Analysis

Resid(COVID) + Joe Choo, Glen Morgenstern, Carrie Wang, and Zhixue (Mary) Wang

Research Question

The coronavirus has hit perhaps no American city quite as hard as New York, where more than 10,000 residents have died due to it. Yet, New Yorkers have heard conflicting advice from Governer Andrew Cuomo and President Donald Trump. Our team decided to look at how the stay-at-home order and other advice from government officials affected how New Yorkers drive. With reduced traffic, are accidents down? Has the lockdown affected New York's six boroughs differently?

Motivating question: **How has government guidance affected New York drivers' motivations and safety?**

Data

The data comes from the City of New York's website. This data set consists of police reports on all motor vehicle collisions in New York City as of April 18. A police report is warranted when there is a fatality or injury or damage of $1000 or more. Our analysis will focus only on collisions in 2020.

You can find the updated data set at: https://data.cityofnewyork.us/Public-Safety/Motor-Vehicle-Collisions-Crashes/h9gi-nx95?fbclid=IwAR1UoErhqzmJrviRZ4zkpYu7cHhOgCAA417o6A3rhrIZSKrXPPNVjdzsWUQ

Set Up

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.2.1 --
```

```
## v tibble  1.4.2     v purrr   0.2.5
## v tidyr   0.8.1     v dplyr   0.7.6
## v readr   1.1.1     v forcats 0.3.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x readr::guess_encoding()  masks rvest::guess_encoding()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x purrr::pluck()           masks rvest::pluck()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

```
library(lubridate)
```

```
data <- read.csv("data/Motor_Vehicle_Collisions_-_Crashes.csv")
```

Dates to look at: Jan 20 - First reported US case Feb 9 - Death toll in China surpasses SARS epidemic (811 deaths) Feb 29 - First death on American soil March 1 - First confirmed case in NYC March 7 - Cuomo declares state of emergency March 11 - WHO declares pandemic, Trump bans travel from 26 European countries March 13 - Trump declares national emergency March 14 - First coronavirus death in NYC March

16 - NYC schools close (ordered by Cuomo) March 17 - De Blasio says NYC considering order, Cuomo says it won't happen; schools, bar, restaurants close in NYC (de Blasio) March 20 - Cuomo orders lockdown March 22 - Lockdown starts (8pm) March 28 - Trump signs $2.2 trillion stimulus bill April 3 - CDC recommends wearing masks April 6 - Cuomo extends stay at home order to April 29 April 14 - Cuomo says he would defy Trump order to reopen New York April 16 - Cuomo extends stay at home order to May 15 April 17 - Trump tweets "Liberate" Minnesota, Michigan, Virginia

Data Wrangling

Make all street names upper case.

```
data$ON.STREET.NAME = toupper(data$ON.STREET.NAME)
data$OFF.STREET.NAME = toupper(data$OFF.STREET.NAME)
data$CROSS.STREET.NAME = toupper(data$CROSS.STREET.NAME)
```

Confirm no duplicate collision IDs.

```
data[duplicated(data$COLLISION_ID), ]
```

```
##  [1] X                              CRASH.DATE
##  [3] CRASH.TIME                     BOROUGH
##  [5] ZIP.CODE                       LATITUDE
##  [7] LONGITUDE                      LOCATION
##  [9] ON.STREET.NAME                 CROSS.STREET.NAME
## [11] OFF.STREET.NAME                NUMBER.OF.PERSONS.INJURED
## [13] NUMBER.OF.PERSONS.KILLED       NUMBER.OF.PEDESTRIANS.INJURED
## [15] NUMBER.OF.PEDESTRIANS.KILLED   NUMBER.OF.CYCLIST.INJURED
## [17] NUMBER.OF.CYCLIST.KILLED       NUMBER.OF.MOTORIST.INJURED
## [19] NUMBER.OF.MOTORIST.KILLED      CONTRIBUTING.FACTOR.VEHICLE.1
## [21] CONTRIBUTING.FACTOR.VEHICLE.2  CONTRIBUTING.FACTOR.VEHICLE.3
## [23] CONTRIBUTING.FACTOR.VEHICLE.4  CONTRIBUTING.FACTOR.VEHICLE.5
## [25] COLLISION_ID                   VEHICLE.TYPE.CODE.1
## [27] VEHICLE.TYPE.CODE.2            VEHICLE.TYPE.CODE.3
## [29] VEHICLE.TYPE.CODE.4            VEHICLE.TYPE.CODE.5
## <0 rows> (or 0-length row.names)
```

No duplicates—all good.

```
data <- data %>%
  select(-X)
```

```
data %>%
   count(CONTRIBUTING.FACTOR.VEHICLE.1)
```

```
## # A tibble: 56 x 2
##    CONTRIBUTING.FACTOR.VEHICLE.1      n
##    <fct>                          <int>
##  1 ""                               160
##  2 Accelerator Defective             22
##  3 Aggressive Driving/Road Rage     226
##  4 Alcohol Involvement              486
##  5 Animals Action                    45
##  6 Backing Unsafely                1723
##  7 Brakes Defective                 152
##  8 Cell Phone (hand-Held)            16
##  9 Cell Phone (hands-free)            1
```

```
## 10 Driver Inattention/Distraction 10710
## # ... with 46 more rows
```

Regroup so that accerlator defective, brakes defective, windshield inadequate, tire failure, tow hitch defective, headlights defective, and other lighting defects are in one group called mechanical defect; using on board navigation device, texting, cell phone (hand-held), cell phone (hands-free), and other electronic device are under distraction; listening is under distracted; fatigued/drowsy, fell asleep, and lost consciousness are under drowsy; pavement defective, shoulders defective, and pavement slippery under environment.

```
data <- data %>%
  mutate(CONTRIBUTING.FACTOR.VEHICLE.1 = as.character(CONTRIBUTING.FACTOR.VEHICLE.1)) %>%
   mutate(CONTRIBUTING.FACTOR.VEHICLE.1 = case_when(
     CONTRIBUTING.FACTOR.VEHICLE.1 == ""  ~ "Unspecified",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Glare"  ~ "View Obstructed/Limited",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Tire Failure/Inadequate"  ~ "Mechanical Defect",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Windshield Inadequate"  ~ "Mechanical Defect",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Tow Hitch Defective"  ~ "Mechanical Defect",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Accelerator Defective"  ~ "Mechanical Defect",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Brakes Defective"  ~ "Mechanical Defect",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Headlights Defective"  ~ "Mechanical Defect",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Other Lighting Defects"  ~ "Mechanical Defect",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Cell Phone (hand-Held)"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Listening/Using Headphones"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Other Electronic Device"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Using On Board Navigation Device"  ~ "Driver Inattention/Distrac
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Cell Phone (hands-free)"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Outside Car Distraction"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Passenger Distraction"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Texting"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Eating or Drinking"  ~ "Driver Inattention/Distraction",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Fatigued/Drowsy"  ~ "Tired",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Fell Asleep"  ~ "Tired",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Lost Consciousness"  ~ "Tired",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Lane Marking Improper/Inadequate"  ~ "Road Environment",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Obstruction/Debris"  ~ "Road Environment",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Pavement Defective"  ~ "Road Environment",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Pavement Slippery"  ~ "Road Environment",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Shoulders Defective/Improper"  ~ "Road Environment",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Prescription Medication"  ~ "Health",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Physical Disability"  ~ "Health",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Illnes"  ~ "Health",
     CONTRIBUTING.FACTOR.VEHICLE.1 == "Traffic Control Device Improper/Non-Working"  ~ "Road Environmen
    CONTRIBUTING.FACTOR.VEHICLE.1 == "Tinted Windows"  ~ "View Obstructed/Limited",
     TRUE ~ CONTRIBUTING.FACTOR.VEHICLE.1
   ))
```

```
data %>%
   count(CONTRIBUTING.FACTOR.VEHICLE.1)
```

```
## # A tibble: 29 x 2
##    CONTRIBUTING.FACTOR.VEHICLE.1      n
##    <chr>                          <int>
##  1 Aggressive Driving/Road Rage     226
##  2 Alcohol Involvement              486
##  3 Animals Action                    45
##  4 Backing Unsafely                1723
```

```
##  5 Driver Inattention/Distraction 10915
##  6 Driver Inexperience              618
##  7 Driverless/Runaway Vehicle        33
##  8 Drugs (illegal)                   21
##  9 Failure to Keep Right             48
## 10 Failure to Yield Right-of-Way   2876
## # ... with 19 more rows
```

```
data %>%
   count(VEHICLE.TYPE.CODE.1)
```

```
## # A tibble: 141 x 2
##    VEHICLE.TYPE.CODE.1     n
##    <fct>               <int>
##  1 ""                    324
##  2 2 dr sedan              8
##  3 3-Door                  5
##  4 4 dr sedan            117
##  5 AMB                     3
##  6 AMBU                    1
##  7 AMBUL                  38
##  8 Amb                     1
##  9 Ambul                   7
## 10 Ambulance             144
## # ... with 131 more rows
```

Regroup so EMERG, AMB, AMBU, AMBUL, amb, ambul are under ambulance; BOX T is under Box Truck; DELIV, DELV, deliv, devli, Deliv are under delivery; DUMP, dump is under dump; FIRE, Fire, Firet, fire, and FIRET and under Firetruck; FLATB is under Flat Bed; FORK and FORKL is under Forlift; PICKU, pick, and Pickup with mounted is under Pick-up Truck; SCHOO, schoo, Schoo under School Bus; all wheeler sednas under Sedan; Tow, tow truck, tow t, under tow truck; UTIL, UTILI, Utili under utility

```
data <- data %>%
  mutate(VEHICLE.TYPE.CODE.1 = as.character(VEHICLE.TYPE.CODE.1)) %>%
   mutate(VEHICLE.TYPE.CODE.1 = case_when(
     VEHICLE.TYPE.CODE.1 == "van" ~ "Van",
     VEHICLE.TYPE.CODE.1 == "UTIL" ~ "Utility",
     VEHICLE.TYPE.CODE.1 == "UTILI" ~ "Utility",
     VEHICLE.TYPE.CODE.1 == "Utili" ~ "Utility",
     VEHICLE.TYPE.CODE.1 == "" ~ "Unknown",
     VEHICLE.TYPE.CODE.1 == "TOW T" ~ "Tow Truck",
     VEHICLE.TYPE.CODE.1 == "Tow" ~ "Tow Truck",
     VEHICLE.TYPE.CODE.1 == "Tow t" ~ "Tow Truck",
     VEHICLE.TYPE.CODE.1 == "Tow Truck / Wrecker" ~ "Tow Truck",
     VEHICLE.TYPE.CODE.1 == "TRK" ~ "Truck",
     VEHICLE.TYPE.CODE.1 == "TRUCK" ~ "Truck",
     VEHICLE.TYPE.CODE.1 == "2 dr sedan" ~ "Sedan",
     VEHICLE.TYPE.CODE.1 == "3-Door" ~ "Sedan",
     VEHICLE.TYPE.CODE.1 == "4 dr sedan" ~ "Sedan",
     VEHICLE.TYPE.CODE.1 == "REFG" ~ "Refrigerated Van",
     VEHICLE.TYPE.CODE.1 == "pick" ~ "Pick-up Truck",
     VEHICLE.TYPE.CODE.1 == "PICKU" ~ "Pick-up Truck",
     VEHICLE.TYPE.CODE.1 == "Pickup with mounted Camper" ~ "Pick-up Truck",
     VEHICLE.TYPE.CODE.1 == "FORKL" ~ "Forklift",
     VEHICLE.TYPE.CODE.1 == "FORK" ~ "Forklift",
     VEHICLE.TYPE.CODE.1 == "FLATB" ~ "Flat Bed",
```

```
    VEHICLE.TYPE.CODE.1 == "TRAC" ~ "Tractor Truck",
    VEHICLE.TYPE.CODE.1 == "TRACT" ~ "Tractor Truck",
    VEHICLE.TYPE.CODE.1 == "Tract" ~ "Tractor Truck",
    VEHICLE.TYPE.CODE.1 == "tract" ~ "Tractor Truck",
    VEHICLE.TYPE.CODE.1 == "Tractor Truck Diesel" ~ "Tractor Truck",
    VEHICLE.TYPE.CODE.1 == "Tractor Truck Gasoline" ~ "Tractor Truck",
    VEHICLE.TYPE.CODE.1 == "SCHOO" ~ "School Bus",
    VEHICLE.TYPE.CODE.1 == "schoo" ~ "School Bus",
    VEHICLE.TYPE.CODE.1 == "Schoo" ~ "School Bus",
    VEHICLE.TYPE.CODE.1 == "Fire" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "Firet" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "FIRE" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "FIRET" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "fire" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "FIRE" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "EMERG" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "AMB" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "AMBU" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "AMBUL" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "ambul" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "Amb" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "Ambul" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "Ambulance" ~ "Ambulance",
    VEHICLE.TYPE.CODE.1 == "deliv" ~ "Delivery",
    VEHICLE.TYPE.CODE.1 == "delvi" ~ "Delivery",
    VEHICLE.TYPE.CODE.1 == "DELIV" ~ "Delivery",
    VEHICLE.TYPE.CODE.1 == "DELV" ~ "Delivery",
    VEHICLE.TYPE.CODE.1 == "Deliv" ~ "Delivery",
    VEHICLE.TYPE.CODE.1 == "DUMP" ~ "Dump",
    VEHICLE.TYPE.CODE.1 == "dump" ~ "Dump",
    VEHICLE.TYPE.CODE.1 == "BOX T" ~ "Box Truck",
    VEHICLE.TYPE.CODE.1 == "Bobca" ~ "Station Wagon/Sport Utility Vehicle",
    VEHICLE.TYPE.CODE.1 == "Armored Truck" ~ "Truck",
    VEHICLE.TYPE.CODE.1 == "Beverage Truck" ~ "Truck",
    VEHICLE.TYPE.CODE.1 == "truck" ~ "Truck",
    VEHICLE.TYPE.CODE.1 == "COM" ~ "Commercial",
    VEHICLE.TYPE.CODE.1 == "Comme" ~ "Commercial",
    VEHICLE.TYPE.CODE.1 == "FDNY" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "Fdny" ~ "Firetruck",
    VEHICLE.TYPE.CODE.1 == "ford" ~ "Ford",
    VEHICLE.TYPE.CODE.1 == "DIRT" ~ "Station Wagon/Sport Utility Vehicle",
    VEHICLE.TYPE.CODE.1 == "Unkno" ~ "Unknown",
    VEHICLE.TYPE.CODE.1 == "Van Camper" ~ "Van",
    VEHICLE.TYPE.CODE.1 == "moped" ~ "Moped",
    VEHICLE.TYPE.CODE.1 == "GOLF" ~ "Carry All",
    VEHICLE.TYPE.CODE.1 == "H1" ~ "Station Wagon/Sport Utility Vehicle",
    VEHICLE.TYPE.CODE.1 == "Hears" ~ "Hearse",
    VEHICLE.TYPE.CODE.1 == "Hopper" ~ "Freight",
    VEHICLE.TYPE.CODE.1 == "MTA" ~ "City",
    VEHICLE.TYPE.CODE.1 == "OMR" ~ "Omnibus",
    VEHICLE.TYPE.CODE.1 == "OMS" ~ "Omnibus",
    VEHICLE.TYPE.CODE.1 == "SCOOT" ~ "Motorscooter",
    VEHICLE.TYPE.CODE.1 == "TR-Tr" ~ "Trailer",
```

```
    VEHICLE.TYPE.CODE.1 == "TRAIL" ~ "Trailer",
    VEHICLE.TYPE.CODE.1 == "TRL" ~ "Trailer",
    VEHICLE.TYPE.CODE.1 == "Trail" ~ "Trailer",
    VEHICLE.TYPE.CODE.1 == "UNK" ~ "Unknown",
    VEHICLE.TYPE.CODE.1 == "US PO" ~ "USPS",
    VEHICLE.TYPE.CODE.1 == "posta" ~ "USPS",
    VEHICLE.TYPE.CODE.1 == "POSTA" ~ "USPS",
    VEHICLE.TYPE.CODE.1 == "MAIL" ~ "USPS",
    VEHICLE.TYPE.CODE.1 == "FREIG" ~ "Freight",
    TRUE ~ VEHICLE.TYPE.CODE.1
  ))
```

```
data %>%
  count(VEHICLE.TYPE.CODE.1, sort = TRUE)
```

```
## # A tibble: 72 x 2
##    VEHICLE.TYPE.CODE.1                     n
##    <chr>                               <int>
##  1 Sedan                               18037
##  2 Station Wagon/Sport Utility Vehicle 15734
##  3 Taxi                                 1870
##  4 Pick-up Truck                        1163
##  5 Box Truck                             782
##  6 Bus                                   675
##  7 Tractor Truck                         361
##  8 Unknown                               326
##  9 Van                                   225
## 10 Bike                                  222
## # ... with 62 more rows
```

```
data <- data %>%
  mutate(weekday = wday(mdy(CRASH.DATE), label = TRUE)) %>%
  separate(CRASH.DATE, c("month", "day", "year"), "/")
data <- data %>%
  mutate(day = as.numeric(day)) %>%
  mutate(month = as.numeric(month))
data <- data %>%
  mutate(daytot = case_when(
    month == "1" ~ day,
    month == "2" ~ day + 31,
    month == "3" ~ day + 60,
    TRUE ~ day + 91
  ))
data <- data %>%
  mutate(VEHICLE.TYPE.CODE.1new = fct_lump(VEHICLE.TYPE.CODE.1, n = 14, other_level = "Other"))

#New indicator variable saying pre- or post-lockdown:

data <- data %>%
  mutate(lockdown=case_when(daytot>81 ~ 1,
                            daytot<=81 ~ 0))
```

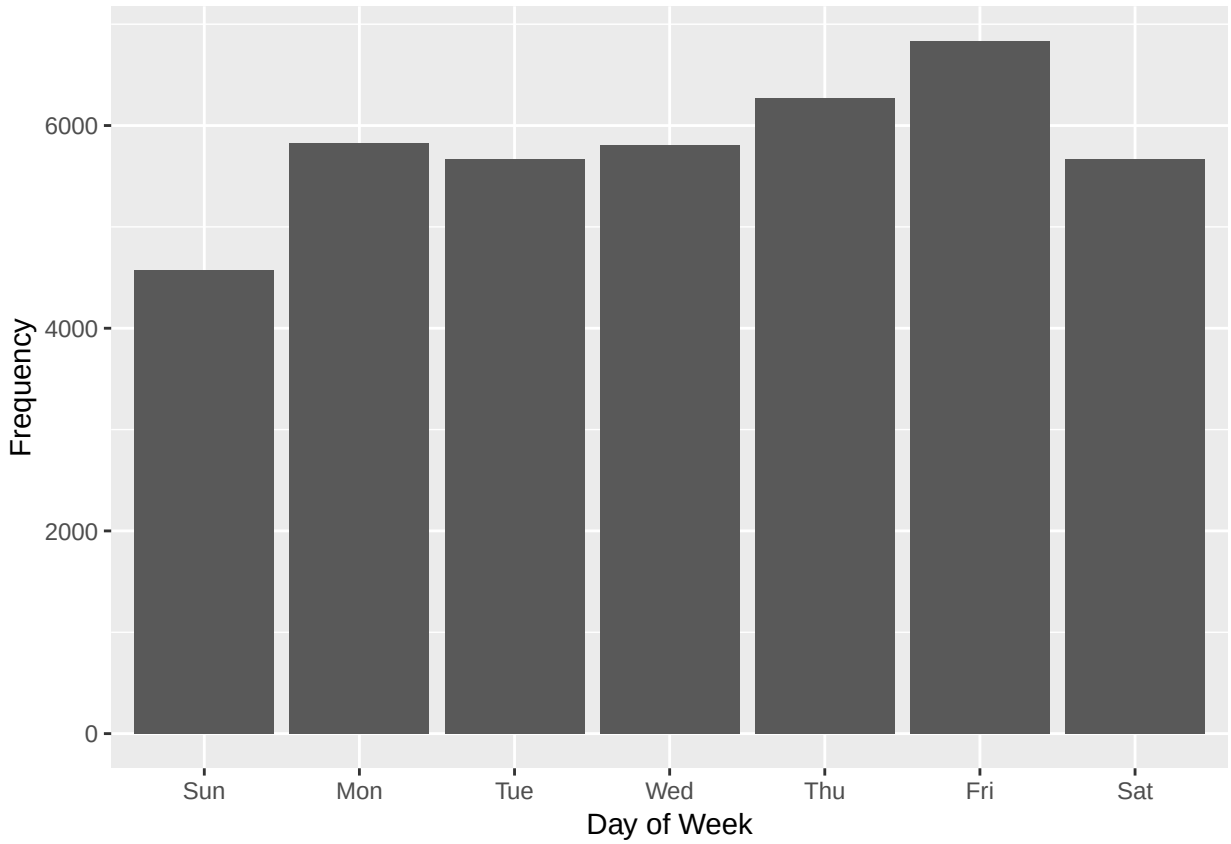
```
data %>%
  count(VEHICLE.TYPE.CODE.1new, sort = TRUE)
```

```
## # A tibble: 15 x 2
##    VEHICLE.TYPE.CODE.1new                 n
##    <fct>                              <int>
##  1 Sedan                              18037
##  2 Station Wagon/Sport Utility Vehicle 15734
##  3 Taxi                                1870
##  4 Pick-up Truck                       1163
##  5 Box Truck                            782
##  6 Other                                696
##  7 Bus                                  675
##  8 Tractor Truck                        361
##  9 Unknown                              326
## 10 Van                                  225
## 11 Bike                                 222
## 12 Ambulance                            201
## 13 Dump                                 121
## 14 Motorcycle                           113
## 15 Convertible                          109
```

```
data <- data %>%
   mutate(time_of_day = case_when(
      hm(CRASH.TIME) < hm("12:00") ~ "AM",
      TRUE ~ "PM"
   )) %>%
   mutate(person_tot = NUMBER.OF.PERSONS.INJURED + NUMBER.OF.PERSONS.KILLED)
```

Exploratory Data Analysis

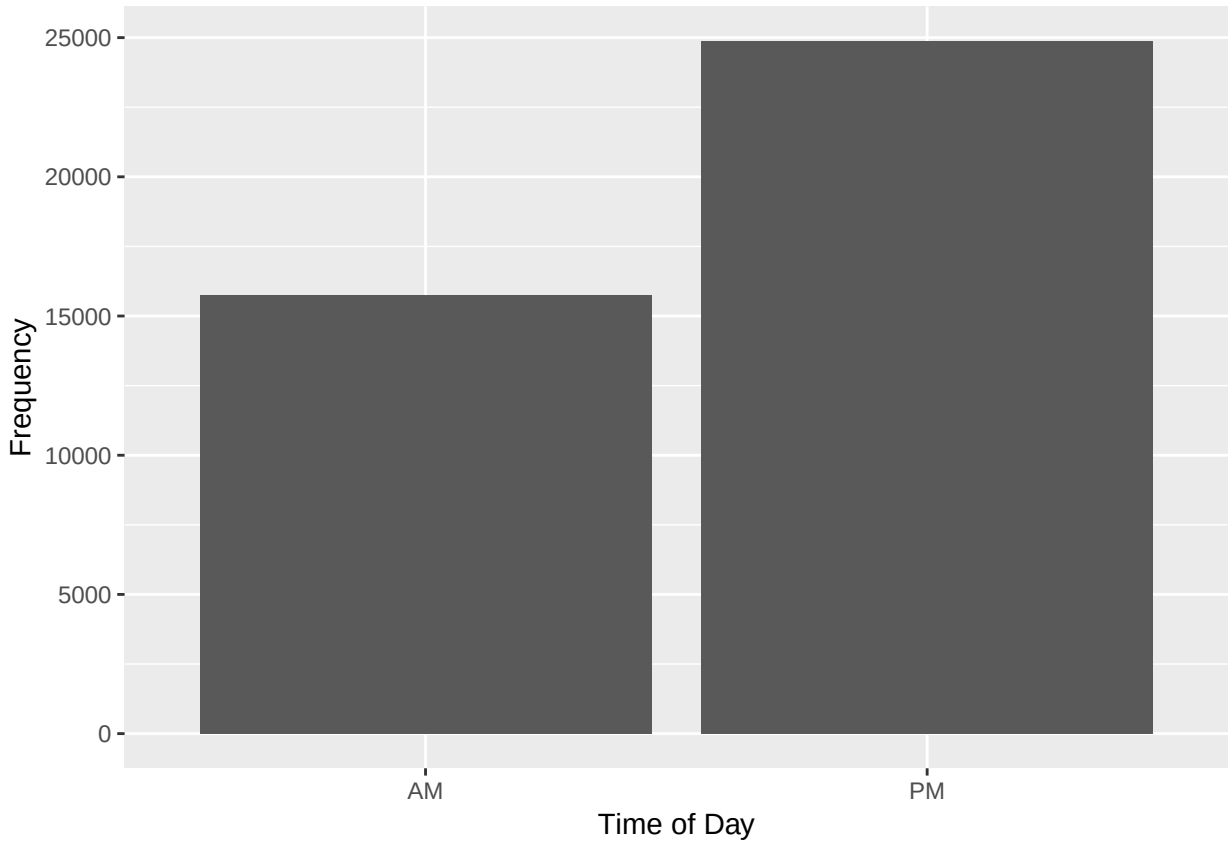
```
data %>%
   ggplot(aes(x = weekday)) +
   geom_bar() +
   labs(x = "Day of Week",
        y = "Frequency")
```




```
data %>%
   count(weekday)
```

```
## # A tibble: 7 x 2
##   weekday     n
##   <ord>   <int>
## 1 Sun      4572
## 2 Mon      5826
## 3 Tue      5669
## 4 Wed      5803
## 5 Thu      6266
## 6 Fri      6835
## 7 Sat      5664
```

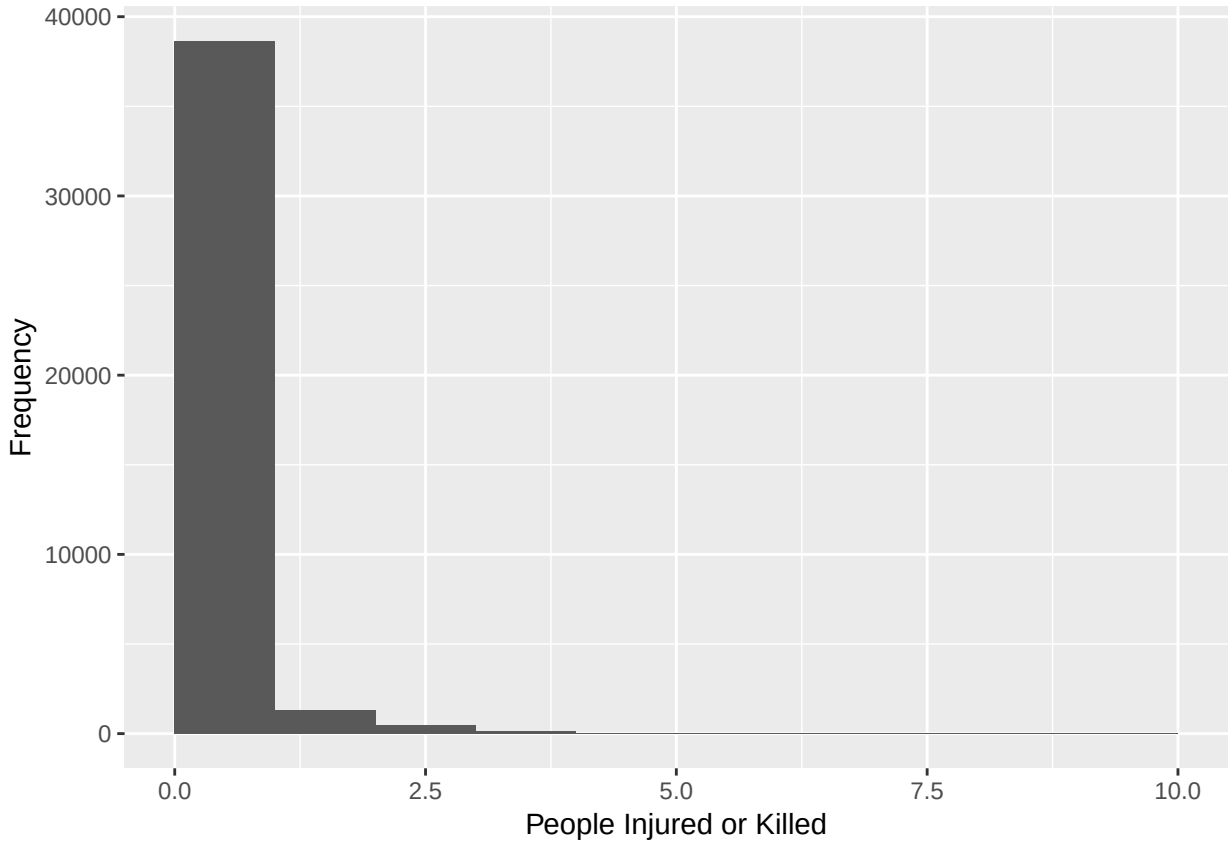
```
data %>%
   ggplot(aes(x = time_of_day)) +
   geom_bar() +
   labs(x = "Time of Day",
        y = "Frequency")
```




```
data %>%
   count(time_of_day)
```

```
## # A tibble: 2 x 2
##   time_of_day     n
##   <chr>       <int>
## 1 AM          15753
## 2 PM          24882
```

```
data %>%
   ggplot(aes(x = person_tot)) +
   geom_histogram(breaks = seq(0, 10, by = 1)) +
   labs(x = "People Injured or Killed",
        y = "Frequency")
```




```
data %>%
   count(person_tot)
```

```
## # A tibble: 11 x 2
##    person_tot     n
##         <int> <int>
##  1          0 31645
##  2          1  7005
##  3          2  1320
##  4          3   450
##  5          4   142
##  6          5    45
##  7          6    15
##  8          7     4
##  9          8     5
## 10          9     3
## 11         10     1
```

```
data %>%
   summarise(mean = mean(person_tot),
             sd = sd(person_tot))
```

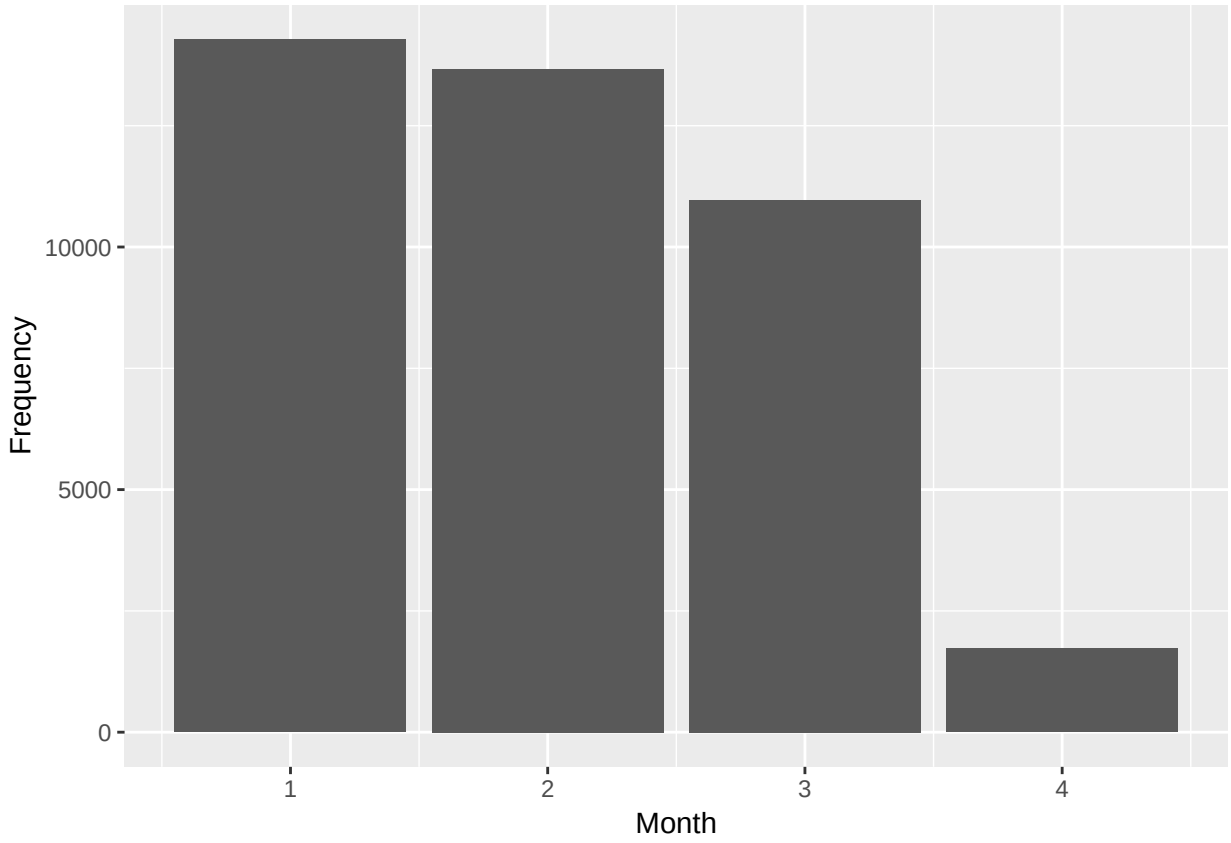
```
##        mean        sd
## 1 0.2948936 0.6580791
```

```
data %>%
   summarise(mean = mean(person_tot),
             sd = sd(person_tot),
```

```
            med = median(person_tot),
            iqr = IQR(person_tot))
```

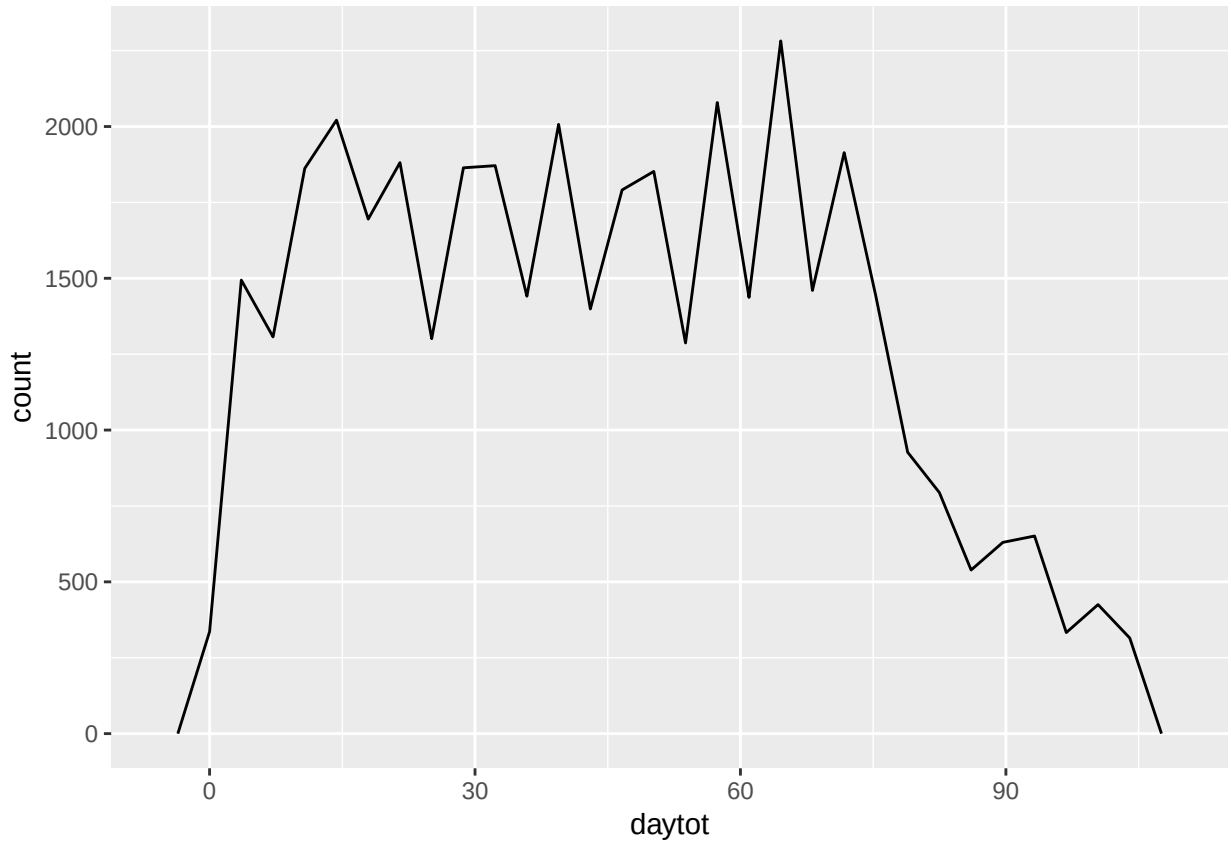
```
##        mean        sd med iqr
## 1 0.2948936 0.6580791   0   0
```

```
data %>%
   ggplot(aes(x = month)) +
   geom_bar() +
   labs(x = "Month",
        y = "Frequency")
```




```
### March lower, April not done yet but still proportionally even lower
data %>%
   ggplot(aes(x=daytot)) +
   geom_freqpoly()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



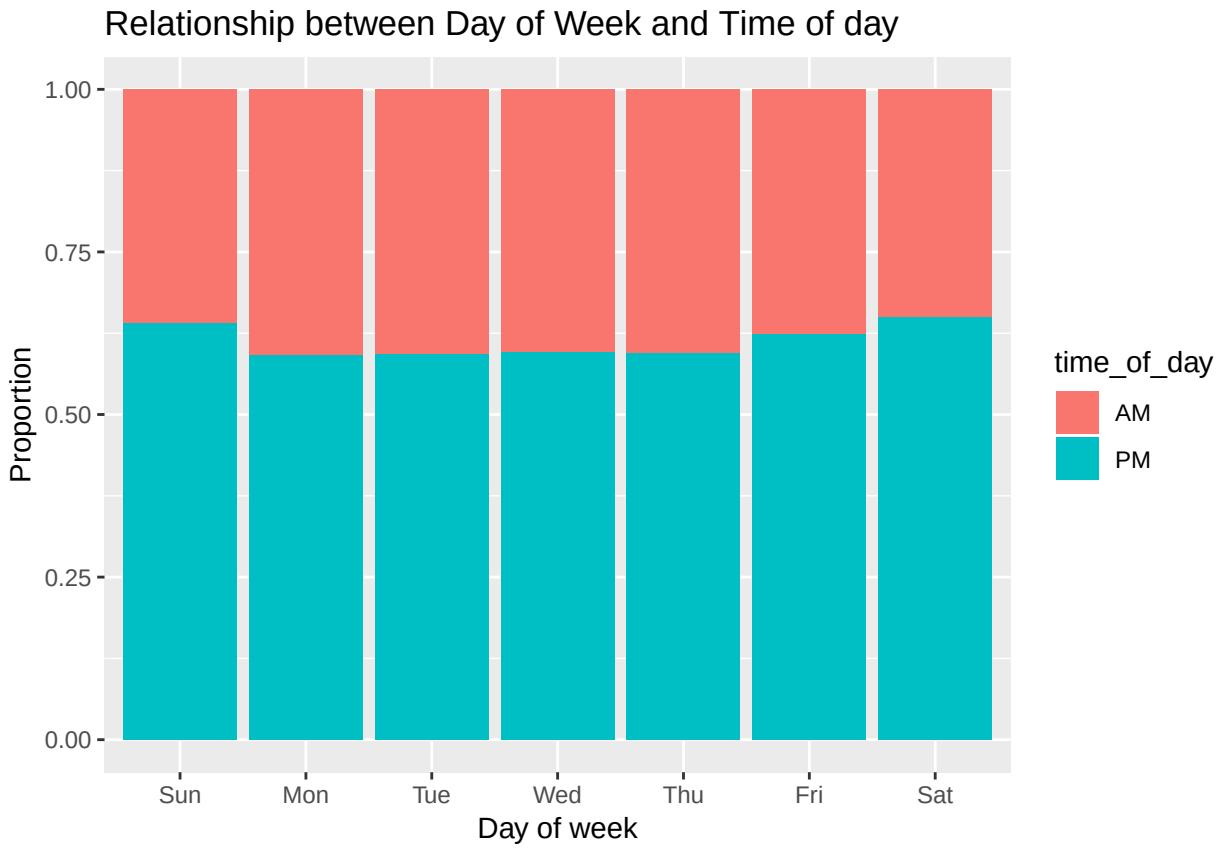

```
data %>%
   count(month)
```

```
## # A tibble: 4 x 2
##   month     n
##   <dbl> <int>
## 1     1 14277
## 2     2 13667
## 3     3 10967
## 4     4  1724
```

```
data %>%
   count(month)
```

```
## # A tibble: 4 x 2
##   month     n
##   <dbl> <int>
## 1     1 14277
## 2     2 13667
## 3     3 10967
## 4     4  1724
```

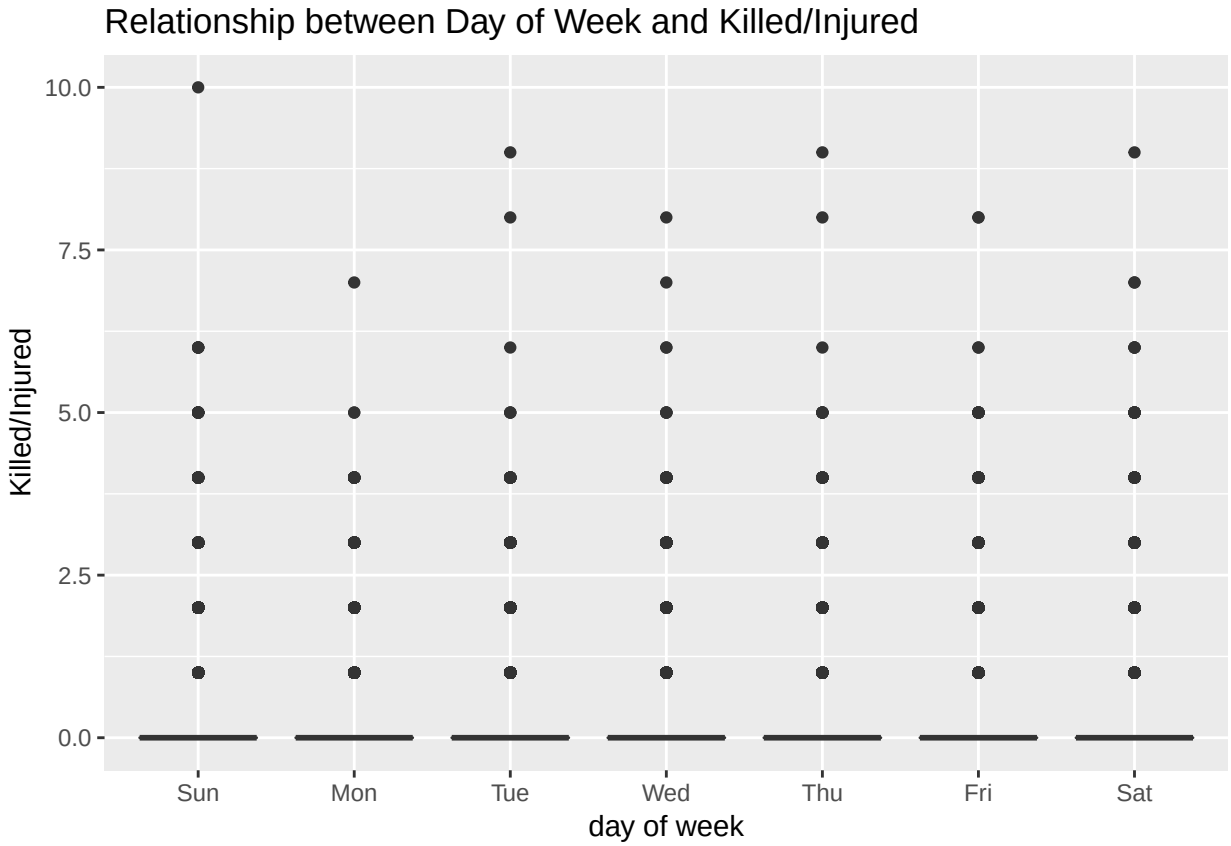
```
data %>%
  ggplot(aes(x = weekday, fill = time_of_day)) +
  geom_bar(position = "fill") +
  labs(title = "Relationship between Day of Week and Time of day",
       x = "Day of week", y = "Proportion")
```




```
table <- table(data$time_of_day, data$weekday)
prop.table(table, margin = 2)
```

```
##
##            Sun       Mon       Tue       Wed       Thu       Fri       Sat
##   AM 0.3589239 0.4086852 0.4065973 0.4041013 0.4052027 0.3752743 0.3490466
##   PM 0.6410761 0.5913148 0.5934027 0.5958987 0.5947973 0.6247257 0.6509534
```

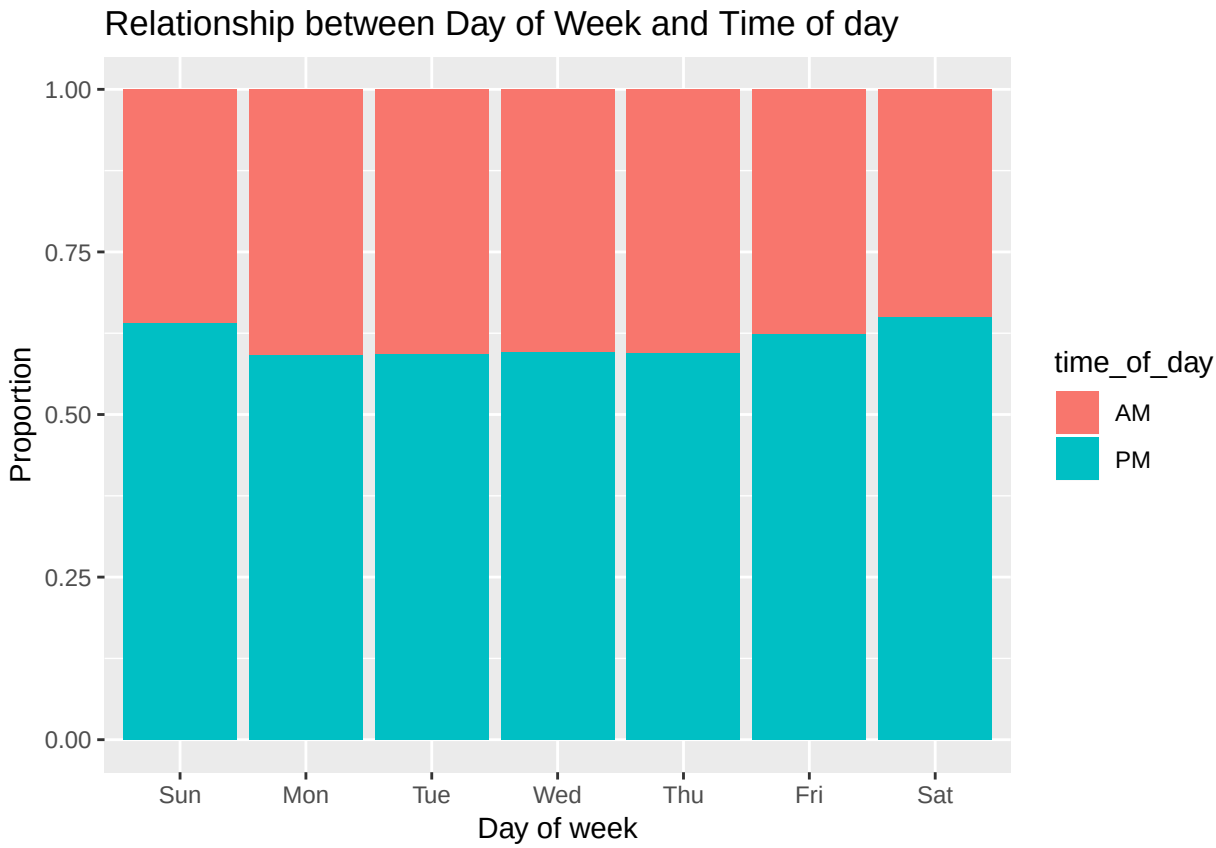
```
data %>%
   ggplot(aes(x = weekday, y = person_tot)) +
   geom_boxplot() +
   labs(title = "Relationship between Day of Week and Killed/Injured",
       x = "day of week", y = "Killed/Injured")
```



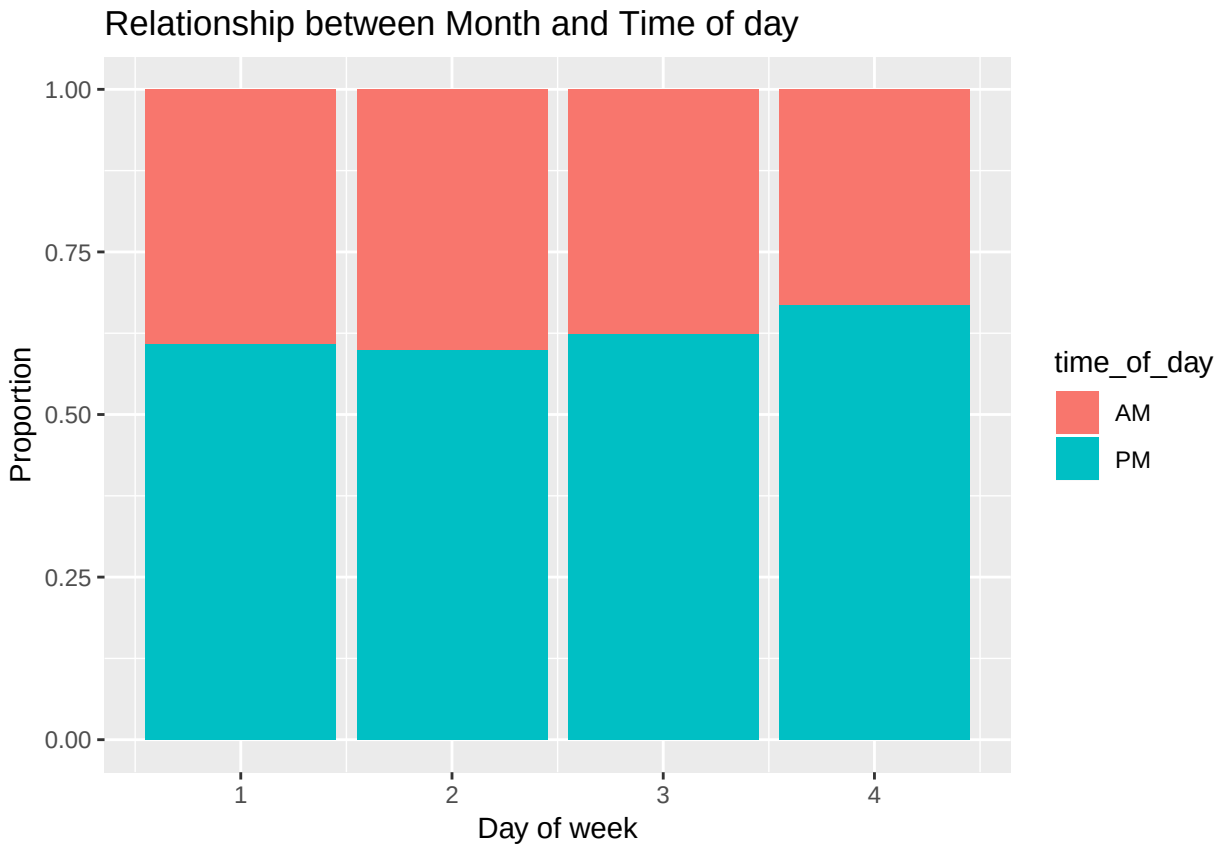

```
data %>%
   group_by(weekday) %>%
   summarise(mean = mean(person_tot),
             sd = sd(person_tot),
             med = median(person_tot),
             iqr = IQR(person_tot))
```

```
## # A tibble: 7 x 5
##   weekday  mean    sd   med   iqr
##   <ord>   <dbl> <dbl> <dbl> <dbl>
## 1 Sun     0.340 0.747     0     0
## 2 Mon     0.281 0.616     0     0
## 3 Tue     0.297 0.649     0     0
## 4 Wed     0.283 0.637     0     0
## 5 Thu     0.295 0.643     0     0
## 6 Fri     0.274 0.631     0     0
## 7 Sat     0.308 0.699     0     0
```

```
data %>%
  ggplot(aes(x = weekday, fill = time_of_day)) +
  geom_bar(position = "fill") +
  labs(title = "Relationship between Day of Week and Time of day",
       x = "Day of week", y = "Proportion")
```



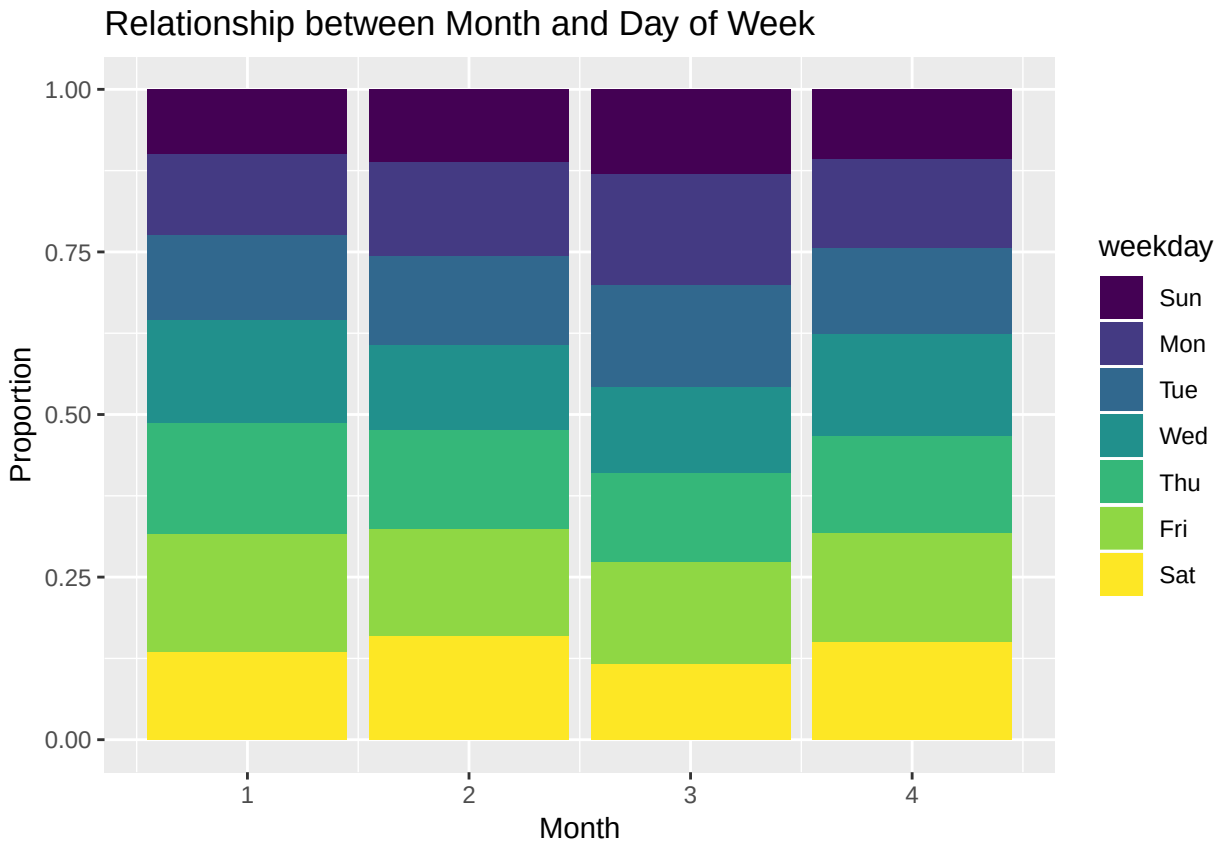

```
data %>%
  ggplot(aes(x = month, fill = time_of_day)) +
  geom_bar(position = "fill") +
  labs(title = "Relationship between Month and Time of day",
       x = "Day of week", y = "Proportion")
```




```
table <- table(data$time_of_day, data$month)
prop.table(table, margin = 2)
```

```
##
##              1         2         3         4
##   AM 0.3915388 0.4004536 0.3754901 0.3317865
##   PM 0.6084612 0.5995464 0.6245099 0.6682135
```

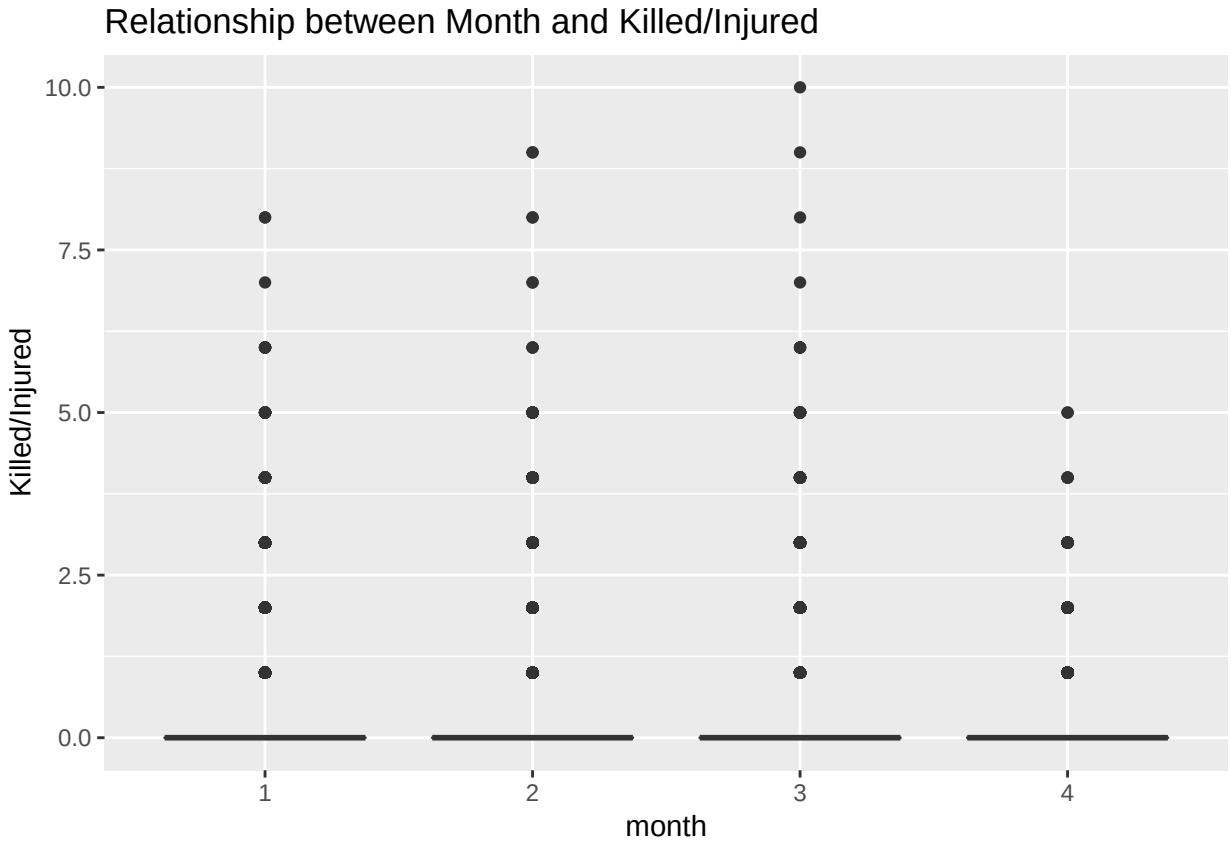
```
data %>%
  ggplot(aes(x = month, fill = weekday)) +
  geom_bar(position = "fill") +
  labs(title = "Relationship between Month and Day of Week",
       x = "Month", y = "Proportion")
```



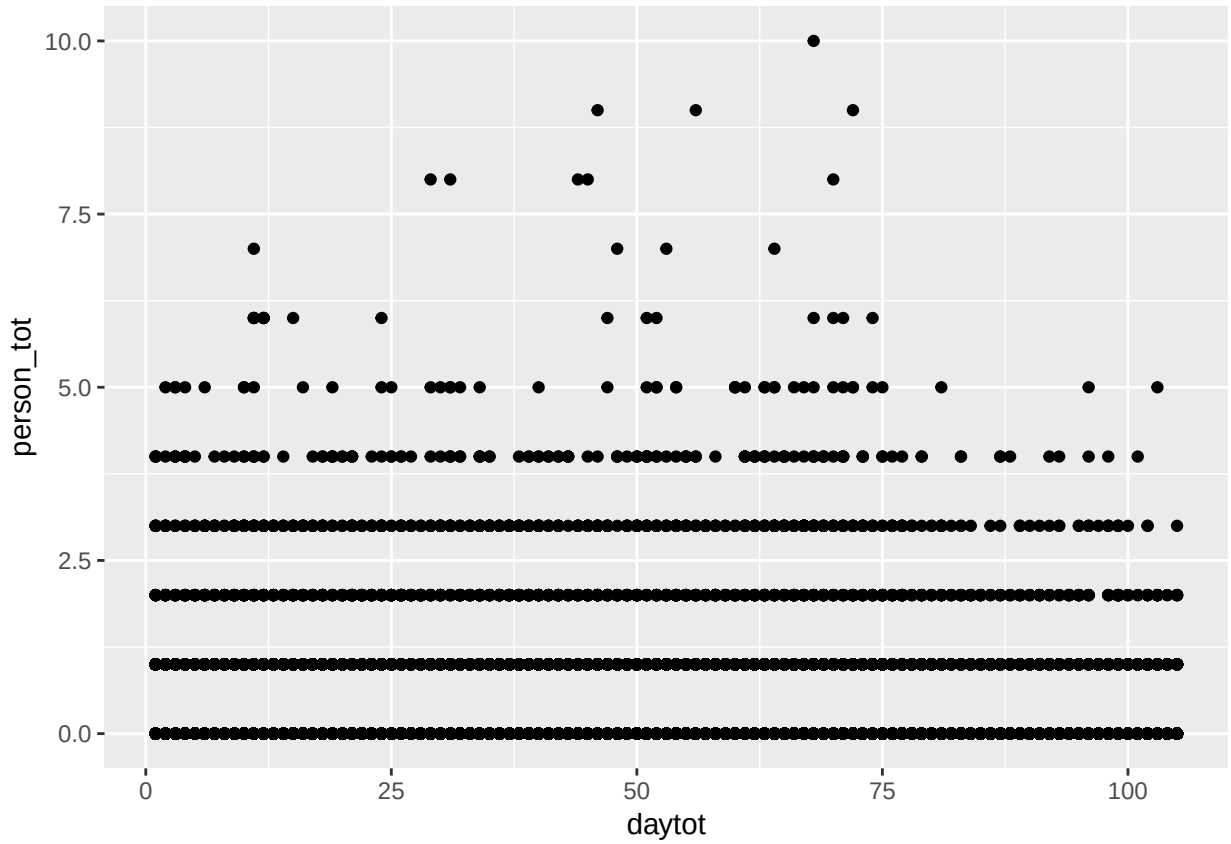

```
table <- table(data$weekday, data$month)
prop.table(table, margin = 2)
```

```
##
##                1          2          3          4
##   Sun 0.09988093 0.11209483 0.13048236 0.10614849
##   Mon 0.12376550 0.14311846 0.17005562 0.13805104
##   Tue 0.12992926 0.13697227 0.15637823 0.13167053
##   Wed 0.15899699 0.13141143 0.13367375 0.15719258
##   Thu 0.17111438 0.15167923 0.13622686 0.14849188
##   Fri 0.18043006 0.16550816 0.15573995 0.16763341
##   Sat 0.13588289 0.15921563 0.11744324 0.15081206
```

```
data %>%
   ggplot(aes(x = as.character(month), y = person_tot)) +
   geom_boxplot() +
   labs(title = "Relationship between Month and Killed/Injured",
       x = "month", y = "Killed/Injured")
```



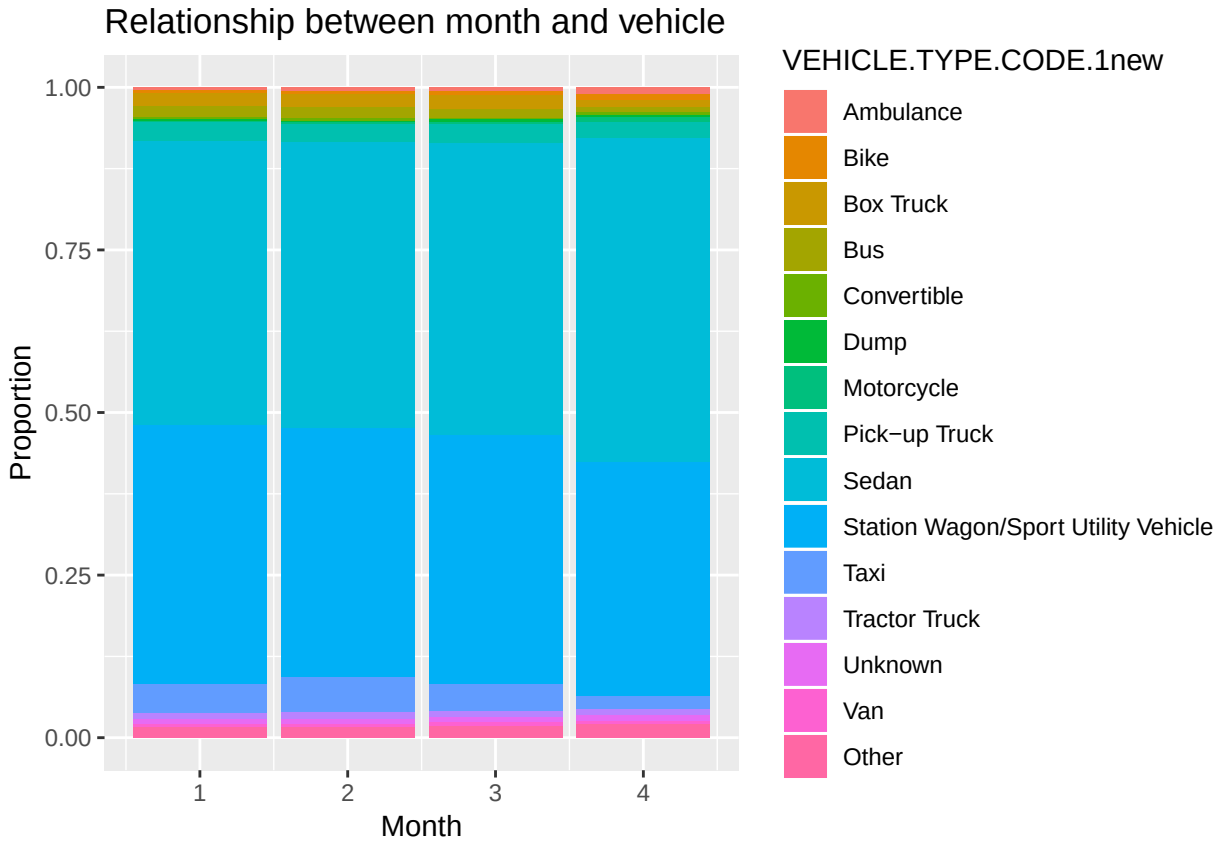

```
data %>%
   ggplot(aes(x = daytot, y = person_tot)) +
   geom_point()
```




```
data %>%
   group_by(month) %>%
   summarise(mean = mean(person_tot),
             sd = sd(person_tot),
             med = median(person_tot),
             iqr = IQR(person_tot))
```

```
## # A tibble: 4 x 5
##   month  mean    sd   med   iqr
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     1 0.296 0.659     0     0
## 2     2 0.297 0.658     0     0
## 3     3 0.287 0.658     0     0
## 4     4 0.325 0.651     0     0
```

```
data %>%
  ggplot(aes(x = month, fill = VEHICLE.TYPE.CODE.1new)) +
  geom_bar(position = "fill") +
  labs(title = "Relationship between month and vehicle",
       x = "Month", y = "Proportion")
```




```
table <- table(data$VEHICLE.TYPE.CODE.1new, data$month)
prop.table(table, margin = 2)
```

```
##
##                                             1           2           3
##   Ambulance                         0.004482734 0.005048657 0.004741497
##   Bike                              0.004342649 0.004609644 0.007203428
##   Box Truck                         0.019051621 0.019828785 0.020151363
##   Bus                               0.018421237 0.017706885 0.014315674
##   Convertible                       0.002241367 0.003146265 0.002370749
##   Dump                              0.002941794 0.002707251 0.003464940
##   Motorcycle                        0.001821111 0.002414575 0.003556123
##   Pick-up Truck                     0.028507390 0.028096876 0.030181453
##   Sedan                             0.436576312 0.440769737 0.448527400
##   Station Wagon/Sport Utility Vehicle 0.398613154 0.382819931 0.382146439
##   Taxi                              0.044827345 0.053486500 0.042582292
##   Tractor Truck                     0.008615255 0.009292456 0.008571168
##   Unknown                           0.008124956 0.008048584 0.007659342
##   Van                               0.005113119 0.005926685 0.005744506
##   Other                             0.016319955 0.016097168 0.018783624
##
##                                             4
##   Ambulance                         0.009280742
##   Bike                              0.010440835
##   Box Truck                         0.010440835
##   Bus                               0.007540603
##   Convertible                       0.004640371
```

```
##   Dump                                0.002320186
##   Motorcycle                          0.008700696
##   Pick-up Truck                       0.023781903
##   Sedan                               0.499419954
##   Station Wagon/Sport Utility Vehicle 0.359628770
##   Taxi                                0.018561485
##   Tractor Truck                       0.009860789
##   Unknown                             0.009280742
##   Van                                 0.004640371
##   Other                               0.021461717
```

The above plot demonstrates the preliminary exploratory data analysis for our research.

The frequency of over the weekdays seem to be evenly distributed. However, weekend crashes differ: Fridays seem to have the most number of crashes, whereas Saturdays and Sundays (especially Sundays) seem to result in far fewer crashes. There seem to be considerably more crashes that occur in PM hours than AM hours. The vast majority of crashes result in 0 injuries and fatalities. The histogram for the number of total injuries and fatalities is highly skewed right. The average number of injuries/fatalities per crash is .294 with a standard deviation of .658, whereas both the median and IQR are 0. January and February seem to have an almost equal number of crashes. March has considerably less crashes than both January or February, and April has considerably less crashes than March, even considering the fact that approximately only half of April's total crashes are represented, as the dataset only records observations through April 14th. The frequency polygraph graphs the number of crashes per day against time. We can see a constant fluctuation around 1750 throughout March, which starts to drop signficantly around day 70 (which is around March 10th). Most of the vehicle types recorded are Sedans, followed by Sports cars.

Comparing the day of week and time of day for the crashes, the weekend days seem to have a slightly higher proportion of PM crashes than do the weekday days. There doesn't seem to be much relationship between day of week and total number of injuries and fatalities for the crashes, although it is hard to tell because of the extreme skewness of the plot. Summary statistics reveal slightly higher injury/fatality rates for Saturdays and Sundays (although the median and IQR are 0 for each day of the week), but we don't know yet if this is significant. There seem to be a higher proportion in PM crashes instead of AM crashes in March and April, compared to January and February. There doesn't seem to be much difference in the distribution of the days of the week of the crashes over the four months. Comparing month of crashes and number of injuries/fatalities per crash, there seem to be less crashes in April with a higher number of injuries/fatalities, but it is important to keep in mind that only half of April's days are observed. However, summary statistics demonstrate that there doesn't seem to be much significant difference in the average number of people injured/killed per crash, or the standard deviation (April has a slightly higher average, but especially because it has less datapoints, we would need to do more tests to determine the significance of this difference.) Graphing the number of fatalities/injuries in each crash over time, we can wee that the number of injuries and fatalities in each crash have much less scatter after around day 74 (about March 14th), which demonstrates that there have been virtually no crashes with a considerably high number (5+) of injuries/fatalities since then. Comparing month with type of vehicle, there seems to be a slightly higher proportion of Sedans in crashes for March and more so April, though it is of note that there are fewer April dates.

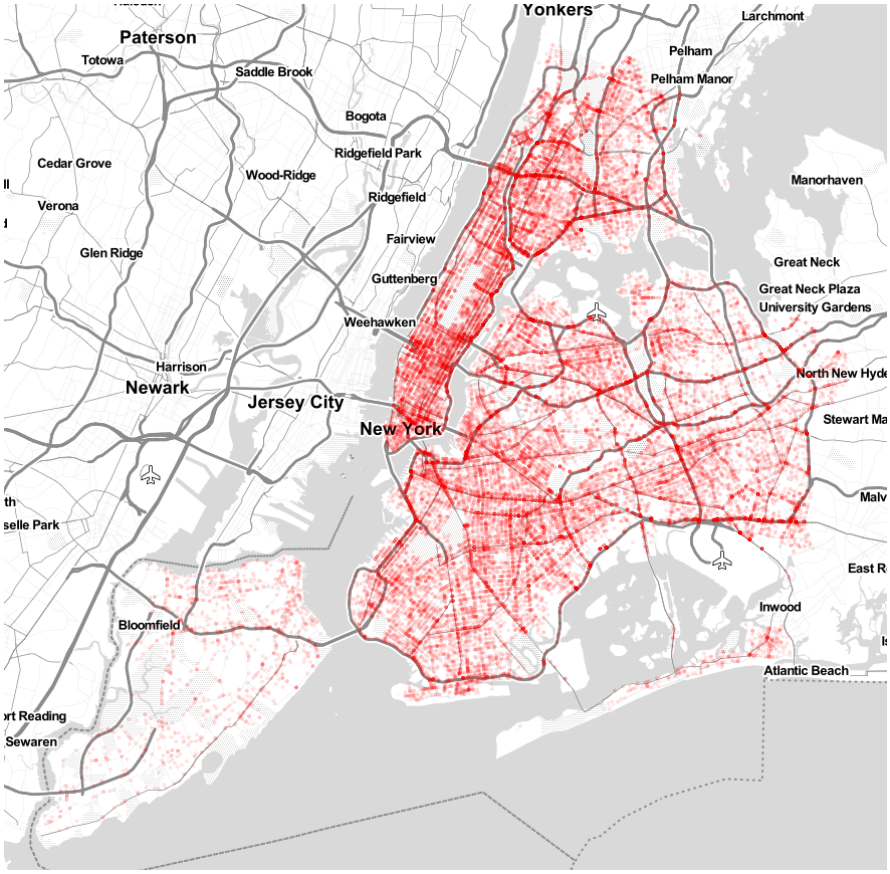
The following plot shows the geographic distribution of all NYC car crashes in 2020 from the dataset.

```
library(naniar)
library(ggmap)
data <- data %>%
  replace_with_na_at(.vars = c("LONGITUDE","LATITUDE"),
                     condition = ~.x == 0)
qmplot(LONGITUDE, LATITUDE, data = data, color = I("red"), alpha = I(.1), size = I(.01))
```

```
## Using zoom = 11...
```

```
## Map from URL : http://tile.stamen.com/toner-lite/11/601/768.png
```

```
## Map from URL : http://tile.stamen.com/toner-lite/11/602/768.png
## Map from URL : http://tile.stamen.com/toner-lite/11/603/768.png
## Map from URL : http://tile.stamen.com/toner-lite/11/604/768.png
## Map from URL : http://tile.stamen.com/toner-lite/11/601/769.png
## Map from URL : http://tile.stamen.com/toner-lite/11/602/769.png
## Map from URL : http://tile.stamen.com/toner-lite/11/603/769.png
## Map from URL : http://tile.stamen.com/toner-lite/11/604/769.png
## Map from URL : http://tile.stamen.com/toner-lite/11/601/770.png
## Map from URL : http://tile.stamen.com/toner-lite/11/602/770.png
## Map from URL : http://tile.stamen.com/toner-lite/11/603/770.png
## Map from URL : http://tile.stamen.com/toner-lite/11/604/770.png
## Map from URL : http://tile.stamen.com/toner-lite/11/601/771.png
## Map from URL : http://tile.stamen.com/toner-lite/11/602/771.png
## Map from URL : http://tile.stamen.com/toner-lite/11/603/771.png
## Map from URL : http://tile.stamen.com/toner-lite/11/604/771.png
## Warning: `panel.margin` is deprecated. Please use `panel.spacing` property
## instead
## Warning: Removed 3263 rows containing missing values (geom_point).
## Theme element panel.border missing
## Theme element axis.line.x.bottom missing
## Theme element axis.ticks.x.bottom missing
## Theme element axis.line.x.top missing
## Theme element axis.ticks.x.top missing
## Theme element axis.line.y.left missing
## Theme element axis.ticks.y.left missing
## Theme element axis.line.y.right missing
## Theme element axis.ticks.y.right missing
## Theme element plot.title missing
## Theme element plot.subtitle missing
## Theme element plot.tag missing
## Theme element plot.caption missing
```



To try to spot patterns in scatter over time, we divide the dataset into monthly periods, and plot the geographic distribution by month.

```
data.s1 <- data[data$month == 1 ,]
data.s2 <- data[data$month == 2 ,]
data.s3 <- data[data$month == 3 ,]
data.s4 <- data[data$month == 4 ,]

qmplot(LONGITUDE, LATITUDE, data = data.s1, color = I("red"), alpha = I(.2), size = I(.01))
```

```
## Using zoom = 11...

## Warning: `panel.margin` is deprecated. Please use `panel.spacing` property
## instead

## Warning: Removed 1124 rows containing missing values (geom_point).

## Theme element panel.border missing

## Theme element axis.line.x.bottom missing

## Theme element axis.ticks.x.bottom missing

## Theme element axis.line.x.top missing

## Theme element axis.ticks.x.top missing

## Theme element axis.line.y.left missing

## Theme element axis.ticks.y.left missing

## Theme element axis.line.y.right missing
```

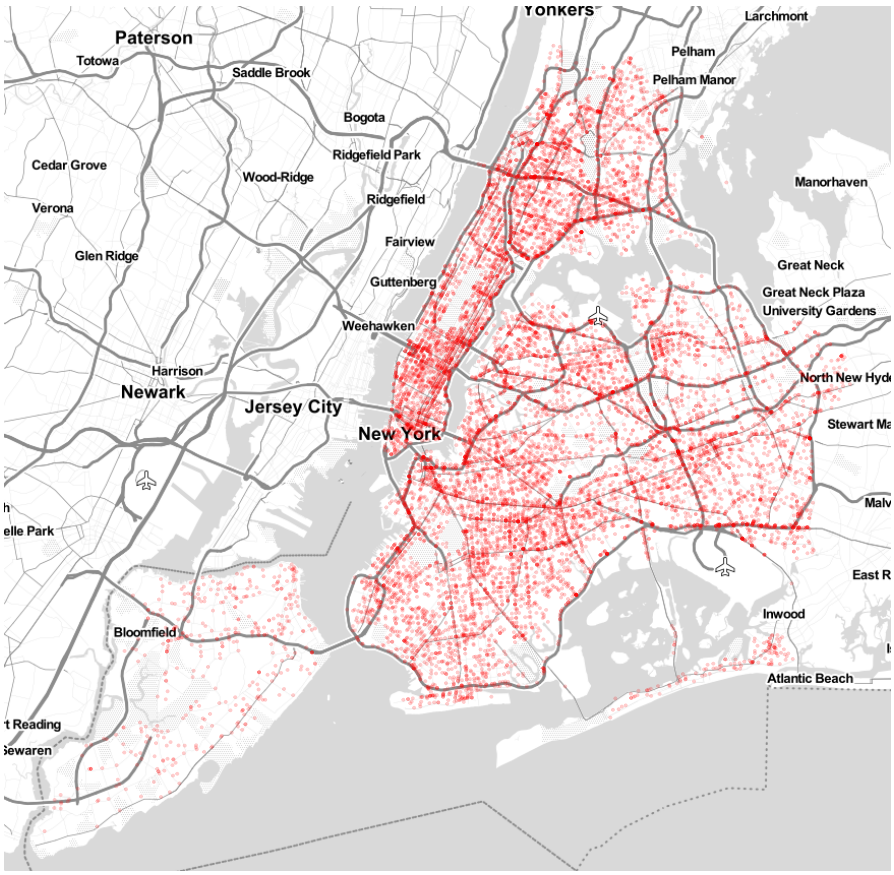
```
## Theme element axis.ticks.y.right missing
```

```
## Theme element plot.title missing
```

```
## Theme element plot.subtitle missing
```

```
## Theme element plot.tag missing
```

```
## Theme element plot.caption missing
```




```
qmplot(LONGITUDE, LATITUDE, data = data.s2, color = I("red"), alpha = I(.2), size = I(.01))
```

```
## Using zoom = 11...
```

```
## Warning: `panel.margin` is deprecated. Please use `panel.spacing` property
## instead
```

```
## Warning: Removed 1112 rows containing missing values (geom_point).
```

```
## Theme element panel.border missing
```

```
## Theme element axis.line.x.bottom missing
```

```
## Theme element axis.ticks.x.bottom missing
```

```
## Theme element axis.line.x.top missing
```

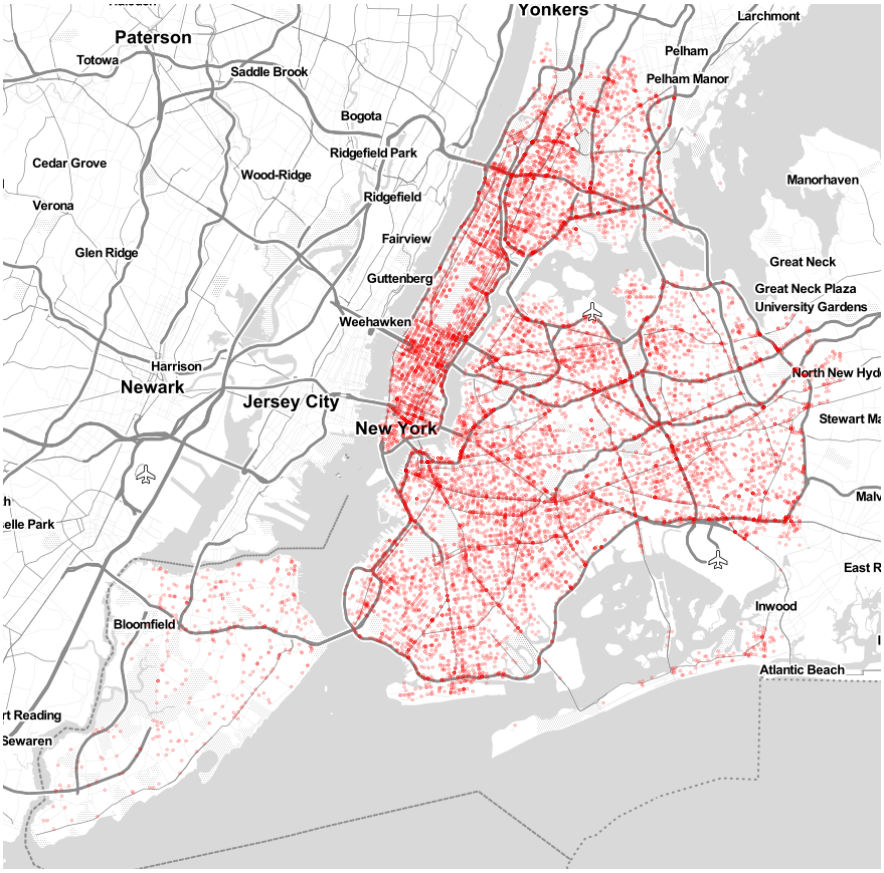
```
## Theme element axis.ticks.x.top missing
```

```
## Theme element axis.line.y.left missing
```

```
## Theme element axis.ticks.y.left missing
```

```
## Theme element axis.line.y.right missing
```

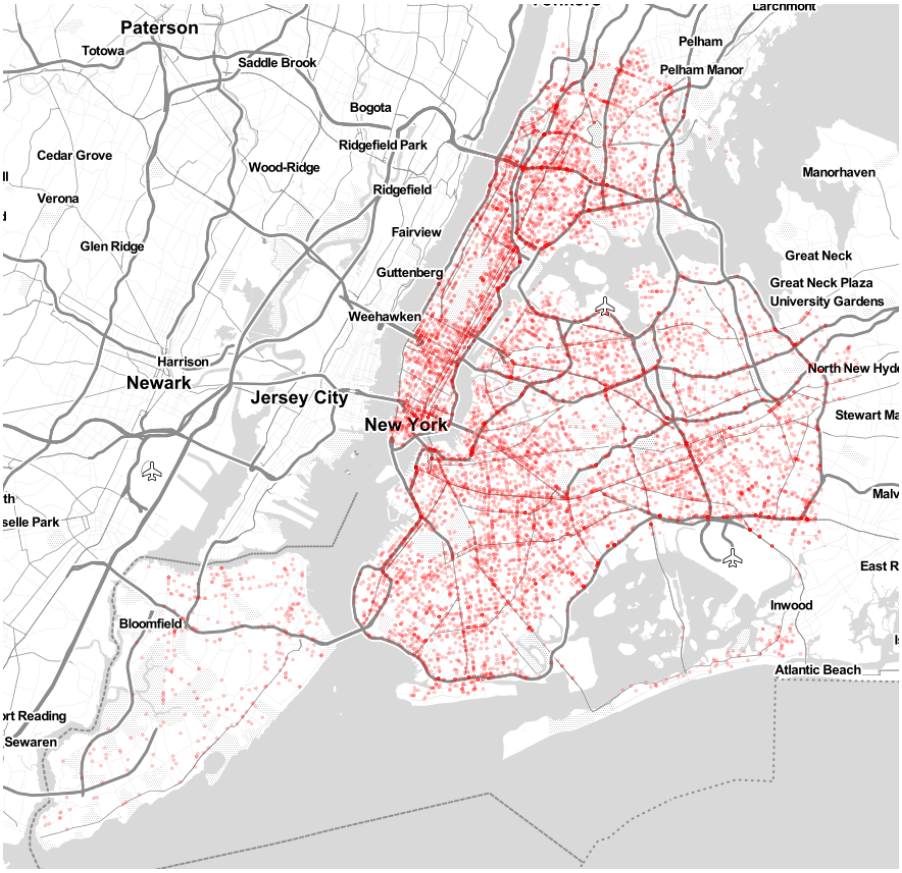
```
## Theme element axis.ticks.y.right missing
## Theme element plot.title missing
## Theme element plot.subtitle missing
## Theme element plot.tag missing
## Theme element plot.caption missing
```




```
qmplot(LONGITUDE, LATITUDE, data = data.s3, color = I("red"), alpha = I(.2), size = I(.01))
```

```
## Using zoom = 11...
## Warning: `panel.margin` is deprecated. Please use `panel.spacing` property
## instead
## Warning: Removed 882 rows containing missing values (geom_point).
## Theme element panel.border missing
## Theme element axis.line.x.bottom missing
## Theme element axis.ticks.x.bottom missing
## Theme element axis.line.x.top missing
## Theme element axis.ticks.x.top missing
## Theme element axis.line.y.left missing
## Theme element axis.ticks.y.left missing
## Theme element axis.line.y.right missing
```

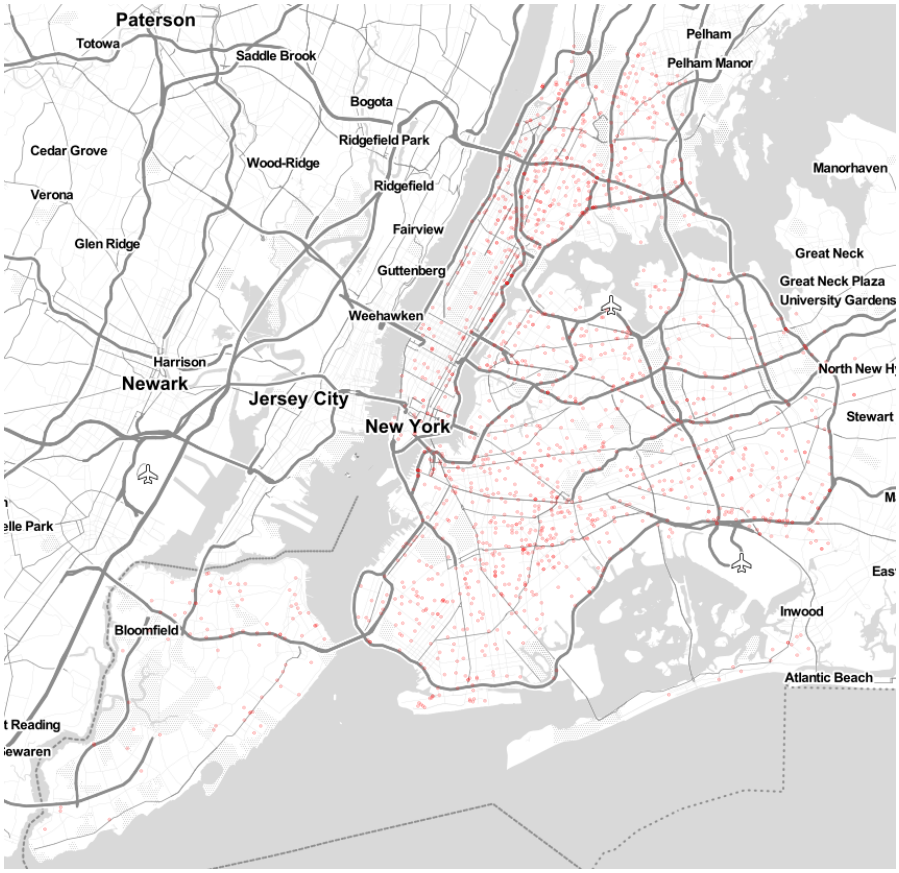
```
## Theme element axis.ticks.y.right missing
## Theme element plot.title missing
## Theme element plot.subtitle missing
## Theme element plot.tag missing
## Theme element plot.caption missing
```




```
qmplot(LONGITUDE, LATITUDE, data = data.s4, color = I("red"), alpha = I(.2), size = I(.01))
```

```
## Using zoom = 11...
## Warning: `panel.margin` is deprecated. Please use `panel.spacing` property
## instead
## Warning: Removed 145 rows containing missing values (geom_point).
## Theme element panel.border missing
## Theme element axis.line.x.bottom missing
## Theme element axis.ticks.x.bottom missing
## Theme element axis.line.x.top missing
## Theme element axis.ticks.x.top missing
## Theme element axis.line.y.left missing
## Theme element axis.ticks.y.left missing
## Theme element axis.line.y.right missing
```

```
## Theme element axis.ticks.y.right missing
## Theme element plot.title missing
## Theme element plot.subtitle missing
## Theme element plot.tag missing
## Theme element plot.caption missing
```

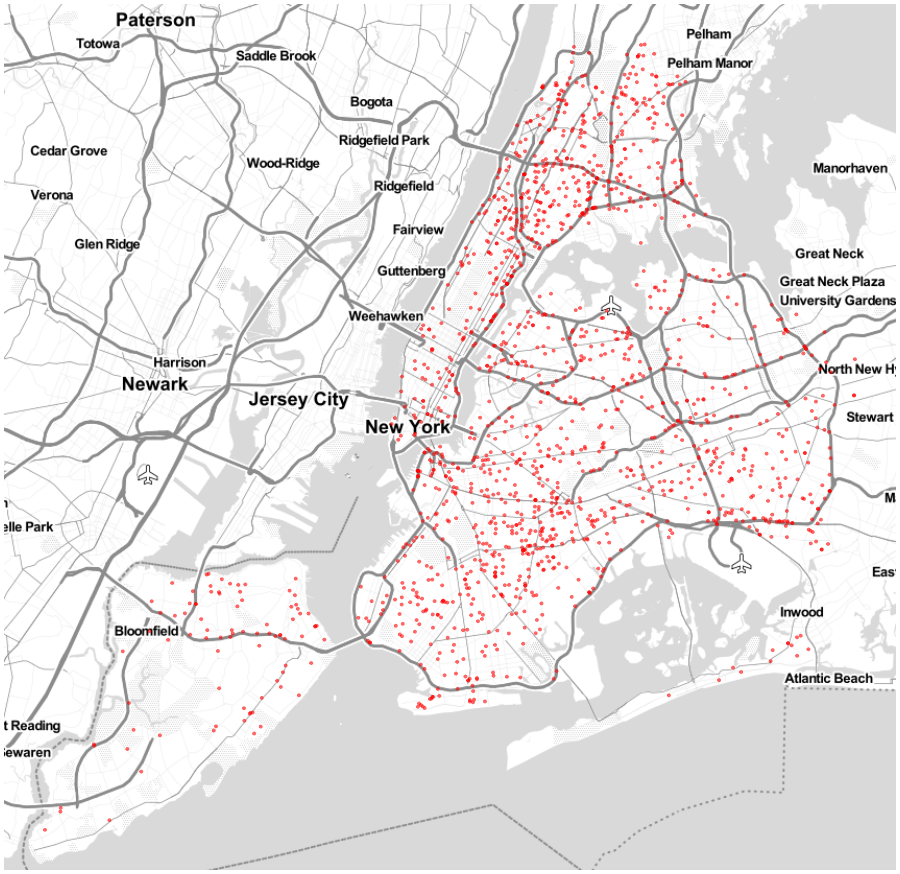


From the above plots, we can see that the March and April have far less density of crashes in all of the city as a whole. We note, however, that only half of April is accounted for in the dataset. To account for this, we decrease the transparency of points and plot April again to see if there are notable differences in geographic location.

```
qmplot(LONGITUDE, LATITUDE, data = data.s4, color = I("red"), alpha = I(.6), size = I(.01))
```

```
## Using zoom = 11...
## Warning: `panel.margin` is deprecated. Please use `panel.spacing` property
## instead
## Warning: Removed 145 rows containing missing values (geom_point).
## Theme element panel.border missing
## Theme element axis.line.x.bottom missing
## Theme element axis.ticks.x.bottom missing
## Theme element axis.line.x.top missing
## Theme element axis.ticks.x.top missing
## Theme element axis.line.y.left missing
```

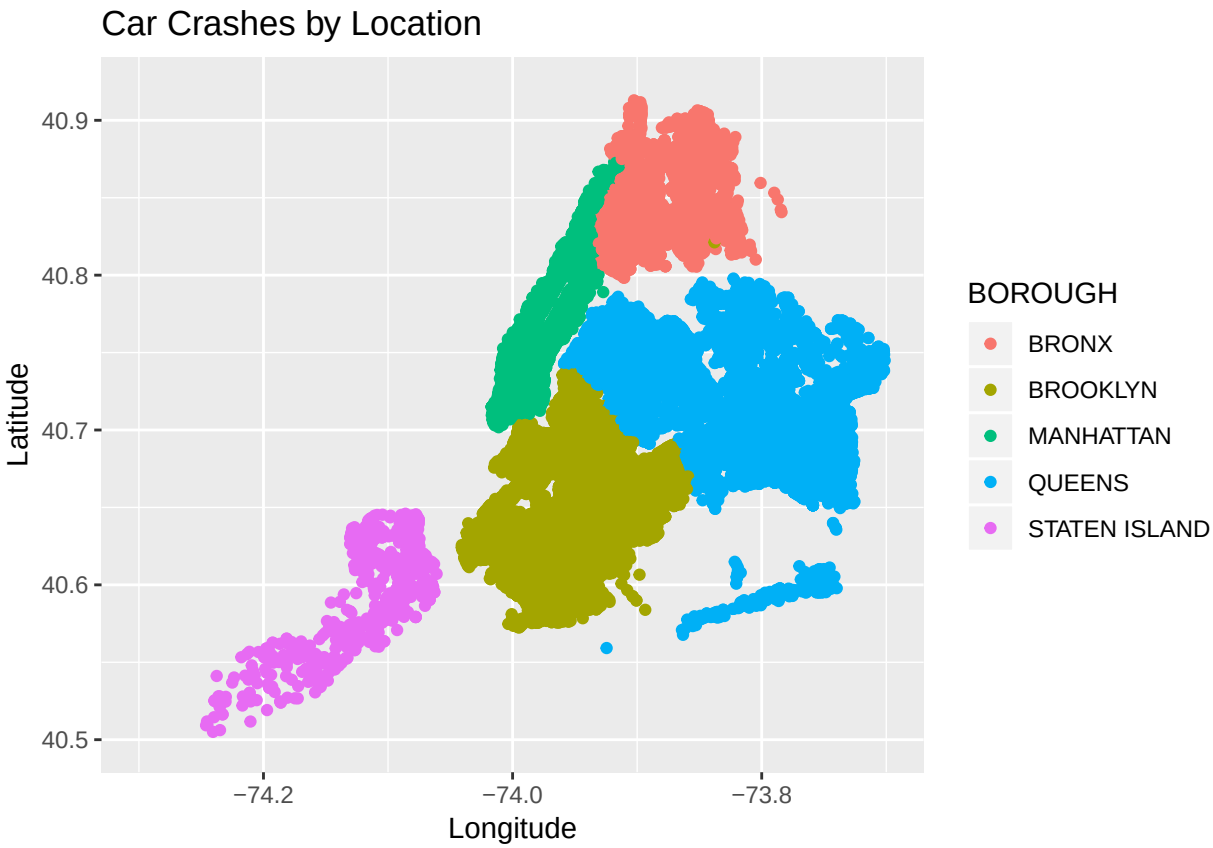
```
## Theme element axis.ticks.y.left missing
## Theme element axis.line.y.right missing
## Theme element axis.ticks.y.right missing
## Theme element plot.title missing
## Theme element plot.subtitle missing
## Theme element plot.tag missing
## Theme element plot.caption missing
```



Whereas the geographic scatterplots for January through March showed a higher concentration of crashes in the lower Manhattan area, the plot for April so far seems to show crashes that are much more evenly distributed throughout New York City (a trend that we can see starting in March, though to a lesser extent). In fact, the higher concentrated areas now seem to be the area in between Manhattan and Bronx, as well as the middle of Brooklyn. However, we must keep in mind that the April plot is constructed with fewer data points- both because of the shorter time period, and because of the overall lower frequency of crashes. Furthermore, we must keep in mind that there are several missing values that weren't included in these geographic scatterplots, which may be missing systematically.

```
boroughs_included <- data %>% filter(BOROUGH=="BROOKLYN" | BOROUGH=="MANHATTAN" | BOROUGH=="QUEENS" | BO
ggplot(data=boroughs_included, mapping=aes(x=LONGITUDE, y=LATITUDE, colour = BOROUGH)) + geom_point() +
```

```
## Warning: Removed 785 rows containing missing values (geom_point).
```



Analyze by lockdown and borough:

```
boroughs_included %>% group_by(lockdown, BOROUGH) %>% summarise(n())
```

```
## # A tibble: 10 x 3
## # Groups:   lockdown [?]
##    lockdown BOROUGH       `n()`
##       <dbl> <fct>         <int>
##  1        0 BRONX          4280
##  2        0 BROOKLYN       8191
##  3        0 MANHATTAN      4217
##  4        0 QUEENS         7234
##  5        0 STATEN ISLAND   579
##  6        1 BRONX           455
##  7        1 BROOKLYN        792
##  8        1 MANHATTAN       242
##  9        1 QUEENS          688
## 10        1 STATEN ISLAND    89
```

```
boroughs_included %>% group_by(lockdown) %>% summarise(n())
```

```
## # A tibble: 2 x 2
##   lockdown `n()`
##      <dbl> <int>
## 1        0 24501
## 2        1  2266
```

```
4280/24501 #Bronx pre
```

```
## [1] 0.1746867
```

```
8191/24501 #Brooklyn pre
```

```
## [1] 0.3343129
```

```
4217/24501 #Manhattan pre
```

```
## [1] 0.1721154
```

```
7234/24501 #Queens pre
```

```
## [1] 0.2952533
```

```
579/24501 #Staten Island pre
```

```
## [1] 0.02363169
```

```
455/2266
```

```
## [1] 0.2007944
```

```
792/2266
```

```
## [1] 0.3495146
```

```
242/2266
```

```
## [1] 0.1067961
```

```
688/2266
```

```
## [1] 0.3036187
```

```
89/2266
```

```
## [1] 0.03927626
```