U.S. Road Accidents (2016–2022) EDA Report

Suraj Shrestha

Executive Summary

This analysis explores a sample of **500,000** U.S. traffic accidents (2016–March 2023)** to uncover patterns across **time, geography, weather, and road environments**. The goal is to uncover insights and trends that can guide safety strategies, urban planning, and policy decisions.

Key Insights:

This analysis of U.S. traffic accidents (2016–2022) reveals three main themes:

- **Time Patterns:** Accidents increased steadily from 2016 to 2022, more than quadrupling over the period, with particularly sharp rises after 2020. Crashes cluster around **commuting hours (7–9 AM, 4–6 PM)**. Winter months, particularly **December**, consistently show **higher crash counts**. Summer months (June–July) generally record the fewest accidents, while late autumn and winter (October–December) consistently show elevated levels, highlighting strong seasonal effects.
- **Geographic Patterns:** Large states like **California, Texas, and Florida** report the most accidents, but smaller states such as **South Carolina and Montana** rank much higher when normalized by population.
- **Environmental Factors:** Most accidents occur in clear weather, but severity increases in rain, snow, and fog. Traffic signals and crossing dominate crash counts, while railways and junctions are less frequent but carry higher severity.

Conclusion - Recommendations

Based on the insights, the following actions are recommended:

- Improve high-risk environments like busy intersections and junctions with better signage, adaptive signals, speed enforcement, etc.
- Target rush hour & winter safety campaigns.
- Deploy weather-responsive roadway measures.
- States with high per-capita accident rates should receive tailored interventions, not just high-volume states.
- Future Research: Incorporate traffic density, hospital outcomes, and fatality data to better understand accident consequences and prevention strategies.

"This EDA project demonstrates my ability to clean messy real-world data, uncover meaningful insights across time, geography, and environment, and translate findings into actionable recommendations."

Limitations

- Missing weather data could bias severity analysis.
- Accident reporting may vary by jurisdiction.
- Population normalization was done for 2022 only; extending it across years would improve accuracy.

Data Description

- **Source:** U.S. Accidents (2016-2023) (Kaggle) https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents/data?select=US_Accidents_March23.csv.
- **Size:** 500,000 sampled records.
- **Key variables used:** accident severity (1–4), timestamps, location (state, city, lat/lng), weather conditions, environmental flags (junction, traffic signal, etc.).
- Added **2022 state population estimates** from the U.S. Census data to normalize accident rates. Source: https://www2.census.gov/programs-surveys/popest/datasets/2020-2024/state/totals/.

Data Loading & Initial Checks

```
library(readr)
library(dplyr)
library(tidyverse)
library(tidyr)
library(lubridate)
library(stringr)
```

```
US_Accidents <- read_csv("~/Desktop/US_Accidents Project/US_Accidents_March23_sampled_500k.csv")


# Overview of missing values
colSums(is.na(US_Accidents))
```

```
##                    ID                Source              Severity
##                     0                     0                     0
##            Start_Time              End_Time             Start_Lat
##                     0                     0                     0
##             Start_Lng              End_Lat              End_Lng
##                     0                220377                220377
##          Distance(mi)           Description                Street
##                     0                     1                   691
##                  City                County                 State
##                    19                     0                     0
##               Zipcode               Country              Timezone
##                   116                     0                   507
##          Airport_Code     Weather_Timestamp         Temperature(F)
##                  1446                  7674                 10466
##         Wind_Chill(F)          Humidity(%)          Pressure(in)
##                129017                 11130                  8928
```

```
##        Visibility(mi)        Wind_Direction       Wind_Speed(mph)
##                 11291                 11197                 36987
##     Precipitation(in)     Weather_Condition               Amenity
##                142616                 11101                     0
##                  Bump              Crossing              Give_Way
##                     0                     0                     0
##              Junction              No_Exit               Railway
##                     0                     0                     0
##            Roundabout              Station                  Stop
##                     0                     0                     0
##       Traffic_Calming       Traffic_Signal          Turning_Loop
##                     0                     0                     0
##        Sunrise_Sunset       Civil_Twilight     Nautical_Twilight
##                  1483                  1483                  1483
## Astronomical_Twilight
##                  1483
```

The dataset is generally well-populated, but some fields show substantial missingness:

- **End_Lat / End_Lng (~220K missing):** End coordinates of road segments are often missing. Since *Start_Lat* and *Start_Lng* are complete, we can safely rely on those for geographic analysis.
- **Weather variables:**
  - *Wind_Chill (129K), Precipitation (142K), Humidity (11K), Pressure (9K), Visibility (11K)* — incomplete, might be due to sensor or reporting gaps.
- **Weather_Timestamp (~7.6K missing):** Not critical since *Start_Time* provides temporal context.
- **Twilight/Sunrise fields (~1.5K missing):** Minor impact, can be dropped.
- **Zipcode (~116) and City (~19):** Negligible compared to total dataset size.

Decision: I will keep core features (Severity, Start_Time, Start_Lat/Start_Lng, key weather conditions) and exclude variables with heavy missingness for cleaner, more reliable analysis.

Data Cleaning & Preprocessing

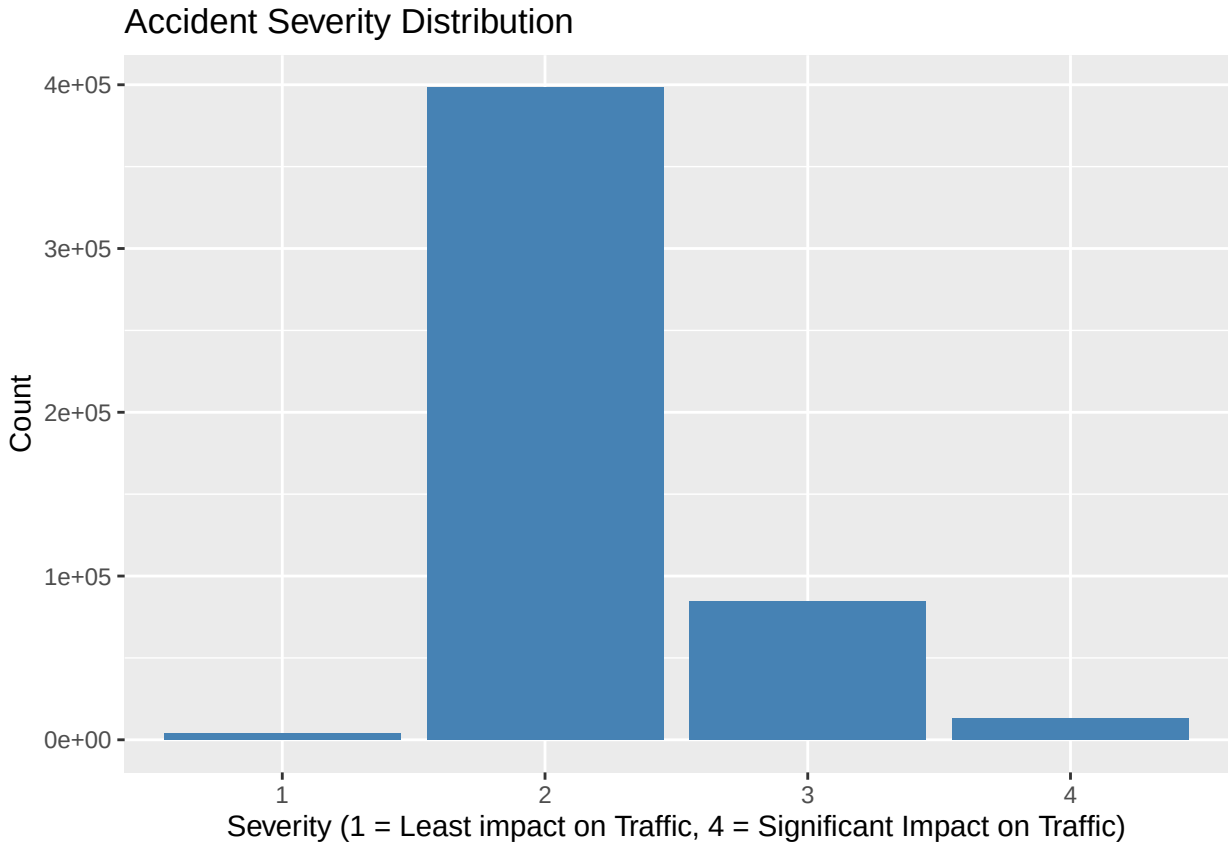
```r
# Select relevant columns
clean_data <- US_Accidents %>%
  select(
    Severity,
    Start_Time,
    End_Time,
    Start_Lat,
    Start_Lng,
    Temperature = `Temperature(F)`,
    Humidity = `Humidity(%)`,
    Pressure = `Pressure(in)`,
    Wind_Speed = `Wind_Speed(mph)`,
    Weather_Condition,
    City, State
  ) %>%
  drop_na(Severity, Start_Time)
```

Focusing on severity, time, weather, and location provides a balanced view without overloading on minor features.

1. Severity of Accidents

1a. Severity Distribution

```
ggplot(clean_data, aes(x = factor(Severity))) +
  geom_bar(fill = "steelblue") +
  labs(
    title = "Accident Severity Distribution",
    x = "Severity (1 = Least impact on Traffic, 4 = Significant Impact on Traffic)",
    y = "Count"
  )
```



Findings

- The data is **highly skewed** for Severity 2.
- Most accidents are at severity level 2 (minor–moderate). Severe accidents (level 4) are relatively rare but critical.

1b. Accident Duration by Severity

```
# Duration Summary by Severity - min, max, mean, median, mode

# calculate duration
accidents_hours <- clean_data %>%
  mutate(Duration_hours = as.numeric(difftime(End_Time, Start_Time, units = "hours")))

# Helper: quick mode function
get_mode <- function(x) {
  x <- na.omit(x)
  if(length(x) == 0) return(NA)
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

# Summaries by severity
duration_summary <- accidents_hours %>%
  group_by(Severity) %>%
  summarise(
    Min    = min(Duration_hours, na.rm = TRUE),
    Max    = max(Duration_hours, na.rm = TRUE),
    Mean   = mean(Duration_hours, na.rm = TRUE),
    Median = median(Duration_hours, na.rm = TRUE),
    Mode   = get_mode(round(Duration_hours, 1)),  # rounded to 0.1 hr bins
    n      = n()
  )

duration_summary
```

```
## # A tibble: 4 x 7
##   Severity    Min      Max   Mean Median  Mode      n
##      <dbl>  <dbl>    <dbl>  <dbl>  <dbl> <dbl>  <int>
## 1        1 0.25       7.48  0.766  0.746   0.5   4274
## 2        2 0.0417 37273.    7.63   1.29    0.5 398142
## 3        3 0.172   4944.    1.28   0.74    0.5  84520
## 4        4 0.167  17329.   27.9    2.15    6    13064
```

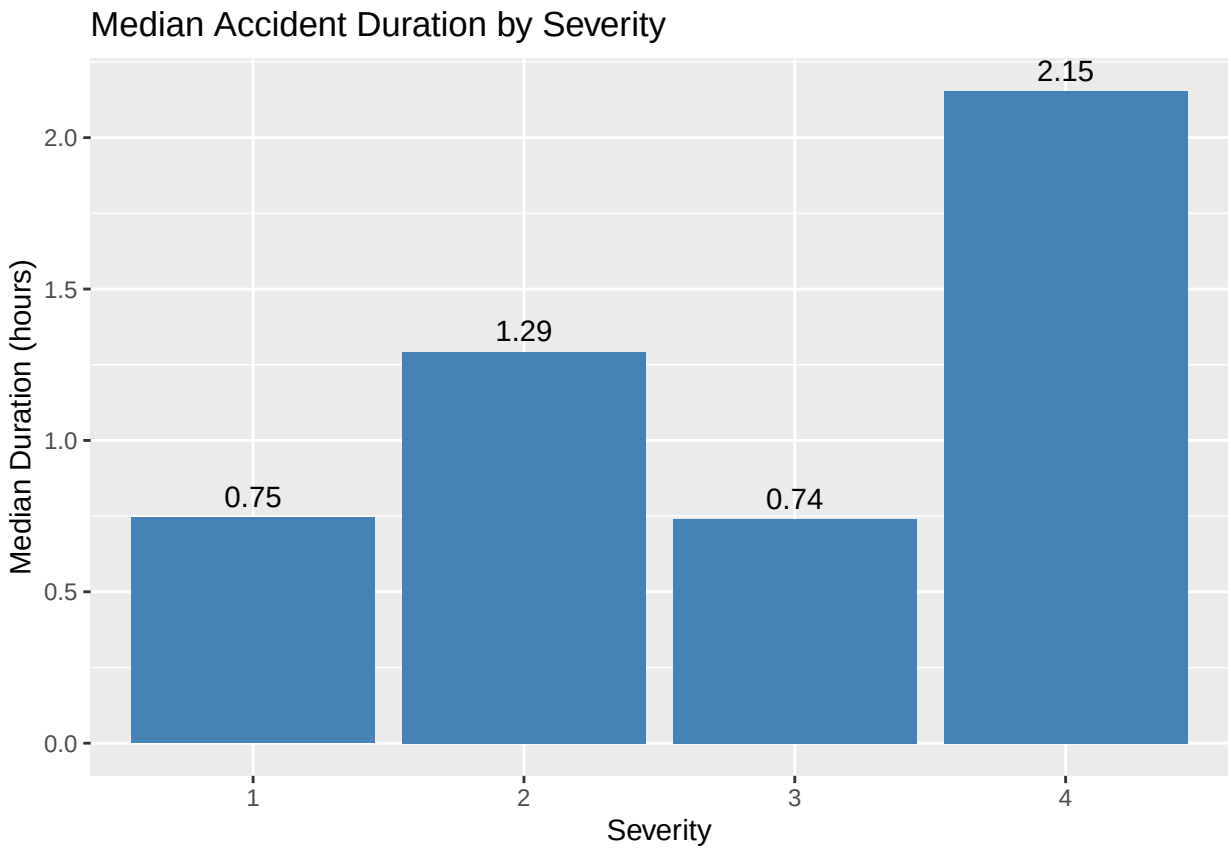
Findings:

- **Extreme outliers** exist (e.g., Severity 2 accidents lasting over 37,000 hours 4 years)
- The mean is heavily inflated by these outliers. Example, Severity 2 has a mean of 7.6 hrs but a median of only 1.3 hrs.
- The median is stable and much closer to the typical accident duration.
- The mode shows that many accidents are resolved within 30–60 minutes, but it does not capture the spread across severities.

Thus Median provides a robust summary and is unaffected by extreme outliers. It avoids exaggeration and gives a clearer picture of realistic, day-to-day durations.

```r
median_duration <- accidents_hours %>%
  group_by(Severity) %>%
  summarise(Median_Duration = median(Duration_hours, na.rm = TRUE))

ggplot(median_duration, aes(x = factor(Severity), y = Median_Duration)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  geom_text(aes(label = round(Median_Duration, 2)), vjust = -0.5) +
  labs(
    title = "Median Accident Duration by Severity",
    x = "Severity",
    y = "Median Duration (hours)"
  )
```



Findings

- Minor accidents (Severity 1–3) usually last ~1 hour or less.
- Severe accidents (Severity 4) typically last ~2 hours, although some extend much longer.

1c. Capping durations at 72 hours to handle extreme outliers

```r
# Cap durations at 72 hours to handle extreme outliers
accidents_capped <- accidents_hours %>%
  mutate(Duration_capped = ifelse(Duration_hours > 72, 72, Duration_hours))
```

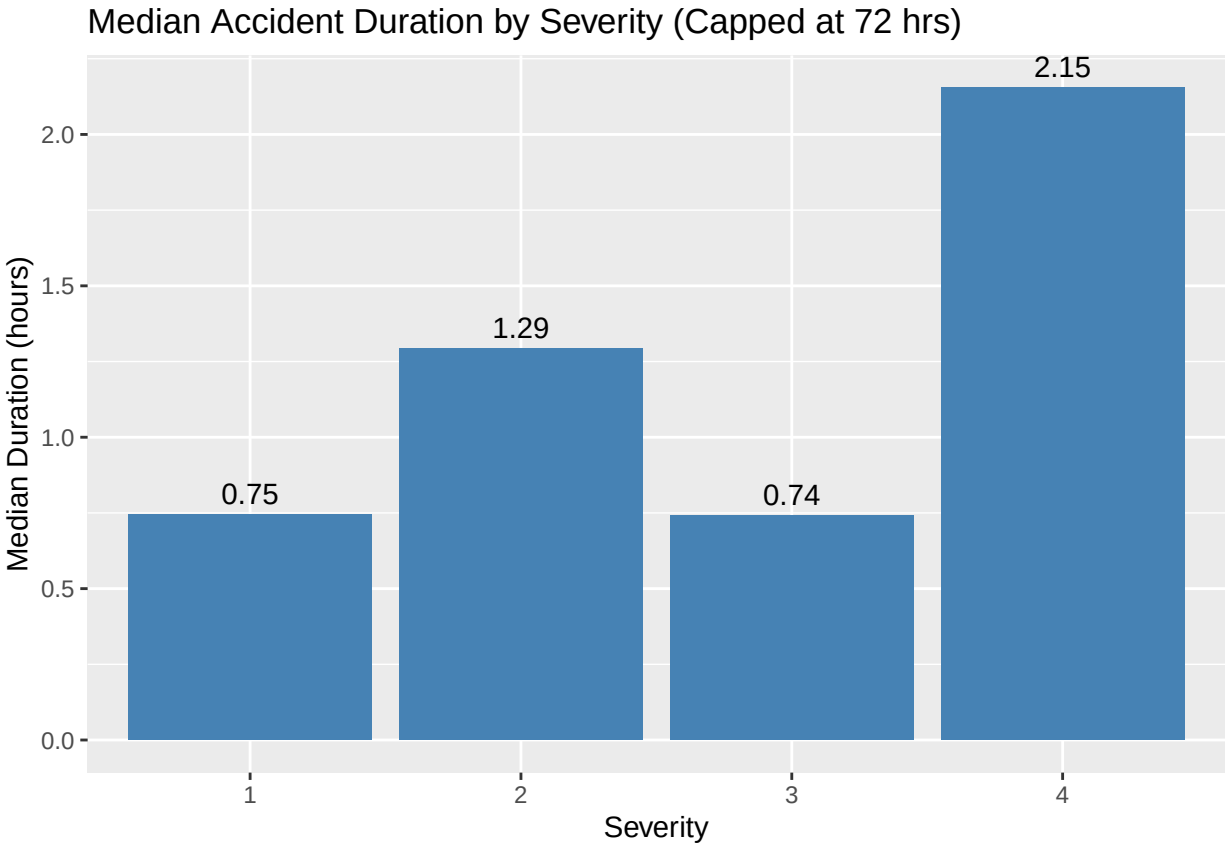
```r
# Summarize again by severity
duration_summary_capped <- accidents_capped %>%
  group_by(Severity) %>%
  summarise(
    Min    = min(Duration_capped, na.rm = TRUE),
    Max    = max(Duration_capped, na.rm = TRUE),
    Mean   = mean(Duration_capped, na.rm = TRUE),
    Median = median(Duration_capped, na.rm = TRUE),
    Mode   = get_mode(round(Duration_capped, 1)),
    n      = n()
  )

duration_summary_capped
```

```
## # A tibble: 4 x 7
##   Severity    Min   Max  Mean Median  Mode      n
##      <dbl>  <dbl> <dbl> <dbl>  <dbl> <dbl>  <int>
## 1        1 0.25    7.48 0.766  0.746   0.5   4274
## 2        2 0.0417 72    2.09   1.29    0.5 398142
## 3        3 0.172  72    1.15   0.74    0.5  84520
## 4        4 0.167  72    3.99   2.15    6    13064
```

```r
median_duration_capped <- accidents_capped %>%
  group_by(Severity) %>%
  summarise(Median_Duration = median(Duration_capped, na.rm = TRUE))

ggplot(median_duration_capped, aes(x = factor(Severity), y = Median_Duration)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  geom_text(aes(label = round(Median_Duration, 2)), vjust = -0.5) +
  labs(
    title = "Median Accident Duration by Severity (Capped at 72 hrs)",
    x = "Severity",
    y = "Median Duration (hours)"
  )
```



Final Decision on Duration Measure.

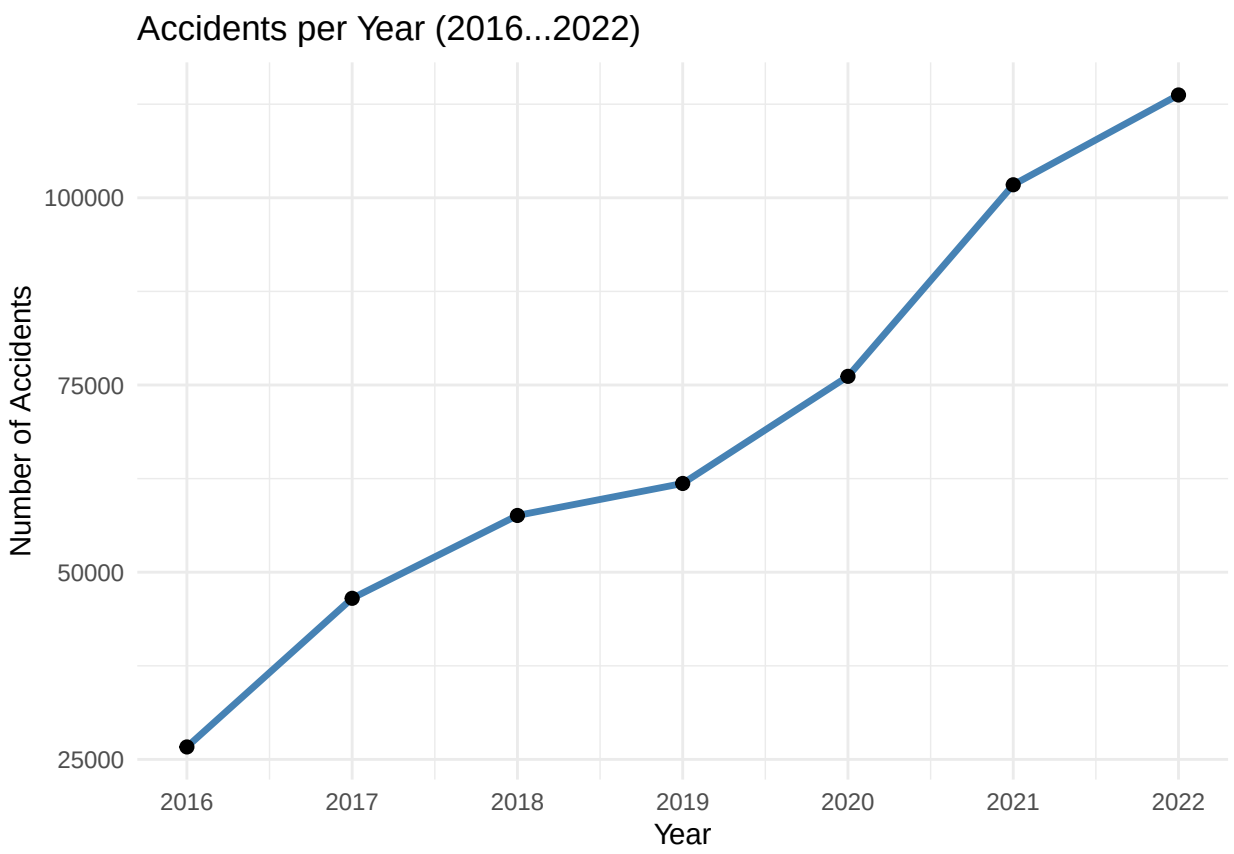
- Even after capping accident durations at 72 hours, the results remained the same. This confirms that the **median** is robust and unaffected by extreme outliers, while the **mean** was inflated.
- Therefore, I will report **median accident duration by severity** as the most reliable measure. This ensures my findings reflect typical accident clearance times rather than being distorted by rare anomalies.

2. Temporal Patterns

2a. Accidents per Year

```
clean_data %>%
  mutate(Year = year(Start_Time)) %>%
  filter(Year <= 2022) %>%  # filterring out 2023 since we have data unitl March 2023 only
  count(Year) %>%
  ggplot(aes(x = Year, y = n)) +
  geom_line(color = "steelblue", linewidth = 1.2) +
  geom_point(size = 2, color = "black") +
  scale_x_continuous(breaks = seq(2016, 2022, 1)) +
  labs(
    title = "Accidents per Year (2016-2022)",
    x = "Year",
    y = "Number of Accidents"
```

```
  ) +
  theme_minimal()
```

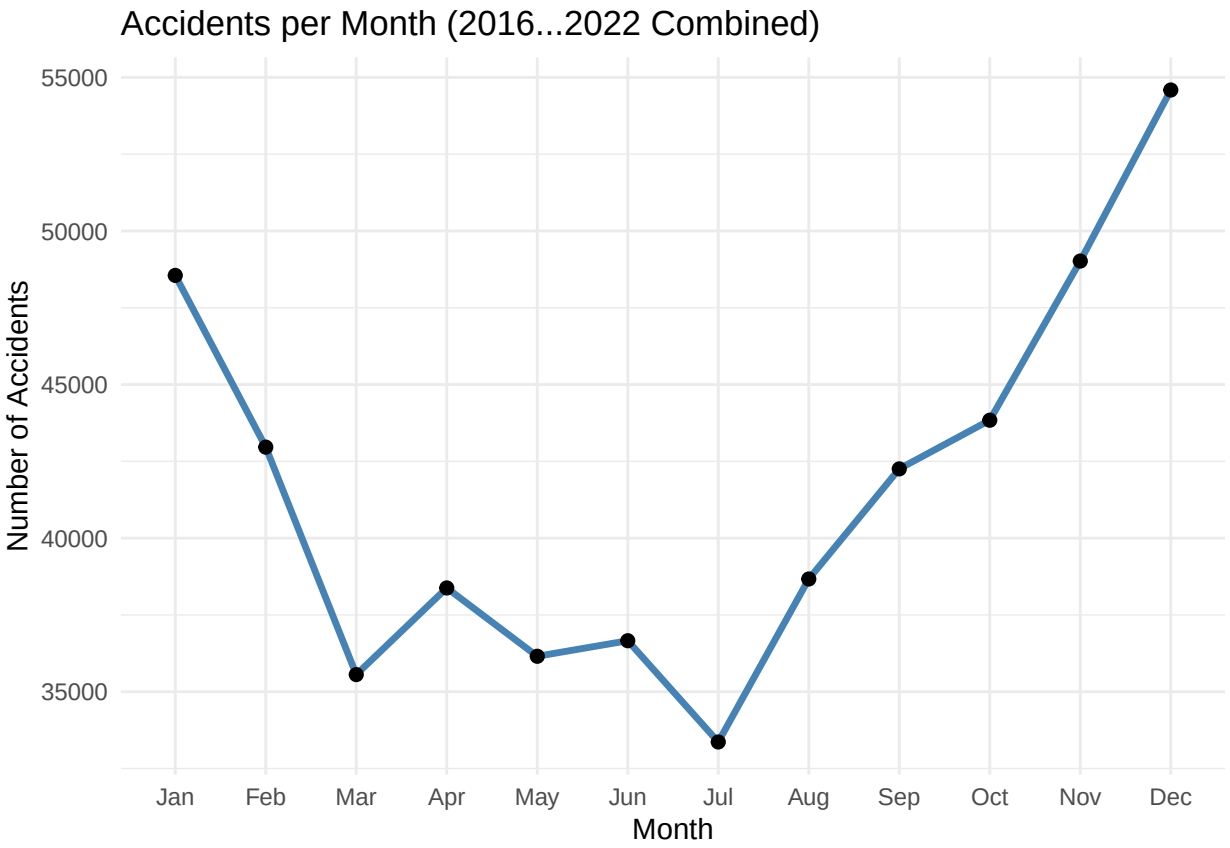


Findings

- Accident counts increased steadily from 2016 to 2022, more than quadrupling over the period, with particularly sharp rises after 2020 — indicating growing traffic exposure and improved reporting over time.

2b.Accidents per Month

```
clean_data %>%
  mutate(Month = month(Start_Time, label = TRUE, abbr = TRUE)) %>%
  count(Month) %>%
  ggplot(aes(x = Month, y = n, group = 1)) +
  geom_line(color = "steelblue", linewidth = 1.2) +
  geom_point(size = 2, color = "black") +
  labs(
    title = "Accidents per Month (2016-2022 Combined)",
    x = "Month",
    y = "Number of Accidents"
  ) +
  theme_minimal()
```

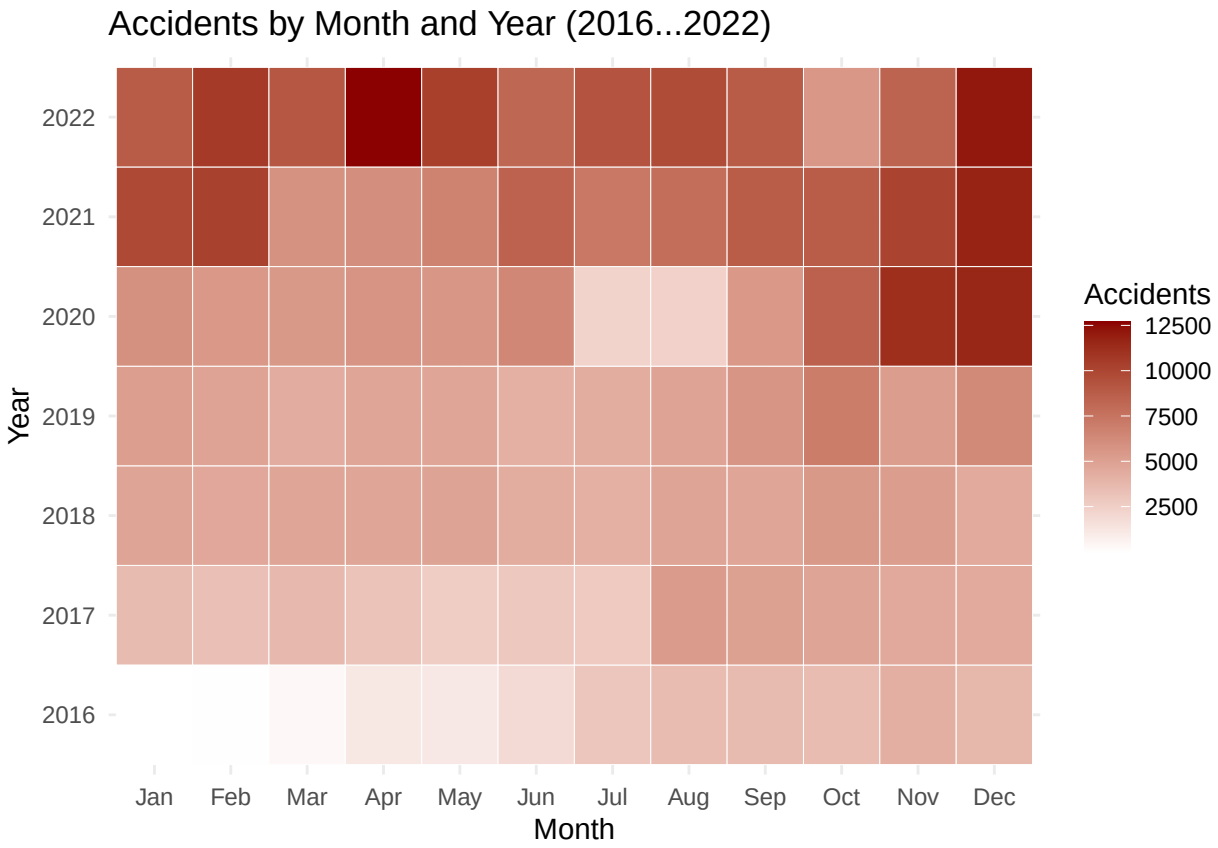


Findings

- Looking at the combined records from 2016 to March 2023, accident frequency dips in spring and summer, but rises steadily through fall, peaking in December, suggesting that adverse winter conditions significantly increase crash risk.

2c. Accidents by Year and Month

```
# Heatmap of Accidents by Year and Month
clean_data %>%
  mutate(
    Year = year(Start_Time),
    Month = month(Start_Time, label = TRUE, abbr = TRUE)
  ) %>%
  filter(Year <= 2022) %>%
  count(Year, Month) %>%
  ggplot(aes(x = Month, y = factor(Year), fill = n)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "white", high = "darkred") +
  labs(
    title = "Accidents by Month and Year (2016-2022)",
    x = "Month",
    y = "Year",
    fill = "Accidents"
  ) +
  theme_minimal()
```

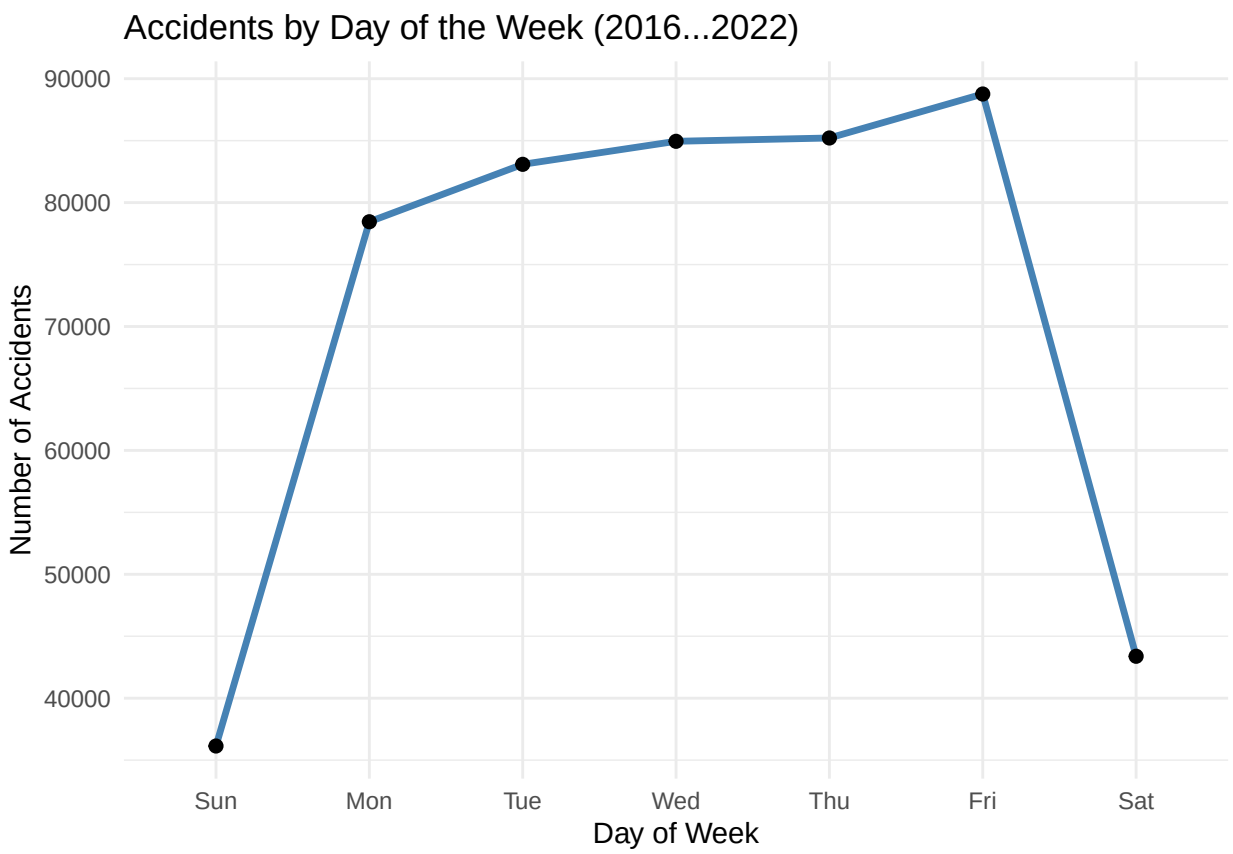


Findings

- Accident counts have increased consistently from 2016 to 2022, with December standing out as the peak month across multiple years. Summer months (June–July) generally record the fewest accidents, while late autumn and winter (October–December) consistently show elevated levels, highlighting strong seasonal effects in road safety.

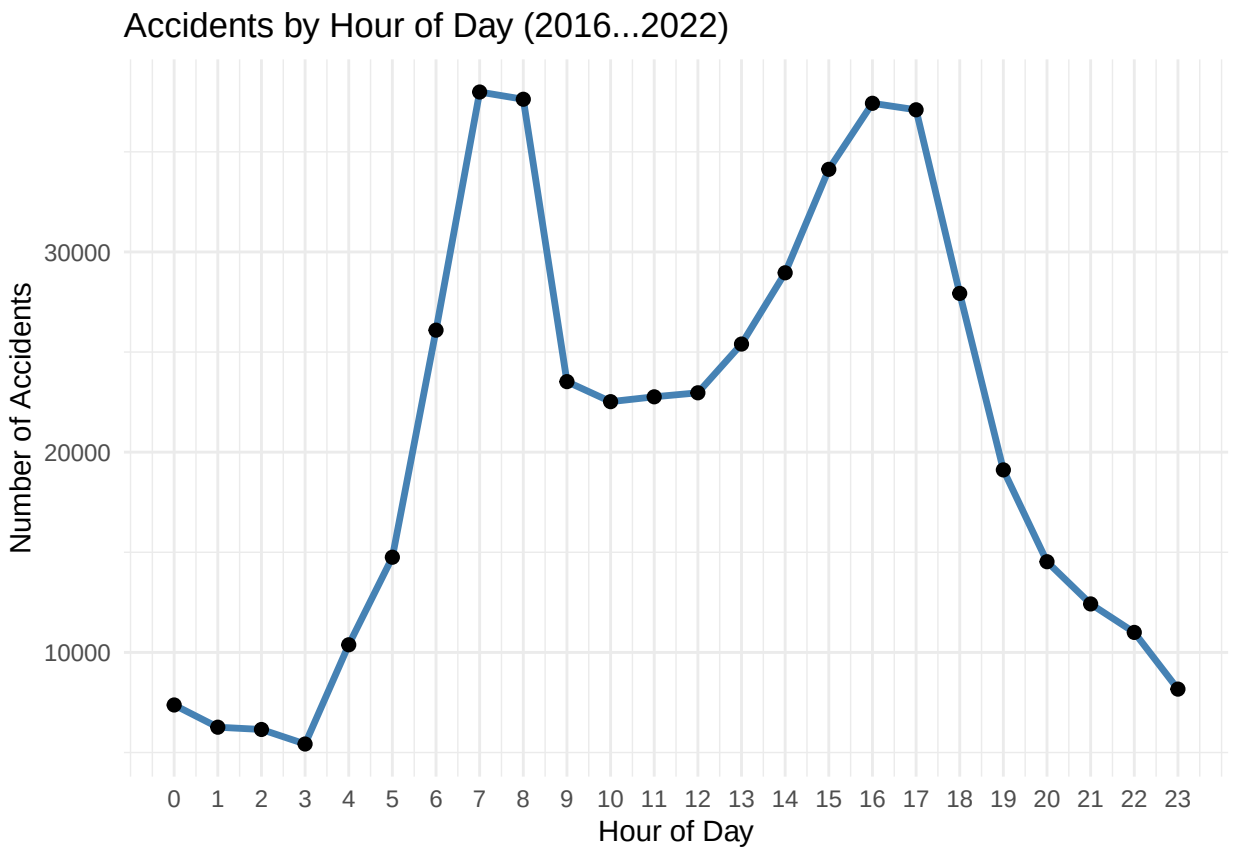
2d. Accidents by Day of the Week

```
clean_data %>%
  mutate(Weekday = wday(Start_Time, label = TRUE, abbr = TRUE)) %>%
  count(Weekday) %>%
  ggplot(aes(x = Weekday, y = n, group = 1)) +
  geom_line(color = "steelblue", linewidth = 1.2) +
  geom_point(size = 2, color = "black") +
  labs(
    title = "Accidents by Day of the Week (2016-2022)",
    x = "Day of Week",
    y = "Number of Accidents"
  ) +
  theme_minimal()
```



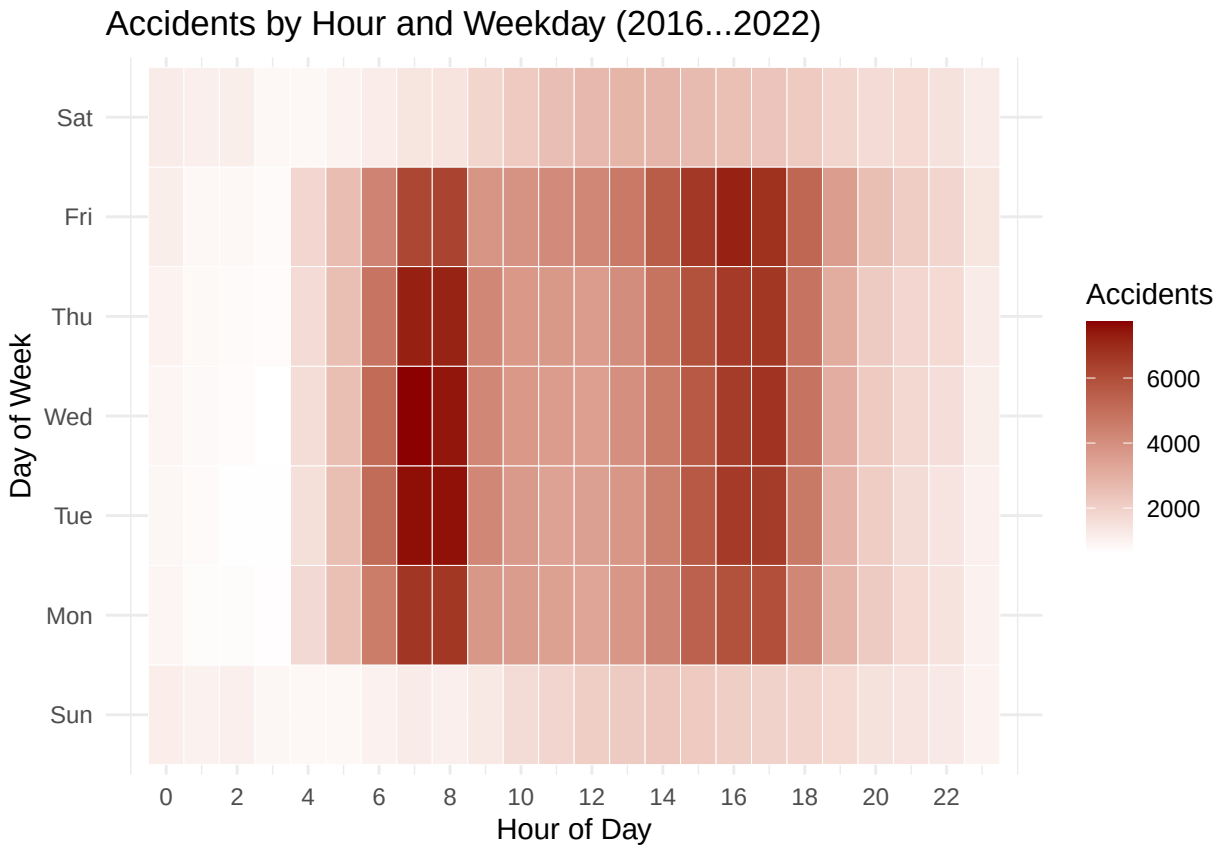
2e. Accidents by Hour of the Day

```
clean_data %>%
  mutate(Hour = hour(Start_Time)) %>%
  count(Hour) %>%
  ggplot(aes(x = Hour, y = n, group = 1)) +
  geom_line(color = "steelblue", linewidth = 1.2) +
  geom_point(size = 2, color = "black") +
  scale_x_continuous(breaks = 0:23) +
  labs(
    title = "Accidents by Hour of Day (2016-2022)",
    x = "Hour of Day",
    y = "Number of Accidents"
  ) +
  theme_minimal()
```



2f. Accidents by Hour and Weekday

```
# Heatmap of Accidents by Hour and Weekday
clean_data %>%
  mutate(
    Weekday = wday(Start_Time, label = TRUE, abbr = TRUE),
    Hour = hour(Start_Time)
  ) %>%
  count(Weekday, Hour) %>%
  ggplot(aes(x = Hour, y = Weekday, fill = n)) +
  geom_tile(color = "white") +
  scale_fill_gradient(low = "white", high = "darkred") +
  scale_x_continuous(breaks = seq(0, 23, 2)) +
  labs(
    title = "Accidents by Hour and Weekday (2016-2022)",
    x = "Hour of Day",
    y = "Day of Week",
    fill = "Accidents"
  ) +
  theme_minimal()
```



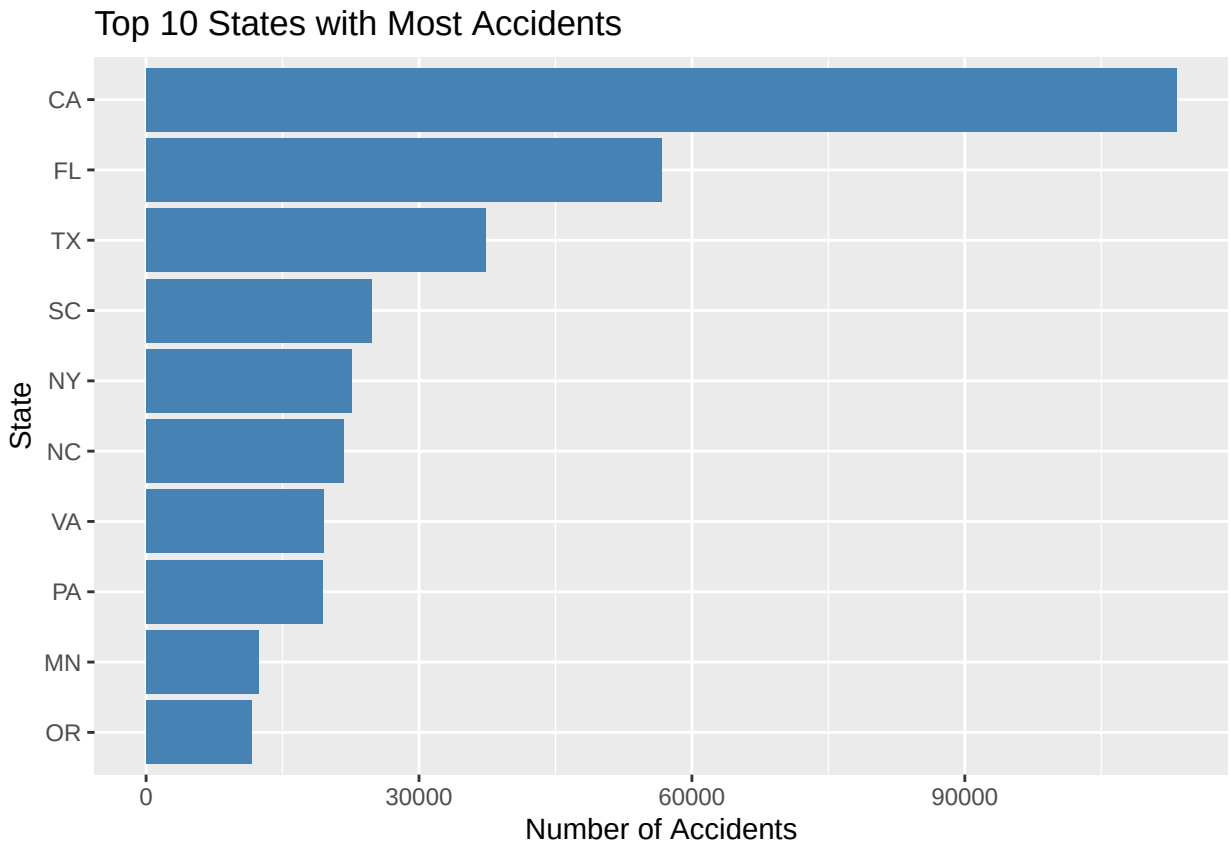
Findings

- Most accidents happen on weekdays during rush hours, especially around 7–9 AM and 4–6 PM. Nights and weekends have fewer accidents.

3. Geographic Analysis

3a. Top States with Most Accidents

```
clean_data %>%
  count(State, sort = TRUE) %>%
  top_n(10, n) %>%
  ggplot(aes(x = reorder(State, n), y = n)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  coord_flip() +
  labs(title = "Top 10 States with Most Accidents",
       x = "State",
       y = "Number of Accidents")
```

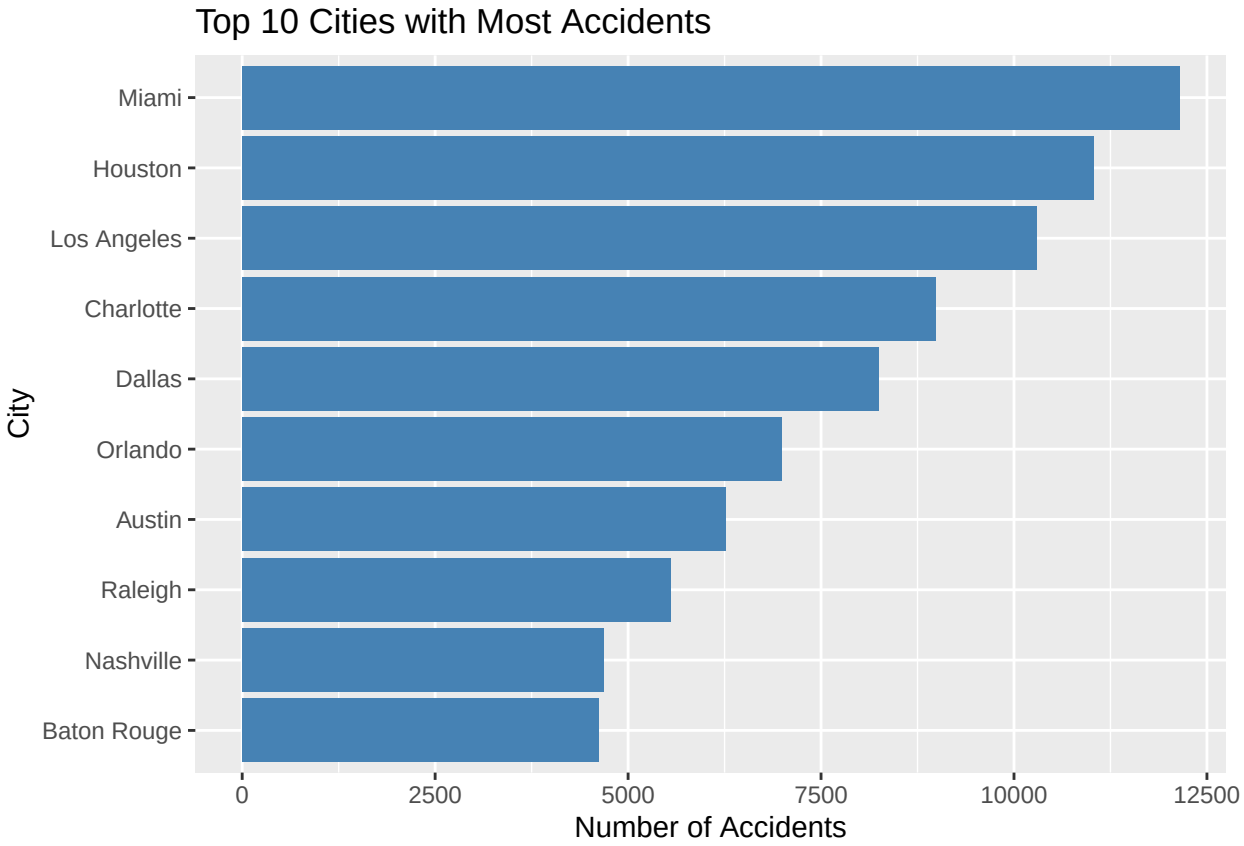


Findings

California, Florida, and Texas dominate accident counts, with California alone far ahead of other states, showing how high population and dense traffic contribute to higher crash volumes.

3b. Top Cities with Most Accidents

```
clean_data %>%
  count(City, sort = TRUE) %>%
  top_n(10, n) %>%
  ggplot(aes(x = reorder(City, n), y = n)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  coord_flip() +
  labs(title = "Top 10 Cities with Most Accidents", x = "City", y = "Number of Accidents")
```



Findings

- Miami, Houston, and Los Angeles lead as the most accident-prone cities, with large metro areas across the South and West consistently appearing in the top 10.

3c. Accident Locations (Point Map)

- with 100k sample points for plot clarity

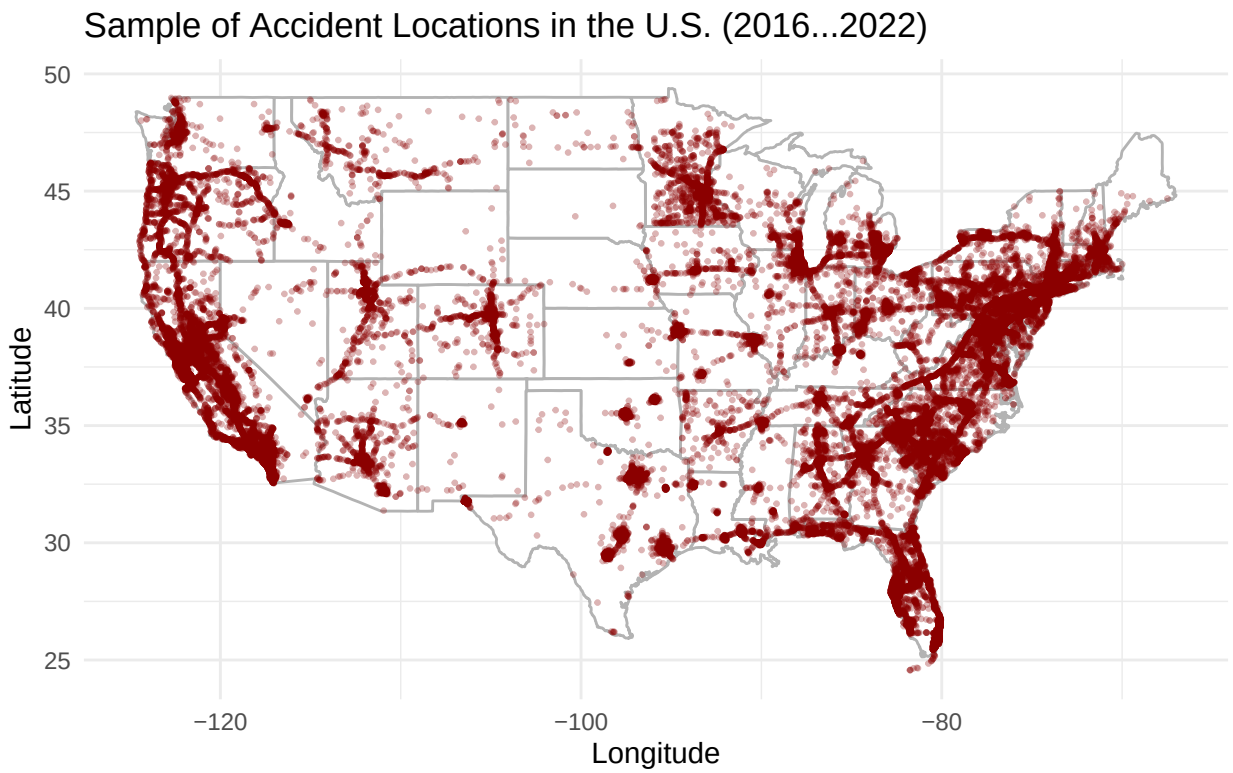
```
### Accident Locations (Point Map)
library(maps)

set.seed(123)  # reproducible sampling
sample_points <- clean_data %>%
  sample_n(100000)   # sample 100k points to keep plot clear

usa_map <- map_data("state")

ggplot() +
  geom_polygon(data = usa_map, aes(x = long, y = lat, group = group),
               fill = "white", color = "gray70") +
  geom_point(data = sample_points,
             aes(x = Start_Lng, y = Start_Lat),
             color = "darkred", alpha = 0.3, size = 0.5) +
  coord_fixed(1.3) +
  labs(
```

```
    title = "Sample of Accident Locations in the U.S. (2016-2022)",
    x = "Longitude",
    y = "Latitude"
  ) +
  theme_minimal()
```

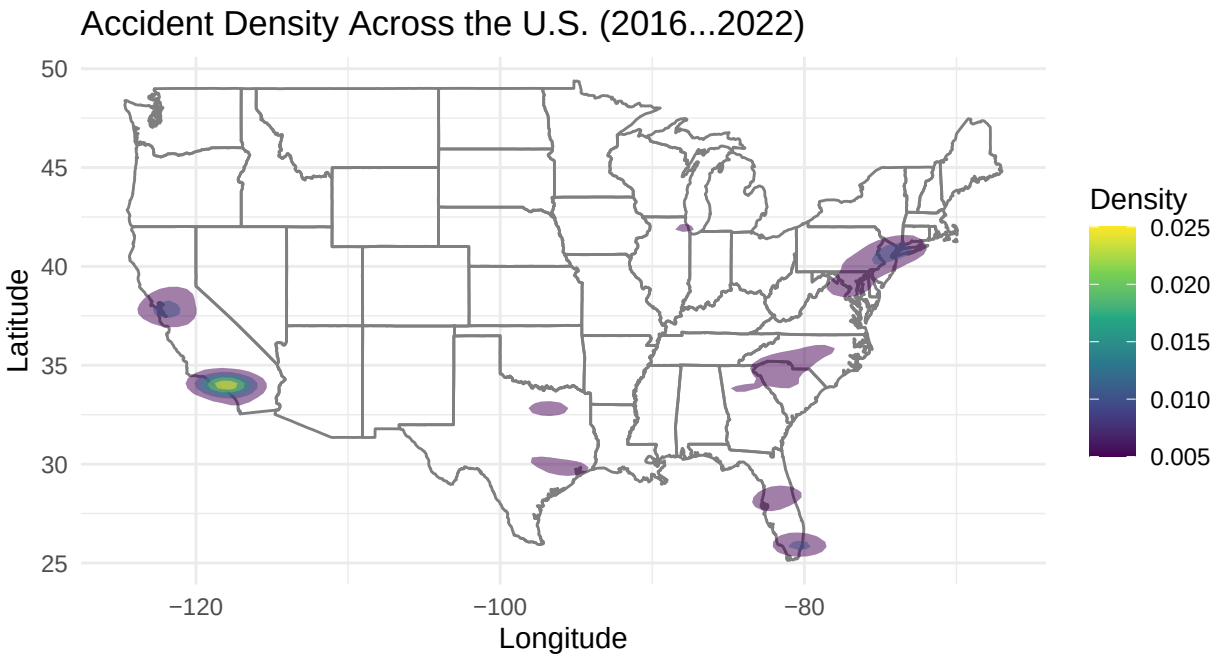


Findings

- Accident locations are concentrated along major highways and densely populated corridors, especially in California, Florida, Texas, and the East Coast.

3d. Accident Density Heatmap

```
#Accident Density Heatmap
ggplot(clean_data, aes(x = Start_Lng, y = Start_Lat)) +
  borders("state") +
  stat_density_2d(aes(fill = ..level..), geom = "polygon", alpha = 0.5) +
  scale_fill_viridis_c() +
  coord_fixed(1.3) +
  labs(
    title = "Accident Density Across the U.S. (2016-2022)",
    x = "Longitude",
    y = "Latitude",
    fill = "Density"
  ) +
  theme_minimal()
```



Findings

- Hotspots emerge in urban clusters such as Southern California, South Florida, Texas cities, and the Northeast corridor, reinforcing that population density and traffic exposure drive accident risk.

3e. Accidents by State (2016–2022)

- Map Plot (Raw Count)

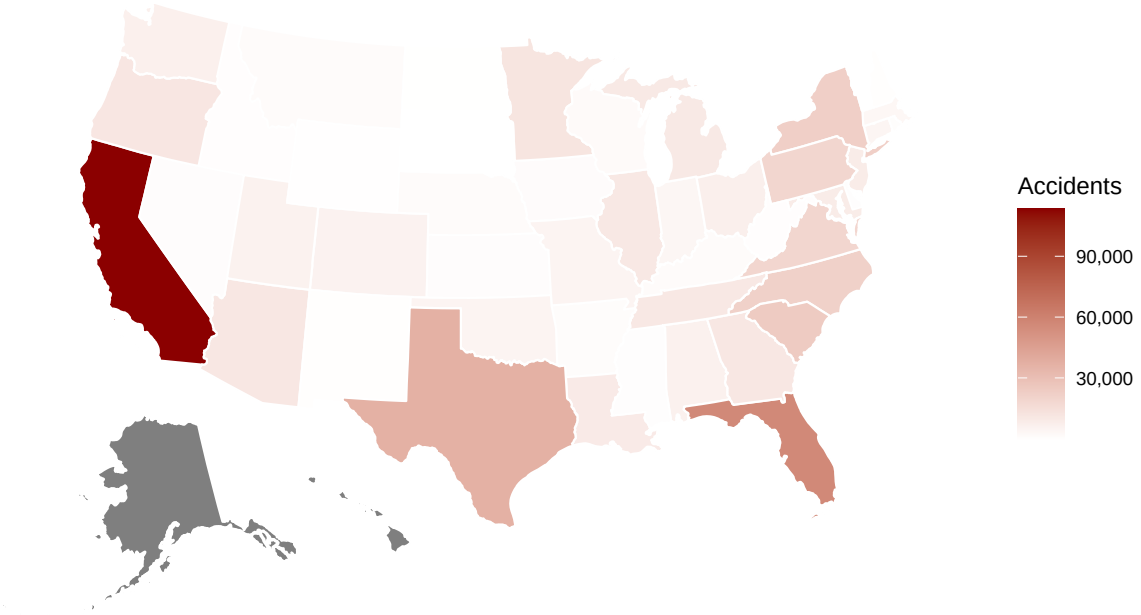
```
library(maps)
library(usmap)

# Convert state abbreviations to full names
state_counts <- clean_data %>%
  count(State) %>%
  mutate(state = state.name[match(State, state.abb)]) %>%
  filter(!is.na(state))  # drop territories like PR if present

plot_usmap(data = state_counts, values = "n", regions = "states", color = "white") +
  scale_fill_continuous(
    low = "white", high = "darkred", name = "Accidents",
    label = scales::comma
  ) +
  labs(
    title = "Accidents by State (2016-2022)",
    subtitle = "Raw Counts"
  ) +
  theme(legend.position = "right")
```

Accidents by State (2016...2022)

Raw Counts



Findings

- Overall, Accidents in the U.S. are not evenly distributed — they cluster in large, high-traffic states and metropolitan regions. California, Florida, and Texas consistently appear as leaders, while dense urban hubs like Miami, Houston, and Los Angeles show the greatest concentration of crashes. The geographic analysis highlights that urbanization and traffic volume are key drivers of accident hotspots.

3f. State-Level Accident in 2022 (Per 100,000 Residents)

```
# population data from 2024 to 2024 from www.census.gov
NST_EST2024_POPCHG2020_2024 <- read_csv("~/Desktop/US_Accidents Project/NST-EST2024-POPCHG2020-2024.csv
pop_data <- NST_EST2024_POPCHG2020_2024
head(pop_data)
```

```
## # A tibble: 6 x 37
##   SUMLEV REGION DIVISION STATE NAME            ESTIMATESBASE2020 POPESTIMATE2020
##   <chr>  <chr>  <chr>    <chr> <chr>                      <dbl>           <dbl>
## 1 010    0      0        00    United States           331515736       331577720
## 2 020    1      0        00    Northeast Regi~          57617706        57431458
## 3 030    1      1        00    New England              15122011        15057350
## 4 030    1      2        00    Middle Atlantic          42495695        42374108
## 5 020    2      0        00    Midwest Region           68998970        68984258
## 6 030    2      3        00    East North Cen~          47381362        47358568
## # i 30 more variables: POPESTIMATE2021 <dbl>, POPESTIMATE2022 <dbl>,
## #   POPESTIMATE2023 <dbl>, POPESTIMATE2024 <dbl>, NPOPCHG_2020 <dbl>,
## #   NPOPCHG_2021 <dbl>, NPOPCHG_2022 <dbl>, NPOPCHG_2023 <dbl>,
## #   NPOPCHG_2024 <dbl>, PPOPCHG_2020 <dbl>, PPOPCHG_2021 <dbl>,
## #   PPOPCHG_2022 <dbl>, PPOPCHG_2023 <dbl>, PPOPCHG_2024 <dbl>,
```

```
## #   NRANK_ESTBASE2020 <chr>, NRANK_POPEST2020 <chr>, NRANK_POPEST2021 <chr>,
## #   NRANK_POPEST2022 <chr>, NRANK_POPEST2023 <chr>, NRANK_POPEST2024 <chr>, ...
```

```
pop_data_clean <- pop_data %>%
  filter(!NAME %in% c("United States", "Northeast Region", "Midwest Region",
                      "South Region", "West Region", "New England",
                      "Middle Atlantic", "East North Central",
                      "West North Central", "South Atlantic",
                      "East South Central", "West South Central",
                      "Mountain", "Pacific", "Puerto Rico")) %>%
  select(State = NAME, Population2022 = POPESTIMATE2022)

head(pop_data_clean)
```

```
## # A tibble: 6 x 2
##   State      Population2022
##   <chr>               <dbl>
## 1 Alabama           5076181
## 2 Alaska             734442
## 3 Arizona           7377566
## 4 Arkansas          3047704
## 5 California       39142414
## 6 Colorado          5850935
```

```
# Filter accidents for 2022
accidents_2022 <- clean_data %>%
  mutate(Year = year(Start_Time)) %>%
  filter(Year == 2022)
```

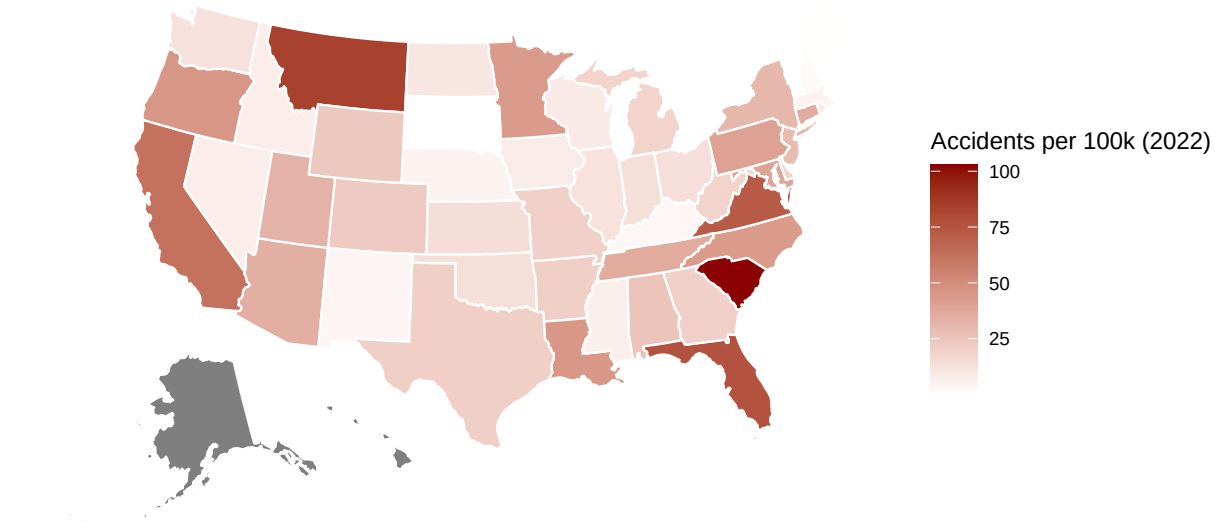
```
# joining datasets
state_rate_2022 <- accidents_2022 %>%
  count(State) %>%
  mutate(state = state.name[match(State, state.abb)]) %>%
  filter(!is.na(state)) %>%
  left_join(pop_data_clean, by = c("state" = "State")) %>%
  mutate(Accidents_per_100k = (n / Population2022) * 100000) %>%
  arrange(desc(Accidents_per_100k))   # sort by total accidents

head(state_rate_2022)
```

```
## # A tibble: 6 x 5
##   State     n state          Population2022 Accidents_per_100k
##   <chr> <int> <chr>                   <dbl>              <dbl>
## 1 SC     5448 South Carolina        5287935              103.
## 2 MT      938 Montana               1122095               83.6
## 3 FL    16946 Florida              22379312               75.7
## 4 VA     6286 Virginia              8683414               72.4
## 5 CA    24487 California           39142414               62.6
## 6 OR     1951 Oregon                4247372               45.9
```

```
library(usmap)
```

```r
plot_usmap(data = state_rate_2022, values = "Accidents_per_100k", regions = "states", color = "white") +
  scale_fill_continuous(
    low = "white", high = "darkred",
    name = "Accidents per 100k (2022)",
    label = scales::comma
  ) +
  labs(
    title = "Accidents per 100k Residents by State (2022)",
    subtitle = "Normalized by 2022 Census Population"
  ) +
  theme(legend.position = "right")
```



Findings

- When normalized by population, some smaller but high-density states (e.g., South Carolina, Montana) appear much riskier than large-population states like California or Texas. This shift shows the importance of normalizing by population: it reveals hidden hotspots where residents face disproportionately high accident risks, even if the total accident count is smaller.

Overall Findings

By normalizing accident counts with 2022 U.S. Census population estimates, this analysis highlights where residents face the highest relative risk of accidents, rather than just showing where the largest totals occur.

- **South Carolina** recorded the highest accident rate, exceeding 100 accidents per 100,000 residents, making it the most accident-prone state in relative terms.
- Other notable hotspots include Montana, Florida, and Virginia, where residents also face disproportionately high risks.
- California, which dominates in raw accident counts, drops in ranking when normalized by population, with about 63 accidents per 100k.

- Smaller states like Oregon also appear in the top tier, underscoring that accident risk is not solely a function of state size.

Key takeaway:

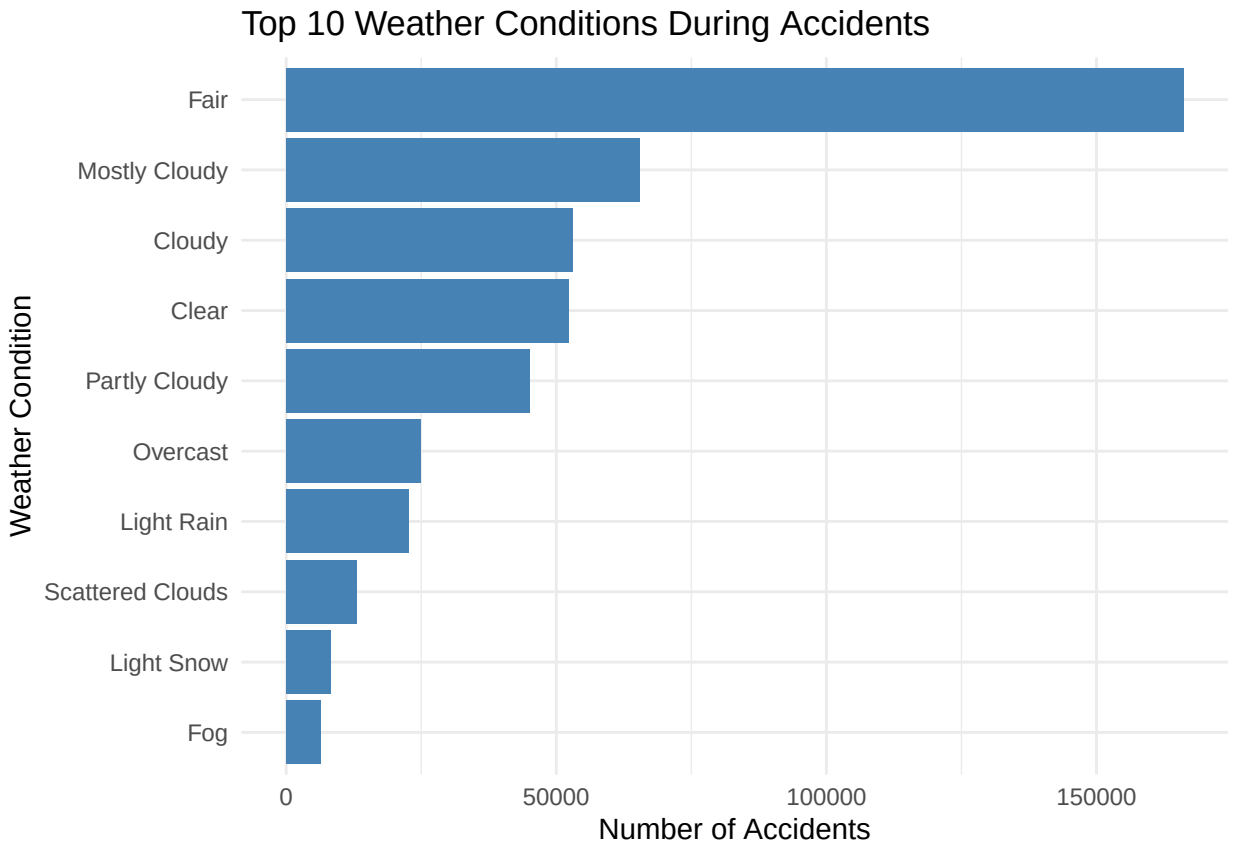
While raw totals point to large, populous states, per-capita analysis uncovers hidden hotspots in smaller states where accident rates are much higher relative to population size. This dual perspective provides a more balanced understanding of geographic risk.

4. Weather & Environmental Analysis

4a. Top Weather Conditions

```r
# Count top 10 weather conditions, ignoring NA
top_weather <- clean_data %>%
  filter(!is.na(Weather_Condition)) %>%
  count(Weather_Condition) %>%
  arrange(desc(n)) %>%
  slice(1:10)
```

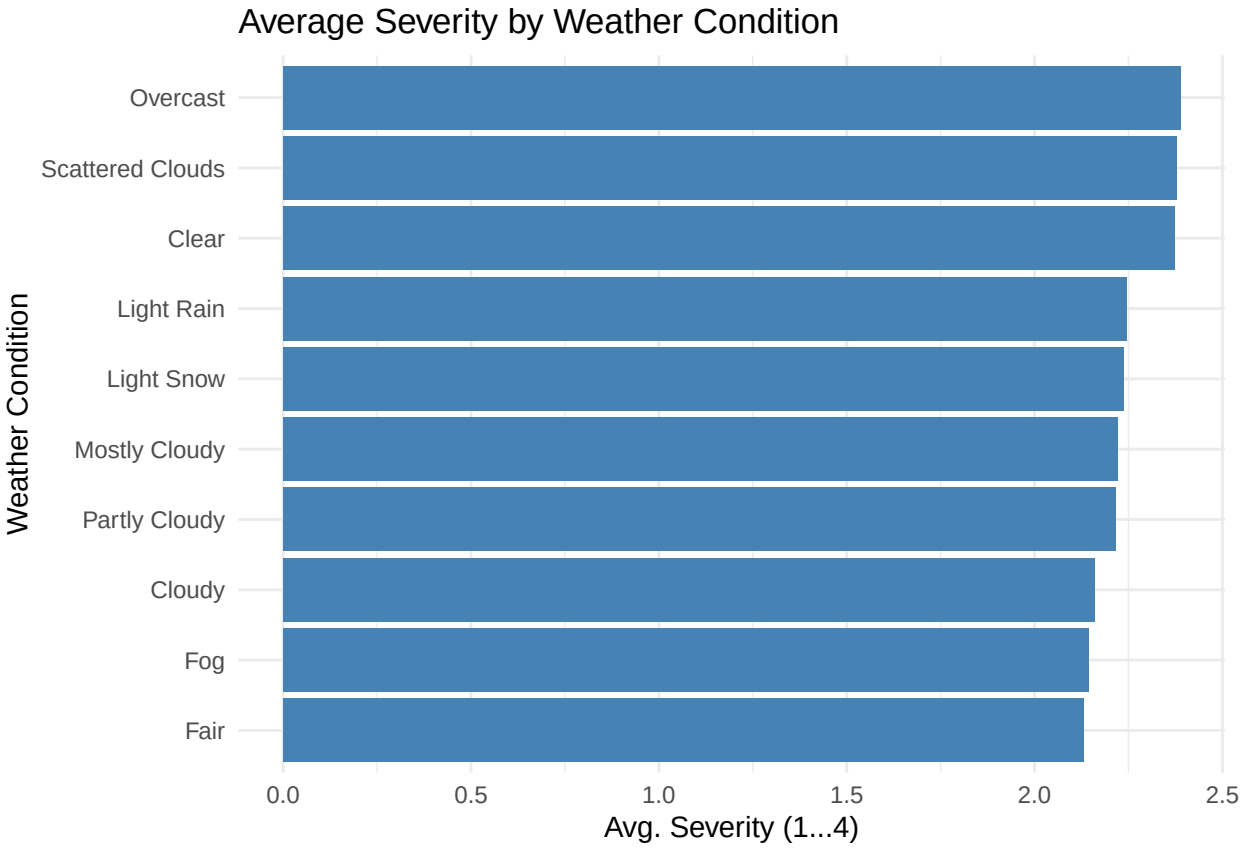
```r
# Plot
ggplot(top_weather, aes(x = reorder(Weather_Condition, n), y = n)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  labs(
    title = "Top 10 Weather Conditions During Accidents",
    x = "Weather Condition",
    y = "Number of Accidents"
  ) +
  theme_minimal()
```



4b. Severity by Weather

```r
# Average severity by top 10 weather conditions
weather_severity <- clean_data %>%
  filter(!is.na(Weather_Condition),
         Weather_Condition %in% top_weather$Weather_Condition) %>%
  group_by(Weather_Condition) %>%
  summarise(Average_Severity = mean(Severity, na.rm = TRUE)) %>%
  arrange(desc(Average_Severity))
```

```r
# Plot
ggplot(weather_severity, aes(x = reorder(Weather_Condition, Average_Severity),
                             y = Average_Severity)) +
  geom_col(fill = "steelblue") +
  coord_flip() +
  labs(
    title = "Average Severity by Weather Condition",
    x = "Weather Condition",
    y = "Avg. Severity (1-4)"
  ) +
  theme_minimal()
```



- Most accidents happen in normal weather — conditions like Fair, Mostly Cloudy, and Clear account for the majority of crashes. This makes sense because people drive more often in good weather, so exposure is higher.
- Adverse weather is less frequent but more dangerous — while categories like Light Snow, Fog, and Overcast appear less often, they are associated with higher average severity scores.
- Overcast conditions lead severity — accidents during Overcast and Scattered Clouds weather had the highest average severity (~2.3 out of 4), suggesting reduced visibility and tricky road conditions increase crash impact.
- Fair weather crashes are least severe — even though they dominate in volume, their severity scores are among the lowest (~2.1), likely due to better visibility and safer road conditions.

Findings: Weather & Environmental Features

Weather analysis shows a clear divide:

- **Good weather = more accidents, less severe.**
- **Bad weather = fewer accidents, more severe.**

5. Road & Environmental Features Analysis

5a. Accidents by Road Features (Binary Variables)

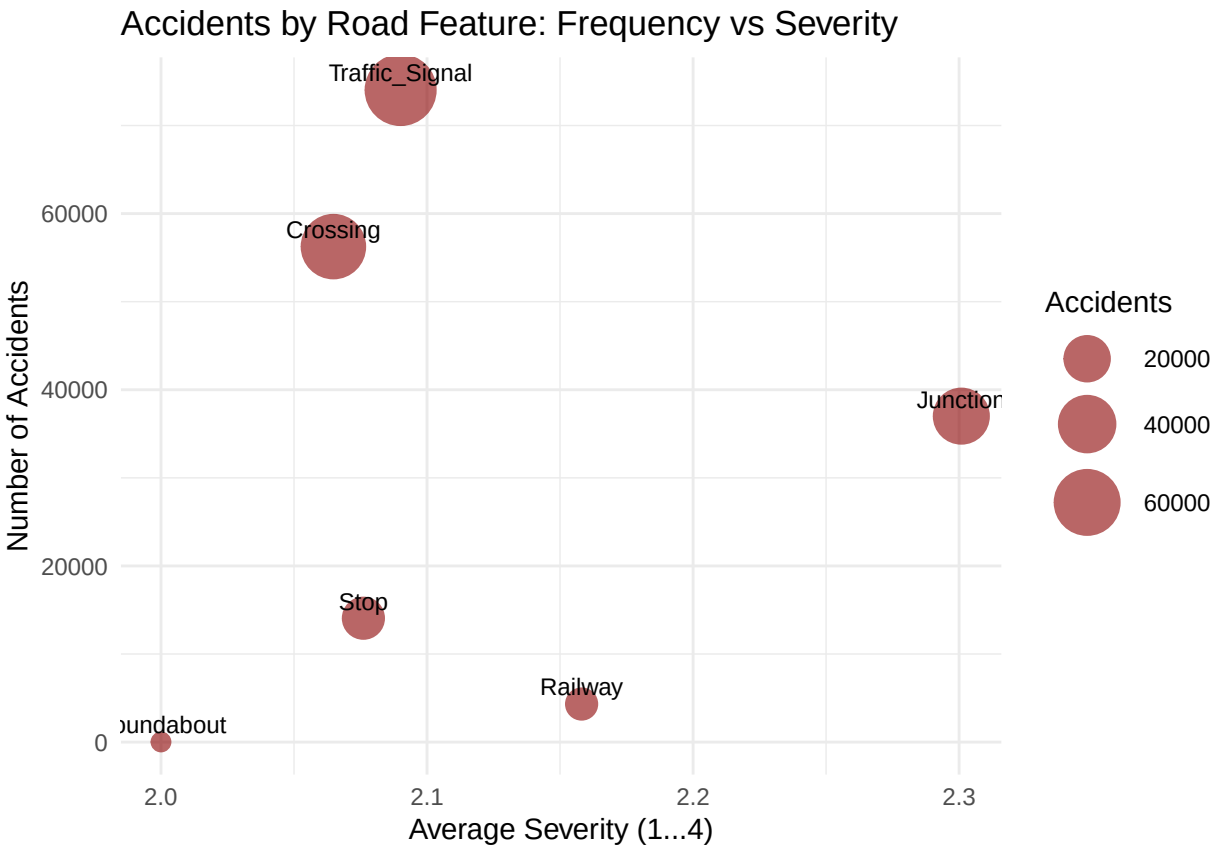
```r
road_data <- US_Accidents %>%
  select(Severity, Junction, Traffic_Signal, Stop, Railway, Crossing, Roundabout)
```

```r
# Accident counts
road_summary <- road_data %>%
  summarise(across(everything(), ~ sum(. == TRUE, na.rm = TRUE))) %>%
  pivot_longer(cols = everything(), names_to = "Feature", values_to = "Accidents")
```

```r
# Severity averages
severity_features <- road_data %>%
  pivot_longer(cols = -Severity, names_to = "Feature", values_to = "Present") %>%
  filter(Present == TRUE) %>%
  group_by(Feature) %>%
  summarise(Average_Severity = mean(Severity, na.rm = TRUE))
```

```r
# Combine into one dataset
road_compare <- road_summary %>%
  left_join(severity_features, by = "Feature") %>%
  pivot_longer(cols = c("Accidents", "Average_Severity"),
               names_to = "Metric", values_to = "Value")
```

```r
ggplot(road_compare %>%
         pivot_wider(names_from = Metric, values_from = Value),
       aes(x = Average_Severity, y = Accidents, size = Accidents, label = Feature)) +
  geom_point(color = "darkred", alpha = 0.6) +
  geom_text(vjust = -0.5, size = 3) +
  scale_size(range = c(3, 12)) +
  labs(
    title = "Accidents by Road Feature: Frequency vs Severity",
    x = "Average Severity (1-4)",
    y = "Number of Accidents"
  ) +
  theme_minimal()
```



Findings: Road & Environmental Features

- Traffic signals, crossing and junctions dominate in accident frequency, with the largest bubbles, reflecting heavy traffic flow and complex interactions.
- Junctions shows highest average severity (~2.3), suggesting crashes here tend to be more dangerous compared to signals or crossings.
- Crossings and stop signs are common in severity for accidents, reflecting lower speed zones.
- Railways has very few accidents, but when they occur, they lean toward higher severity — low frequency, high impact.
- Roundabout has very few accidents but the severity is around 2.

This highlights that accident hotspots differ by metric:

- **Volume: Traffic signals, crossing and junctions drive the most crashes.**
- **Severity: Junctions & railways pose the greatest risk per crash.**