road limits where both vehicle density and pedestrian exposure are higher.

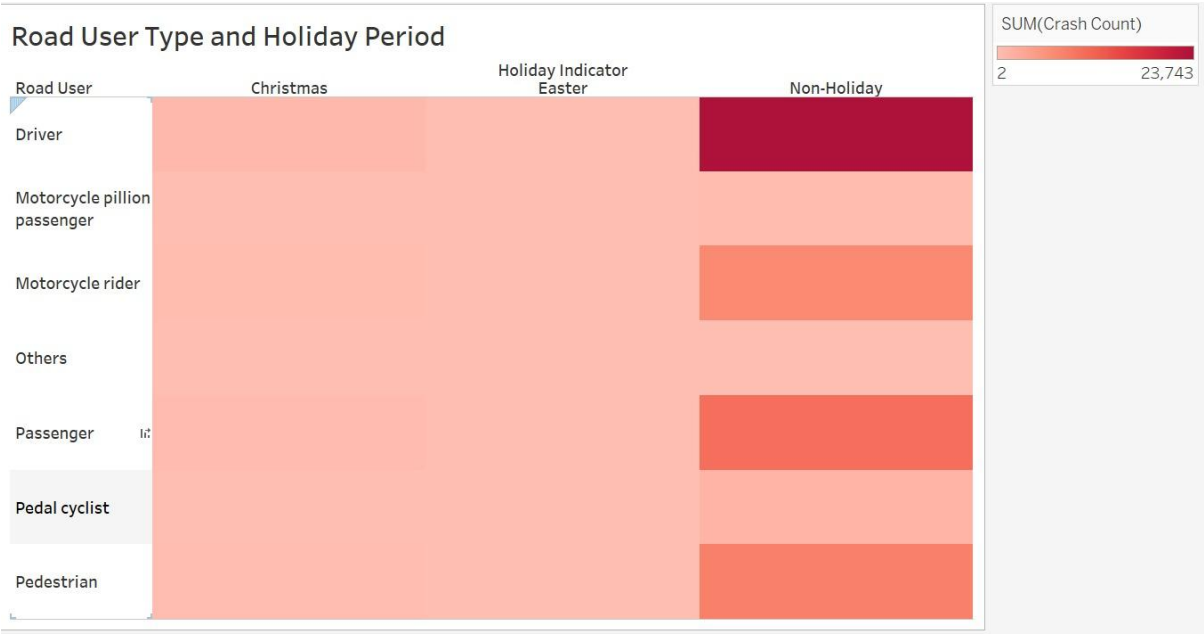
 Risk Rises with Speed (70–100 km/h):

- There's a clear upward trend in average fatalities per crash, reaching 1.16 at 100 km/h.
- This aligns with physics: higher speeds increase impact energy and reduce driver reaction time.

 Highest Severity at 130 km/h:

- Crashes at 130 km/h have the highest average fatalities per crash (1.26).
- Although crash count is low, these crashes are disproportionately fatal, confirming that higher speed = higher lethality.

6. Is there any pattern between road user types and holiday periods in the occurrence of fatal crashes?



This heatmap visualization represents the number of fatal crashes **(Crash Count)** by **Road User Type** across three periods: **Christmas**, **Easter**, and **Non-Holiday**. The darker the shade, the higher the number of crashes.

Insights:

 Drivers dominate fatal crashes across all periods, with 23,743 crashes occurring during non-holiday periods alone. This makes them the most vulnerable or exposed group on the road.

 Passengers also account for a significant number of fatal crashes, especially during non-holidays (10,308 crashes). Combined with drivers, they represent the bulk of road crash victims.

 Pedestrians and Motorcycle Riders show elevated risks:

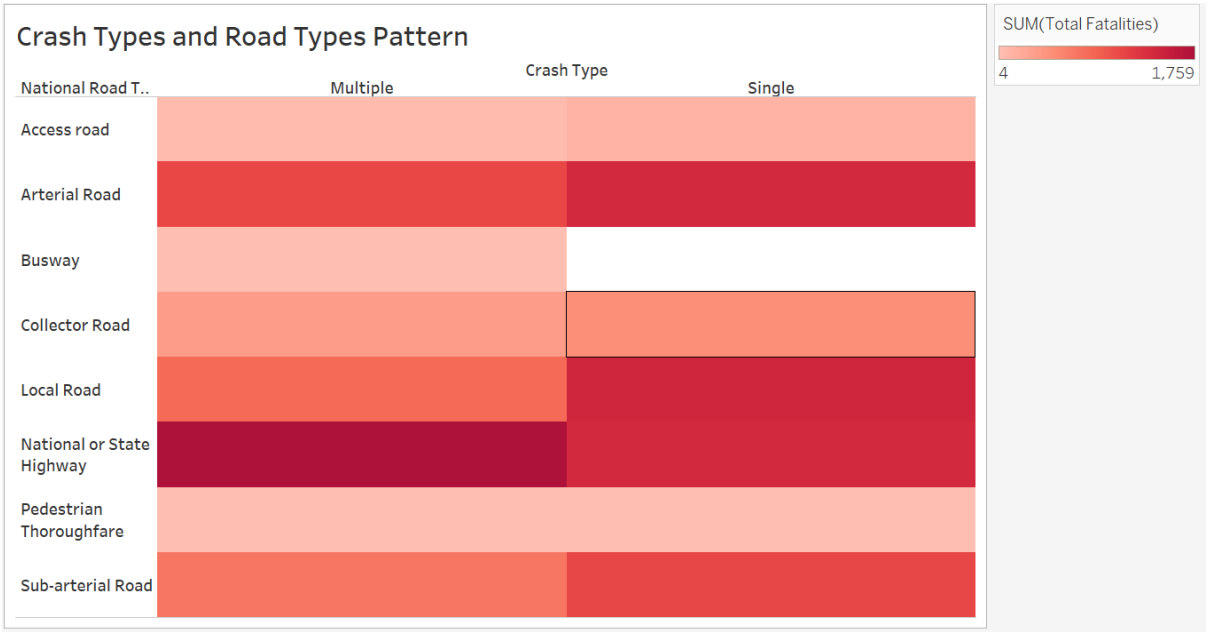
- Over 8,000 fatal pedestrian crashes during non-holidays.
- Motorcycle riders are particularly vulnerable, with a spike during non-holiday periods (7,007 crashes).

 Holiday Periods (Christmas & Easter) account for 1,820 crashes, which is ~3.6% of all fatal crashes:

- Despite being lower in volume, these periods might involve denser traffic, travel surges, or alcohol-related risks, especially affecting vulnerable groups like pedestrians and motorcyclists.

 ‘Others’ and ‘Unknown’ categories contribute very little to the overall crash count but still show some presence, indicating some data incompleteness or categorization ambiguity.

7. Which combinations of crash types and road types contribute most to total fatalities?



This heatmap visualization shows **the intersection of crash types (Single vs. Multiple)** and **national road types**, with **the colour intensity representing the total number of fatalities.**

Insights:

1. Highest Fatalities Occur on National or State Highways

- Multiple-vehicle crashes on highways resulted in the highest fatalities (1,759).
- Single-vehicle crashes also contributed significantly (1,404 fatalities).
- This indicates that highways, likely due to higher speed limits, are high-risk zones for both types of crashes.

2. Local Roads and Arterial Roads are also High-Fatality Zones

- Local Road (Multiple): 813 fatalities
- Local Road (Single): 1,435 fatalities
- Arterial Road (Multiple): 1,108 fatalities
- Arterial Road (Single): 1,406 fatalities
  These roads are typically within suburban or high-traffic urban zones, suggesting the density of vehicles and intersections may contribute to fatal crash risk.

3. Single-Vehicle Crashes Contribute More Fatalities Overall

- Across all road types:
  - Single crashes: 30,626 fatalities
  - Multiple crashes: 24,847 fatalities
- This counters a common assumption that multi-vehicle crashes are deadlier, suggesting that loss of control, speeding, or fatigue in solo driving incidents are very dangerous.

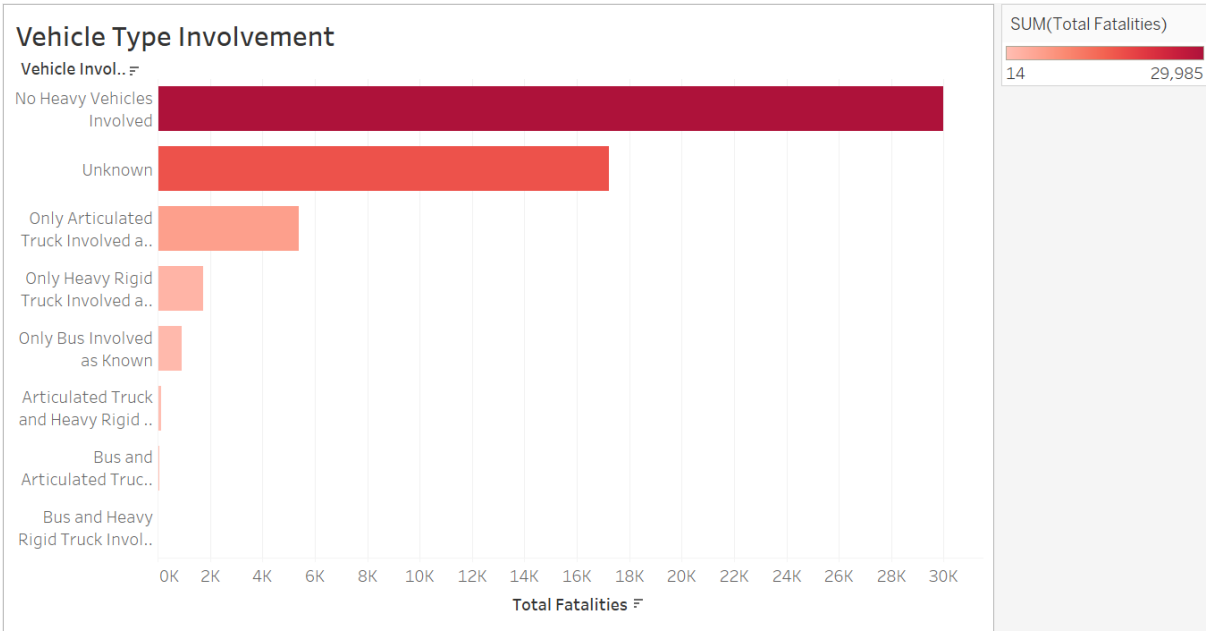
4. Access Roads and Sub-arterial Roads Show Lower But Non-trivial Impact

- These roads are lower-tier in hierarchy but still see hundreds of deaths:
    - Access road (Single + Multiple): 157 fatalities
    - Sub-arterial Road (Single + Multiple): 1,807 fatalities

5. Busways and Pedestrian Thoroughfares Have Minimal but Noticeable Fatalities

- Minimal data:
    - Busway: 4 fatalities (Multiple crashes only)
    - Pedestrian Thoroughfare: 14 total fatalities
- Though the count is low, the presence of fatalities in pedestrian zones is a red flag for vulnerable road users.

8. Which vehicle involvement types are associated with the highest number of fatal crashes and fatalities?



This bar chart illustrates the **total number of fatalities** associated with different categories of **heavy vehicle involvement** in fatal road crashes. The analysis was derived from the *vehicle_involvement_description* attribute of the *dim_vehicle_involvement* dimension and joined with *fact_crash.*

Insights:

 Majority of Fatalities Occurred Without Heavy Vehicles

- The **overwhelming majority (~30,000 fatalities)** occurred in crashes that **did not involve any heavy vehicles**.
- This suggests that passenger vehicles, motorcyclists, and pedestrians are primary contributors to fatal outcomes.

 High Volume of 'Unknown' Involvement (~17K Fatalities)

- A **large portion of records have unspecified or missing involvement types**.
- This lack of data hinders precise trend analysis and highlights the need for improved reporting or data collection standards.

 Articulated Trucks Have Highest Fatalities Among Heavy Vehicles

- With nearly **5,400 fatalities**, **articulated trucks alone** represent the **most hazardous heavy vehicle type** when involved in crashes.
- This may be due to their mass, braking distance, and usage on high-speed rural highways.

 Combined Heavy Vehicle Involvement Is Rare but Deadly

- Scenarios involving **multiple heavy vehicle types** (e.g., buses with trucks) are rare (fewer than 100 crashes) but often severe, indicating complex or high-impact incidents.

 Buses Appear Least Fatal Among Heavy Vehicles

- Crashes involving **only buses** account for 937 fatalities — lower than truck-related categories — possibly due to regulated operation and lower average speeds in urban settings.

Data Mining: Association Rule Mining

1. Algorithm

The Apriori algorithm was employed for association rule mining using the *mlxtend* library. This algorithm identifies frequent itemsets in a transactional dataset based on a minimum support threshold and generates rules based on confidence and lift metrics.[1]

2. Methodology

- **Input Dataset**: *merged_fatalities_crashes.csv* (cleaned dataset in our case) with categorical columns like Gender, Age Group, Crash Type and so on.
- **Preprocessing**: Each record was transformed into a transaction of attribute-value pairs (e.g., *"Gender=Male").*
- **Encoding**: *TransactionEncoder* was used to one-hot encode transactions.
- **Apriori Settings**:
    - *min_support=0.1*
    - *min_threshold for lift=1.0*
- **Focus**: Rules with "*Road User="* on the right-hand side (consequent).
- **Ranking**: Top rules were selected based on ***Lift*** and ***Confidence***.

3. Top Rules with "Road User" as Consequent

#	Antenecents	Consequent	Support	Confidence	Lift
1	Gender=Female	Road User=Passenger	10.7%	37.9%	1.685
2	Gender=Female	Holiday=Non-Holiday, Road User=Passenger	10.2%	36.1%	1.681
3	Gender=Female, Holiday=Non-Holiday	Road User=Passenger	10.2%	37.6%	1.674
4	Crash Type=Single, Holiday=Non-Holiday	Road User=Pedestrian	13.0%	24.5%	1.593
5	Crash Type=Single	Road User=Pedestrian	13.4%	24.3%	1.581

4. Key rules

Top 3 Rules are:

Rule 1

- **If** the person involved is **Female**, **then** there's a **37.9%** chance they are a **Passenger**.
- This is **1.685 times more likely** than a random occurrence, indicating a strong association.

Rule 2

- **If** the person is **Female** and the crash occurred on a **Non-Holiday**, **then** they are likely to be a **Passenger**.
- The **lift** (1.681) again indicates a much higher than average likelihood.

Rule 3

- **If** a crash occurred with a **Single vehicle** on a **Non-Holiday**, the road user is likely to be a **Pedestrian**.
- Suggests isolated pedestrian incidents may be more common on non-holidays.

5. Insights from Rules

- **Gender patterns**: Females are more frequently associated with being **Passengers** than Drivers or other roles.
- **Crash context**: **Single-vehicle crashes** have strong ties to **Pedestrian involvement**, hinting at scenarios like run-overs.
- **Temporal patterns**: Many strong rules emerge during **non-holiday periods**, perhaps indicating routine travel behavior.

6. Recommendations to Government

Based on the mined rules, here are three actionable suggestions:

1. Passenger Safety Campaigns for Women

Given the high frequency of female passengers in fatal crashes, targeted safety education, seatbelt reminders, and safer ride programs for women should be implemented.

2. Pedestrian Zone Safety Upgrades

Single-vehicle crashes involving pedestrians suggest infrastructure weaknesses. Investment in **pedestrian crossings, lighting, and barriers** during regular (non-holiday) periods is crucial.

3. Smart Enforcement Outside Holiday Periods

Non-holiday weekdays show concentrated risk patterns. Governments should **not limit safety efforts to holiday campaigns** but maintain **consistent enforcement and visibility**.

References:

[1] OpenAI, *ChatGPT* (version GPT-4), ChatGPT, [Online]. Available: https://chat.openai.com, Accessed: Apr. 11, 2025.