Traffic Stop EDA

2025-02-23

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.5.1     v purrr   1.0.1
## v tibble  3.2.1     v dplyr   1.1.0
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.3     v forcats 1.0.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
traffic_data <- read_csv("Police_Department_Stop_Data_20250222.csv")
```

```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)
```

```
## Rows: 261874 Columns: 85
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (66): doj_record_id, unique_identifier, lea_record_id, stop_data_record_...
## dbl (11): person_number, duration_of_stop, perceived_age, longitude, latitud...
## num  (3): citation_cjis_off_code, perceived_race_ethnicity_code, perceived_o...
## lgl  (5): nfia_flag, is_location_k12_pub_school, k_12_school_code, education...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

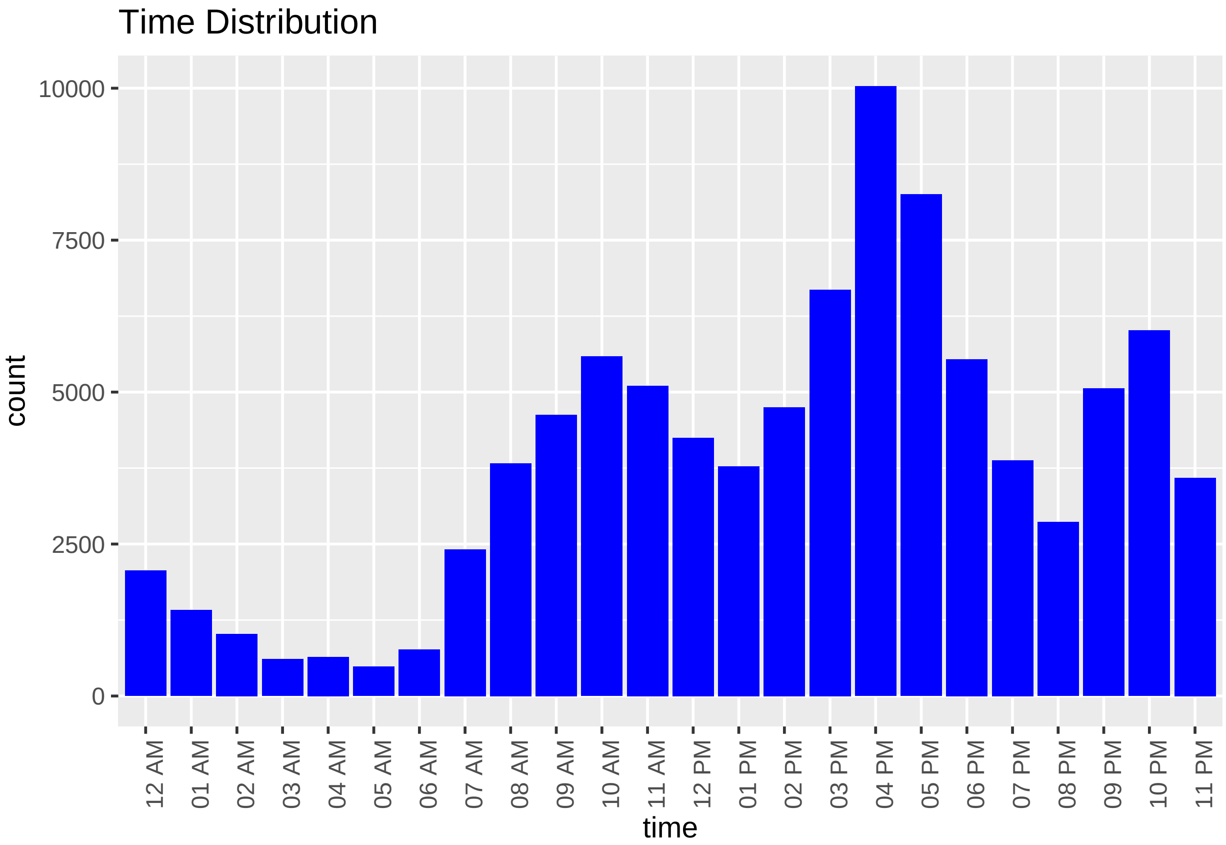
Data Filtering

```r
traffic_data <- traffic_data |>
  mutate(stop_datetime = ymd_hms(stop_datetime),
    date = as.Date(stop_datetime),
    time = format(stop_datetime, "%H:%M:%S"),
    year = year(date),
    month = month(date),
    month_year = ymd(paste(year, month, "01", sep = "-")),
    district = str_to_title(district),
    result = case_when(
      results_of_stop_code %in% c(1, 7, 9) ~ "Non-Arrest Actions",
      results_of_stop_code %in% c(2, 3, 4) ~ "Non-Custodial Actions",
      results_of_stop_code %in% c(5, 6) ~ "Arrest-Related Actions",
      results_of_stop_code %in% c(8, 10) ~ "Health-Related Actions",
      results_of_stop_code %in% c(11, 12, 13) ~ "Referral Actions",
      is.na(results_of_stop_code) ~ "Missing Data",
      TRUE ~ "Other"
    )) |>
  filter(city == "SAN FRANCISCO",
         reason_for_stop == "Traffic violation",
         stop_data_record_status == "Completed - Successful Submission",
         traffic_violation_type == "Moving violation",
         perceived_gender %in% c("Male", "Female"),
         !(district %in% c("#N/A", "Out Of Sf / Unk")) & !is.na(district),
         traffic_viol_off_type == "VC",
         !is.na(perceived_age_group))

nrow(traffic_data)
```

```
## [1] 93256
```

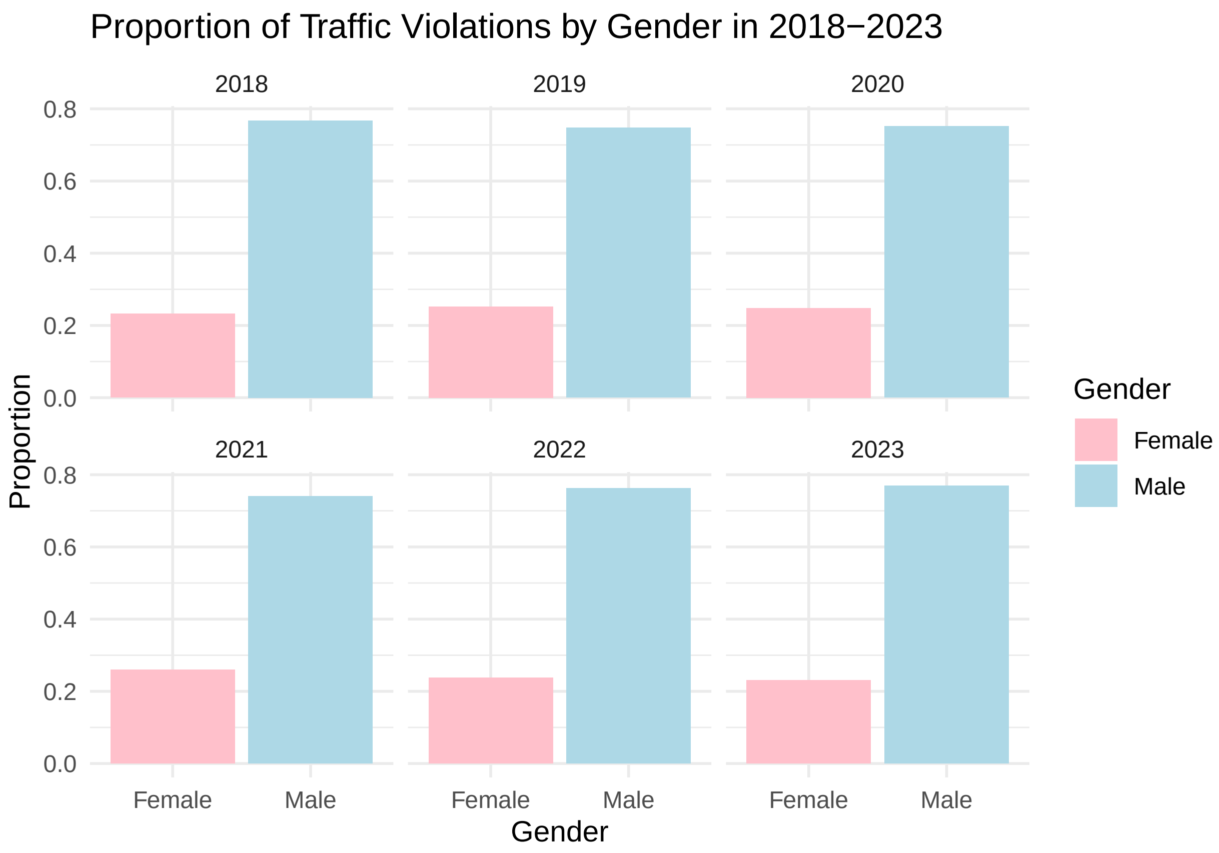
Plotting Traffic Violations by Time

```r
traffic_time <- traffic_data %>% mutate(datetime = ymd_hms(stop_datetime), date = as.Date(datetime),
                 time = hour(datetime), label = format(datetime, '%I %p')) %>%
  select(date, time, label, reason_for_stop)

hour_label <- format(seq(ymd_hms("2024-02-23 00:00:00"), by = "hour", length.out = 24), "%I %p")

#Time Distribution for all dates
ggplot(traffic_time, aes(x=factor(time, levels = 0:23,
                       labels = hour_label))) +
  geom_bar(fill='blue') + labs(title = 'Time Distribution',
                    x='time', y='count') + theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  scale_x_discrete(drop = FALSE)
```



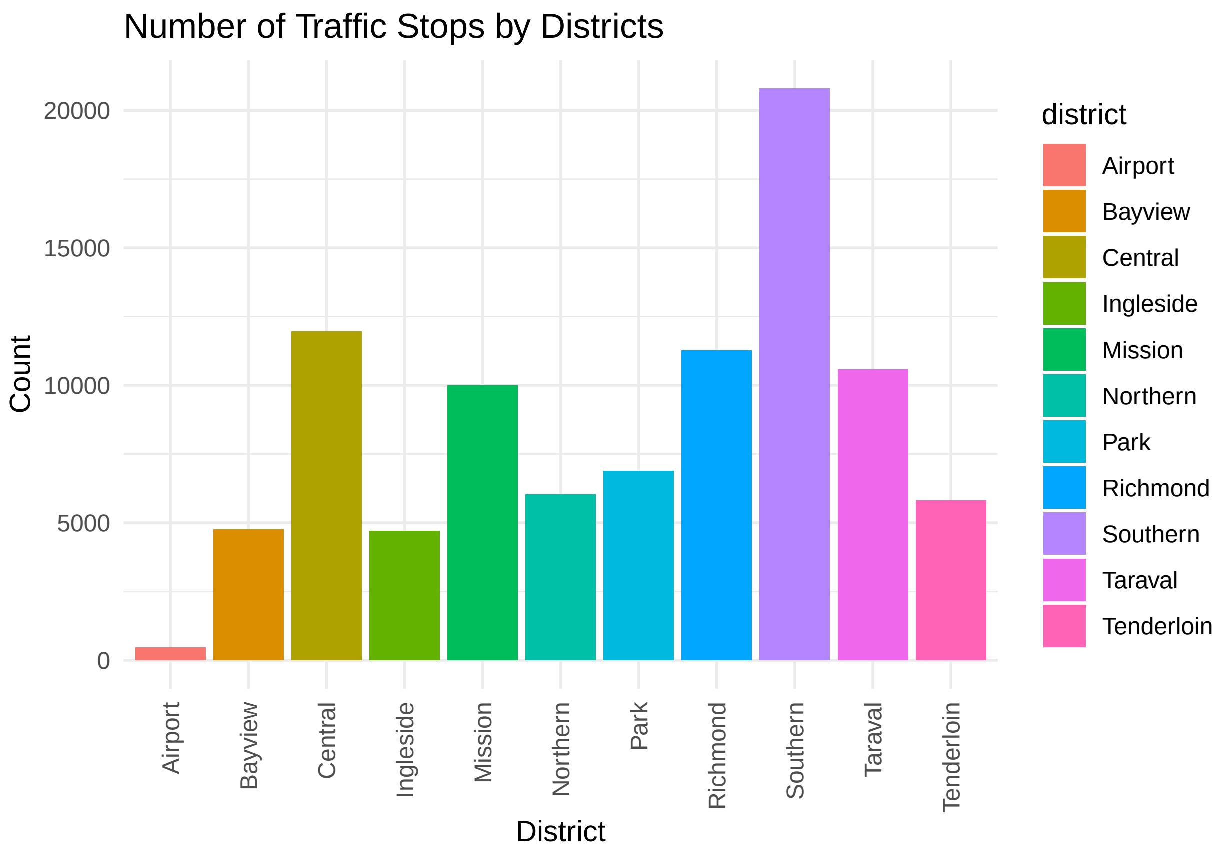
Plotting Traffic Violations by Gender

```r
traffic_data_gender <- traffic_data|> group_by(year, perceived_gender) |>
  summarize(count = n(), .groups = "drop") |>
  group_by(year) |>
  mutate(proportion = count / sum(count))

traffic_data_gender |>
  ggplot(aes(x = perceived_gender, y = proportion, fill = perceived_gender)) +
  geom_col() +
  scale_fill_manual(name = "Gender", values = c("Male" = "lightblue", "Female" = "pink")) +
  labs(x = "Gender", y = "Proportion", title = "Proportion of Traffic Violations by Gender in 2018-2023
  facet_wrap(~ year) +
  theme_minimal()
```



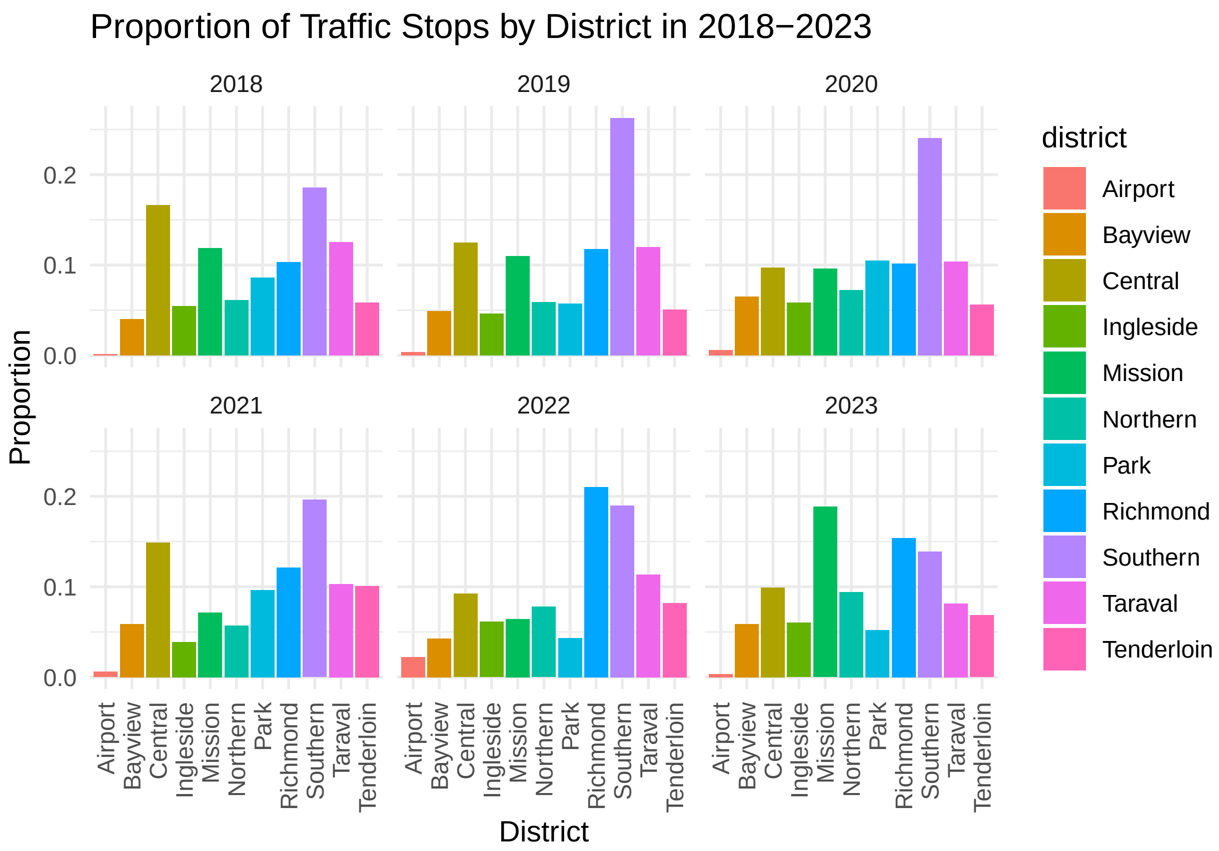
Number Traffic Stops per District from 2018-2023

```
traffic_data |>
  group_by(district) |>
  summarize(count = n()) |>
  ggplot(aes(x = district, y = count, fill = district)) +
  geom_col() +
  labs(x = "District", y = "Count", title = "Number of Traffic Stops by Districts") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



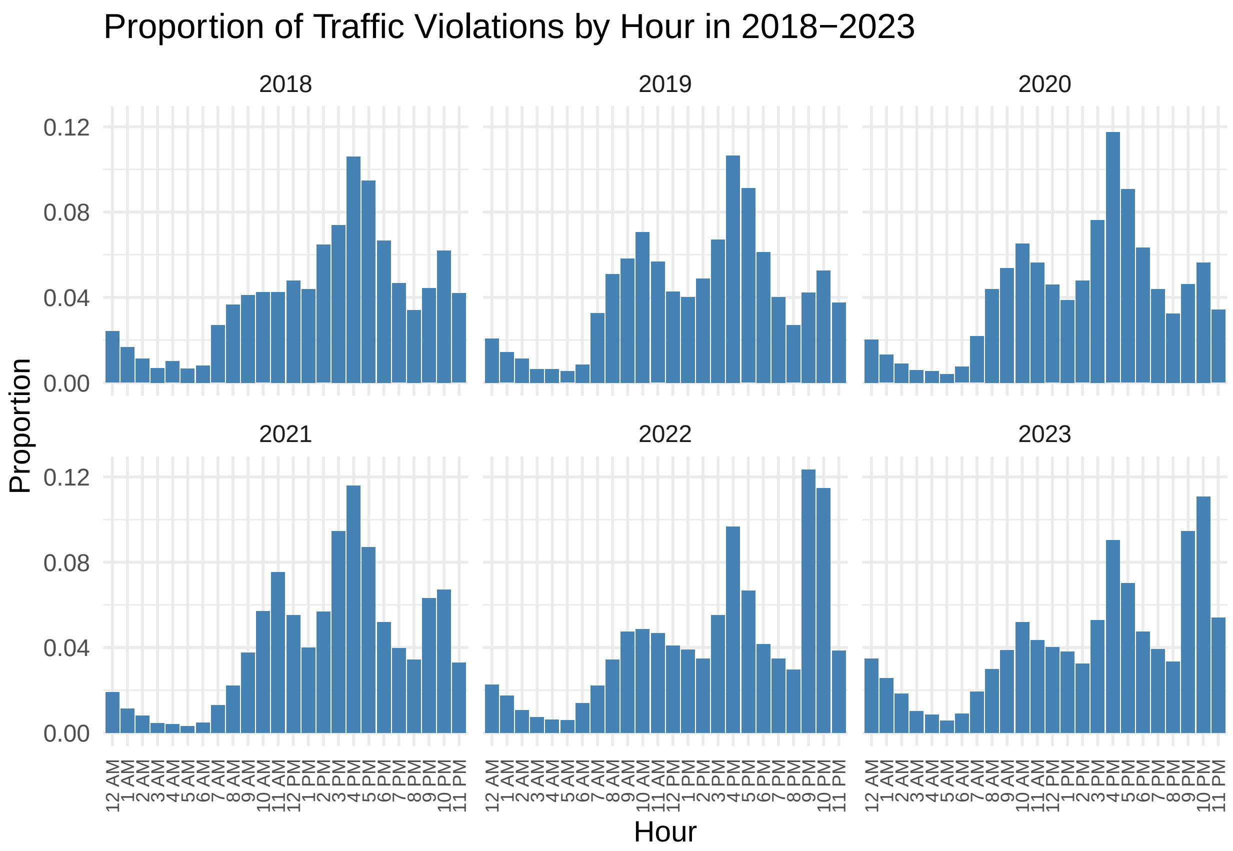
Plotting Proportion of Traffic Spots per District from 2018-2023

```
traffic_data_district <- traffic_data |>
  group_by(year, district) |>
  summarize(count = n(), .groups = "drop") |>
  group_by(year) |>
  mutate(proportion = count / sum(count))

traffic_data_district |>
  ggplot(aes(x = district, y = proportion, fill = district)) +
  geom_col() +
  labs(x = "District", y = "Proportion", title = "Proportion of Traffic Stops by District in 2018-2023"
  facet_wrap(~ year) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



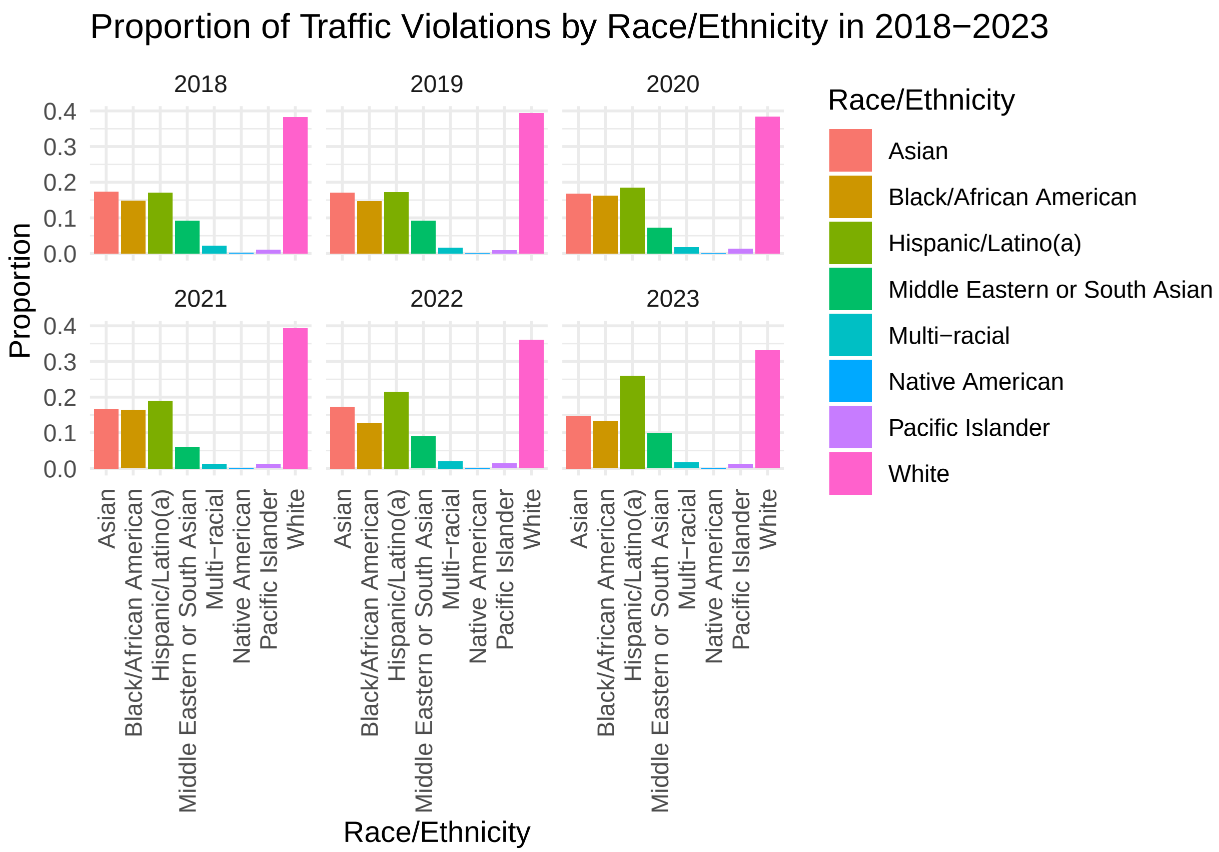
Time of Day Plots 2018-2023

```
traffic_data_time <- traffic_data |>
  mutate(hour = hour(stop_datetime),
         hour_label = case_when(
    hour == 0  ~ "12 AM",
    hour == 12 ~ "12 PM",
    hour < 12  ~ paste0(hour, " AM"),
    hour > 12  ~ paste0(hour - 12, " PM")),
    hour_label = factor(hour_label, levels = c(
           "12 AM", "1 AM", "2 AM", "3 AM", "4 AM", "5 AM", "6 AM", "7 AM",
           "8 AM", "9 AM", "10 AM", "11 AM", "12 PM", "1 PM", "2 PM", "3 PM",
           "4 PM", "5 PM", "6 PM", "7 PM", "8 PM", "9 PM", "10 PM", "11 PM"
         ))) |>
  group_by(year, hour_label) |>
  summarize(count = n(), .groups = "drop") |>
  group_by(year) |>
  mutate(proportion = count / sum(count))

traffic_data_time |>
  ggplot(aes(x = hour_label, y = proportion)) +
  geom_col(fill = "steelblue") +
  labs(x = "Hour", y = "Proportion", title = "Proportion of Traffic Violations by Hour in 2018-2023") +
  facet_wrap(~year) +
  theme_minimal() +
  theme(axis.text.x = element_text(size = 7, angle = 90, vjust = 0.5, hjust = 1))
```



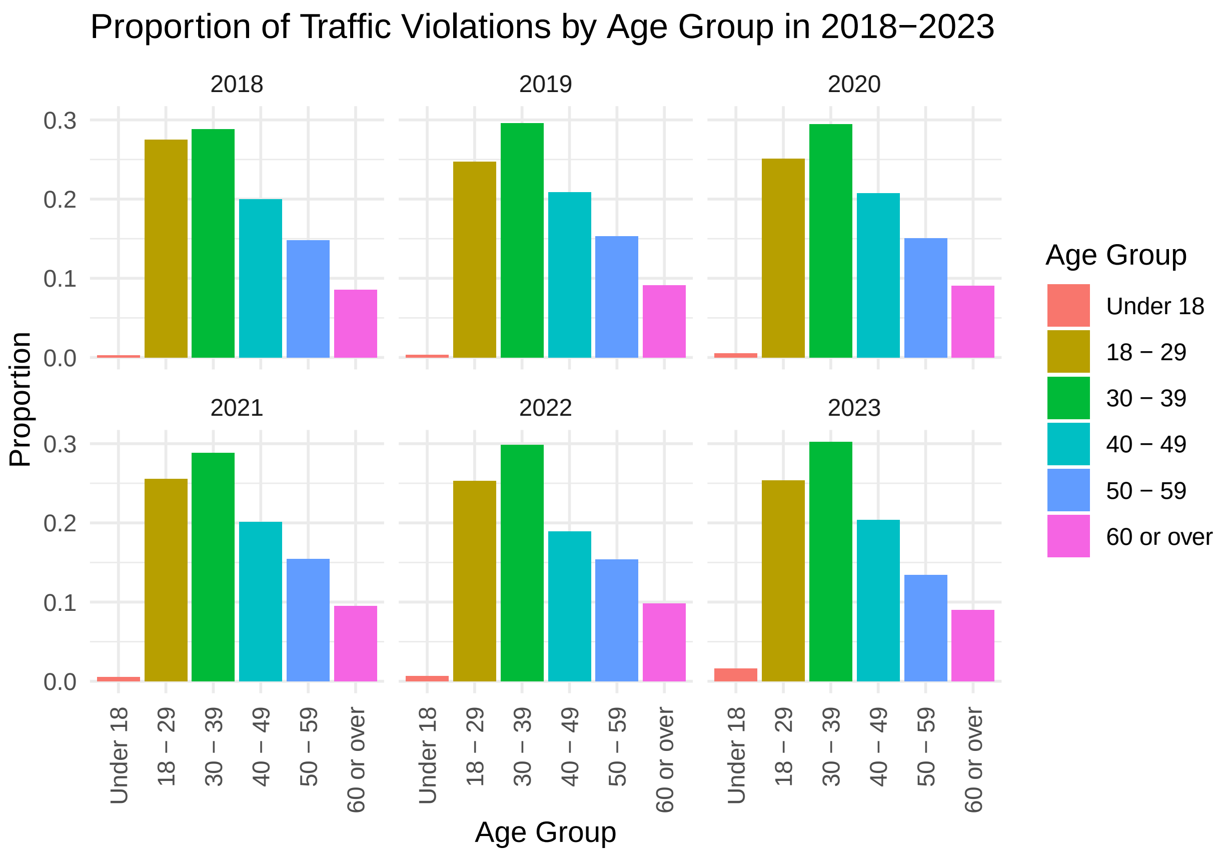
Traffic Violations by Race

```
traffic_data_race <- traffic_data |>
  group_by(year, perceived_race_ethnicity) |>
  summarize(count = n(), .groups = "drop") |>
  group_by(year) |>
  mutate(proportion = count / sum(count))

traffic_data_race |>
  ggplot(aes(x = perceived_race_ethnicity, y = proportion, fill = perceived_race_ethnicity)) +
  geom_col() +
  labs(x = "Race/Ethnicity", y = "Proportion", title = "Proportion of Traffic Violations by Race/Ethnic
  guides(fill = guide_legend(title = "Race/Ethnicity")) +
  facet_wrap(~ year) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



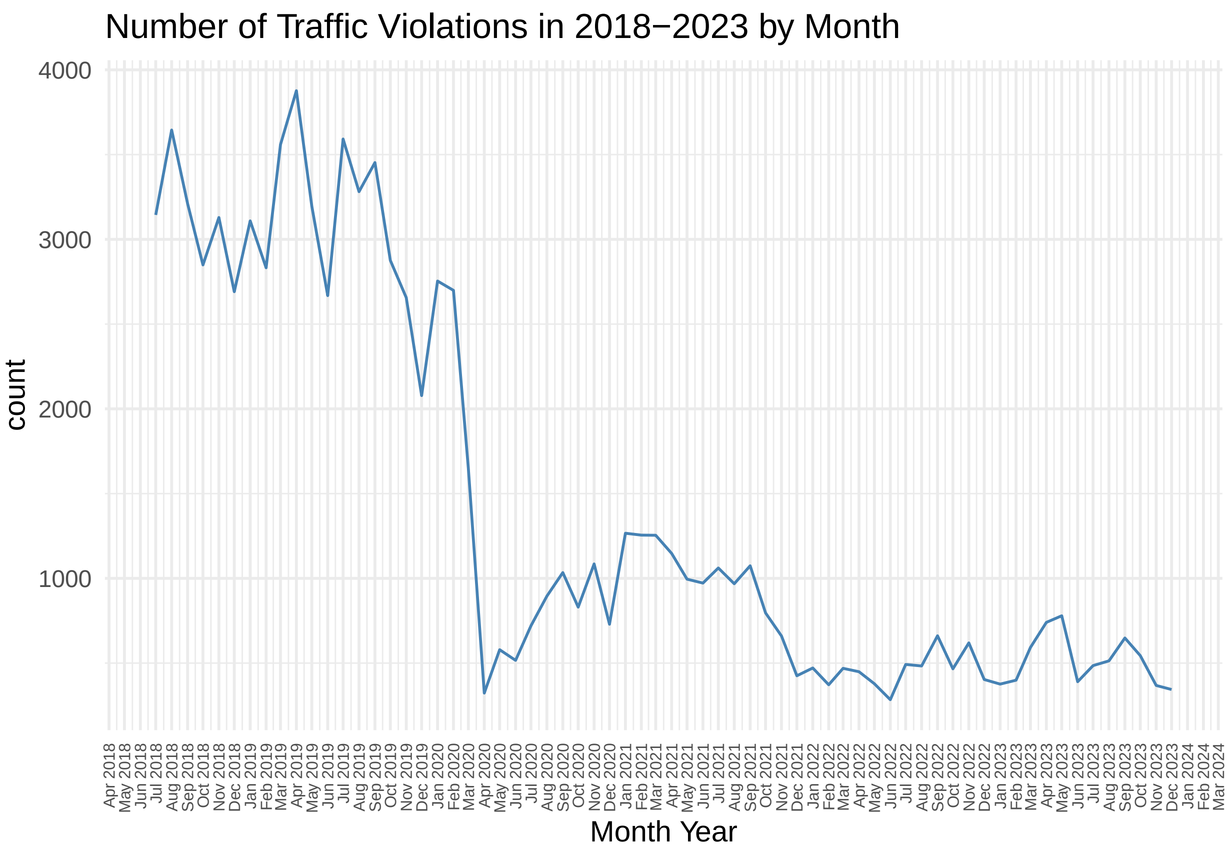
Traffic Violations by Age

```
traffic_data_age <- traffic_data |>
  mutate(perceived_age_group = factor(perceived_age_group, levels = c("Under 18",
                                      "18 - 29",
                                      "30 - 39",
                                        "40 - 49",
                                        "50 - 59",
                                        "60 or over"))) |>
  group_by(year, perceived_age_group) |>
  summarize(count = n(), .groups = "drop") |>
  group_by(year) |>
  mutate(proportion = count / sum(count))

traffic_data_age |>
  ggplot(aes(x = perceived_age_group, y = proportion, fill = perceived_age_group)) +
  geom_col() +
  labs(x = "Age Group", y = "Proportion", title = "Proportion of Traffic Violations by Age Group in 2018
  guides(fill = guide_legend(title = "Age Group")) +
  facet_wrap(~ year) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



Plots of Traffic Violations by Month

```r
traffic_monthly <- traffic_data |>
  group_by(month_year) |>
  summarize(count = n(), .groups = "drop")

traffic_monthly |>
  ggplot(aes(x = month_year,  y = count)) +
  geom_line(color = "steelblue") +
  scale_x_date(date_labels = "%b %Y",  # e.g., Jan 2023
               date_breaks = "1 month") +
  labs(title = "Number of Traffic Violations in 2018-2023 by Month", x = "Month Year") +
  theme_minimal() +
  theme(axis.text.x = element_text(size = 6, angle = 90, vjust = 0.5, hjust=1))
```



Number of Traffic Violations in 2018−2023 by Month
count
4000
3000
2000
1000
Apr 2018
May 2018
Jun 2018
Jul 2018
Aug 2018
Sep 2018
Oct 2018
Nov 2018
Dec 2018
Jan 2019
Feb 2019
Mar 2019
Apr 2019
May 2019
Jun 2019
Jul 2019
Aug 2019
Sep 2019
Oct 2019
Nov 2019
Dec 2019
Jan 2020
Feb 2020
Mar 2020
Apr 2020
May 2020
Jun 2020
Jul 2020
Aug 2020
Sep 2020
Oct 2020
Nov 2020
Dec 2020
Jan 2021
Feb 2021
Mar 2021
Apr 2021
May 2021
Jun 2021
Jul 2021
Aug 2021
Sep 2021
Oct 2021
Nov 2021
Dec 2021
Jan 2022
Feb 2022
Mar 2022
Apr 2022
May 2022
Jun 2022
Jul 2022
Aug 2022
Sep 2022
Oct 2022
Nov 2022
Dec 2022
Jan 2023
Feb 2023
Mar 2023
Apr 2023
May 2023
Jun 2023
Jul 2023
Aug 2023
Sep 2023
Oct 2023
Nov 2023
Dec 2023
Jan 2024
Feb 2024
Mar 2024
Month Year