Project_1_Main

Michael Arango, Yeshwant Chillakuru, Jackson Crum

March 6, 2017

Description of the Dataset

Reading the Data into R

```
getwd()
tickets <- data.frame(read.csv("Muni_Driver_Reported_Transit_Only_Lane_Violations.csv",
    header = T, sep = ","))
```

Checking Dimensions and Structure of the Dataset

```
# Check dimensions of dataset
dim(tickets)
# Check variable names
colnames(tickets)
# Examine structure of the data
str(tickets)
```

Summary of the Dataset

The dataset consists of San Francisco Municipal Transportation Agency (SFMTA) driver reported transit only lane violations from March 2008 to February 2015. The dataset contains 17,178 observations and 18 variables. The data were obtained from the US government's open data site at https://catalog.data.gov/dataset/muni-driver-reported-transit-only-lane-violations.

Description of Variables in the dataset:

- `Object.ID` – Unique key that corresponds to each observation
- `Ticket.Number` – Unique number that corresponds to each ticket catalogued with the Superior Court of San Francisco
- `Citation.Issue.Date` – The date (MM/DD/YY) that the traffic only lane violation citation was issued
- `Citation.Issue.Month` – The month (MM) that the traffic only lane violation citation was issued; This column was completely blank when downloaded
- `Citation.Issue.Time` – The time of day (HH:MM) PST that the traffic only lane violation citation was issued
- `Location` – The street address where the traffic only lane violation citation was issued
- `Violation.Code` – A code that corresponds to the type of traffic only lane violation that was committed to warrant issuing a citation
- `Violation` – Description of the traffic only lane violation issued in the citation
- `Fine.Amount` – The dollar amount of to be paid for the traffic only lane violation
- `Citation.Status` – Status of the citation (Open, Closed, Unapplied)
- `Amount.Paid` – The dollar amount paid for the traffic only lane violation
- `Amount.Due` – Outstanding balance for the traffic only lane violation

- `Suspend.Code` – Code that represents reason for license suspension (Suspended until the driver goes to driving school, pays fine, too many points on license, etc.)
- `Suspend.Process.Date` – Date the license suspension goes into effect
- `Suspend.Until.Date` – Date the license suspension is over
- `Disposition.Code` – The city of San Francisco does not release disposition codes
- `Last.Edited.Date` – Date that represents the last date the corresponding observation was edited; This column was completely blank when downloaded
- `Geom` – The latitude and longitude (latitutde, longitude) of the traffic only lane violation

Explanations of Violation variable:

- 'DBL Park' – Double Parking
- 'TWAWY ZN#1' and 'TWAWY ZN#2' – Towaway Zone 1 and Towaway Zone 2
- 'TRNST ONLY' – Parking Transit-Only
- 'PRK PROHIB' – Tow-Away Zone - Downtown Core
- 'PK PHB OTD' – Tow-Away Zone - Outside Downtown Core
- 'PK FR LN' – Parking in Fire Lane
- 'ON SIDEWLK' – Parking on Sidewalk
- 'DBL PARK' – Double Parking
- 'BUS ZONE' – Parking in Bus Zone
- 'OVR 18" C' – Over 18 inches from Curb

Formatting Data

```
# Format Date and Month columns Change citation issue date from string to
# month/day/year date variable
tickets$Citation.Issue.Date <- mdy(tickets$Citation.Issue.Date)
# Populate blank column for citation issue month with month of citation
# issue date
tickets$Citaton.Issue.Month <- NULL
tickets$Citation.Issue.Month <- month(tickets$Citation.Issue.Date, label = TRUE)

# Convert Amount.Paid from factor to numeric
tickets$Amount.Paid <- as.numeric(tickets$Amount.Paid)
```

Data Cleaning

```
# New Column 'Paid.On.Time' (1 = true, 0 = false). Based on assumption that
# late payments have to pay an extra fine and, thus, Amount.Paid >
# Fine.Ammount
tickets$Paid.On.Time <- ifelse(tickets$Amount.Paid > tickets$Amount.Due, 1,
    0)
tickets$Paid.On.Time <- as.factor(tickets$Paid.On.Time)

# Seperate Geom variable into 2 separate latitude and longitude numeric
# variables remove parentheses from Geom variable
tickets$Geom <- substr(tickets$Geom, 2, nchar(as.character(tickets$Geom)) -
    1)
## split string at comma & create new columns
out <- strsplit(as.character(tickets$Geom), ", ")
```

```
## Drop observations with no Geom data. ***21 OBSERVATIONS DROPPED***
tickets <- tickets[!(is.na(tickets$Geom) | tickets$Geom == ""), ]
tickets <- data.frame(tickets, do.call(rbind, out))
rm(out)
## rename columns and set str to numeric
names(tickets)[names(tickets) == "X1"] <- "Latitude"
names(tickets)[names(tickets) == "X2"] <- "Longitude"
tickets$Latitude <- as.numeric(as.character(tickets$Latitude))
tickets$Longitude <- as.numeric(as.character(tickets$Longitude))
head(tickets)
```

```
##   Object.ID Ticket.Number Citation.Issue.Date Citation.Issue.Time
## 1         1     977404120          2008-02-29               11:57
## 2         2     977404131          2008-03-02               16:32
## 3         3     977404142          2008-03-02               15:00
## 4         4     100000502          2008-03-02               14:26
## 5         5     100000503          2008-03-04               16:42
## 6         6     100000504          2008-03-04                9:36
##        Location Violation.Code Violation Fine.Amount Citation.Status
## 1  561 OFARRELL        V22500H  DBL PARK          65          Closed
## 2  162 GEARY ST        V22500H  DBL PARK          65          Closed
## 3 1055 OFARRELL        V22500H  DBL PARK          65            Open
## 4  431 OFARRELL        V22500H  DBL PARK          65          Closed
## 5     908 GEARY        V22500I  BUS ZONE         250          Closed
## 6  577 OFARRELL        V22500H  DBL PARK          65          Closed
##   Amount.Paid Amount.Due Suspend.Code         Suspend.Process.Date
## 1         163          0 109 109TRNLN 03/11/2008 12:00:00 AM +0000
## 2         152          0 109 109TRNLN 03/11/2008 12:00:00 AM +0000
## 3           1         90
## 4         152          0
## 5          90          0
## 6         152          0
##             Suspend.Until.Date Disposition.Code Last.Edited.Date
## 1 04/01/2008 12:00:00 AM +0000                               NA
## 2 04/01/2008 12:00:00 AM +0000                               NA
## 3                                                            NA
## 4                                                            NA
## 5                                                            NA
## 6                                                            NA
##                           Geom Citation.Issue.Month Paid.On.Time Latitude
## 1  37.7856012607, -122.41396696                 Feb            1 37.78560
## 2 37.7877148223, -122.406035679                 Mar            1 37.78771
## 3 37.7846059732, -122.421865197                 Mar            0 37.78461
## 4 37.7858668852, -122.411853807                 Mar            1 37.78587
## 5 37.7861515263, -122.418291079                 Mar            1 37.78615
## 6 37.7855694751, -122.414219806                 Mar            1 37.78557
##   Longitude
## 1 -122.4140
## 2 -122.4060
## 3 -122.4219
## 4 -122.4119
## 5 -122.4183
## 6 -122.4142
```

```r
# Create column indicating the year the ticket was issued
tickets$Citation.Issue.Year <- year(tickets$Citation.Issue.Date)

# Create column indicating DayOfWeek (Sunday (1) to Saturday (7)) that the
# ticket was issued
tickets$Citation.Issue.DayOfWeek <- wday(tickets$Citation.Issue.Date)

# Create column indicating whether the ticket was issued on: weekday (1),
# weekend (0)
tickets$Citation.Issue.Weekday <- ifelse(tickets$Citation.Issue.DayOfWeek >=
    2 & tickets$Citation.Issue.DayOfWeek <= 6, 1, 0)
tickets$Citation.Issue.Weekday <- as.factor(tickets$Citation.Issue.Weekday)
tickets$Citation.Issue.Weekday <- ifelse(tickets$Citation.Issue.Weekday == 0,
    "Weekend", "Weekday")

# Convert Citation.Issue.DayOfWeek from 1, 2, 3, 4, 5, 6, 7 to Mon, Tues,
# Wed, Thurs, Fri, Sat
tickets$Citation.Issue.DayOfWeek <- wday(tickets$Citation.Issue.Date, label = TRUE)

# Create column for Hour of citation issuance. Drop observations with no
# Citation.Issue.Time data. ***0 OBSERVATIONS DROPPED***
tickets <- tickets[!(is.na(tickets$Citation.Issue.Time) | tickets$Citation.Issue.Time ==
    ""), ]
outTime <- strsplit(as.character(tickets$Citation.Issue.Time), ":")
tickets <- data.frame(tickets, do.call(rbind, outTime))
names(tickets)[names(tickets) == "X1"] <- "Citation.Issue.Hour"
names(tickets)[names(tickets) == "X2"] <- "Citation.Issue.Minute"
tickets$Citation.Issue.Hour <- as.numeric(as.character(tickets$Citation.Issue.Hour))
tickets$Citation.Issue.Minute <- NULL
rm(outTime)

# Create column from Month-Year
tickets <- within(tickets, Citation.Issue.MonthYear <- paste(Citation.Issue.Month,
    "-", Citation.Issue.Year, sep = ""))

# Drop 'Unapplied' and empty/NA observations within Citation.Status.
# Unapplied result is outside the scope of this project. ***84 OBSERVATIONS
# DROPPED***
tickets <- tickets[!(is.na(tickets$Citation.Status) | tickets$Citation.Status ==
    "" | tickets$Citation.Status == "Unapplied"), ]

levels(tickets$Citation.Status) <- list(Open = "Open", Closed = "Closed")

# Drop extraneous variables: Object.ID (just row index), Location (Not need
# street address b/c we have latitude and longitude), Last.Edited.Date
# (empty column), Suspend.Code (NA), Disposition.Code (NA), Violation.Code
# (have this info in Violation variable), Citation.Issue.Time (Not needed
# for analysis)
tickets$Object.ID <- NULL
tickets$Location <- NULL
tickets$Last.Edited.Date <- NULL
tickets$Suspend.Code <- NULL
tickets$Disposition.Code <- NULL
```

```
tickets$Violation.Code <- NULL
tickets$Citation.Issue.Time <- NULL

# Drop 'NO VIOL' observations in Violation variable. ***1 OBSERVATIONS
# DROPPED***
tickets <- tickets[!(tickets$Violation == "NO VIOL"), ]

# Create neighborhood column using Lat/Long and Census Block Data from shp
# file Thanks to Shekeine for instructions:
# http://stackoverflow.com/questions/29872109/binning-longitude-latitude-labeled-data-by-census-block-i
# Load Census tract data for San Fransisco tractSF <- shapefile('Analysis
# Neighborhoods - 2010 census tracts assigned to
# neighborhoods/geo_export_72577790-b6d5-439f-9b51-edc69b942ac6.shp')
```

Dropped Observations

- We dropped all observations with no Geom data
  - This was only 21 of the 17,178 observations
- We dropped all observations with no information on the time of the citation
  - Every observation had information on the time of the citation
- We dropped all observations where citation status was not specified or "unapplied"
  - We determined "Unapplied" citations are outside the scope of this project
  - This only led to 84 observations being dropped
- Dropping extraneous variables did not affect the number of observations in the dataset
- We dropped Violations where the listed violation was determined to be "No Violation"
  - This led to 1 observation being dropped
  - It is worth noting that only 1 violation was deemed not to be a violation

Overall, we dropped 106 of the original 17,178 observations in the dataset. This only accounts for 0.62 percent of the dataset. Thus, we are confident that we did not introduce any inherent biases to the data by dropping the aforementioned observations.

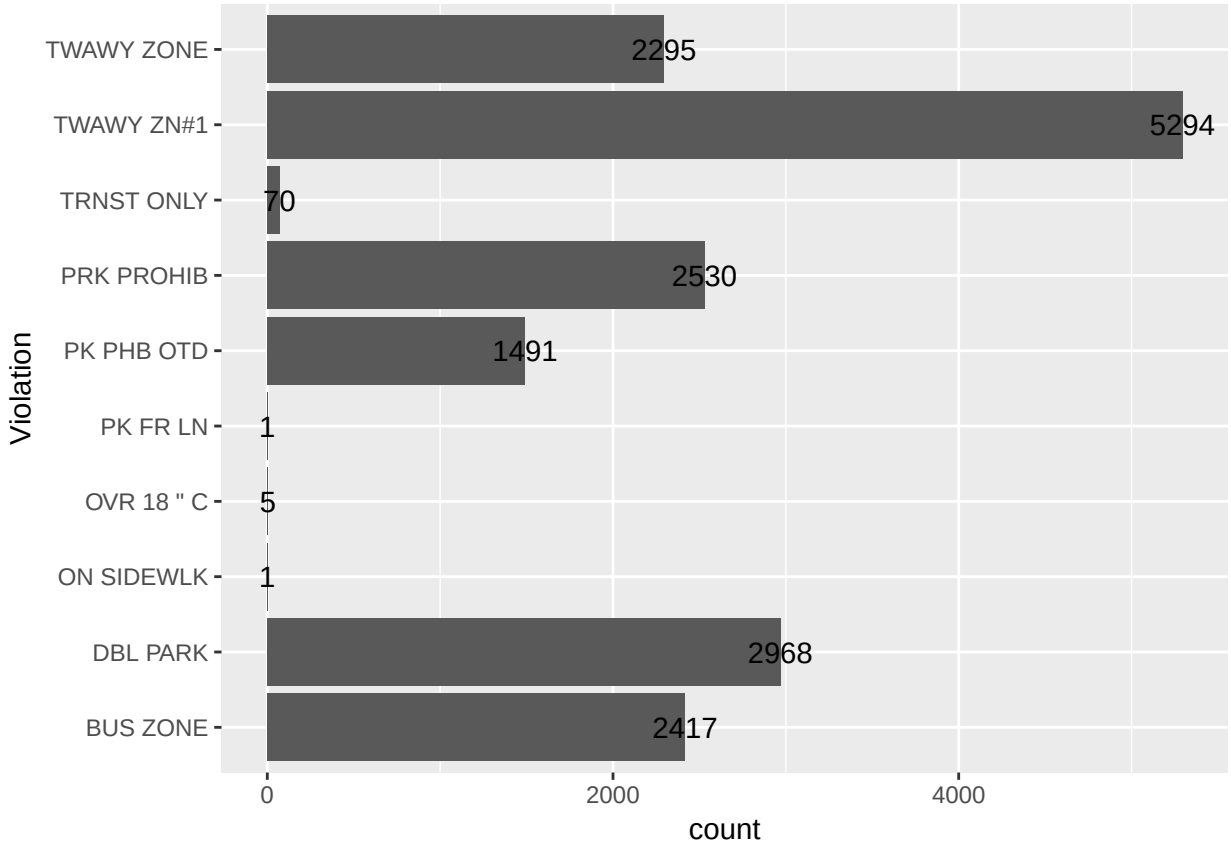
Cleaned Dataset

```
# Check dimensions of cleaned dataset
dim(tickets)
# Check all variable names
colnames(tickets)
# Examine structure of cleaned dataset
str(tickets)
```

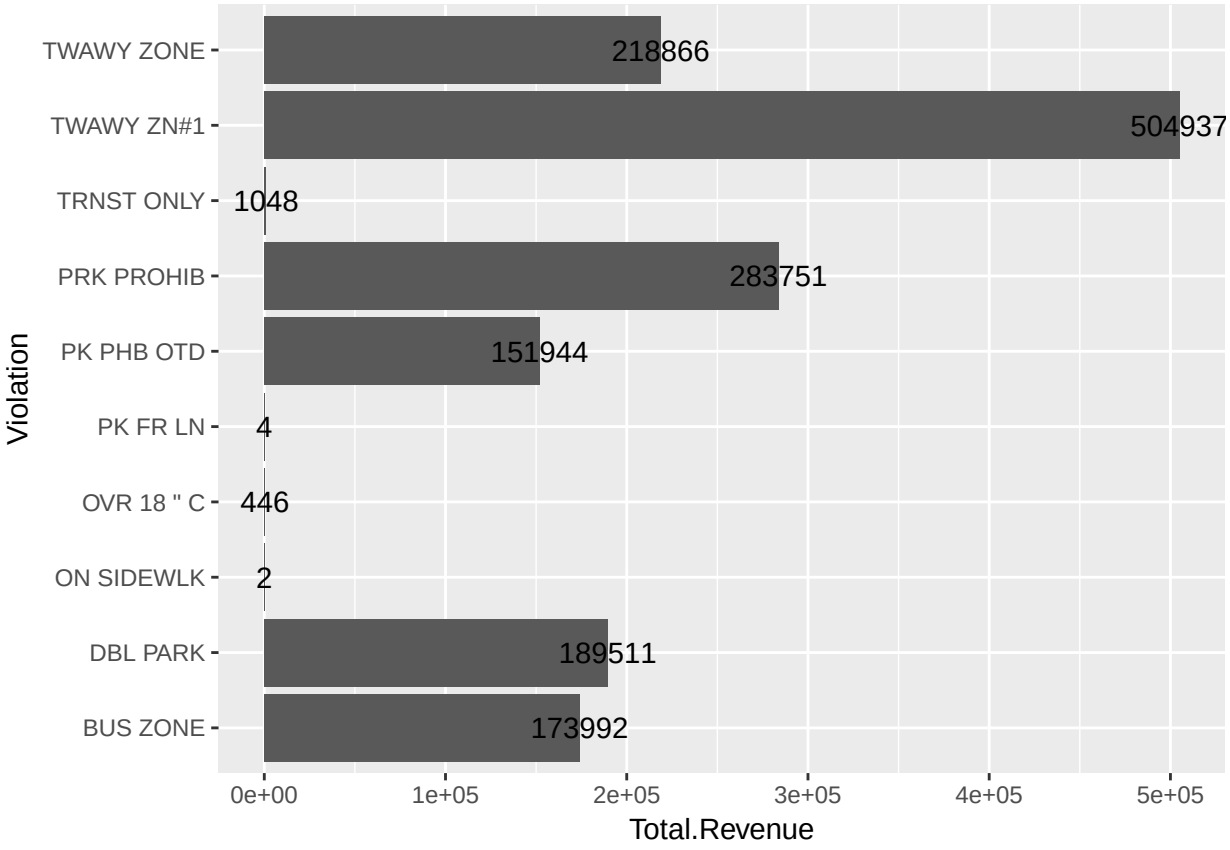
The cleaned dataset that we will be performing the data analysis on has 17072 observations and 19 variables.

Yesh's Stuff

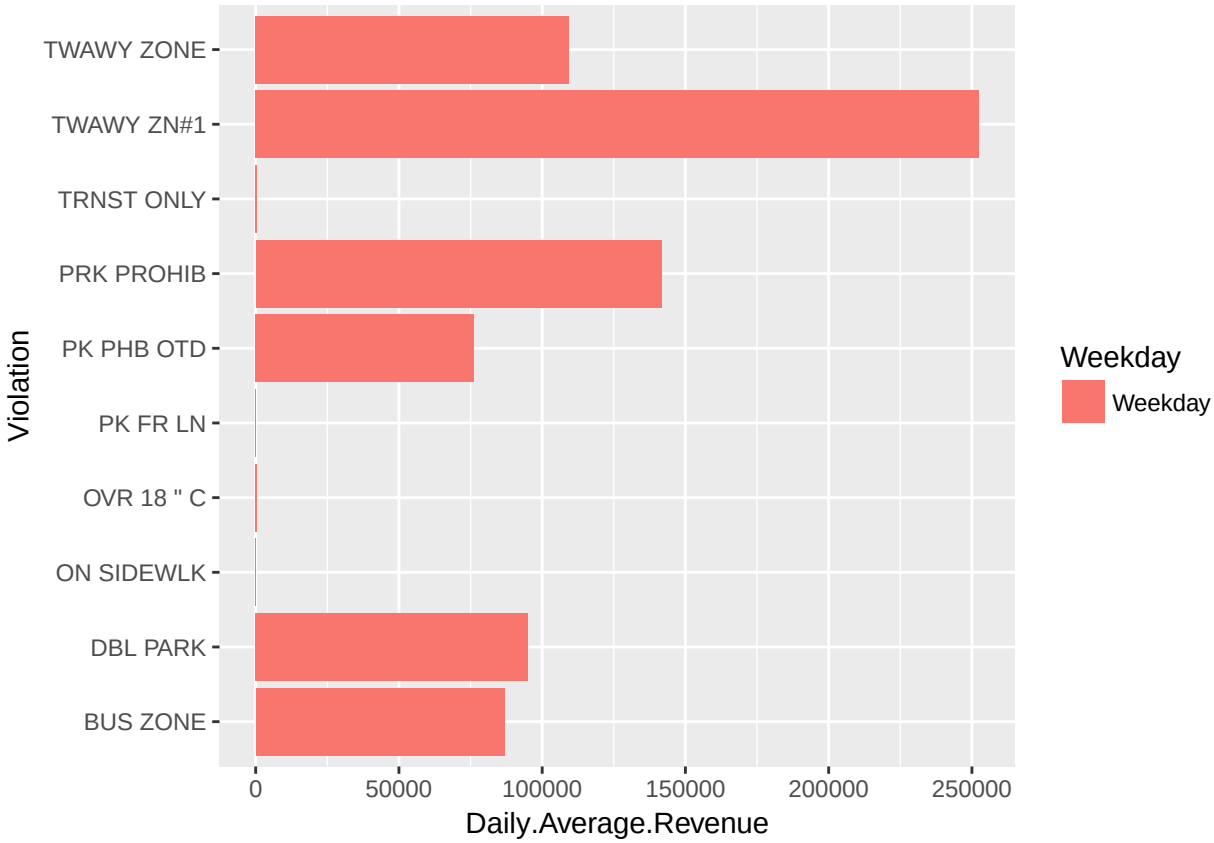
What traffic only lane violations are most common?



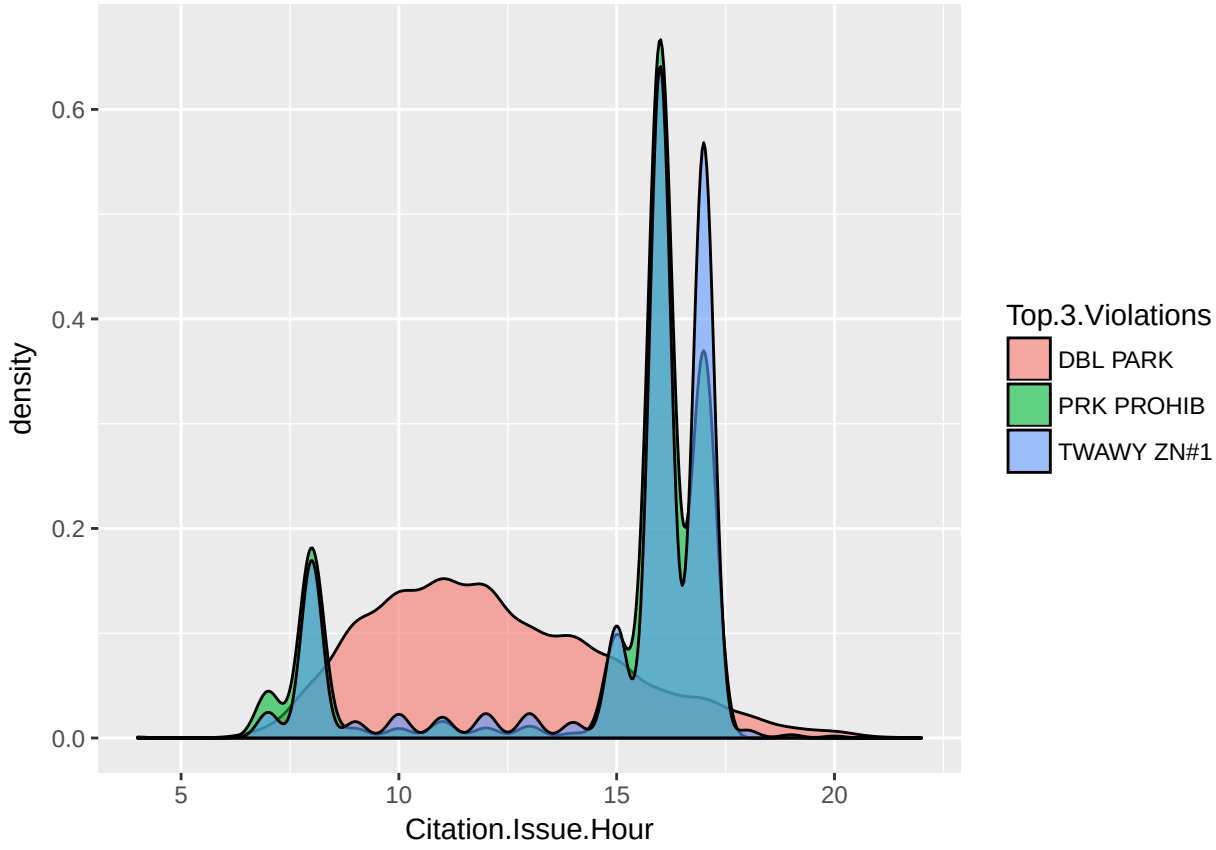
What traffic lane violations make up the most revenue?



What violations produce the most daily revenue on weekends and weekdays?



What times get the most tickets for the top 3 violation types?



Which fees get the most tickets for the top 3 violation types?