Part 1

Kahvi Patel

Loading Libraries

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyvers
e 1.2.1 ──
```

```
## ✔ ggplot2 2.2.1          ✔ purrr   0.3.2
## ✔ tibble  2.1.1          ✔ dplyr   0.8.0.1
## ✔ tidyr   0.8.3.9000     ✔ stringr 1.3.1
## ✔ readr   1.1.1          ✔ forcats 0.3.0
```

```
## Warning: package 'tibble' was built under R version 3.5.2
```

```
## Warning: package 'purrr' was built under R version 3.5.2
```

```
## Warning: package 'dplyr' was built under R version 3.5.2
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conf
licts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(scales)
```

```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:viridis':
##
##     viridis_pal
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

Data Import and Cleaning

```
setwd("~/Desktop/Spring 2019/STAT 123/Final Project")

ds <- read_csv("Data/919report_data.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   Date = col_character(),
##   Time = col_time(format = ""),
##   Involved = col_character(),
##   `Name/Description` = col_character(),
##   Gender = col_character(),
##   Height = col_character(),
##   `Weight (lbs)` = col_character(),
##   `Hair Colour` = col_character(),
##   `Call initiated to: Police (911)` = col_character(),
##   `Call initiated to: Fire (911)` = col_character(),
##   `Vehicle ID` = col_character(),
##   `Incident Teyp(S): Other (Type)` = col_character(),
##   `Other: (Enter)` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
# Functions
clean.up = function(x) {
  if (is.na(x)) {
    x <- FALSE
  } else if (x == 0) {
    x <- FALSE
  } else {
    x <- TRUE
  }
}

# Fix and seperate Time
ds$Time = sapply(strsplit(as.character(ds$Time), " "), tail, n = 1)
ds = ds %>%
  mutate(Time=hms::as.hms(Time))

# Time of Day Column
ds = ds %>%
  mutate(Time_Of_Day =
           ifelse((Time <= 12*60*60 & Time >= 5*60*60), "Morning",
           ifelse((Time <= 17*60*60 & Time > 12*60*60), "Afternoon",
           ifelse((Time < 22*60*60 & Time > 17*60*60), "Evening", "Night"
))))

# Reorder, rename Columns
ds <- ds[c(1,2,3,43,4:42)]
colnames(ds)[17] <- "Attended By: Fire"
colnames(ds)[42] <- "Event Location: Snack Bar"

# Tranforms with semicolon
for (i in c(12:17, 19:26, 30:42)) {
  ds[i] <- apply(ds[i], MARGIN = 1, FUN = clean.up)
  ds[i] <- transform(ifelse(ds[i] == TRUE, paste(str_split(colnames(ds)
[i], ": ")[[1]][2]), NA))
}

# Transforms with no semicolon
for (i in c(11, 27:29)) {
  ds[i] <- apply(ds[i], MARGIN = 1, FUN = clean.up)
  ds[i] <- transform(ifelse(ds[i] == TRUE, paste(colnames(ds)[i]), NA))
}

# Unite transformed Columns
ds <- unite(ds, col = "Incident_Type", matches("Incident"), sep=",")
ds$Incident_Type <- gsub("NA[,]?","", ds$Incident_Type)
ds$Incident_Type <- gsub("^$", NA, ds$Incident_Type)
```

```
ds <- unite(ds, col = "Attended_By", matches("Attended"), sep=",")
ds$Attended_By <- gsub("NA[,]?","", ds$Attended_By)
ds$Attended_By <- gsub("^$", NA, ds$Attended_By)

ds <- unite(ds, col = "Init_Call", matches("Call"), sep=",")
ds$Init_Call <- gsub("NA[,]?","", ds$Init_Call)
ds$Init_Call <- gsub("^$", NA, ds$Init_Call)

ds <- unite(ds, col = "Location", matches("Event"), sep=",")
ds$Location <- gsub("NA[,]?","", ds$Location)
ds$Location <- gsub("^$", NA, ds$Location)

ds = ds %>%
  separate(Incident_Type, "Incident Type", sep = ",", remove = TRUE) %>%
  separate(Attended_By, "Attended By", sep = ",", remove = TRUE) %>%
  separate(Init_Call, "Initial Call", sep = ",", remove = TRUE) %>%
  separate(Location, "Location", sep = ",", remove = TRUE)

ds$Location <- factor(ds$Location)
ds$Location <- fct_infreq(ds$Location)
```

1. Summary statistics

```
summary(ds)
```

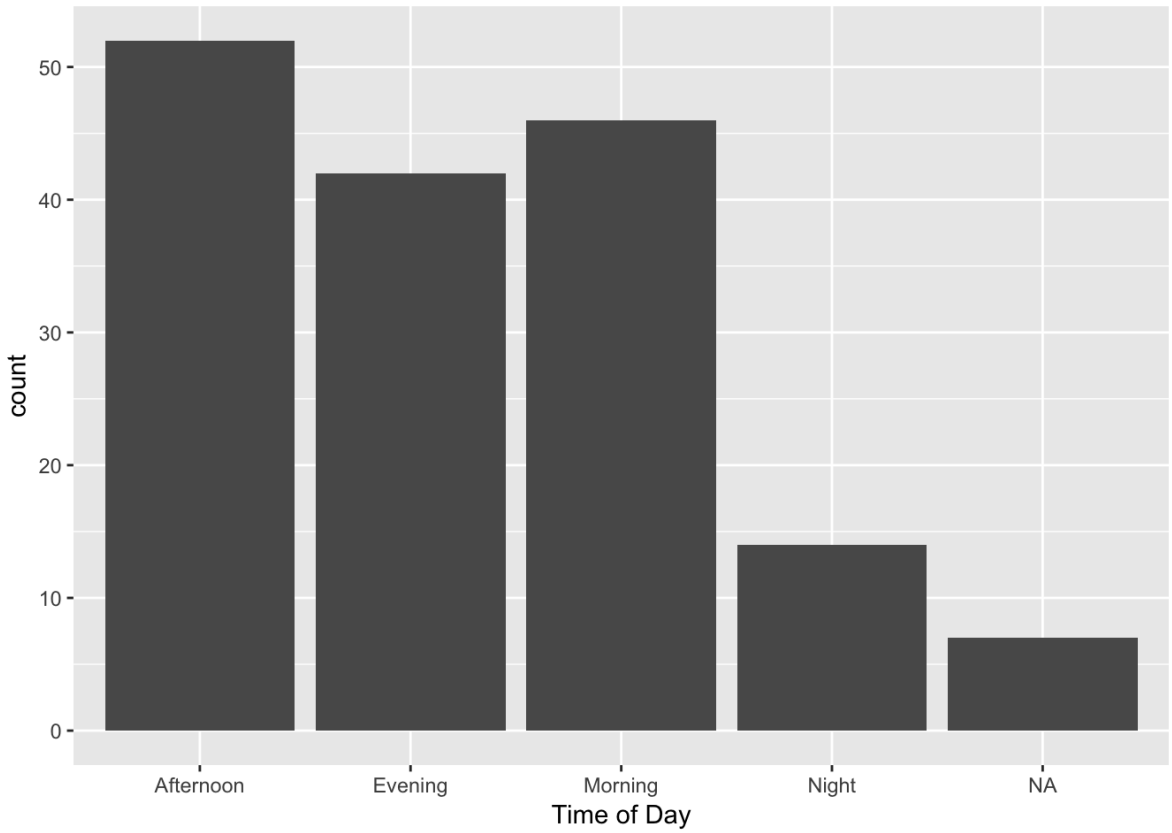
```
##      Case #           Date                Time          Time_Of_Day
##  Min.   :  1.00   Length:161         Length:161        Length:161
##  1st Qu.:  7.00   Class :character   Class1:hms        Class :characte
r
##  Median : 21.00   Mode  :character   Class2:difftime   Mode  :characte
r
##  Mean   : 52.43                      Mode  :numeric
##  3rd Qu.:117.00
##  Max.   :188.00
##
##    Involved         Name/Description      Gender
##  Length:161         Length:161         Length:161
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     Height          Weight (lbs)       Hair Colour
##  Length:161         Length:161         Length:161
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##               Emergency Services Required Initial Call
##  Emergency Services Required: 58          Length:161
##  NA's                        :103         Class :character
##                                           Mode  :character
##
##
##
##
##  Attended By         Vehicle ID        Incident Type
##  Length:161         Length:161         Length:161
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##           First Aid required     Overdose       Naloxone        Locati
on
##  First Aid required: 19      Overdose: 19   Naloxone: 20   Drop In  :5
0
##  NA's              :142      NA's    :142   NA's    :141   Courtyard:1
```

```
8
##                                                                  Perimeter:1
2
##                                                                  Lobby    :1
1
##                                                                  Hygiene  :
8
##                                                                  (Other)  :1
7
##                                                                  NA's     :4
5
##  Other: (Enter)
##  Length:161
##  Class :character
##  Mode  :character
##
##
##
##
```

2. Are there more incidents at the end of the day?

Yes. There are more incidents in the afternoon/evening than in the morning.

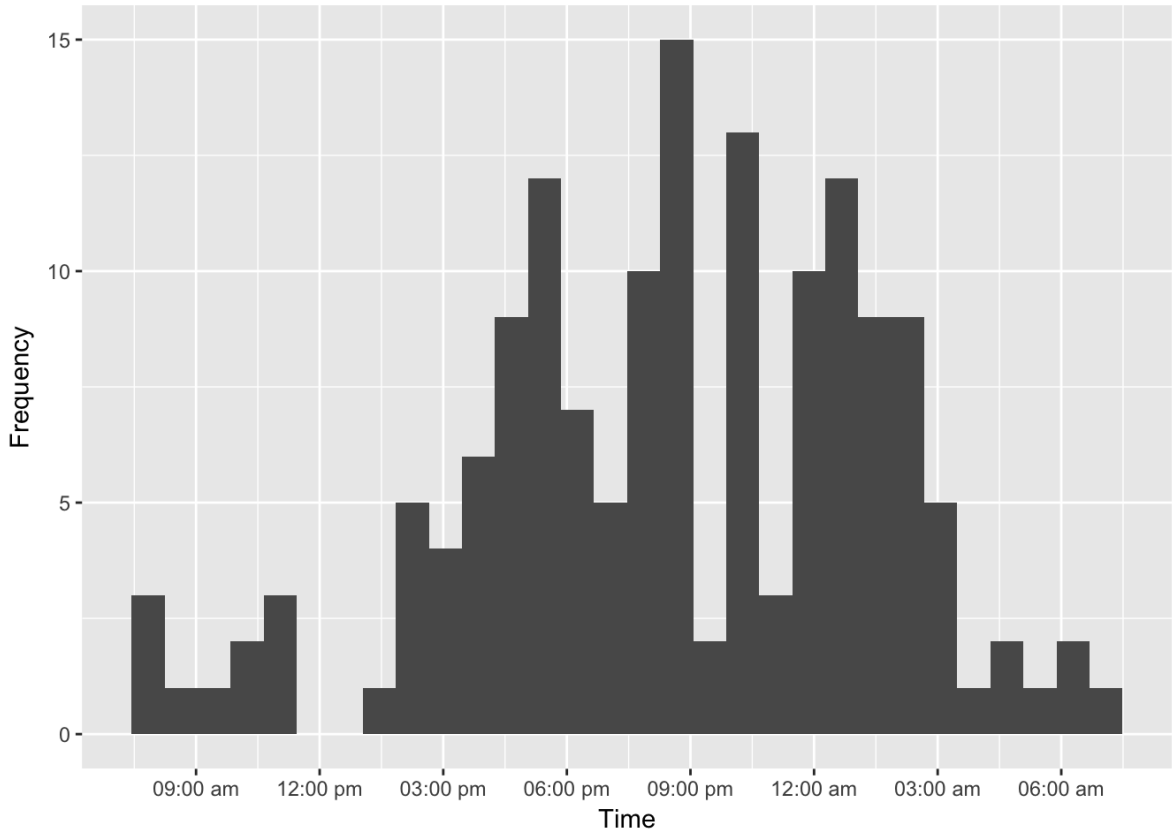
```
ggplot(data.frame(ds$Time_Of_Day), aes(x=ds$Time_Of_Day)) +
  geom_bar() +
  xlab("Time of Day")
```




```
ggplot(data.frame(ds$Time), aes(x=as.POSIXct(strptime(ds$Time, format="%
H:%M:%S")))) +
  geom_histogram() +
  xlab("Time") +
  ylab("Frequency") +
  scale_x_datetime(breaks = date_breaks("3 hour"), labels = date_format(
"%I:%M %p"))
```

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

```
## Warning: Removed 7 rows containing non-finite values (stat_bin).
```



3. Number of Overdose Calls to ambulance

```
overdose_calls <- sum(ds$Overdose == "Overdose", na.rm = TRUE)
print(overdose_calls)
```

```
## [1] 19
```

4. Numbers of calls to first responders

```
calls <- sum(ds$`Emergency Services Required` == "Emergency Services Requ
ired", na.rm = TRUE)
print(calls)
```

```
## [1] 58
```

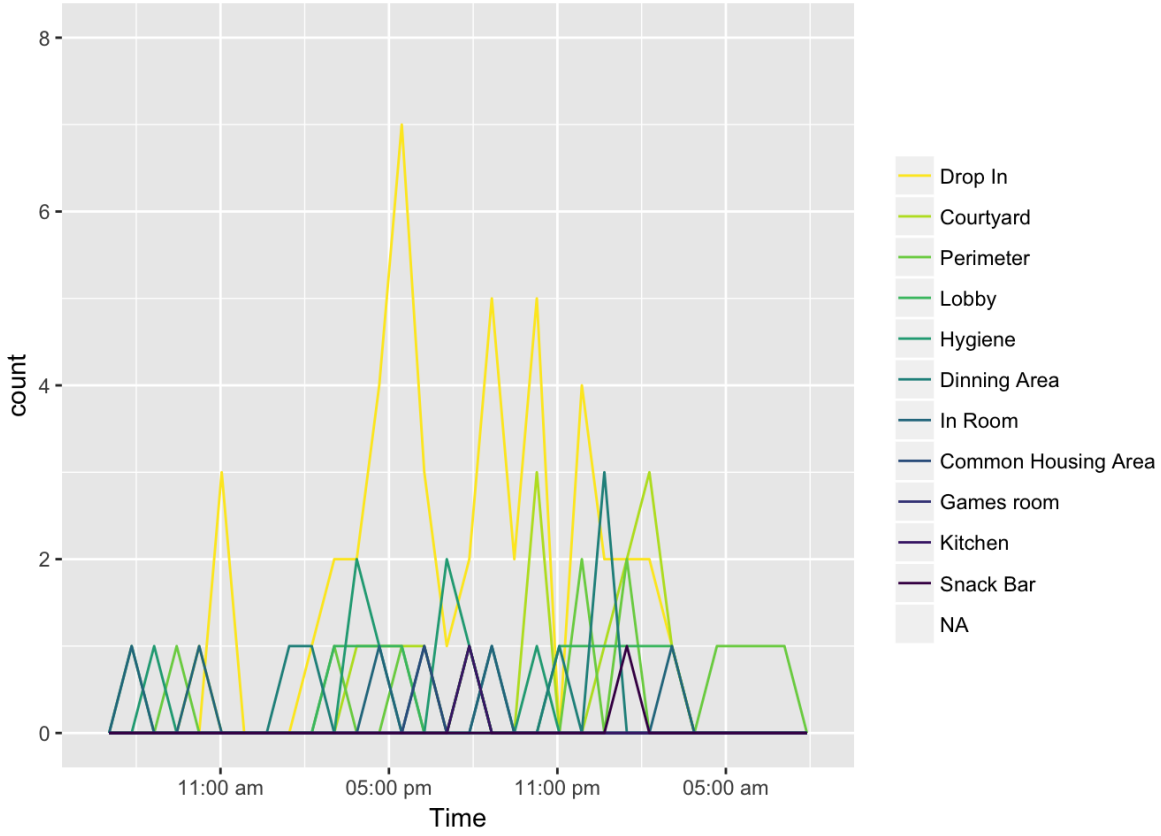
5. Is there a correlation between the time of incident and the site of incident?

It seems like there are more incidents after 5PM when the patient is a Drop-In. There are more incidents in the night when the patient is in the Courtyard. Aside from those two locations, the frequency of incidents in the other locations is moderately constant throughout the day.

```
ggplot(ds, aes(x = as.POSIXct(strptime(ds$Time, format="%H:%M:%S")),
               fill = Location,
               color = factor(Location))) +
  geom_freqpoly() +
  xlab("Time") +
  scale_x_datetime(breaks = date_breaks("6 hour"), labels = date_format(
"%I:%M %p")) +
  scale_color_viridis(discrete = TRUE, "", direction = -1)
```

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

```
## Warning: Removed 7 rows containing non-finite values (stat_bin).
```



```
ggplot(ds, aes(x = as.POSIXct(strptime(ds$Time, format="%H:%M:%S")),
               y=Location,
               color = factor(Location))) +
  geom_point() +
  scale_x_datetime(breaks = date_breaks("3 hour"), labels = date_format(
"%I:%M %p")) +
  xlab("Time")
```

```
## Warning: Removed 7 rows containing missing values (geom_point).
```