

NYC Analyst Exercise

John Perez

3/28/2019

Analysis of NYC 311 Service Requests

Below we will load the relevant libraries and the csv downloaded from NYC 311 website. The CSV was downloaded without any additional filtering besides date range from 1/1/2018 - present.

```
#load libraries
library(ggplot2)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

library(lubridate)

##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##   date

#df <- read.csv("data/311_Service_Requests_from_2010_to_Present.csv", header=TRUE)
df <- read.csv("data/littlefile.csv", header=TRUE)
agencies <- c(as.character(unique(df$Agency)), "ALL") #get list of agencies
```

Exploration

First we'll perform some basic exploration:

For each Agency, how many total complaints between 1/1/2018 and 12/31/2018?

```
#total complaints in 2018
date_range_complaints <- df %>% subset(mdy_hms(Created.Date)>=mdy("1/1/2018") & mdy_hms(Created.Date)<=
  group_by(Agency) %>% summarize(Complaints=n()) %>% arrange(desc(Complaints))

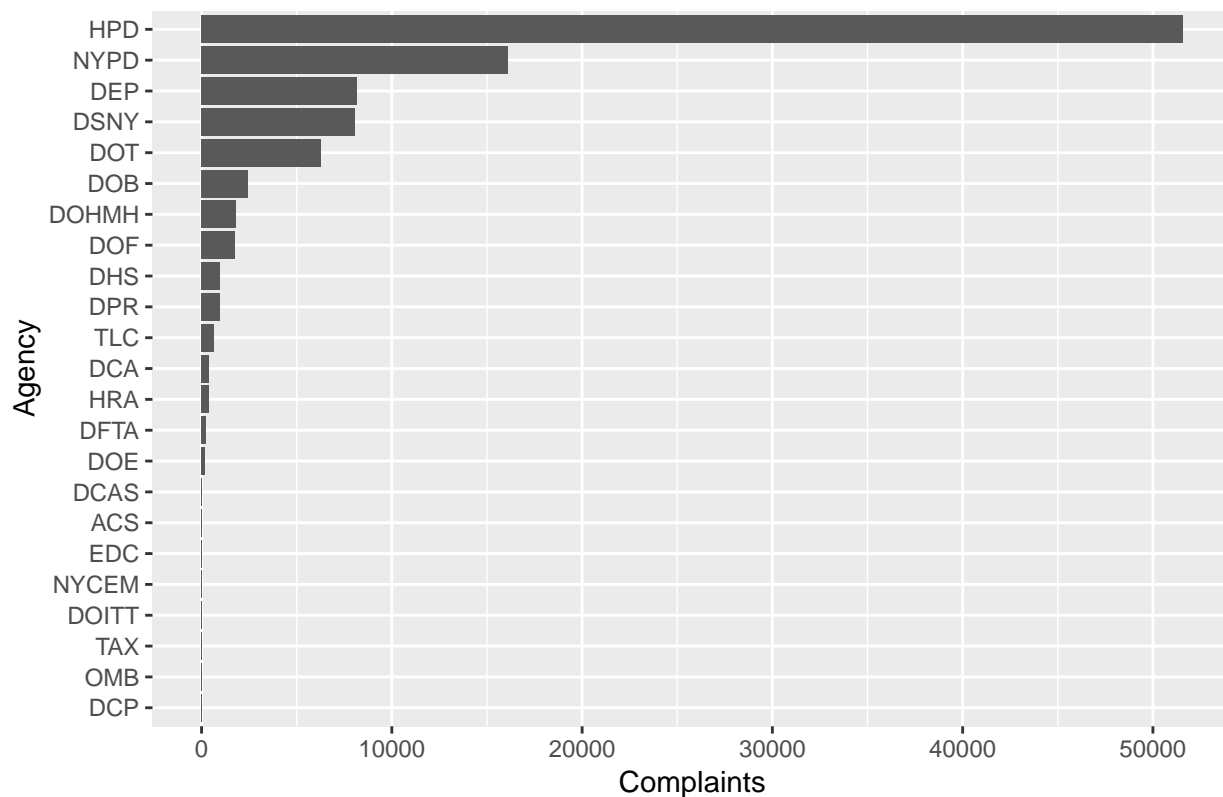
knitr::kable(date_range_complaints)
```

Agency	Complaints
HPD	51583
NYPD	16111
DEP	8163

Agency	Complaints
DSNY	8077
DOT	6267
DOB	2443
DOHMH	1822
DOF	1735
DHS	965
DPR	949
TLC	650
DCA	392
HRA	373
DFTA	223
DOE	187
DCAS	18
ACS	12
EDC	10
NYCEM	8
DOITT	7
TAX	2
DCP	1
OMB	1

```
ggplot(date_range_complaints, aes(x=reorder(Agency, Complaints), y= Complaints)) +
  xlab("Agency") +
  geom_col() +
  coord_flip() +
  labs(title="311 Complaints by Agency")+
  theme(plot.title = element_text(hjust = 0.4))
```

311 Complaints by Agency



```
#complaints breakdown by weekday and hour, by agency between date range
heatmap <- function(dept,start,end){
  startDate <- mdy(start)
  endDate <- mdy(end)

  if(!(dept %in% agencies)){ #simple error handling for incorrect agency entered
    print("Error, please enter a correct agency abbreviation ie: NYPD")
  }

  #Analyze overall data or limited to a specific agency
  if(dept=="ALL"){
    times <- df[,c("Unique.Key","Created.Date")] %>% #keep just complaint ID and date
      mutate(datetime=mdy_hms(Created.Date)) %>% #convert string date to date object
      subset(datetime>=startDate & datetime<=endDate) #keep just those complaints within selected
  } else {
    times <- df[df$Agency==dept,c("Unique.Key","Created.Date")] %>% #keep just complaint ID and date
      mutate(datetime=mdy_hms(Created.Date)) %>% #convert string date to date object
      subset(datetime>=startDate & datetime<=endDate) #keep just those complaints within selected
  }

  times$day <- weekdays(as.Date(times$datetime)) #get the day of the week
  times$hour <- hour(times$datetime) #get the hour of the day
  dayHour <- plyr::ddply(times, c("hour","day"),
    summarise,N=length(datetime)) #create new summary dataframe with counts by day and hour

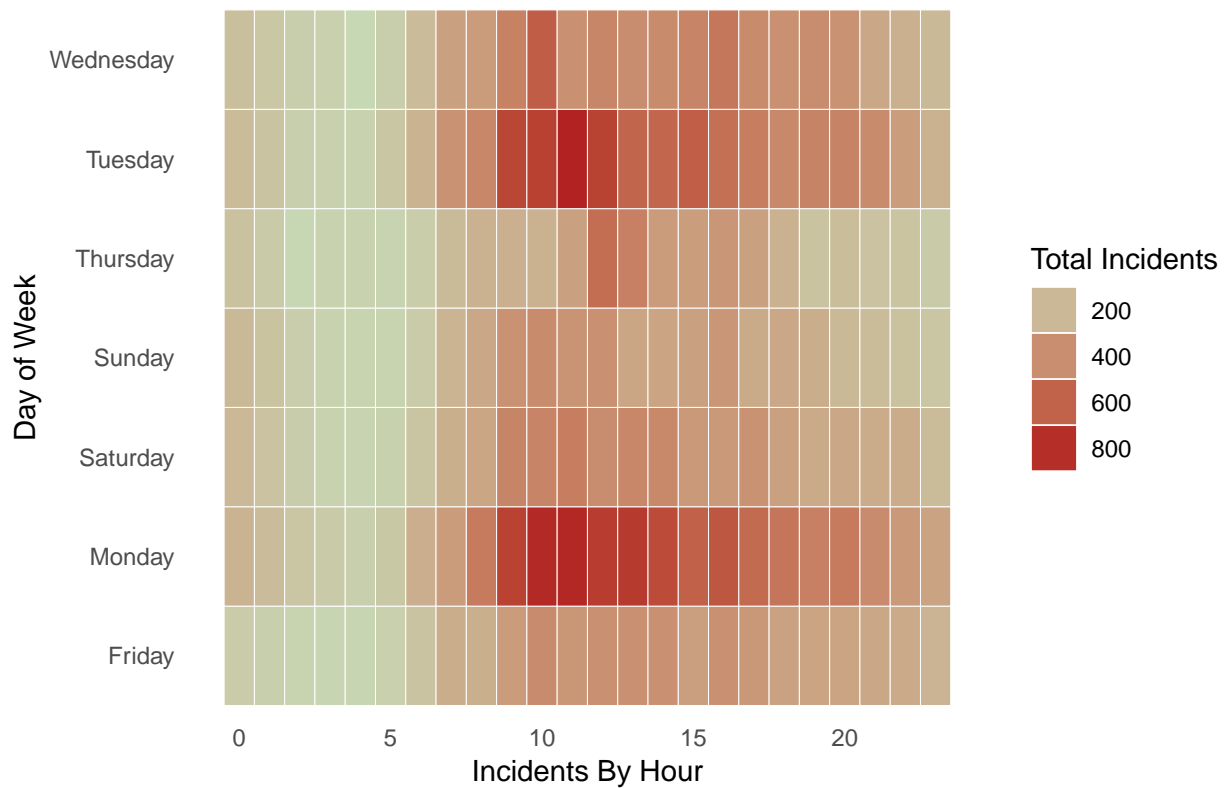
  col1 = "#C7D8B5"
  col2 = "#B22222"
}
```

```

htitle=paste(dept," 311 Complaints by Day between ",start, " - ", end)
ggplot(dayHour, aes(hour, day)) +
  geom_tile(aes(fill = N),colour = "white", na.rm = TRUE) +
  scale_fill_gradient(low = col1, high = col2) +
  guides(fill=guide_legend(title="Total Incidents")) +
  theme_bw() + theme_minimal() +
  labs(title = htitle,
       x = "Incidents By Hour", y = "Day of Week") +
  theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())
}
heatmap("HPD", "1/1/2018", "1/31/2018")

```

HPD 311 Complaints by Day between 1/1/2018 – 1/31/2018



```
#heatmap("HPD", "1/1/2019", "1/31/2019")
```

```

# startDate <- mdy('1/1/2018')
# endDate <- mdy('3/1/2018')
# times <- df[,c("Unique.Key", "Created.Date")] %>%
#   mutate(datetime=mdy_hms(Created.Date))
# times <- subset(times, select=-c(Created.Date))
# times <- subset(times, times$datetime>=startDate & times$datetime<=endDate)
#
# times$day <- weekdays(as.Date(times$datetime))
# times$hour <- hour(times$datetime)
# times$month <- month(times$datetime)
# times$year <- year(times$datetime)
# dayHour <- plyr::ddply(times, c("hour", "day"), summarise,
#   N=length(datetime))

```

```

# #heatmap colors
# col1 = "#C7D8B5"
# col2 = "#438484"
#
# ggplot(dayHour, aes(hour, day)) +
#   geom_tile(aes(fill = N), colour = "white", na.rm = TRUE) +
#   scale_fill_gradient(low = col1, high = col2) +
#   guides(fill=guide_legend(title="Total Incidents")) +
#   theme_bw() + theme_minimal() +
#   labs(title = "Histogram of 311 Complaints by Day of Week and Hour",
#         x = "Incidents Per Hour", y = "Day of Week") +
#   theme(panel.grid.major = element_blank(), panel.grid.minor = element_blank())

```

Call volume by day/hour between 1/1/2019 & 1/31/2019 Histogram of complaint types for NYCHA between 1/1/2019 & 1/31/2019