Insert title here

Insert subtitle here

Introduction

Dataset

Our dataset is composed of harvested and compiled data from New York City Police Department (NYPD) open access data on all police reported motor vehicle collisions (MVC) in all five boroughs of New York City from July 1st, 2012 through April 24th, 2023. The police report from which individual MVC observations in our dataset hail (MV104-AN) is required to be filled out for MVC where someone is injured or killed, or which result in at least $1,000 of total property damage. Notably, only one MV104-AN form is filled out for all involved in an accident, meaning each observation in our dataset represents a unique MVC.

The following are the variables of interest from our dataset:

`crash_date`: the date on which the MVC occurred.

`crash_time`: the hour and minute (24-hr time) at which the MVC occurred.

`number_of_persons_injured`: The number of people who were injured in an individual MVC (not including people who were killed in the MVC).

`number_of_persons_killed`: The number of people who were killed in an individual MVC.

`crash_date`

Read Data/Libraries

```
library(tidyverse)
library(tidymodels)
library(lubridate)
```

```
crashes <- read_csv("data/crashes_10k.csv")
```

Methodology

- MCAR
- 2 or fewer vehicles only
- response: person casualty (injured + killed)
- removed rows with >2 vehicles
- add column to track whether or not there are injuries/fatalities 5:12 12:5 5:9 9:5

Data Wrangling

```
crashes <- crashes |>
  # remove accidents involving greater than 2 vehicles
  filter(is.na(contributing_factor_vehicle_3), is.na(vehicle_type_code_3)) |>

  select(!c(vehicle_type_code_3, vehicle_type_code_4, vehicle_type_code_5,
            contributing_factor_vehicle_3, contributing_factor_vehicle_4,
            contributing_factor_vehicle_5)) |>

  mutate(
    # add combined casualty column (injuries + fatalities)
    num_casualties = number_of_persons_killed + number_of_persons_injured,

    # create factors
    contributing_factor_vehicle_1 = as.factor(contributing_factor_vehicle_1),
    contributing_factor_vehicle_2 = as.factor(contributing_factor_vehicle_2),
    vehicle_type_code_1 = as.factor(vehicle_type_code_1),
    vehicle_type_code_2 = as.factor(vehicle_type_code_2),
    zip_code = as.factor(zip_code),
    borough = as.factor(borough),

    # add time of day categories
    time_day = case_when(
      hms(crash_time) > hm("5:00") & hms(crash_time) <= hm("12:00") ~
        "morning",
      hms(crash_time) > hm("12:00") & hms(crash_time) <= hm("17:00") ~
        "afternoon",
      hms(crash_time) > hm("17:00") & hms(crash_time) <= hm("21:00") ~
        "evening",
      hms(crash_time) > hm("21:00") | hms(crash_time) <= hm("5:00") ~
        "night"
    ),

    # add ordinal column for injuries v. fatalities
    has_injury = number_of_persons_injured > 0,
    has_fatality = number_of_persons_killed >0,
    has_casualty = has_injury | has_fatality,
    severity = case_when(
      has_fatality ~ "fatal",
      has_injury ~ "injury",
      T ~ "no casualties"
    ),
    severity = factor(severity, levels = c("no casualties", "injury", "fatal")),

    # add Julian date column
    crash_date = as.Date(crash_date, format = "%m/%d/%Y"),
    crash_day = weekdays(crash_date, abbreviate = F),
    yday = yday(crash_date)
  )
```

```
ped_bike <- "Pedestrian/Bicyclist/Other Pedestrian Error/Confusion"
crashes <- crashes %>%
  mutate(across(starts_with("contributing_factor_vehicle"),
                ~ case_when(
                  . %in% c("Alcohol Involvement", "Drugs (Illegal)",
                           "Prescription Medication") ~ "Impaired",
                  . %in% c("Driver Inattention/Distraction", "Fatigued/Drowsy",
                           "Lost Consciousness", "Other Electronic Device",
                           "Outside Car Distraction", "Passenger Distraction")
                           ~ "Distraction/Inattention/Fatigue",
                  . %in% c("Following Too Closely", "Passing Too Closely",
                           "Unsafe Lane Changing",
                           "Unsafe Speed",
                           "Backing Unsafely") ~ "Aggressive/Reckless Driving",
                  . %in% c("Failure to Yield Right-of-Way",
                           ped_bike,
                           "Reaction to Other Uninvolved Vehicle",
                           "Reaction to Uninvolved Vehicle",
                           "Turning Improperly")
                           ~ "Failure to Obey Traffic Signs/Signals/Rules",
                  . %in% c("Lane Marking Improper/Inadequate",
                           "Obstruction/Debris", "Other Vehicular",
                           "Oversized Vehicle", "Pavement Defective",
                           "Pavement Slippery",
                           "Traffic Control Device Improper/Non-Working",
                           "Traffic Control Disregarded",
                           "View Obstructed/Limited")
                           ~ "Other Technical/Mechanical Factors",
                  TRUE ~ "Other/Unknown"
                )
  )) |>
  rename(factor1 = contributing_factor_vehicle_1,
         factor2 = contributing_factor_vehicle_2)

crashes <- crashes %>%
  mutate(across(starts_with("vehicle_type_code"),
                ~ case_when(
                  . %in% c("2 dr sedan", "3-Door", "4 dr sedan", "4dsd",
                        "Convertible", "Sedan", "SEDONA",
                        "Station Wagon/Sport Utility Vehicle",
                        "SPORT UTILITY / STATION WAGON", "LIMO",
                        "LIVERY VEHICLE") ~ "Passenger vehicles",
                  . %in% c("AMBU", "AMBUL", "Ambulance", "AMBULANCE",
                        "AMBULETTE", "Armored Truck", "Beverage Truck",
                        "Box Truck", "Bulk Agriculture", "Bus", "BUS",
                        "Carry All", "Chassis Cab", "DELIV", "DELV",
                        "Dump", "FDNY", "Fire", "FIRE TRUCK", "FIRETRUCK",
                        "Flat Bed", "Flat Rack", "Garbage or Refuse",
                        "Hopper", "SMALL COM VEH(4 TIRES)",
                        "LARGE COM VEH(6 OR MORE TIRES)",
                        "Mack Truck",
                        "PICK-UP TRUCK", "PK", "Refrigerated Van", "schoo",
                        "SCHOO", "Snow Plow", "Tow Truck",
                        "Tow Truck / Wrecker", "TRACT",
                        "Tractor Truck Diesel", "Tractor Truck Gasoline",
                        "TRAIL", "TRAILER", "TRALI", "TRUCK",
                        "USPS", "UTIL", "Van", "VAN",
                        "VAN TRUCK") ~ "Commercial vehicles",
                  . %in% c("MOTOR SCOO", "Motorcycle",
                        "MOTORCYCLE", "Motorscooter", "SCOOTER", "Moped") ~
                    "Motorcycles",
                  . %in% c("E-Bike", "BICYCLE", "Bike", "E-Scooter") ~
                    "Non-Motor Vehicle",
                  . %in% c("FORK", "unk", "UNKNOWN", "OTHER") ~ "Other/Unknown",
                  TRUE ~ "Other/Unknown"
          ))) |>
      rename(vtype1 = vehicle_type_code_1, vtype2 = vehicle_type_code_2)
```

```
class(crashes$contributing_factor_vehicle_1)
```

```
[1] "NULL"
```

```
levels(crashes$contributing_factor_vehicle_1)
```

```
NULL
```

```
levels(crashes$contributing_factor_vehicle_2)
```

```
NULL
```

```
crashes %>%
  filter(is.na(borough)) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1  2823
```

```
crashes %>%
  filter(is.na(borough)) %>%
  filter(!is.na(latitude)) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1  1920
```

```
crashes %>%
  filter(number_of_persons_killed != 0) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1     8
```

```
crashes %>%
  filter(num_casualties > 1) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1   374
```

```
crashes %>%
  filter(num_casualties > 2) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1   112
```

```
crashes %>%
  filter(num_casualties > 3) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1    37
```

```r
crashes %>%
  filter(num_casualties > 4) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1    15
```

```r
crashes %>%
  filter(num_casualties > 5) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1     8
```

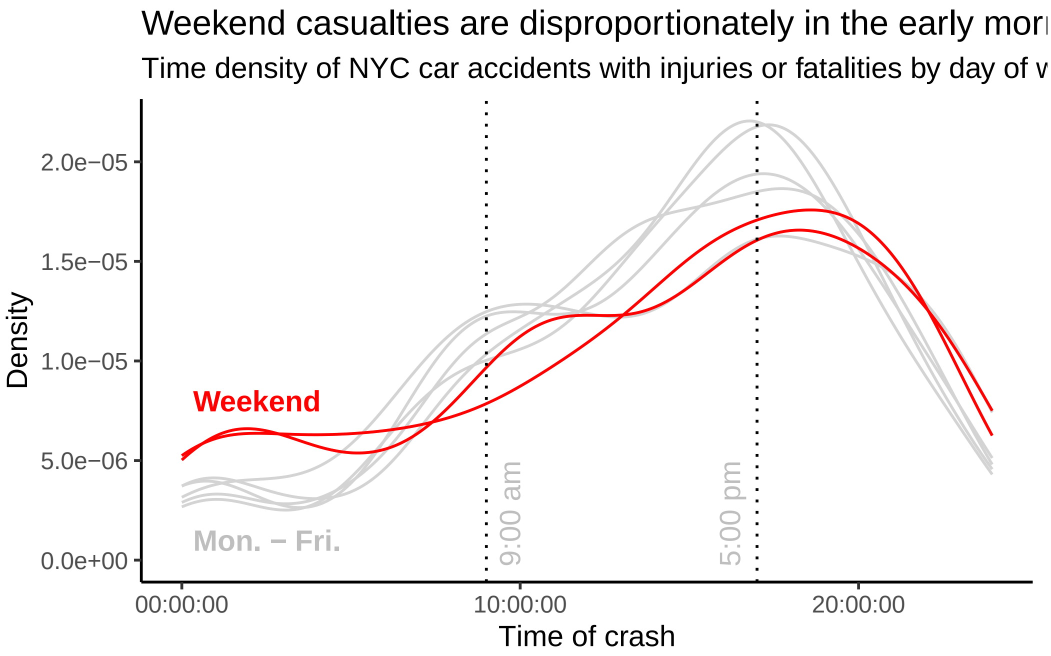
```r
crashes %>%
  filter(num_casualties > 6) %>%
  count()
```

```
# A tibble: 1 x 1
      n
  <int>
1     4
```

Exploratory Analysis

Time of Day

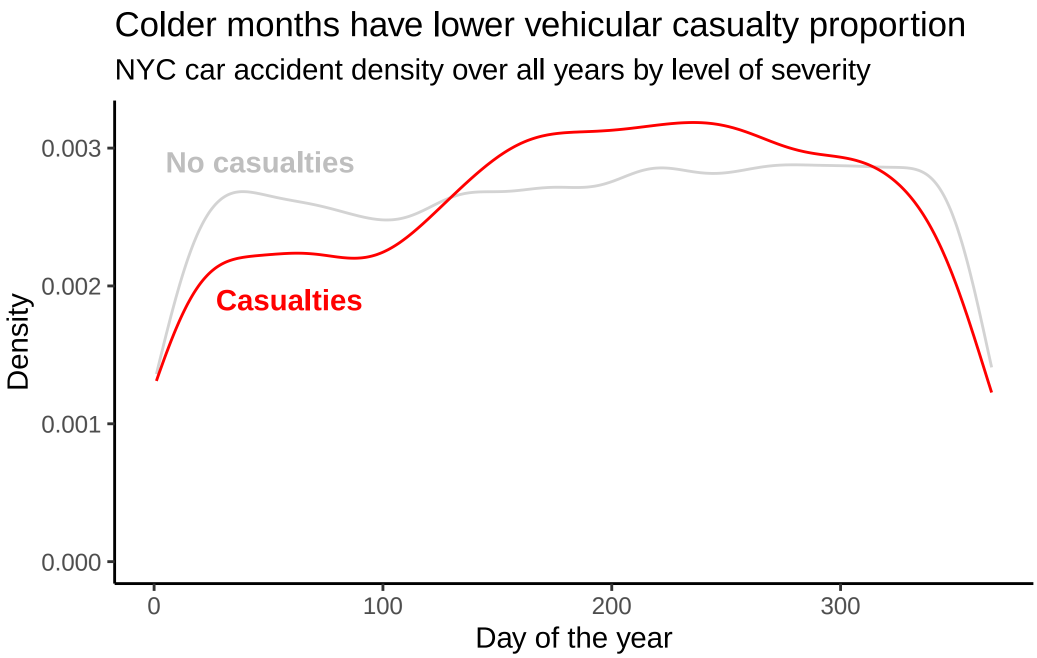
```r
crashes |>
  filter(num_casualties > 0) |>
  mutate(color = if_else(crash_day == "Sunday" | crash_day == "Saturday",
         "red", "gray")) |>
  ggplot(aes(x = crash_time, group = crash_day)) +
  geom_density(color = "lightgray") +
  geom_density(
    data = filter(crashes, num_casualties > 0,
                  crash_day == "Sunday" | crash_day == "Saturday"),
    color = "red"
  ) +
  geom_vline(xintercept = hm("9:00"), linetype = 3) +
  geom_vline(xintercept = hm("17:00"), linetype = 3) +
  annotate("text", label = "9:00 am", color = "gray", angle = 90,
           x = hm("10:00"), y = 5e-06, hjust = "right", vjust = "bottom") +
  annotate("text", label = "5:00 pm", color = "gray", angle = 90,
           x = hm("16:30"), y = 5e-06, hjust = "right", vjust = "bottom") +
  annotate("text", color = "red", label = "Weekend", x = hm("00:20"),
           hjust = "left", fontface = "bold", y = 8e-06) +
  annotate("text", color = "gray", label = "Mon. - Fri.", x = hm("00:20"),
           hjust = "left", fontface = "bold", y = 1e-06) +
  scale_color_identity() +
  theme_classic() +
  labs(
    x = "Time of crash",
    y = "Density",
    title = "Weekend casualties are disproportionately in the early morning",
    subtitle = paste("Time density of NYC car accidents with injuries or",
                     "fatalities by day of week"),
    color = "Weekday"
  )
```



Of accidents with one or more casualties, those happening on Saturdays and Sundays appear to occur disproportionately late at night and with a notably smaller peak around 5:00 pm. While there is variance around the weekend, the general pattern stays the same between all seven days. To account for this, we will introduce a random effect based on the days of the week to allow information sharing between them. We anticipate that our model's performance will benefit from such a change, and this random effect will end up in the final model after performance evaluation.

Day of the year

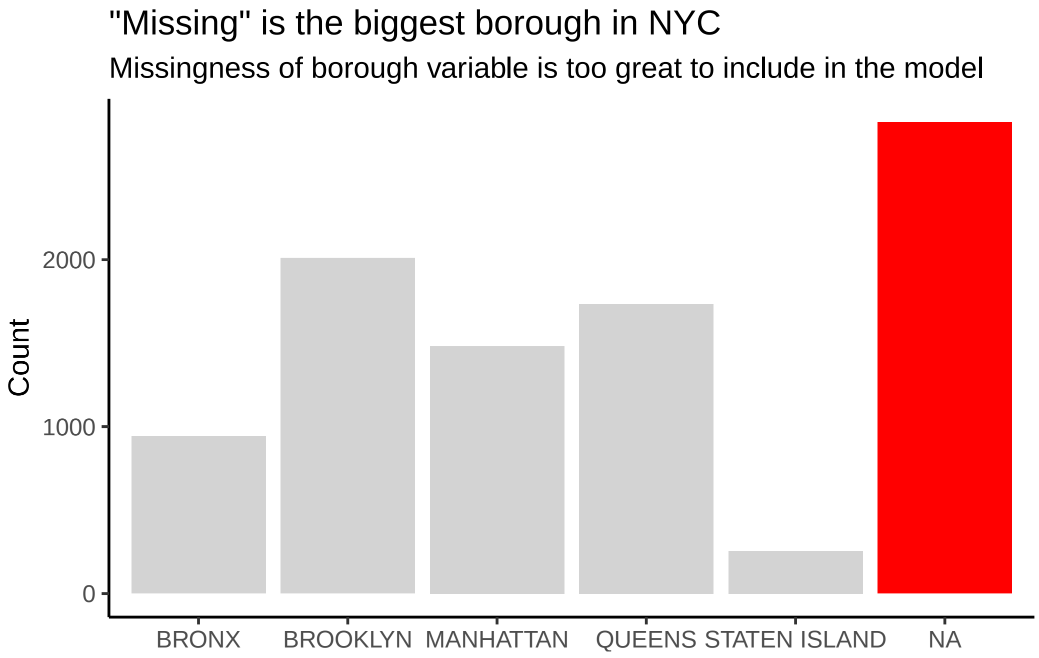
```
crashes |>
  ggplot(aes(x = yday)) +
  geom_density(data = filter(crashes, severity == "no casualties"),
               aes(group = severity), color = "lightgray") +
  geom_density(data = filter(crashes, severity == "injury"), color = "red") +
  annotate("text", color = "red", label = "Casualties", x = 27, y = 0.0019,
           fontface = "bold", hjust = "left") +
  annotate("text", color = "gray", label = "No casualties", x = 5,
           y = 0.0029, fontface = "bold", hjust = "left") +
  theme_classic() +
  labs(
    x = "Day of the year",
    y = "Density",
    title = "Colder months have lower vehicular casualty proportion",
    subtitle = "NYC car accident density over all years by level of severity"
  )
```



The variance in density between levels of severity of accidents through all years is relatively low. However, we see disproportionately severe accidents around the late summer/early spring months and minor accidents in the early winter months. We will include the day of the year in our initial model, but it may get cut during model evaluation.

Borough

```
crashes |>
  group_by(borough) |>
  summarise(count = n()) |>
  mutate(color = if_else(is.na(borough), "red", "lightgray")) |>
  ggplot(aes(x = borough, y = count, fill = color)) +
  geom_col() +
  scale_fill_identity() +
  theme_classic() +
  labs(
    x = NULL,
    y = "Count",
    title = "\"Missing\" is the biggest borough in NYC",
    subtitle = paste("Missingness of borough variable is too great to include",
                     "in the model")
  )
```



We have far too many missing values to include borough in our model. If we could verify that borough was Missing Completely At Random (MCAR), we could include it in our complete-case model. Still, unfortunately, we are unable to verify the missingness mechanism and are therefore uncomfortable with possible performance impacts by using borough in our final model.

Model Construction

```
model1 <- glm(has_casualty ~ yday + time_day + factor1 + factor2 +
                vtype1 + vtype2,
              data = crashes, family = "binomial")
print(tidy(model1), n = 23)
```

```
# A tibble: 23 x 5
   term                                        estimate std.e~1 stati~2  p.value
   <chr>                                          <dbl>   <dbl>   <dbl>    <dbl>
 1 (Intercept)                                 -3.22e+0 2.72e-1 -11.9   1.98e-32
 2 yday                                         8.71e-5 2.62e-4   0.333 7.39e- 1
 3 time_dayevening                              2.43e-1 7.44e-2   3.26  1.10e- 3
 4 time_daymorning                             -1.21e-1 7.43e-2  -1.63  1.02e- 1
 5 time_daynight                                3.72e-1 7.94e-2   4.68  2.80e- 6
 6 factor1Distraction/Inattention/Fatigue       2.66e-1 9.47e-2   2.80  5.06e- 3
 7 factor1Failure to Obey Traffic Signs/Signa~  6.47e-1 1.08e-1   5.99  2.05e- 9
 8 factor1Impaired                              4.38e-1 2.15e-1   2.04  4.13e- 2
 9 factor1Other Technical/Mechanical Factors    1.08e-1 1.36e-1   0.794 4.27e- 1
10 factor1Other/Unknown                        -6.45e-2 9.04e-2  -0.714 4.76e- 1
11 factor2Distraction/Inattention/Fatigue       5.11e-2 2.37e-1   0.215 8.29e- 1
12 factor2Failure to Obey Traffic Signs/Signa~  3.09e-1 2.88e-1   1.07  2.83e- 1
13 factor2Impaired                              1.21e-1 7.49e-1   0.162 8.71e- 1
14 factor2Other Technical/Mechanical Factors    1.56e-1 2.84e-1   0.549 5.83e- 1
15 factor2Other/Unknown                         2.92e-1 2.03e-1   1.44  1.50e- 1
16 vtype1Motorcycles                            2.72e+0 2.74e-1   9.91  3.61e-23
17 vtype1Non-Motor Vehicle                      3.35e+0 3.24e-1  10.3   4.59e-25
18 vtype1Other/Unknown                          4.54e-1 1.31e-1   3.47  5.23e- 4
19 vtype1Passenger vehicles                     6.40e-1 1.25e-1   5.13  2.83e- 7
20 vtype2Motorcycles                            3.06e+0 3.19e-1   9.59  8.40e-22
21 vtype2Non-Motor Vehicle                      3.59e+0 2.04e-1  17.6   5.31e-69
22 vtype2Other/Unknown                          8.60e-1 1.35e-1   6.37  1.84e-10
23 vtype2Passenger vehicles                     4.88e-1 1.36e-1   3.59  3.34e- 4
# ... with abbreviated variable names 1: std.error, 2: statistic
```

Results