Homework vi

Sunakshi Sharma

2018-10-12

Contents

NYC 311 Data

The data represents the 311 complaints registered by the citizens. It basically provides all the information regarding the complaint like where did the incident take place, the exact address, latitude, longitude, type of address etc. At this point we are cleaning the data and removing the unnecessary rows or columns from it. Also we are loading another dataset of NYC collision data having something common with our previous dataset and perform the appropriate cleaning in the new data.

```
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
```

Data loading, cleaning and transformation for analysis

Loading the CSV file data-frame

```
nyc.df <- fread(file="./311_Service_Requests_from_2010_to_Present.csv", na.strings=c("","NA"), header=TR
```

```
kable(head(nyc.df[, 1:9]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip
30387854	04/14/2015 02:14:40 AM	04/14/2015 03:03:22 AM	NYPD	New York City Police Department	Vending	In Prohibited Area	Street/Sidewalk	10465
30388338	04/14/2015 02:10:12 AM	NA	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11234
30395236	04/14/2015 02:03:01 AM	NA	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	11204
30394595	04/14/2015 02:02:40 AM	NA	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Talking	Street/Sidewalk	11211
30390517	04/14/2015 02:00:04 AM	04/14/2015 02:47:33 AM	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Talking	Street/Sidewalk	10025
30389560	04/14/2015 01:52:15 AM	04/14/2015 02:11:10 AM	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Talking	Street/Sidewalk	11205

```
kable(head(nyc.df[, 10:18]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

Incident Address	Street Name	Cross Street 1	Cross Street 2	Intersection Street 1	Intersection Street 2	Address Type	City	Landmark
3775 EAST TREMONT AVENUE	EAST TREMONT AVENUE	RANDALL AVENUE	ROOSEVELT AVENUE	NA	NA	ADDRESS	BRONX	NA
1524 RYDER STREET	RYDER STREET	FLATLANDS AVENUE	AVENUE P	NA	NA	ADDRESS	BROOKLYN	NA
NA	NA	NA	NA	71 STREET	16 AVENUE	INTERSECTION	BROOKLYN	NA
361 METROPOLITAN AVENUE	METROPOLITAN AVENUE	HAVEMEYER STREET	HAVEMEYER STREET	NA	NA	ADDRESS	BROOKLYN	NA
NA	NA	NA	NA	WEST 104 STREET	COLUMBUS AVENUE	INTERSECTION	NEW YORK	NA
NA	NA	NA	NA	ST JAMES PLACE	LAFAYETTE AVENUE	INTERSECTION	BROOKLYN	NA

```
kable(head(nyc.df[, 19:29]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

Facility Type	Status	Due Date	Resolution Action Updated Date	Community Board	Borough	X Coordinate (State Plane)	Y Coordinate (State Plane)	Park Facility Name	Park Borough	School Name
Precinct	Closed	04/14/2015 10:14:40 AM	04/14/2015 03:03:05 AM	10 BRONX	BRONX	1033758	240162	Unspecified	BRONX	Unspecified
Precinct	Open	04/14/2015 10:10:12 AM	NA	18 BROOKLYN	BROOKLYN	1001544	164726	Unspecified	BROOKLYN	Unspecified
Precinct	Open	04/14/2015 10:03:01 AM	NA	11 BROOKLYN	BROOKLYN	984678	164647	Unspecified	BROOKLYN	Unspecified
Precinct	Assigned	04/14/2015 10:02:40 AM	04/14/2015 02:10:32 AM	01 BROOKLYN	BROOKLYN	996477	199445	Unspecified	BROOKLYN	Unspecified
Precinct	Closed	04/14/2015 10:00:04 AM	04/14/2015 02:04:59 AM	07 MANHATTAN	MANHATTAN	994260	229982	Unspecified	MANHATTAN	Unspecified
Precinct	Closed	04/14/2015 09:52:15 AM	04/14/2015 02:11:10 AM	02 BROOKLYN	BROOKLYN	994009	190054	Unspecified	BROOKLYN	Unspecified

```
kable(head(nyc.df[, 30:38]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

School Number	School Region	School Code	School Phone Number	School Address	School City	School State	School Zip	School Not Found
Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	N
Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	N
Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	N
Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	N
Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	N
Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	Unspecified	N

```
kable(head(nyc.df[, 39:48]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

School or Citywide Complaint	Vehicle Type	Taxi Company Borough	Taxi Pick Up Location	Bridge Highway Name	Bridge Highway Direction	Road Ramp	Bridge Highway Segment	Garage Lot Name	Ferry Direction
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

```
kable(head(nyc.df[, 49:52]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

Ferry Terminal Name	Latitude	Longitude	Location
NA	40.82573	-73.82111	(40.8257259931145, -73.82111429330192)
NA	40.61879	-73.93771	(40.618794391821936, -73.93770589155426)
NA	40.61859	-73.99846	(40.61859442131066, -73.99845832101916)
NA	40.71410	-73.95589	(40.71409874640673, -73.95589458206499)
NA	40.79792	-73.96385	(40.79791780509379, -73.96384631347463)
NA	40.68833	-73.96481	(40.68832571866554, -73.96481079590191)

Data Summary

```
summary(nyc.df)
```

```
##    Unique Key       Created Date       Closed Date
##  Min.   :     147   Length:9124937     Length:9124937
##  1st Qu.:20215235   Class :character   Class :character
##  Median :24011025   Mode  :character   Mode  :character
##  Mean   :23654275
##  3rd Qu.:27273071
##  Max.   :30395290
##
##     Agency          Agency Name        Complaint Type
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   Descriptor        Location Type      Incident Zip
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Incident Address   Street Name        Cross Street 1
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
```

```
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Cross Street 2     Intersection Street 1 Intersection Street 2
##  Length:9124937     Length:9124937        Length:9124937
##  Class :character   Class :character      Class :character
##  Mode  :character   Mode  :character      Mode  :character
##
##
##
##
##  Address Type           City             Landmark
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Facility Type         Status            Due Date
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Resolution Action Updated Date Community Board      Borough
##  Length:9124937                 Length:9124937     Length:9124937
##  Class :character               Class :character   Class :character
##  Mode  :character               Mode  :character   Mode  :character
##
##
##
##
##  X Coordinate (State Plane) Y Coordinate (State Plane) Park Facility Name
##  Min.   : 913357            Min.   : 121032            Length:9124937
##  1st Qu.: 993610            1st Qu.: 181731            Class :character
##  Median :1004500            Median : 202458            Mode  :character
##  Mean   :1005300            Mean   : 206665
##  3rd Qu.:1017838            3rd Qu.: 235453
##  Max.   :1067302            Max.   :1065949
##  NA's   :831617             NA's   :831617
##  Park Borough       School Name        School Number
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  School Region      School Code        School Phone Number
```

```
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  School Address     School City        School State
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   School Zip        School Not Found   School or Citywide Complaint
##  Length:9124937     Length:9124937     Length:9124937
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Vehicle Type       Taxi Company Borough Taxi Pick Up Location
##  Length:9124937     Length:9124937       Length:9124937
##  Class :character   Class :character     Class :character
##  Mode  :character   Mode  :character     Mode  :character
##
##
##
##
##  Bridge Highway Name Bridge Highway Direction  Road Ramp
##  Length:9124937      Length:9124937           Length:9124937
##  Class :character    Class :character         Class :character
##  Mode  :character    Mode  :character         Mode  :character
##
##
##
##
##  Bridge Highway Segment Garage Lot Name    Ferry Direction
##  Length:9124937         Length:9124937     Length:9124937
##  Class :character       Class :character   Class :character
##  Mode  :character       Mode  :character   Mode  :character
##
##
##
##
##  Ferry Terminal Name    Latitude        Longitude        Location
##  Length:9124937      Min.   :40.5     Min.   :-74.3    Length:9124937
##  Class :character    1st Qu.:40.7     1st Qu.:-74.0    Class :character
##  Mode  :character    Median :40.7     Median :-73.9    Mode  :character
##                      Mean   :40.7     Mean   :-73.9
##                      3rd Qu.:40.8     3rd Qu.:-73.9
##                      Max.   :43.1     Max.   :-73.7
```

```
##                     NA's   :831617   NA's   :831617
```

Glimpse of data

```
glimpse(nyc.df)
```

```
## Observations: 9,124,937
## Variables: 52
## $ `Unique Key`                     <int> 30387854, 30388338, 30395236,...
## $ `Created Date`                   <chr> "04/14/2015 02:14:40 AM", "04...
## $ `Closed Date`                    <chr> "04/14/2015 03:03:22 AM", NA,...
## $ Agency                           <chr> "NYPD", "NYPD", "NYPD", "NYPD...
## $ `Agency Name`                    <chr> "New York City Police Departm...
## $ `Complaint Type`                 <chr> "Vending", "Blocked Driveway"...
## $ Descriptor                       <chr> "In Prohibited Area", "No Acc...
## $ `Location Type`                  <chr> "Street/Sidewalk", "Street/Si...
## $ `Incident Zip`                   <chr> "10465", "11234", "11204", "1...
## $ `Incident Address`               <chr> "3775 EAST TREMONT AVENUE", "...
## $ `Street Name`                    <chr> "EAST TREMONT AVENUE", "RYDER...
## $ `Cross Street 1`                 <chr> "RANDALL AVENUE", "FLATLANDS ...
## $ `Cross Street 2`                 <chr> "ROOSEVELT AVENUE", "AVENUE P...
## $ `Intersection Street 1`          <chr> NA, NA, "71 STREET", NA, "WES...
## $ `Intersection Street 2`          <chr> NA, NA, "16 AVENUE", NA, "COL...
## $ `Address Type`                   <chr> "ADDRESS", "ADDRESS", "INTERS...
## $ City                             <chr> "BRONX", "BROOKLYN", "BROOKLY...
## $ Landmark                         <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Facility Type`                  <chr> "Precinct", "Precinct", "Prec...
## $ Status                           <chr> "Closed", "Open", "Open", "As...
## $ `Due Date`                       <chr> "04/14/2015 10:14:40 AM", "04...
## $ `Resolution Action Updated Date` <chr> "04/14/2015 03:03:05 AM", NA,...
## $ `Community Board`                <chr> "10 BRONX", "18 BROOKLYN", "1...
## $ Borough                          <chr> "BRONX", "BROOKLYN", "BROOKLY...
## $ `X Coordinate (State Plane)`     <int> 1033758, 1001544, 984678, 996...
## $ `Y Coordinate (State Plane)`     <int> 240162, 164726, 164647, 19944...
## $ `Park Facility Name`             <chr> "Unspecified", "Unspecified",...
## $ `Park Borough`                   <chr> "BRONX", "BROOKLYN", "BROOKLY...
## $ `School Name`                    <chr> "Unspecified", "Unspecified",...
## $ `School Number`                  <chr> "Unspecified", "Unspecified",...
## $ `School Region`                  <chr> "Unspecified", "Unspecified",...
## $ `School Code`                    <chr> "Unspecified", "Unspecified",...
## $ `School Phone Number`            <chr> "Unspecified", "Unspecified",...
## $ `School Address`                 <chr> "Unspecified", "Unspecified",...
## $ `School City`                    <chr> "Unspecified", "Unspecified",...
## $ `School State`                   <chr> "Unspecified", "Unspecified",...
## $ `School Zip`                     <chr> "Unspecified", "Unspecified",...
## $ `School Not Found`               <chr> "N", "N", "N", "N", "N", "N",...
## $ `School or Citywide Complaint`   <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Vehicle Type`                   <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Taxi Company Borough`           <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Taxi Pick Up Location`          <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Bridge Highway Name`            <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Bridge Highway Direction`       <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Road Ramp`                      <chr> NA, NA, NA, NA, NA, NA, NA, N...
```

```
## $ `Bridge Highway Segment`         <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Garage Lot Name`                <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Ferry Direction`                <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ `Ferry Terminal Name`            <chr> NA, NA, NA, NA, NA, NA, NA, N...
## $ Latitude                         <dbl> 40.82573, 40.61879, 40.61859,...
## $ Longitude                        <dbl> -73.82111, -73.93771, -73.998...
## $ Location                         <chr> "(40.8257259931145, -73.82111...
```

```
colnames(nyc.df)
```

```
##  [1] "Unique Key"                    "Created Date"
##  [3] "Closed Date"                   "Agency"
##  [5] "Agency Name"                   "Complaint Type"
##  [7] "Descriptor"                    "Location Type"
##  [9] "Incident Zip"                  "Incident Address"
## [11] "Street Name"                   "Cross Street 1"
## [13] "Cross Street 2"                "Intersection Street 1"
## [15] "Intersection Street 2"         "Address Type"
## [17] "City"                          "Landmark"
## [19] "Facility Type"                 "Status"
## [21] "Due Date"                      "Resolution Action Updated Date"
## [23] "Community Board"               "Borough"
## [25] "X Coordinate (State Plane)"    "Y Coordinate (State Plane)"
## [27] "Park Facility Name"            "Park Borough"
## [29] "School Name"                   "School Number"
## [31] "School Region"                 "School Code"
## [33] "School Phone Number"           "School Address"
## [35] "School City"                   "School State"
## [37] "School Zip"                    "School Not Found"
## [39] "School or Citywide Complaint"  "Vehicle Type"
## [41] "Taxi Company Borough"          "Taxi Pick Up Location"
## [43] "Bridge Highway Name"           "Bridge Highway Direction"
## [45] "Road Ramp"                     "Bridge Highway Segment"
## [47] "Garage Lot Name"               "Ferry Direction"
## [49] "Ferry Terminal Name"           "Latitude"
## [51] "Longitude"                     "Location"
```

Lets first check whether there are duplicate rows based on the column Unique Key. We can see that in the original nyc.df data-frame there are 9124937 observations. Whereas when we find the length of column Unique Key based on how many unique rows there are, it comes to be around 9124915. Which means there are around (9124937-9124915) = 22 Observations that are not unique. Removing them will provide us with unique rows!

```
length(unique(nyc.df$`Unique Key`))
```

```
## [1] 9124915
```

Removing observation with duplicate Unique Key. This will remove redundancy from the dataset. And we will be better able to understand it.

```
nyc.df <- nyc.df[!duplicated(nyc.df$`Unique Key`),]
```

Doing some analysis to better understand our data.

Here we are trying to find the columnns and their corresponding counts. We can see there are a few columns where there are a lot of empty, null and na values. We find out using this particular r code. if there is column

which has a lot of empty, null or na values it will have less count and thus deems useless for our purpose.

```
lapply(nyc.df, function(x) length(which(!is.na(x))))
```

```
## $`Unique Key`
## [1] 9124915
##
## $`Created Date`
## [1] 9124915
##
## $`Closed Date`
## [1] 8084368
##
## $Agency
## [1] 9124915
##
## $`Agency Name`
## [1] 9124915
##
## $`Complaint Type`
## [1] 9124915
##
## $Descriptor
## [1] 9122323
##
## $`Location Type`
## [1] 6393568
##
## $`Incident Zip`
## [1] 8334994
##
## $`Incident Address`
## [1] 7158438
##
## $`Street Name`
## [1] 7131980
##
## $`Cross Street 1`
## [1] 7272440
##
## $`Cross Street 2`
## [1] 7228146
##
## $`Intersection Street 1`
## [1] 1637979
##
## $`Intersection Street 2`
## [1] 1637654
##
## $`Address Type`
## [1] 8650528
##
## $City
## [1] 8356386
##
```

```
## $Landmark
## [1] 8465
##
## $`Facility Type`
## [1] 9104172
##
## $Status
## [1] 9124915
##
## $`Due Date`
## [1] 2631722
##
## $`Resolution Action Updated Date`
## [1] 8745781
##
## $`Community Board`
## [1] 9124915
##
## $Borough
## [1] 9124915
##
## $`X Coordinate (State Plane)`
## [1] 8293298
##
## $`Y Coordinate (State Plane)`
## [1] 8293298
##
## $`Park Facility Name`
## [1] 9124915
##
## $`Park Borough`
## [1] 9124915
##
## $`School Name`
## [1] 9124915
##
## $`School Number`
## [1] 9123228
##
## $`School Region`
## [1] 9078721
##
## $`School Code`
## [1] 9078722
##
## $`School Phone Number`
## [1] 9124915
##
## $`School Address`
## [1] 9124911
##
## $`School City`
## [1] 9124915
##
```

```
## $`School State`
## [1] 9124915
##
## $`School Zip`
## [1] 9124915
##
## $`School Not Found`
## [1] 2569034
##
## $`School or Citywide Complaint`
## [1] 3256
##
## $`Vehicle Type`
## [1] 7772
##
## $`Taxi Company Borough`
## [1] 9384
##
## $`Taxi Pick Up Location`
## [1] 90940
##
## $`Bridge Highway Name`
## [1] 32276
##
## $`Bridge Highway Direction`
## [1] 35663
##
## $`Road Ramp`
## [1] 31768
##
## $`Bridge Highway Segment`
## [1] 43585
##
## $`Garage Lot Name`
## [1] 4038
##
## $`Ferry Direction`
## [1] 6384
##
## $`Ferry Terminal Name`
## [1] 13788
##
## $Latitude
## [1] 8293298
##
## $Longitude
## [1] 8293298
##
## $Location
## [1] 8293298
```

Working on Incident Zip

We can see from the analysis we did on overall data finding NA values, we found out that Incident Zip is the only column from one of many important columns having a lot on NA values. Whereas Agency and Borough does not have NA values. We will want to remove rows with NA values for Incident Zip. Lets remove these rows and then will dig further.

```
nyc.df <- nyc.df[!is.na(nyc.df$`Incident Zip`), ]
```

We can see after removing Incident Zip with NA values we are left with 8334994 observations.

Understanding and tidying Borough

Getting the summary

```
summary(nyc.df$Borough)
```

```
##    Length     Class      Mode
##   8334994 character character
```

Getting glimpse

```
glimpse(nyc.df$Borough)
```

```
##  chr [1:8334994] "BRONX" "BROOKLYN" "BROOKLYN" "BROOKLYN" "MANHATTAN" ...
```

Looking at Boroughs to see unique values

```
unique(nyc.df$Borough)
length(unique(nyc.df$Borough))
```

From above we can see that there are total 6 Boroughs, out of these unique 6, one is unspecified. The 'unspecified' might not be of much use to us for our exploratory data analysis. We will remove it.

Removing rows where Borough is unspecified.

```
nyc.df <- nyc.df[!(nyc.df$Borough == "Unspecified"), ]
```

Lets run again and see if "Unspecified" Boroughs have been removed.

```
unique(nyc.df$Borough)
```

```
## [1] "BRONX"         "BROOKLYN"      "MANHATTAN"     "QUEENS"
## [5] "STATEN ISLAND"
```

```
length(unique(nyc.df$Borough))
```

```
## [1] 5
```

We can see the "Unspecified" Boroughs have been removed from the data frame. Now we are left with 7507949 observations.

Now converting Borough to Factor.

```
nyc.df$Borough <- as.factor(nyc.df$Borough)
```

Lets summarize and see whether the Boroughs have been converted to factors?

```
summary(nyc.df$Borough)
```

```
##         BRONX      BROOKLYN     MANHATTAN        QUEENS STATEN ISLAND
##       1373164       2374507       1533429       1805467        421382
```

Understanding and tidying Agency

Getting the summary

```
summary(nyc.df$Agency)
```

```
##    Length     Class      Mode
##   7507949 character character
```

Getting the glimpse

```
glimpse(nyc.df$Agency)
```

```
##  chr [1:7507949] "NYPD" "NYPD" "NYPD" "NYPD" "NYPD" "NYPD" "NYPD" ...
```

Looking at Agency to see unique values

```
unique(nyc.df$Agency)
```

```
##  [1] "NYPD"  "TLC"   "DPR"   "DOT"   "DOHMH" "EDC"   "DCA"   "DSNY"
##  [9] "HPD"   "DEP"   "FDNY"  "DHS"   "DOE"   "DOB"   "DOF"   "DOITT"
## [17] "DFTA"  "3-1-1" "HRA"
```

```
length(unique(nyc.df$Agency))
```

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

As can be seen there are 19 Agencies. There aren't any NA or random values. Agency looks like doesn't need much cleaning! Thankfully!!

Now converting Agency to Factor too!

```
nyc.df$Agency <- as.factor(nyc.df$Agency)
```

Lets summarize and see whether the Agencies have been converted to factors?

```
summary(nyc.df$Agency)
```

```
##   3-1-1     DCA     DEP    DFTA     DHS     DOB     DOE     DOF   DOHMH
##   19891   99917  792783    1623    3152  465124   11007   18213  262690
##   DOITT     DOT     DPR    DSNY     EDC    FDNY     HPD     HRA    NYPD
##    4663 1118420  402332  664230    4749   23784 2477621       1 1022346
##     TLC
##  115403
```

Understanding and tidying City

Getting the summary

```
summary(nyc.df$City)
```

```
##    Length     Class      Mode
##   7507949 character character
```

Getting the glimpse

```
glimpse(nyc.df$City)
```

```
##  chr [1:7507949] "BRONX" "BROOKLYN" "BROOKLYN" "BROOKLYN" "NEW YORK" ...
```

Looking at City to see unique values

```
unique(nyc.df$City)
```

```
##   [1] "BRONX"                 "BROOKLYN"
##   [3] "NEW YORK"              "RIDGEWOOD"
##   [5] "STATEN ISLAND"         "LITTLE NECK"
##   [7] "KEW GARDENS"           "JACKSON HEIGHTS"
##   [9] "ARVERNE"               "CORONA"
##  [11] "WOODSIDE"              "FLUSHING"
##  [13] "MASPETH"               "ASTORIA"
##  [15] "OZONE PARK"            "Hollis"
##  [17] "Cambria Heights"       "FRESH MEADOWS"
##  [19] "Astoria"               "COLLEGE POINT"
##  [21] "South Ozone Park"      "Ridgewood"
##  [23] "EAST ELMHURST"         "Woodside"
##  [25] "Corona"                "ELMHURST"
##  [27] "Maspeth"               "Flushing"
##  [29] "HOLLIS"                "Fresh Meadows"
##  [31] "WHITESTONE"            "Jamaica"
##  [33] "FOREST HILLS"          "WOODHAVEN"
##  [35] "SOUTH OZONE PARK"      "SPRINGFIELD GARDENS"
##  [37] "Bayside"               "Arverne"
##  [39] "SOUTH RICHMOND HILL"   "Forest Hills"
##  [41] "QUEENS VILLAGE"        "Sunnyside"
```

```
##  [43] "MIDDLE VILLAGE"         "JAMAICA"
##  [45] "BAYSIDE"                "Ozone Park"
##  [47] "East Elmhurst"          "Jackson Heights"
##  [49] "Howard Beach"           "REGO PARK"
##  [51] "Whitestone"             "ROSEDALE"
##  [53] "Bellerose"              "Richmond Hill"
##  [55] "RICHMOND HILL"          "Rego Park"
##  [57] "Little Neck"            "Far Rockaway"
##  [59] "Rosedale"               "Queens Village"
##  [61] "Saint Albans"           "ROCKAWAY PARK"
##  [63] "Woodhaven"              "LONG ISLAND CITY"
##  [65] "GLEN OAKS"              "Middle Village"
##  [67] "Elmhurst"               "South Richmond Hill"
##  [69] "BELLEROSE"              "HOWARD BEACH"
##  [71] "Glen Oaks"              "Rockaway Park"
##  [73] "Long Island City"       "BREEZY POINT"
##  [75] "FAR ROCKAWAY"           "College Point"
##  [77] "OAKLAND GARDENS"        "Kew Gardens"
##  [79] "Springfield Gardens"    "SUNNYSIDE"
##  [81] "CAMBRIA HEIGHTS"        "FLORAL PARK"
##  [83] "Oakland Gardens"        "SAINT ALBANS"
##  [85] "NEW HYDE PARK"          "CENTRAL PARK"
##  [87] "New Hyde Park"          "QUEENS"
##  [89] "Floral Park"            "MANHATTAN"
##  [91] NA                       "Breezy Point"
##  [93] "VALLEYSTREAM"           "BELLMORE"
##  [95] "NY"                     "WOOSIDE"
##  [97] "N/A"                    "BRONXS"
##  [99] "BROOKLY"                "LIC"
## [101] "ROSELAND"               "DOUGLASTON"
## [103] "STATEN"                 "BROOKYLN"
## [105] "GLENDALE"               "NEW YORK CITY"
## [107] "BK"                     "NYC"
## [109] "BONX"                   "QNS"
## [111] "QUENNS"                 "UNKNOWN"
## [113] "NEW ROSCHELLE"          "VALLEY STREAM"
## [115] "RICHMOND  HILL"         "LOCAL"
## [117] "BROKLYN"                "ROCKAWAY  BLVD"
## [119] "FOREST HILL"            "LONG CITY"
## [121] "HICKSVILLE"             "BRIARWOOD"
## [123] "LONG ISLAND"            "NEW YOR"
## [125] "INWOOD"                 "QUEEN"
## [127] "GREAT  NECK"            "BRICKTOWN"
## [129] "INDIAN VILLAGE"         "BROOKYN"
## [131] "BKLYN"                  "SOUTH OZONE"
## [133] "CO-OP CITY SECTION 5"   "FLUSING"
## [135] "MANATTAN"               "N/"
## [137] "QUUENS"                 "FRANLIN SQUARE"
## [139] "EIMHURST"               "ST. LOUIS"
## [141] "HERIETTA"               "EASTE ELMHURST"
## [143] "RICHMOND"               "STATE ISLAND"
## [145] "SOUTH OZONE PK"         "SATEN ISLAND"
## [147] "BELLEOSE"               "STATEN  ISLAND"
## [149] "LONG ISLAND  CITY"      "LAURENS"
```

```
## [151] "JAMICA"                "ISLANDIA"
## [153] "MIDDLE VILLIAGE"       "NEWYOK"
## [155] "LAWRENCE"              "BROOLYN"
## [157] "FRESH MEDOWS RD"       "STATEN ISAD"
## [159] "MANHATAN"              "NEW YORK NEW YORK"
## [161] "ROCKAWAY BECH"         "YONKERS"
## [163] "NEW"                   "MIDDLE VILLAGE, QUEENS"
## [165] "STATN ISLAND"          "BROOKILYN"
## [167] "FARMINGDALE"           "FLUSHING MEADOWS"
## [169] "CLERMONT"              "ANONYMOUS"
## [171] "BRONX NY"              "FORESTHILL"
## [173] "ASTOIA"                "BRONXVILLE"
## [175] "UKNOWN"                "VALLYSTREAM"
## [177] "BROX"                  "LITTELE NECK"
## [179] "MAHATTAN"              "BELLROSE"
## [181] "ELMONT"                "FLUHNG"
## [183] "RIDGEWOD"              "QUEEENS"
## [185] "ST.ALBANS"             "CAMBRIA HEIGHS"
## [187] "JAMAICA ESTATE"        "FAR ROCKWA"
## [189] "STATEN ISLAND NY"      "MANHATTEN"
## [191] "GREAT NECK"            "BROOKLN"
## [193] "FLUSHING QUEENS"       "NEWYORK"
## [195] "PARKVILLE STATION"     "ELMHERST"
## [197] "JERICHO"               "BROOLKYN"
## [199] "BX"                    "QUEENS/BROOKLYN"
## [201] "WILLIAMSVILLE"         "STATEN ISLAD"
## [203] "RIVERDALE"             "N.Y"
## [205] "QUENS"                 "CITY ISLAND"
## [207] "BRIGHTON BEACH"        "CROTONA PARK"
## [209] "BAY RIDGE"             "KEW GARDEN"
## [211] "FARMING DALE"          "ST ALBANS"
## [213] "STATEN ISAND"          "JACKSON HEIGHT"
## [215] "RICHMOND COUNTY"       "KE GARDENS HILLS"
## [217] "STATEN ISNALND"        "New York"
## [219] "ROCKAWAY"              "BRRROKLYN"
## [221] "FLUSH MEADOWS"         "NE W HYDE PARK"
## [223] "ELMUNT"                "NEW  YORK"
## [225] "RICHMOND NY"           "NEWYORK,NY"
## [227] "MEMPHIS"               "OZONE"
## [229] "000000"                "SPRINGFEILD GARDENS"
## [231] "RIDGWOOD"              "FORT GREENE"
## [233] "QUEENS, N.Y."          "CARONA"
## [235] "STATEN ISLANE"         "STATTEN ISLAND"
## [237] "TOTTENVILLE"           "STATEN ISALND"
```

```
length(unique(nyc.df$City))
```

```
## [1] 238
```

Removing NA values from City

```
nyc.df <- nyc.df[!is.na(nyc.df$City), ]
```

Converting all Cities to Uppercase. We see that there is no set way

```
nyc.df$City <- toupper(nyc.df$City)
```

Lets see how far we have reached

```
unique(nyc.df$City)
```

```
##   [1] "BRONX"                 "BROOKLYN"
##   [3] "NEW YORK"              "RIDGEWOOD"
##   [5] "STATEN ISLAND"         "LITTLE NECK"
##   [7] "KEW GARDENS"           "JACKSON HEIGHTS"
##   [9] "ARVERNE"               "CORONA"
##  [11] "WOODSIDE"              "FLUSHING"
##  [13] "MASPETH"               "ASTORIA"
##  [15] "OZONE PARK"            "HOLLIS"
##  [17] "CAMBRIA HEIGHTS"       "FRESH MEADOWS"
##  [19] "COLLEGE POINT"         "SOUTH OZONE PARK"
##  [21] "EAST ELMHURST"         "ELMHURST"
##  [23] "WHITESTONE"            "JAMAICA"
##  [25] "FOREST HILLS"          "WOODHAVEN"
##  [27] "SPRINGFIELD GARDENS"   "BAYSIDE"
##  [29] "SOUTH RICHMOND HILL"   "QUEENS VILLAGE"
##  [31] "SUNNYSIDE"             "MIDDLE VILLAGE"
##  [33] "HOWARD BEACH"          "REGO PARK"
##  [35] "ROSEDALE"              "BELLEROSE"
##  [37] "RICHMOND HILL"         "FAR ROCKAWAY"
##  [39] "SAINT ALBANS"          "ROCKAWAY PARK"
##  [41] "LONG ISLAND CITY"      "GLEN OAKS"
##  [43] "BREEZY POINT"          "OAKLAND GARDENS"
##  [45] "FLORAL PARK"           "NEW HYDE PARK"
##  [47] "CENTRAL PARK"          "QUEENS"
##  [49] "MANHATTAN"             "VALLEYSTREAM"
##  [51] "BELLMORE"              "NY"
##  [53] "WOOSIDE"               "N/A"
##  [55] "BRONXS"                "BROOKLY"
##  [57] "LIC"                   "ROSELAND"
##  [59] "DOUGLASTON"            "STATEN"
##  [61] "BROOKYLN"              "GLENDALE"
##  [63] "NEW YORK CITY"         "BK"
##  [65] "NYC"                   "BONX"
##  [67] "QNS"                   "QUENNS"
##  [69] "UNKNOWN"               "NEW ROSCHELLE"
##  [71] "VALLEY STREAM"         "RICHMOND  HILL"
##  [73] "LOCAL"                 "BROKLYN"
##  [75] "ROCKAWAY  BLVD"        "FOREST HILL"
##  [77] "LONG CITY"             "HICKSVILLE"
##  [79] "BRIARWOOD"             "LONG ISLAND"
##  [81] "NEW YOR"               "INWOOD"
##  [83] "QUEEN"                 "GREAT  NECK"
##  [85] "BRICKTOWN"             "INDIAN VILLAGE"
##  [87] "BROOKYN"               "BKLYN"
```

```
##  [89] "SOUTH OZONE"            "CO-OP CITY SECTION 5"
##  [91] "FLUSING"                "MANATTAN"
##  [93] "N/"                     "QUUENS"
##  [95] "FRANLIN SQUARE"         "EIMHURST"
##  [97] "ST. LOUIS"              "HERIETTA"
##  [99] "EASTE ELMHURST"         "RICHMOND"
## [101] "STATE ISLAND"           "SOUTH OZONE PK"
## [103] "SATEN ISLAND"           "BELLEOSE"
## [105] "STATEN  ISLAND"         "LONG ISLAND  CITY"
## [107] "LAURENS"                "JAMICA"
## [109] "ISLANDIA"               "MIDDLE VILLIAGE"
## [111] "NEWYOK"                 "LAWRENCE"
## [113] "BROOLYN"                "FRESH MEDOWS RD"
## [115] "STATEN ISAD"            "MANHATAN"
## [117] "NEW YORK NEW YORK"      "ROCKAWAY BECH"
## [119] "YONKERS"                "NEW"
## [121] "MIDDLE VILLAGE, QUEENS" "STATN ISLAND"
## [123] "BROOKILYN"              "FARMINGDALE"
## [125] "FLUSHING MEADOWS"       "CLERMONT"
## [127] "ANONYMOUS"              "BRONX NY"
## [129] "FORESTHILL"             "ASTOIA"
## [131] "BRONXVILLE"             "UKNOWN"
## [133] "VALLYSTREAM"            "BROX"
## [135] "LITTELE NECK"           "MAHATTAN"
## [137] "BELLROSE"               "ELMONT"
## [139] "FLUHNG"                 "RIDGEWOD"
## [141] "QUEEENS"                "ST.ALBANS"
## [143] "CAMBRIA HEIGHS"         "JAMAICA ESTATE"
## [145] "FAR ROCKWA"             "STATEN ISLAND NY"
## [147] "MANHATTEN"              "GREAT NECK"
## [149] "BROOKLN"                "FLUSHING QUEENS"
## [151] "NEWYORK"                "PARKVILLE STATION"
## [153] "ELMHERST"               "JERICHO"
## [155] "BROOLKYN"               "BX"
## [157] "QUEENS/BROOKLYN"        "WILLIAMSVILLE"
## [159] "STATEN ISLAD"           "RIVERDALE"
## [161] "N.Y"                    "QUENS"
## [163] "CITY ISLAND"            "BRIGHTON BEACH"
## [165] "CROTONA PARK"           "BAY RIDGE"
## [167] "KEW GARDEN"             "FARMING DALE"
## [169] "ST ALBANS"              "STATEN ISAND"
## [171] "JACKSON HEIGHT"         "RICHMOND COUNTY"
## [173] "KE GARDENS HILLS"       "STATEN ISNALND"
## [175] "ROCKAWAY"               "BRRROKLYN"
## [177] "FLUSH MEADOWS"          "NE W HYDE PARK"
## [179] "ELMUNT"                 "NEW  YORK"
## [181] "RICHMOND NY"            "NEWYORK,NY"
## [183] "MEMPHIS"                "OZONE"
## [185] "000000"                 "SPRINGFEILD GARDENS"
## [187] "RIDGWOOD"               "FORT GREENE"
## [189] "QUEENS, N.Y."           "CARONA"
## [191] "STATEN ISLANE"          "STATTEN ISLAND"
## [193] "TOTTENVILLE"            "STATEN ISALND"
```

```
length(unique(nyc.df$City))
```

```
## [1] 194
```

We can see a lot of spelling errors like STATEN ISLAND is written as STATEN ISLANE. Also, there is are values like 000000, N/. These values need to be cleaned up, as they make no sense. We will proceed cleaning up the data.We are keeping Unknown value, as they could be of use for analytical purpose.

```
nyc.df <- nyc.df[!(nyc.df$City == "000000" | nyc.df$City == "N/"), ]
```

Lets order the cities, to understand them better and further correct any issues with them.

```
# Getting Staten Island, and replacing it
nyc.df$City <- sub("^STA.*AND$|^STA.*", "STATEN ISLAND", nyc.df$City)
```

```
unique(nyc.df$City)
```

```
##   [1] "BRONX"                  "BROOKLYN"
##   [3] "NEW YORK"               "RIDGEWOOD"
##   [5] "STATEN ISLAND"          "LITTLE NECK"
##   [7] "KEW GARDENS"            "JACKSON HEIGHTS"
##   [9] "ARVERNE"                "CORONA"
##  [11] "WOODSIDE"               "FLUSHING"
##  [13] "MASPETH"                "ASTORIA"
##  [15] "OZONE PARK"             "HOLLIS"
##  [17] "CAMBRIA HEIGHTS"        "FRESH MEADOWS"
##  [19] "COLLEGE POINT"          "SOUTH OZONE PARK"
##  [21] "EAST ELMHURST"          "ELMHURST"
##  [23] "WHITESTONE"             "JAMAICA"
##  [25] "FOREST HILLS"           "WOODHAVEN"
##  [27] "SPRINGFIELD GARDENS"    "BAYSIDE"
##  [29] "SOUTH RICHMOND HILL"    "QUEENS VILLAGE"
##  [31] "SUNNYSIDE"              "MIDDLE VILLAGE"
##  [33] "HOWARD BEACH"           "REGO PARK"
##  [35] "ROSEDALE"               "BELLEROSE"
##  [37] "RICHMOND HILL"          "FAR ROCKAWAY"
##  [39] "SAINT ALBANS"           "ROCKAWAY PARK"
##  [41] "LONG ISLAND CITY"       "GLEN OAKS"
##  [43] "BREEZY POINT"           "OAKLAND GARDENS"
##  [45] "FLORAL PARK"            "NEW HYDE PARK"
##  [47] "CENTRAL PARK"           "QUEENS"
##  [49] "MANHATTAN"              "VALLEYSTREAM"
##  [51] "BELLMORE"               "NY"
##  [53] "WOOSIDE"                "N/A"
##  [55] "BRONXS"                 "BROOKLY"
##  [57] "LIC"                    "ROSELAND"
##  [59] "DOUGLASTON"             "BROOKYLN"
##  [61] "GLENDALE"               "NEW YORK CITY"
##  [63] "BK"                     "NYC"
##  [65] "BONX"                   "QNS"
##  [67] "QUENNS"                 "UNKNOWN"
##  [69] "NEW ROSCHELLE"          "VALLEY STREAM"
##  [71] "RICHMOND  HILL"         "LOCAL"
##  [73] "BROKLYN"                "ROCKAWAY  BLVD"
##  [75] "FOREST HILL"            "LONG CITY"
##  [77] "HICKSVILLE"             "BRIARWOOD"
```

```
##  [79] "LONG ISLAND"            "NEW YOR"
##  [81] "INWOOD"                 "QUEEN"
##  [83] "GREAT  NECK"            "BRICKTOWN"
##  [85] "INDIAN VILLAGE"         "BROOKYN"
##  [87] "BKLYN"                  "SOUTH OZONE"
##  [89] "CO-OP CITY SECTION 5"   "FLUSING"
##  [91] "MANATTAN"               "QUUENS"
##  [93] "FRANLIN SQUARE"         "EIMHURST"
##  [95] "ST. LOUIS"              "HERIETTA"
##  [97] "EASTE ELMHURST"         "RICHMOND"
##  [99] "SOUTH OZONE PK"         "SATEN ISLAND"
## [101] "BELLEOSE"               "LONG ISLAND  CITY"
## [103] "LAURENS"                "JAMICA"
## [105] "ISLANDIA"               "MIDDLE VILLIAGE"
## [107] "NEWYOK"                 "LAWRENCE"
## [109] "BROOLYN"                "FRESH MEDOWS RD"
## [111] "MANHATAN"               "NEW YORK NEW YORK"
## [113] "ROCKAWAY BECH"          "YONKERS"
## [115] "NEW"                    "MIDDLE VILLAGE, QUEENS"
## [117] "BROOKILYN"              "FARMINGDALE"
## [119] "FLUSHING MEADOWS"       "CLERMONT"
## [121] "ANONYMOUS"              "BRONX NY"
## [123] "FORESTHILL"             "ASTOIA"
## [125] "BRONXVILLE"             "UKNOWN"
## [127] "VALLYSTREAM"            "BROX"
## [129] "LITTELE NECK"           "MAHATTAN"
## [131] "BELLROSE"               "ELMONT"
## [133] "FLUHNG"                 "RIDGEWOD"
## [135] "QUEEENS"                "ST.ALBANS"
## [137] "CAMBRIA HEIGHS"         "JAMAICA ESTATE"
## [139] "FAR ROCKWA"             "MANHATTEN"
## [141] "GREAT NECK"             "BROOKLN"
## [143] "FLUSHING QUEENS"        "NEWYORK"
## [145] "PARKVILLE STATION"      "ELMHERST"
## [147] "JERICHO"                "BROOLKYN"
## [149] "BX"                     "QUEENS/BROOKLYN"
## [151] "WILLIAMSVILLE"          "RIVERDALE"
## [153] "N.Y"                    "QUENS"
## [155] "CITY ISLAND"            "BRIGHTON BEACH"
## [157] "CROTONA PARK"           "BAY RIDGE"
## [159] "KEW GARDEN"             "FARMING DALE"
## [161] "ST ALBANS"              "JACKSON HEIGHT"
## [163] "RICHMOND COUNTY"        "KE GARDENS HILLS"
## [165] "ROCKAWAY"               "BRRROKLYN"
## [167] "FLUSH MEADOWS"          "NE W HYDE PARK"
## [169] "ELMUNT"                 "NEW  YORK"
## [171] "RICHMOND NY"            "NEWYORK,NY"
## [173] "MEMPHIS"                "OZONE"
## [175] "SPRINGFEILD GARDENS"    "RIDGWOOD"
## [177] "FORT GREENE"            "QUEENS, N.Y."
## [179] "CARONA"                 "TOTTENVILLE"
```

We can still see, “SATEN ISLAND” as our Regex could not match it. We can convert is manually. Not a good idea though! I hope there isnt a place like SATEN ISLAND!! Also,

```
nyc.df$City[nyc.df$City == "SATEN ISLAND"] <- "STATEN ISLAND"
```

```
unique(nyc.df$City)
```

```
##   [1] "BRONX"                   "BROOKLYN"
##   [3] "NEW YORK"                "RIDGEWOOD"
##   [5] "STATEN ISLAND"           "LITTLE NECK"
##   [7] "KEW GARDENS"             "JACKSON HEIGHTS"
##   [9] "ARVERNE"                 "CORONA"
##  [11] "WOODSIDE"                "FLUSHING"
##  [13] "MASPETH"                 "ASTORIA"
##  [15] "OZONE PARK"              "HOLLIS"
##  [17] "CAMBRIA HEIGHTS"         "FRESH MEADOWS"
##  [19] "COLLEGE POINT"           "SOUTH OZONE PARK"
##  [21] "EAST ELMHURST"           "ELMHURST"
##  [23] "WHITESTONE"              "JAMAICA"
##  [25] "FOREST HILLS"            "WOODHAVEN"
##  [27] "SPRINGFIELD GARDENS"     "BAYSIDE"
##  [29] "SOUTH RICHMOND HILL"     "QUEENS VILLAGE"
##  [31] "SUNNYSIDE"               "MIDDLE VILLAGE"
##  [33] "HOWARD BEACH"            "REGO PARK"
##  [35] "ROSEDALE"                "BELLEROSE"
##  [37] "RICHMOND HILL"           "FAR ROCKAWAY"
##  [39] "SAINT ALBANS"            "ROCKAWAY PARK"
##  [41] "LONG ISLAND CITY"        "GLEN OAKS"
##  [43] "BREEZY POINT"            "OAKLAND GARDENS"
##  [45] "FLORAL PARK"             "NEW HYDE PARK"
##  [47] "CENTRAL PARK"            "QUEENS"
##  [49] "MANHATTAN"               "VALLEYSTREAM"
##  [51] "BELLMORE"                "NY"
##  [53] "WOOSIDE"                 "N/A"
##  [55] "BRONXS"                  "BROOKLY"
##  [57] "LIC"                     "ROSELAND"
##  [59] "DOUGLASTON"              "BROOKYLN"
##  [61] "GLENDALE"                "NEW YORK CITY"
##  [63] "BK"                      "NYC"
##  [65] "BONX"                    "QNS"
##  [67] "QUENNS"                  "UNKNOWN"
##  [69] "NEW ROSCHELLE"           "VALLEY STREAM"
##  [71] "RICHMOND  HILL"          "LOCAL"
##  [73] "BROKLYN"                 "ROCKAWAY  BLVD"
##  [75] "FOREST HILL"             "LONG CITY"
##  [77] "HICKSVILLE"              "BRIARWOOD"
##  [79] "LONG ISLAND"             "NEW YOR"
##  [81] "INWOOD"                  "QUEEN"
##  [83] "GREAT  NECK"             "BRICKTOWN"
##  [85] "INDIAN VILLAGE"          "BROOKYN"
##  [87] "BKLYN"                   "SOUTH OZONE"
##  [89] "CO-OP CITY SECTION 5"    "FLUSING"
##  [91] "MANATTAN"                "QUUENS"
##  [93] "FRANLIN SQUARE"          "EIMHURST"
##  [95] "ST. LOUIS"               "HERIETTA"
##  [97] "EASTE ELMHURST"          "RICHMOND"
##  [99] "SOUTH OZONE PK"          "BELLEOSE"
```

```
## [101] "LONG ISLAND  CITY"      "LAURENS"
## [103] "JAMICA"                 "ISLANDIA"
## [105] "MIDDLE VILLIAGE"        "NEWYOK"
## [107] "LAWRENCE"               "BROOLYN"
## [109] "FRESH MEDOWS RD"        "MANHATAN"
## [111] "NEW YORK NEW YORK"      "ROCKAWAY BECH"
## [113] "YONKERS"                "NEW"
## [115] "MIDDLE VILLAGE, QUEENS" "BROOKILYN"
## [117] "FARMINGDALE"            "FLUSHING MEADOWS"
## [119] "CLERMONT"               "ANONYMOUS"
## [121] "BRONX NY"               "FORESTHILL"
## [123] "ASTOIA"                 "BRONXVILLE"
## [125] "UKNOWN"                 "VALLYSTREAM"
## [127] "BROX"                   "LITTELE NECK"
## [129] "MAHATTAN"               "BELLROSE"
## [131] "ELMONT"                 "FLUHNG"
## [133] "RIDGEWOD"               "QUEEENS"
## [135] "ST.ALBANS"              "CAMBRIA HEIGHS"
## [137] "JAMAICA ESTATE"         "FAR ROCKWA"
## [139] "MANHATTEN"              "GREAT NECK"
## [141] "BROOKLN"                "FLUSHING QUEENS"
## [143] "NEWYORK"                "PARKVILLE STATION"
## [145] "ELMHERST"               "JERICHO"
## [147] "BROOLKYN"               "BX"
## [149] "QUEENS/BROOKLYN"        "WILLIAMSVILLE"
## [151] "RIVERDALE"              "N.Y"
## [153] "QUENS"                  "CITY ISLAND"
## [155] "BRIGHTON BEACH"         "CROTONA PARK"
## [157] "BAY RIDGE"              "KEW GARDEN"
## [159] "FARMING DALE"           "ST ALBANS"
## [161] "JACKSON HEIGHT"         "RICHMOND COUNTY"
## [163] "KE GARDENS HILLS"       "ROCKAWAY"
## [165] "BRRROKLYN"              "FLUSH MEADOWS"
## [167] "NE W HYDE PARK"         "ELMUNT"
## [169] "NEW  YORK"              "RICHMOND NY"
## [171] "NEWYORK,NY"             "MEMPHIS"
## [173] "OZONE"                  "SPRINGFEILD GARDENS"
## [175] "RIDGWOOD"               "FORT GREENE"
## [177] "QUEENS, N.Y."           "CARONA"
## [179] "TOTTENVILLE"
```

Now lets improve it further. We can see that people who are collecting data are not good at typing in correct spellings. Especially looking at QUEENS, I am flabbergasted at the variance of its spellings: "QNS QUENNS QUUENS QUEEENS QUENS"? Anyway, as a thoughtful and a diligent data anlyst, I will take it as a challenge and try to come up with a smart regex! We see all these words start with Q and end with S. Also, looking at the data above, there isnt any other place that starts with Q and ends with S, which further makes it easy for us to tackle the problem! But wait? What about QUEENS, N.Y.? This ends with a dot?

```
nyc.df$City <- sub("^Q.*S$|^QU.*Y\\.", "QUEENS", nyc.df$City)
```

Running to see our changes!

```
unique(nyc.df$City)
```

```
##   [1] "BRONX"                "BROOKLYN"
##   [3] "NEW YORK"             "RIDGEWOOD"
```

```
##   [5] "STATEN ISLAND"          "LITTLE NECK"
##   [7] "KEW GARDENS"            "JACKSON HEIGHTS"
##   [9] "ARVERNE"                "CORONA"
##  [11] "WOODSIDE"               "FLUSHING"
##  [13] "MASPETH"                "ASTORIA"
##  [15] "OZONE PARK"             "HOLLIS"
##  [17] "CAMBRIA HEIGHTS"        "FRESH MEADOWS"
##  [19] "COLLEGE POINT"          "SOUTH OZONE PARK"
##  [21] "EAST ELMHURST"          "ELMHURST"
##  [23] "WHITESTONE"             "JAMAICA"
##  [25] "FOREST HILLS"           "WOODHAVEN"
##  [27] "SPRINGFIELD GARDENS"    "BAYSIDE"
##  [29] "SOUTH RICHMOND HILL"    "QUEENS VILLAGE"
##  [31] "SUNNYSIDE"              "MIDDLE VILLAGE"
##  [33] "HOWARD BEACH"           "REGO PARK"
##  [35] "ROSEDALE"               "BELLEROSE"
##  [37] "RICHMOND HILL"          "FAR ROCKAWAY"
##  [39] "SAINT ALBANS"           "ROCKAWAY PARK"
##  [41] "LONG ISLAND CITY"       "GLEN OAKS"
##  [43] "BREEZY POINT"           "OAKLAND GARDENS"
##  [45] "FLORAL PARK"            "NEW HYDE PARK"
##  [47] "CENTRAL PARK"           "QUEENS"
##  [49] "MANHATTAN"              "VALLEYSTREAM"
##  [51] "BELLMORE"               "NY"
##  [53] "WOOSIDE"                "N/A"
##  [55] "BRONXS"                 "BROOKLY"
##  [57] "LIC"                    "ROSELAND"
##  [59] "DOUGLASTON"             "BROOKYLN"
##  [61] "GLENDALE"               "NEW YORK CITY"
##  [63] "BK"                     "NYC"
##  [65] "BONX"                   "UNKNOWN"
##  [67] "NEW ROSCHELLE"          "VALLEY STREAM"
##  [69] "RICHMOND  HILL"         "LOCAL"
##  [71] "BROKLYN"                "ROCKAWAY  BLVD"
##  [73] "FOREST HILL"            "LONG CITY"
##  [75] "HICKSVILLE"             "BRIARWOOD"
##  [77] "LONG ISLAND"            "NEW YOR"
##  [79] "INWOOD"                 "QUEEN"
##  [81] "GREAT  NECK"            "BRICKTOWN"
##  [83] "INDIAN VILLAGE"         "BROOKYN"
##  [85] "BKLYN"                  "SOUTH OZONE"
##  [87] "CO-OP CITY SECTION 5"   "FLUSING"
##  [89] "MANATTAN"               "FRANLIN SQUARE"
##  [91] "EIMHURST"               "ST. LOUIS"
##  [93] "HERIETTA"               "EASTE ELMHURST"
##  [95] "RICHMOND"               "SOUTH OZONE PK"
##  [97] "BELLEOSE"               "LONG ISLAND  CITY"
##  [99] "LAURENS"                "JAMICA"
## [101] "ISLANDIA"               "MIDDLE VILLIAGE"
## [103] "NEWYOK"                 "LAWRENCE"
## [105] "BROOLYN"                "FRESH MEDOWS RD"
## [107] "MANHATAN"               "NEW YORK NEW YORK"
## [109] "ROCKAWAY BECH"          "YONKERS"
## [111] "NEW"                    "MIDDLE VILLAGE, QUEENS"
```

```
## [113] "BROOKILYN"              "FARMINGDALE"
## [115] "FLUSHING MEADOWS"       "CLERMONT"
## [117] "ANONYMOUS"              "BRONX NY"
## [119] "FORESTHILL"             "ASTOIA"
## [121] "BRONXVILLE"             "UKNOWN"
## [123] "VALLYSTREAM"            "BROX"
## [125] "LITTELE NECK"           "MAHATTAN"
## [127] "BELLROSE"               "ELMONT"
## [129] "FLUHNG"                 "RIDGEWOD"
## [131] "ST.ALBANS"              "CAMBRIA HEIGHS"
## [133] "JAMAICA ESTATE"         "FAR ROCKWA"
## [135] "MANHATTEN"              "GREAT NECK"
## [137] "BROOKLN"                "FLUSHING QUEENS"
## [139] "NEWYORK"                "PARKVILLE STATION"
## [141] "ELMHERST"               "JERICHO"
## [143] "BROOLKYN"               "BX"
## [145] "QUEENS/BROOKLYN"        "WILLIAMSVILLE"
## [147] "RIVERDALE"              "N.Y"
## [149] "CITY ISLAND"            "BRIGHTON BEACH"
## [151] "CROTONA PARK"           "BAY RIDGE"
## [153] "KEW GARDEN"             "FARMING DALE"
## [155] "ST ALBANS"              "JACKSON HEIGHT"
## [157] "RICHMOND COUNTY"        "KE GARDENS HILLS"
## [159] "ROCKAWAY"               "BRRROKLYN"
## [161] "FLUSH MEADOWS"          "NE W HYDE PARK"
## [163] "ELMUNT"                 "NEW  YORK"
## [165] "RICHMOND NY"            "NEWYORK,NY"
## [167] "MEMPHIS"                "OZONE"
## [169] "SPRINGFEILD GARDENS"    "RIDGWOOD"
## [171] "FORT GREENE"            "CARONA"
## [173] "TOTTENVILLE"
```

Wow! There is still QUEEN out there. But we will handle this manually

```
nyc.df$City[nyc.df$City == "QUEEN"] <- "QUEENS"
```

```
unique(nyc.df$City)
```

```
##   [1] "BRONX"                 "BROOKLYN"
##   [3] "NEW YORK"              "RIDGEWOOD"
##   [5] "STATEN ISLAND"         "LITTLE NECK"
##   [7] "KEW GARDENS"           "JACKSON HEIGHTS"
##   [9] "ARVERNE"               "CORONA"
##  [11] "WOODSIDE"              "FLUSHING"
##  [13] "MASPETH"               "ASTORIA"
##  [15] "OZONE PARK"            "HOLLIS"
##  [17] "CAMBRIA HEIGHTS"       "FRESH MEADOWS"
##  [19] "COLLEGE POINT"         "SOUTH OZONE PARK"
##  [21] "EAST ELMHURST"         "ELMHURST"
##  [23] "WHITESTONE"            "JAMAICA"
##  [25] "FOREST HILLS"          "WOODHAVEN"
##  [27] "SPRINGFIELD GARDENS"   "BAYSIDE"
##  [29] "SOUTH RICHMOND HILL"   "QUEENS VILLAGE"
##  [31] "SUNNYSIDE"             "MIDDLE VILLAGE"
##  [33] "HOWARD BEACH"          "REGO PARK"
```

```
##  [35] "ROSEDALE"                "BELLEROSE"
##  [37] "RICHMOND HILL"           "FAR ROCKAWAY"
##  [39] "SAINT ALBANS"            "ROCKAWAY PARK"
##  [41] "LONG ISLAND CITY"        "GLEN OAKS"
##  [43] "BREEZY POINT"            "OAKLAND GARDENS"
##  [45] "FLORAL PARK"             "NEW HYDE PARK"
##  [47] "CENTRAL PARK"            "QUEENS"
##  [49] "MANHATTAN"               "VALLEYSTREAM"
##  [51] "BELLMORE"                "NY"
##  [53] "WOOSIDE"                 "N/A"
##  [55] "BRONXS"                  "BROOKLY"
##  [57] "LIC"                     "ROSELAND"
##  [59] "DOUGLASTON"              "BROOKYLN"
##  [61] "GLENDALE"                "NEW YORK CITY"
##  [63] "BK"                      "NYC"
##  [65] "BONX"                    "UNKNOWN"
##  [67] "NEW ROSCHELLE"           "VALLEY STREAM"
##  [69] "RICHMOND  HILL"          "LOCAL"
##  [71] "BROKLYN"                 "ROCKAWAY  BLVD"
##  [73] "FOREST HILL"             "LONG CITY"
##  [75] "HICKSVILLE"              "BRIARWOOD"
##  [77] "LONG ISLAND"             "NEW YOR"
##  [79] "INWOOD"                  "GREAT  NECK"
##  [81] "BRICKTOWN"               "INDIAN VILLAGE"
##  [83] "BROOKYN"                 "BKLYN"
##  [85] "SOUTH OZONE"             "CO-OP CITY SECTION 5"
##  [87] "FLUSING"                 "MANATTAN"
##  [89] "FRANLIN SQUARE"          "EIMHURST"
##  [91] "ST. LOUIS"               "HERIETTA"
##  [93] "EASTE ELMHURST"          "RICHMOND"
##  [95] "SOUTH OZONE PK"          "BELLEOSE"
##  [97] "LONG ISLAND  CITY"       "LAURENS"
##  [99] "JAMICA"                  "ISLANDIA"
## [101] "MIDDLE VILLIAGE"         "NEWYOK"
## [103] "LAWRENCE"                "BROOLYN"
## [105] "FRESH MEDOWS RD"         "MANHATAN"
## [107] "NEW YORK NEW YORK"       "ROCKAWAY BECH"
## [109] "YONKERS"                 "NEW"
## [111] "MIDDLE VILLAGE, QUEENS" "BROOKILYN"
## [113] "FARMINGDALE"             "FLUSHING MEADOWS"
## [115] "CLERMONT"                "ANONYMOUS"
## [117] "BRONX NY"                "FORESTHILL"
## [119] "ASTOIA"                  "BRONXVILLE"
## [121] "UKNOWN"                  "VALLYSTREAM"
## [123] "BROX"                    "LITTELE NECK"
## [125] "MAHATTAN"                "BELLROSE"
## [127] "ELMONT"                  "FLUHNG"
## [129] "RIDGEWOD"                "ST.ALBANS"
## [131] "CAMBRIA HEIGHS"          "JAMAICA ESTATE"
## [133] "FAR ROCKWA"              "MANHATTEN"
## [135] "GREAT NECK"              "BROOKLN"
## [137] "FLUSHING QUEENS"         "NEWYORK"
## [139] "PARKVILLE STATION"       "ELMHERST"
## [141] "JERICHO"                 "BROOLKYN"
```

```
## [143] "BX"                      "QUEENS/BROOKLYN"
## [145] "WILLIAMSVILLE"           "RIVERDALE"
## [147] "N.Y"                     "CITY ISLAND"
## [149] "BRIGHTON BEACH"          "CROTONA PARK"
## [151] "BAY RIDGE"               "KEW GARDEN"
## [153] "FARMING DALE"            "ST ALBANS"
## [155] "JACKSON HEIGHT"          "RICHMOND COUNTY"
## [157] "KE GARDENS HILLS"        "ROCKAWAY"
## [159] "BRRROKLYN"               "FLUSH MEADOWS"
## [161] "NE W HYDE PARK"          "ELMUNT"
## [163] "NEW  YORK"               "RICHMOND NY"
## [165] "NEWYORK,NY"              "MEMPHIS"
## [167] "OZONE"                   "SPRINGFEILD GARDENS"
## [169] "RIDGWOOD"                "FORT GREENE"
## [171] "CARONA"                  "TOTTENVILLE"
```

Looking carefully at the data, it looks like QUEENS has been resolved. Also, we see BRONX, BROOKLYN has a lot of spelling mistakes. JAMAICA and MANHATTAN on the other hand only has one mistake. We will resolve BRONX, BROOKLYN, MANHATTAN AND JAMAICA respectively.

```
nyc.df$City <- sub("BRONXS|BONX|BRONX NY|BROX", "BRONX", nyc.df$City)
```

```
nyc.df$City <- sub("^B.*N$", "BROOKLYN", nyc.df$City)
```

```
nyc.df$City <- sub("^MAN.*N$", "MANHATTAN", nyc.df$City)
```

```
nyc.df$City[nyc.df$City == "JAMICA"] <- "JAMAICA"
```

```
nyc.df$City <- sub("^NE.*K$|^NEW.*Y", "NEW YORK", nyc.df$City)
```

Finally all the major cities have been improved as far spelling mistakes are concerned! Lets check how the dataframe looks like now!

```
unique(nyc.df$City)
```

```
##   [1] "BRONX"                 "BROOKLYN"
##   [3] "NEW YORK"              "RIDGEWOOD"
##   [5] "STATEN ISLAND"         "LITTLE NECK"
##   [7] "KEW GARDENS"           "JACKSON HEIGHTS"
##   [9] "ARVERNE"               "CORONA"
##  [11] "WOODSIDE"              "FLUSHING"
##  [13] "MASPETH"               "ASTORIA"
##  [15] "OZONE PARK"            "HOLLIS"
##  [17] "CAMBRIA HEIGHTS"       "FRESH MEADOWS"
##  [19] "COLLEGE POINT"         "SOUTH OZONE PARK"
##  [21] "EAST ELMHURST"         "ELMHURST"
##  [23] "WHITESTONE"            "JAMAICA"
##  [25] "FOREST HILLS"          "WOODHAVEN"
##  [27] "SPRINGFIELD GARDENS"   "BAYSIDE"
##  [29] "SOUTH RICHMOND HILL"   "QUEENS VILLAGE"
##  [31] "SUNNYSIDE"             "MIDDLE VILLAGE"
##  [33] "HOWARD BEACH"          "REGO PARK"
##  [35] "ROSEDALE"              "BELLEROSE"
##  [37] "RICHMOND HILL"         "FAR ROCKAWAY"
##  [39] "SAINT ALBANS"          "ROCKAWAY PARK"
##  [41] "LONG ISLAND CITY"      "GLEN OAKS"
##  [43] "BREEZY POINT"          "OAKLAND GARDENS"
```

```
##  [45] "FLORAL PARK"            "CENTRAL PARK"
##  [47] "QUEENS"                 "MANHATTAN"
##  [49] "VALLEYSTREAM"           "BELLMORE"
##  [51] "NY"                     "WOOSIDE"
##  [53] "N/A"                    "BROOKLY"
##  [55] "LIC"                    "ROSELAND"
##  [57] "DOUGLASTON"             "GLENDALE"
##  [59] "BK"                     "NYC"
##  [61] "UNKNOWN"                "NEW ROSCHELLE"
##  [63] "VALLEY STREAM"          "RICHMOND  HILL"
##  [65] "LOCAL"                  "ROCKAWAY  BLVD"
##  [67] "FOREST HILL"            "LONG CITY"
##  [69] "HICKSVILLE"             "BRIARWOOD"
##  [71] "LONG ISLAND"            "NEW YORKOR"
##  [73] "INWOOD"                 "GREAT  NECK"
##  [75] "INDIAN VILLAGE"         "SOUTH OZONE"
##  [77] "CO-OP CITY SECTION 5"   "FLUSING"
##  [79] "FRANLIN SQUARE"         "EIMHURST"
##  [81] "ST. LOUIS"              "HERIETTA"
##  [83] "EASTE ELMHURST"         "RICHMOND"
##  [85] "SOUTH OZONE PK"         "BELLEOSE"
##  [87] "LONG ISLAND  CITY"      "LAURENS"
##  [89] "ISLANDIA"               "MIDDLE VILLIAGE"
##  [91] "LAWRENCE"               "FRESH MEDOWS RD"
##  [93] "ROCKAWAY BECH"          "YONKERS"
##  [95] "NEW"                    "MIDDLE VILLAGE, QUEENS"
##  [97] "FARMINGDALE"            "FLUSHING MEADOWS"
##  [99] "CLERMONT"               "ANONYMOUS"
## [101] "FORESTHILL"             "ASTOIA"
## [103] "BRONXVILLE"             "UKNOWN"
## [105] "VALLYSTREAM"            "LITTELE NECK"
## [107] "MAHATTAN"               "BELLROSE"
## [109] "ELMONT"                 "FLUHNG"
## [111] "RIDGEWOD"               "ST.ALBANS"
## [113] "CAMBRIA HEIGHS"         "JAMAICA ESTATE"
## [115] "FAR ROCKWA"             "GREAT NECK"
## [117] "FLUSHING QUEENS"        "PARKVILLE STATION"
## [119] "ELMHERST"               "JERICHO"
## [121] "BX"                     "QUEENS/BROOKLYN"
## [123] "WILLIAMSVILLE"          "RIVERDALE"
## [125] "N.Y"                    "CITY ISLAND"
## [127] "BRIGHTON BEACH"         "CROTONA PARK"
## [129] "BAY RIDGE"              "KEW GARDEN"
## [131] "FARMING DALE"           "ST ALBANS"
## [133] "JACKSON HEIGHT"         "RICHMOND COUNTY"
## [135] "KE GARDENS HILLS"       "ROCKAWAY"
## [137] "FLUSH MEADOWS"          "ELMUNT"
## [139] "RICHMOND NY"            "MEMPHIS"
## [141] "OZONE"                  "SPRINGFEILD GARDENS"
## [143] "RIDGWOOD"               "FORT GREENE"
## [145] "CARONA"                 "TOTTENVILLE"
```

Converting Dates to Date objects. This will allow us to better manipulate the date and time.

```
nyc.df$`Created Date` = as.POSIXct(nyc.df$`Created Date`, format="%m/%d/%Y %I:%M:%S %p", tz="UTC")
```

```
## Warning in strptime(x, format, tz = tz): unknown timezone 'zone/tz/2018e.
## 1.0/zoneinfo/America/New_York'
```

```
nyc.df$`Closed Date` = as.POSIXct(nyc.df$`Closed Date`, format="%m/%d/%Y %I:%M:%S %p", tz="UTC")
```

Data Exploration

We find the number of complaints by Borough.

We can see that highest number of complaints were filed from Brooklyn.

```
ggplot(data = nyc.df) +
  geom_bar(mapping = aes(x = nyc.df$Borough), fill="orange") + ylab("Count of Complaint") +xlab("Boroug
```

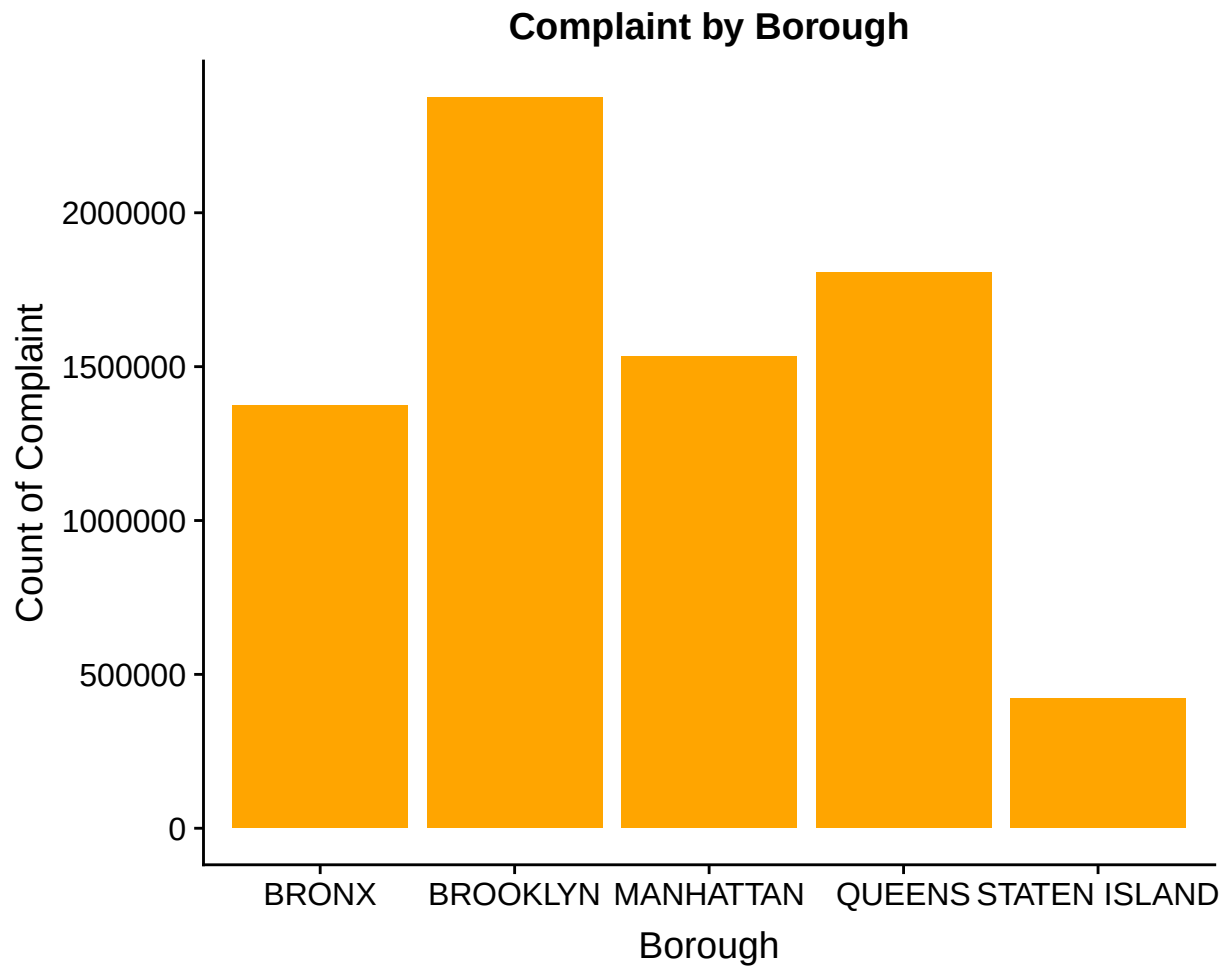


Figure 1: Complaint by Borough

pie chart showing complaint by Status

```
counts_freqs_df <- data.frame(table(nyc.df$Borough))
counts_freqs_df %>%
arrange(desc(counts_freqs_df$Freq)) %>%
mutate(prop = percent(counts_freqs_df$Freq / sum(counts_freqs_df$Freq))) -> counts_freqs_df
```

```
## Warning: package 'bindrcpp' was built under R version 3.4.4
```

```
pie <- ggplot(counts_freqs_df, aes(x = "", y = counts_freqs_df$Freq, fill = fct_inorder(counts_freqs_df$
       geom_bar(width = 1, stat = "identity") +
       scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
       coord_polar("y", start = 0) +
       geom_label_repel(aes(label = prop), size=3, show.legend = F, nudge_y = 20) +
       guides(fill = guide_legend(title = "Boroughs")) + ylab("Frequency")
pie
```

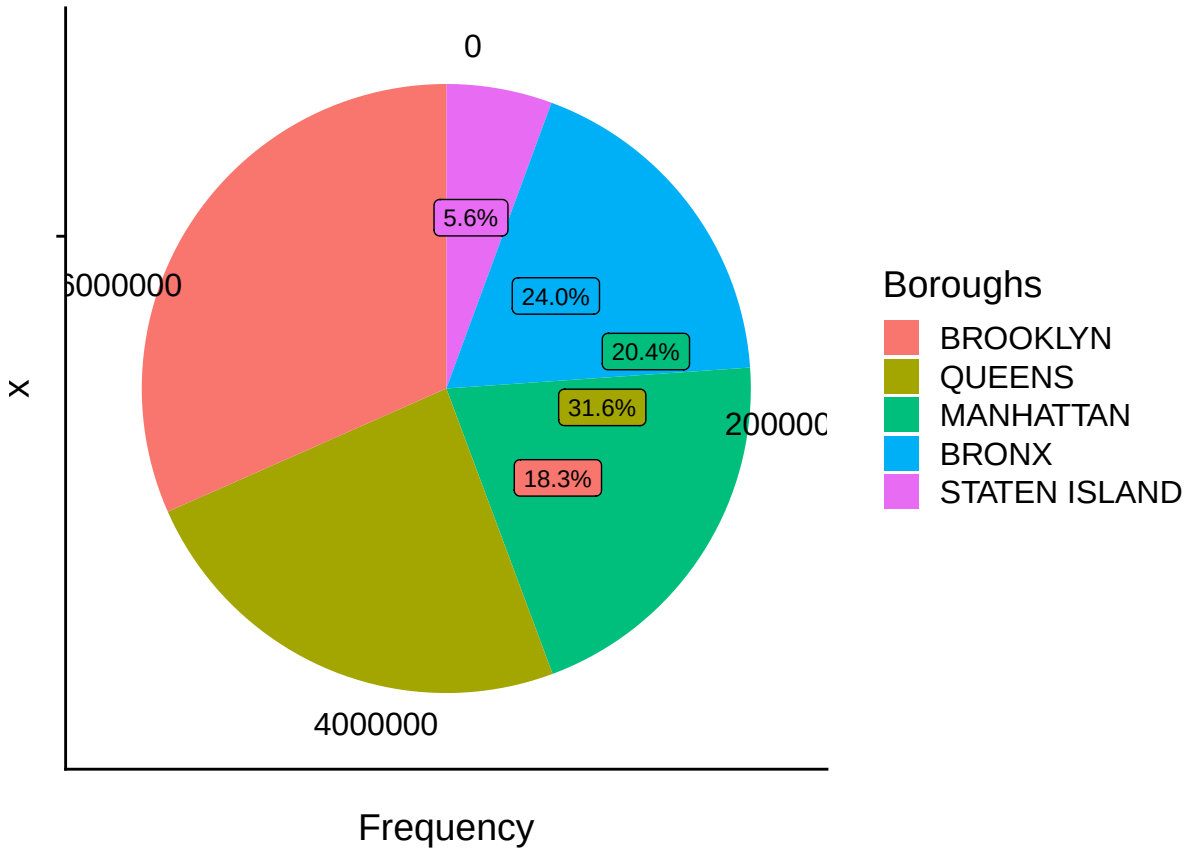


Figure 2: Complaint by Status

This shows top 20 agencies with respect to complaints! HPD, DOT and NYPD are doing a good job here! (Not really.)

```
top_20_agencies_by_complaints <- data.frame(table(nyc.df$Agency))[order(-data.frame(table(nyc.df$Agency)
kable(top_20_agencies_by_complaints, "latex", booktabs = T, caption = "Top 20 Agencies by complaint fre
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Table 1: Top 20 Agencies by complaint frequency

	Var1	Freq
16	HPD	2477621
11	DOT	1118188
18	NYPD	1021879
3	DEP	792783
13	DSNY	664199
6	DOB	465119
12	DPR	402302
9	DOHMH	262683
19	TLC	115381
2	DCA	99917
15	FDNY	23784
1	3-1-1	19866
8	DOF	18213
7	DOE	11007
14	EDC	4749
10	DOITT	4663
5	DHS	3152
4	DFTA	1623
17	HRA	1
NA	NA	NA

Here we can see that which type of agency handles which type of complaints and can deduce the maximum number of complaints that falls into a particular agency. This is basically showing top 20 agencies handling maximum number of complaints.

```
ggplot(top_20_agencies_by_complaints, aes(x = reorder(top_20_agencies_by_complaints$Var1, -top_20_agenci
         geom_bar(stat = "identity", fill="brown") +
         scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
         coord_flip() + xlab("Agency") + ylab("Frequency of Complaints by Agency") + ggtitle("Frequency
```

```
## Warning: Removed 1 rows containing missing values (position_stack).
```

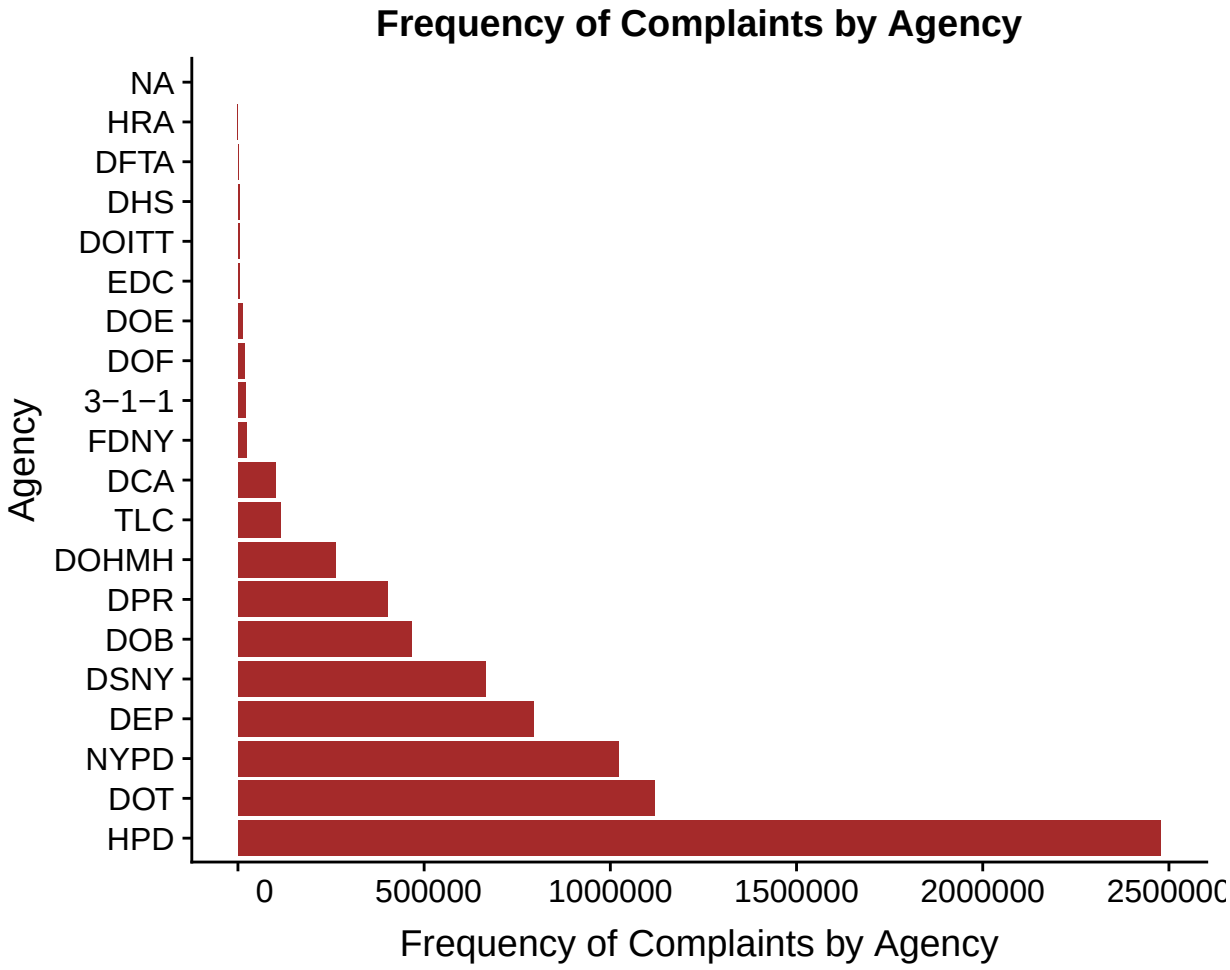


Figure 3: Frequency of Complaints by Agency

We would now like to know the Complaint types. There are a total of 220 complaints.

```
length(unique(nyc.df$`Complaint Type`))
```

```
## [1] 220
```

We will want to see the ticket status of these complaints.

Here we are finding Status of complaints. We can see that there are a total of 9 unique statuses of complaints.

```
unique(nyc.df$Status)
```

```
## [1] "Closed"           "Open"             "Assigned"
## [4] "Pending"          "Started"          "Unassigned"
## [7] "Unspecified"      "Draft"            "Closed - Testing"
```

```
sprintf("Total Number of Status: %i ",length(unique(nyc.df$Status)))
```

```
## [1] "Total Number of Status: 9 "
```

Frequency of complaint tickets. Complaints Status Frequency can be seen by running this particular code chunk. We can see a lot of complaints have been closed. And a lot of them have been assigned and looked after. That is positive! But still there are around 835089 complaints that are 'Open'.

```
kable(data.frame(table(nyc.df$Status)), "latex", booktabs = T, caption = "Complaints Status Frequency")
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Table 2: Complaints Status Frequency

Var1	Freq
Assigned	129582
Closed	6345055
Closed - Testing	12
Draft	1
Open	835089
Pending	186121
Started	11235
Unassigned	18
Unspecified	17

Now lets take a better look at visualizing Complaint Status. The diagram/plot will give us a better intuitive sense. Can be clearly seen what we described above.

```
ggplot(data.frame(table(nyc.df$Status)), aes(x = reorder(data.frame(table(nyc.df$Status))$Var1, -data.fr
         geom_bar(stat = "identity", fill="blue") +
         scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
         coord_flip() + ylab("Count of Complaint status") + xlab("Status")+ ggtitle("Complaint Status")
```

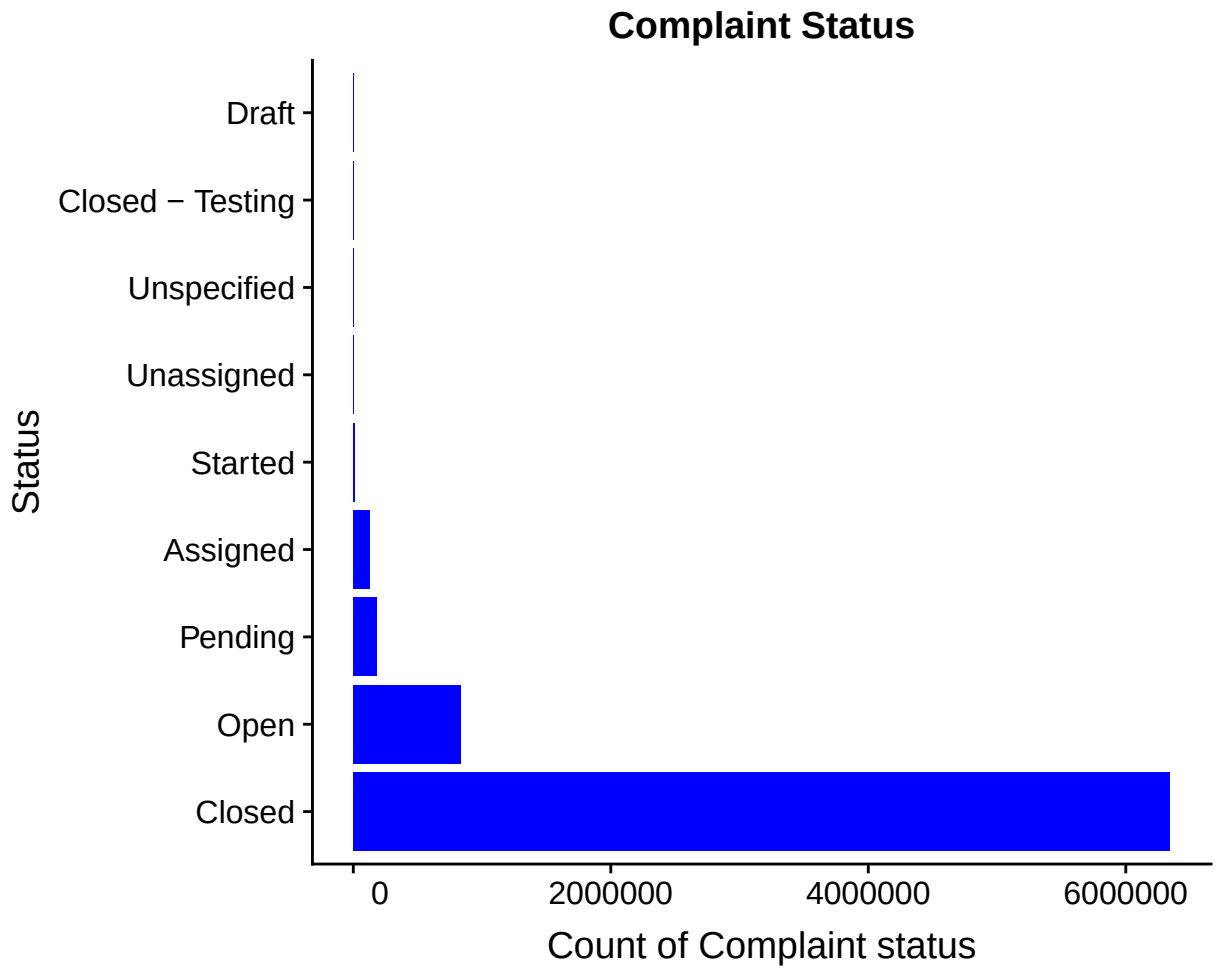


Figure 4: Complaint Status

Lets find top 20 Complaints. Its fun!

```
nyc.df.complaints.ordered <- data.frame(table(nyc.df$`Complaint Type`))[order(data.frame(table(nyc.df$`
top_20_complaints <- nyc.df.complaints.ordered[1:20,]
```

Maximum complaint type

Can be seen complaints that were maximum were for 'Heating'. Obviously its east coast. Run this code if you don't believe.

```
ggplot(top_20_complaints, aes(x = reorder(top_20_complaints$Var1, -top_20_complaints$Freq), y = top_20_
        geom_bar(stat = "identity", fill="red") +
        scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
        coord_flip() + xlab("Complaint Type") + ylab("Frequency") + ggtitle("Complaint Type Frequency"
```

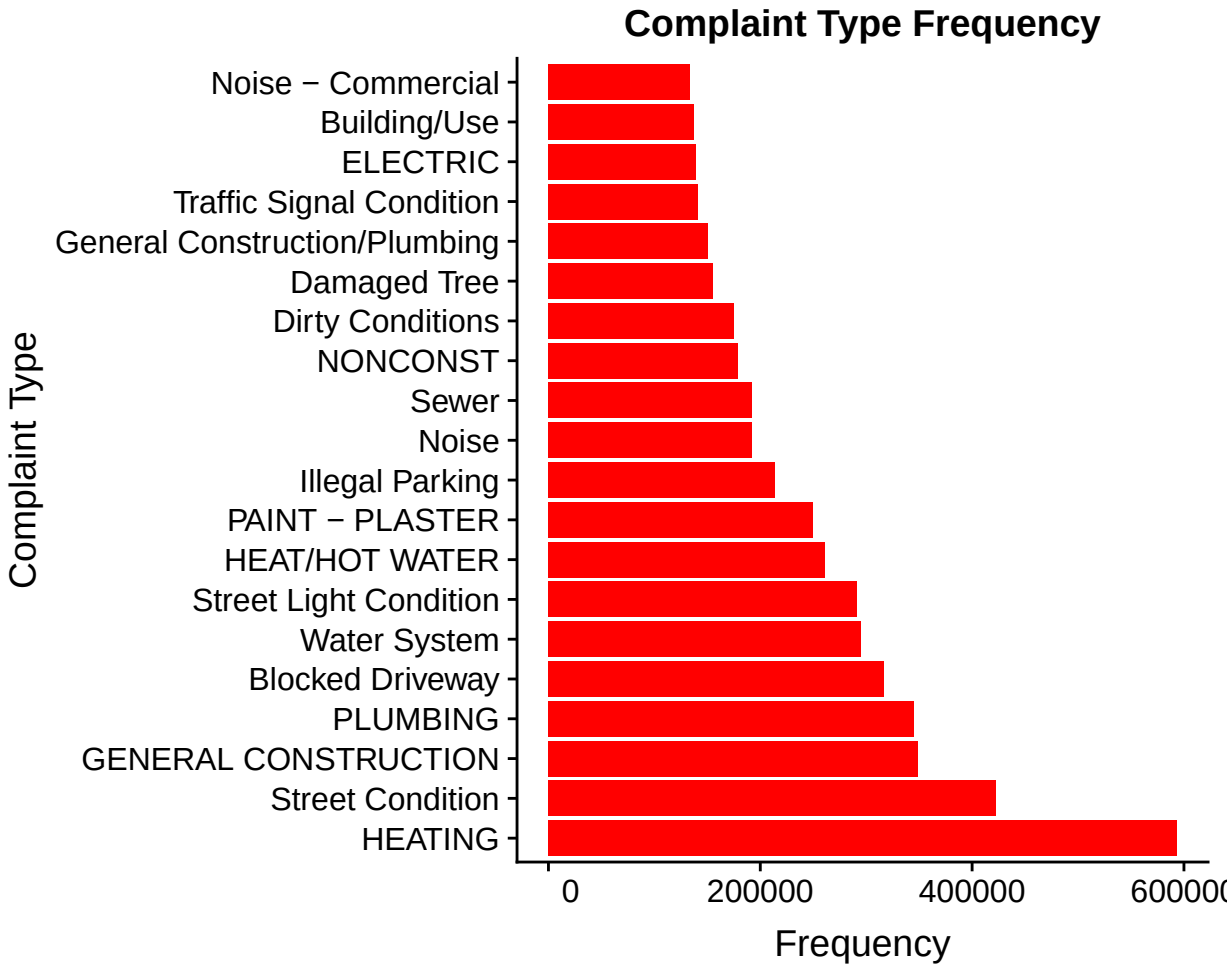


Figure 5: Complaint Type Frequency

Lets try to find out which complaints are most famous and prevalent, even though we can see from the above graph that Heating is the most sought after but it is more enticing to see that using a word cloud for our understanding. Visual treat is always better. And what this shows? Heating is a problem! East coast my friend. East Coast. I believe heating is not the problem, but winters are!

```
nyc.df.complaints.ordered <- data.frame(table(nyc.df$`Complaint Type`))[order(data.frame(table(nyc.df$`
wordcloud(words = nyc.df.complaints.ordered$Var1, freq = nyc.df.complaints.ordered$Freq, min.freq = 1,
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```

Here we are trying to group by cities to find the complaints.

Doing this makes us understand the data here more.

```
Freqs <- nyc.df %>%
  group_by(nyc.df$City) %>%
  summarize(Freq = n())
kable(head(Freqs), "latex", booktabs = T, caption = "Cities grouped by complaint frequency") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Figure 6: Wordcloud

Table 3: Cities grouped by complaint frequency

nyc.df$City	Freq
ANONYMOUS	1
ARVERNE	13999
ASTOIA	1
ASTORIA	120715
BAY RIDGE	2
BAYSIDE	34921

Now analyzing the complaints by cities. And bummer, people in Brooklyn complain alot! Also, second spot is taken by New York City. Not surprising!!

```
Freqs.complaints_by_city <- Freqs[order(Freqs$Freq, decreasing = TRUE),]
Freqs.complaints_by_city
```

```
## # A tibble: 146 x 2
##    `nyc.df$City`    Freq
##    <chr>           <int>
##  1 BROOKLYN      2374146
##  2 NEW YORK      1534981
##  3 BRONX         1373241
##  4 STATEN ISLAND  421324
##  5 JAMAICA        198986
##  6 FLUSHING       154937
##  7 ASTORIA        120715
##  8 RIDGEWOOD       86945
##  9 WOODSIDE        62329
## 10 CORONA          61400
## # ... with 136 more rows
```

```
Freqs.complaints_by_city_top_20 <- Freqs.complaints_by_city[1:20, ]
Freqs.complaints_by_city_top_20
```

```
## # A tibble: 20 x 2
##    `nyc.df$City`          Freq
##    <chr>                 <int>
##  1 BROOKLYN            2374146
##  2 NEW YORK            1534981
##  3 BRONX               1373241
##  4 STATEN ISLAND        421324
##  5 JAMAICA              198986
##  6 FLUSHING             154937
##  7 ASTORIA              120715
##  8 RIDGEWOOD             86945
##  9 WOODSIDE              62329
## 10 CORONA                61400
## 11 FAR ROCKAWAY          58976
## 12 ELMHURST              54295
## 13 QUEENS VILLAGE        53121
## 14 OZONE PARK            52039
## 15 FOREST HILLS          51920
## 16 LONG ISLAND CITY      49153
## 17 SOUTH RICHMOND HILL   48496
```

```
## 18 EAST ELMHURST          48368
## 19 JACKSON HEIGHTS        44594
## 20 FRESH MEADOWS          41764
```

It can be seen from top 20 cities complaining. Brooklyn is at number 1 spot!

```
ggplot(Freqs.complaints_by_city_top_20, aes(x = reorder(Freqs.complaints_by_city_top_20$`nyc.df$City`, 
        geom_bar(stat = "identity") +
         scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
        coord_flip()+ ylab("Frequency") + xlab("City") + ggtitle("Top 20 Complaints")
```

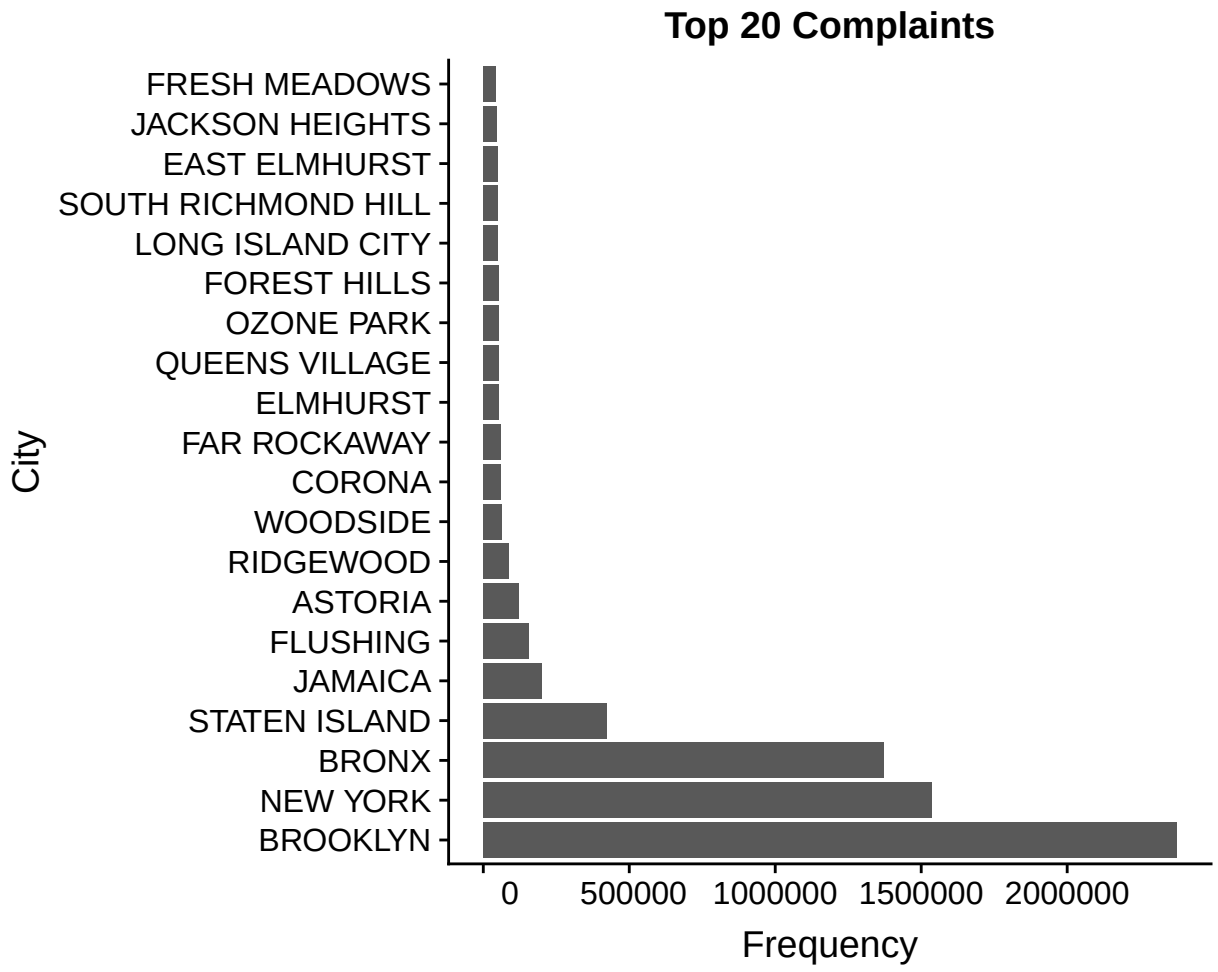


Figure 7: Top 20 Complaints

Getting the frequency of complaints by Borough and Complaints Type.

Grouping the data by Borough and Complaint type. Then arranging the data frame by Borough and Complaint Type. This will help us calculating the complaints by Borough and Complaints Type. This will provide a better insight. Further creating a new data-frame.

```
# Group by Borough and Complaints type
newdf <- nyc.df %>% group_by(nyc.df$Borough, nyc.df$`Complaint Type`) %>% arrange(nyc.df$Borough, nyc.d
```

Finding the frequency of complaints by Borough and Complaints Type.

```
dat.new <- dcast(newdf, newdf$Borough ~ newdf$`Complaint Type`, fun.aggregate = length)
```

```
## Using 'nyc.df$`Complaint Type`' as value column. Use 'value.var' to override
```

```
dat.new
```

```
##   newdf$Borough Adopt-A-Basket Air Quality Animal Abuse
## 1         BRONX             28        3122         3205
## 2      BROOKLYN             49        9071         3650
## 3     MANHATTAN             48       15949         1997
## 4        QUEENS             47        5545         3312
## 5 STATEN ISLAND             10         997          957
##   Animal Facility - No Permit Animal in a Park APPLIANCE Asbestos
## 1                          37             1062     11536     1236
## 2                          95             1873     12347     2831
## 3                         105             3105      6392     3761
## 4                         102             1540      4685     2014
## 5                          39              503       853      340
##   Beach/Pool/Sauna Complaint Benefit Card Replacement BEST/Site Safety
## 1                         76                         0              331
## 2                        214                         0             1407
## 3                        202                         1             1744
## 4                        233                         0             1122
## 5                        160                         0              110
##   Bike Rack Condition Bike/Roller/Skate Chronic Blocked Driveway Boilers
## 1                   6                        128            48228    2284
## 2                 258                        498           117833    2864
## 3                 268                       1300             9891    2643
## 4                  25                        230           130851    1618
## 5                   5                         53            10132     193
##   Bottled Water Bridge Condition Broken Muni Meter Broken Parking Meter
## 1            11               90              7612                 1971
## 2            20              409             18962                 6073
## 3            18              345             70086                 2825
## 4            26              127             21570                 4136
## 5             2               34              1708                  237
##   Building Condition Building/Use Bus Stop Shelter Placement
## 1                  0        15813                       145
## 2                  2        41127                       391
## 3                 13         9779                       373
## 4                  0        61572                       358
## 5                  0         8978                       115
##   Calorie Labeling City Vehicle Placard Complaint Collection Truck Noise
## 1                5                           303                    49
## 2               12                           511                   198
## 3               74                           458                   318
## 4               15                           388                   191
## 5               11                            50                    40
##   Construction CONSTRUCTION Consumer Complaint Cranes and Derricks
## 1           92          857              14622                182
## 2          361         1427              28625                402
## 3          163          613              27701                716
## 4        22162          488              25526                288
```

```
## 5           16          116               3443                  45
##   Curb Condition Damaged Tree DCA Literature Request Dead Tree
## 1            309        12901                     0      4303
## 2           1124        40955                     3     11737
## 3            144         6395                    13      3737
## 4            841        75089                     0     22449
## 5             64        19798                     0      5749
##   Derelict Bicycle Derelict Vehicle Derelict Vehicles Dirty Conditions
## 1               21             7591              7628            26577
## 2             1507            18203             23172            54396
## 3             1660             2473              1758            24108
## 4              263            32799             22093            58843
## 5                7             8553              3808            11284
##   Discipline and Suspension Disorderly Youth DOE Complaint or Compliment
## 1                       142              682                         129
## 2                       150              932                         189
## 3                        85             1186                         126
## 4                        83              655                         127
## 5                        19              357                          44
##   DOF Literature Request DOF Parking - Request Status
## 1                      0                            0
## 2                      1                            1
## 3                      2                            0
## 4                      1                            0
## 5                      0                            0
##   DOF Parking - Tax Exemption DOF Property - City Rebate
## 1                           6                          2
## 2                          19                          5
## 3                        2499                          2
## 4                          20                          5
## 5                           7                          2
##   DOF Property - Payment Issue DOF Property - Reduction Issue
## 1                           78                            461
## 2                          283                            539
## 3                          191                            471
## 4                          201                            425
## 5                           56                             20
##   DOF Property - RPIE Issue DOOR/WINDOW DPR Internal Drinking
## 1                         2       12683          216     1054
## 2                         4       15104          285     1621
## 3                         7        7778          917     1675
## 4                         1        4696          316     1688
## 5                         0         877          152      314
##   Drinking Water DWD EAP Inspection - F59 ELECTRIC Electrical Elevator
## 1             97   1                  96    38571       2600    12655
## 2            119   1                 274    54664       7471    10713
## 3            121   0                4086    22099       4315    11339
## 4            119   0                 261    20872       3512     6088
## 5             33   0                  32     3436        961      745
##   ELEVATOR Emergency Response Team (ERT) Ferry Inquiry
## 1      310                          157             1
## 2      251                          635             0
## 3      212                          774             0
## 4      177                          923             0
```

```
## 5        5                           97             0
##   Fire Alarm - Addition Fire Alarm - Modification Fire Alarm - New System
## 1                     7                        34                     79
## 2                     4                        13                    137
## 3                    61                       422                    320
## 4                    15                        23                    101
## 5                     1                         3                     12
##   Fire Alarm - Reinspection Fire Alarm - Replacement
## 1                       142                       4
## 2                       202                       7
## 3                       714                      32
## 4                       179                      17
## 5                        27                       1
##   Fire Safety Director - F58 FLOORING/STAIRS Food Establishment
## 1                        352            8907               3876
## 2                        970           10156               9489
## 3                      13198            6018              13638
## 4                       1450            2829               7810
## 5                        166             607               1564
##   Food Poisoning For Hire Vehicle Complaint For Hire Vehicle Report
## 1           1721                        3312                     17
## 2           3678                        3513                     55
## 3           5871                        3708                     44
## 4           3147                        3151                     36
## 5            636                         436                      2
##   Forensic Engineering Found Property Gas Station Discharge Lines GENERAL
## 1                    0             29                           1    8971
## 2                  160            294                           0   10369
## 3                  161           2986                           0    8843
## 4                    2            247                           0    3770
## 5                    0              1                           0     518
##   GENERAL CONSTRUCTION General Construction/Plumbing Graffiti
## 1               107097                          16103    19212
## 2               131693                          50769    30420
## 3                61178                          36688    17261
## 4                40953                          39097    16759
## 5                 8270                           8309     1534
##   Harboring Bees/Wasps Hazardous Materials Hazmat Storage/Use Health
## 1                  168                1278                   2     52
## 2                  298                3506                   5     53
## 3                   45                3241                   9     36
## 4                  296                3066                   2     46
## 5                  130                 908                   0     28
##   HEAT/HOT WATER HEATING Highway Condition Highway Sign - Damaged
## 1          87390  191143              3681                     63
## 2          78268  186911              4312                     44
## 3          59292  134768              2357                     35
## 4          33448   74390              8268                     77
## 5           2497    5911              1719                     16
##   Highway Sign - Dangling Highway Sign - Missing
## 1                       8                     22
## 2                      16                     27
## 3                       4                     19
## 4                      15                     25
```

```
## 5                       3                    13
##   Home Care Provider Complaint Homeless Encampment
## 1                            0                762
## 2                            1               2602
## 3                            0               6790
## 4                            0               1541
## 5                            0                179
##   Homeless Person Assistance IGR Illegal Animal - Sold/Kept
## 1                        159   0                          0
## 2                        703  90                          2
## 3                       2008  39                          0
## 4                        268 120                          0
## 5                         14  34                          0
##   Illegal Animal Kept as Pet Illegal Animal Sold Illegal Fireworks
## 1                        225                  15               148
## 2                        404                  70               324
## 3                        106                  45               218
## 4                        350                  70               265
## 5                        137                   7               103
##   Illegal Parking Illegal Tree Damage Indoor Air Quality Indoor Sewage
## 1           22786                 882               3389           772
## 2           74897                3005               7030          1718
## 3           37736                2462               9854           701
## 4           61429                2870               3813           887
## 5           16833                 870                510           233
##   Industrial Waste Interior Demo Internal Code
## 1              884             0            56
## 2             2232             0            62
## 3             1466             1           103
## 4             2333             0           104
## 5              874             0            36
##   Investigations and Discipline (IAD) Laboratory Lead
## 1                                 543          0 1864
## 2                                2104          0 7885
## 3                                1462          0 5831
## 4                                1633          1 5557
## 5                                 333          0 1167
##   Legal Services Provider Complaint Lifeguard Literature Request
## 1                                10         4                  3
## 2                                45        10                  6
## 3                                29        29                  6
## 4                                22        18                  8
## 5                                 5         5                  3
##   Litter Basket / Request Lost Property Maintenance or Facility
## 1                    1317             0                   5752
## 2                    4567             1                  14259
## 3                    3383             6                  12843
## 4                    2853             0                   9957
## 5                    1085             0                   2827
##   Micro Switch Miscellaneous Categories Missed Collection (All Materials)
## 1            0                     296                              7143
## 2            0                      98                             27269
## 3            1                     397                              7984
## 4            0                      49                             31744
```

```
## 5           0                      10                            18577
##   Mold Municipal Parking Facility No Child Left Behind Noise
## 1  155                         80                    2 11633
## 2  469                        105                    4 46999
## 3  487                         24                    3 96130
## 4  341                        171                    3 30927
## 5  165                         50                    5  6735
##   Noise - Commercial Noise - Helicopter Noise - House of Worship
## 1                 8967                 95                      540
## 2                41007                1796                    2310
## 3                58346                2401                    1402
## 4                22605                 378                    1147
## 5                 3114                  80                      68
##   Noise - Park Noise - Street/Sidewalk Noise - Vehicle Noise Survey
## 1         1516                    20100           10842         2035
## 2         4410                    31447           22037         4391
## 3         5574                    51810           24449         6206
## 4         2590                    13717           11250         2846
## 5          235                     2311            1800          662
##   Non-Residential Heat NONCONST OEM Disabled Vehicle Open Flame Permit
## 1                 1291    50274                   67                 0
## 2                 2263    69473                  969                24
## 3                 2676    30981                   60                85
## 4                 1669    24626                  652                 6
## 5                  182     3859                  122                 0
##   Other Enforcement OUTSIDE BUILDING Overflowing Litter Baskets
## 1              1977              431                        608
## 2              8708              716                       1881
## 3              6088              404                       1271
## 4              6584              313                       1615
## 5              1947               76                        662
##   Overflowing Recycling Baskets Overgrown Tree/Branches PAINT - PLASTER
## 1                             2                    8414           78713
## 2                             5                   23613           93546
## 3                            32                    3243           45410
## 4                            16                   29117           28082
## 5                             4                   10740            4155
##   PAINT/PLASTER Panhandling Parent Leadership Plant Plumbing PLUMBING
## 1         26696          43                 26    39     2516   103553
## 2         29002         117                 12   866     5683   127763
## 3         16980         415                 15   121     3577    62857
## 4          9448         146                 18   363     6050    43307
## 5          1374          45                  1   135     1640     7468
##   Poison Ivy Portable Toilet Posting Advertisement Public Assembly
## 1         70               2                    96              14
## 2         76               5                   366              24
## 3         15               5                   234              65
## 4        186               1                   281              22
## 5        202               0                    37               1
##   Public Assembly - Temporary Public Payphone Complaint Public Toilet
## 1                           1                       432            13
## 2                           3                       911            75
## 3                          10                      2322            80
## 4                           0                       826            29
```

```
## 5                          0                       172             7
##   Radioactive Material Rangehood Recycling Enforcement
## 1                    4         0                  416
## 2                   12         2                 1780
## 3                   12         7                 1595
## 4                    9         7                 1229
## 5                   12         1                  297
##   Registration and Transfers Request Xmas Tree Collection Rodent
## 1                         48                           78  24241
## 2                         61                          252  38369
## 3                         23                           48  26996
## 4                         47                          180  21318
## 5                          9                          114  12679
##   Root/Sewer/Sidewalk Condition Safety SAFETY Sanitation Condition
## 1                          5242    199   2718                15257
## 2                         15142    255   3942                50410
## 3                          1045     93   1763                12046
## 4                         20469    138   1624                40322
## 5                          7374     26    233                11939
##   Scaffold Safety School Maintenance SCRIE SDEP SDSC
## 1             157               1538  2179   14   24
## 2             316               2696  3474   68   27
## 3             569               1816  4143    6   34
## 4             167               1655  2925   36   24
## 5              21                534   185   11   14
##   Senior Center Complaint Sewer SG-51 SG-71 SG-98 SG-99 Sidewalk Condition
## 1                     257 18317     2     0     4     1               3106
## 2                     377 52524     2     0    16     0              12198
## 3                     460 17097     2     2     2     0              10542
## 4                     293 81072     1     0    11     0               8260
## 5                      63 23392     4     0     3     0               2310
##   Smoking  Snow Special Enforcement Special Natural Area District (SNAD)
## 1    1416  6379                3984                                    2
## 2    2819 17533                5906                                    5
## 3    3809  4293                5226                                    2
## 4    2270 16649               13770                                   16
## 5     334  7778                1225                                  284
##   Special Projects Inspection Team (SPIT) Sprinkler - Mechanical Squeegee
## 1                                     911                       3        1
## 2                                    3357                       2        2
## 3                                    6476                      25       20
## 4                                    4746                       2        4
## 5                                     607                       1        0
##   Stalled Sites Standing Water Standpipe - Mechanical Street Condition
## 1            51            760                       4            46167
## 2           293           1859                       3           121551
## 3           138            899                      11            84528
## 4           137           2094                       2           116402
## 5            42           1688                       2            53849
##   Street Light Condition Street Sign - Damaged Street Sign - Dangling
## 1                  42936                  2318                    865
## 2                  89755                 18574                   2608
## 3                  11474                  4208                   1392
## 4                 115760                  9107                   2624
```

```
## 5                  31393                  2122                   824
##   Street Sign - Missing STRUCTURAL STSK Summer Camp Sweeping/Inadequate
## 1                  2278          4    5          49                  34
## 2                  7057          4   19         126                 315
## 3                  2161          3    4          39                  47
## 4                  6619          4   21          93                 102
## 5                  1800          1    6          16                   1
##   Sweeping/Missed Sweeping/Missed-Inadequate Tanning Tattooing
## 1             103                       362       1       129
## 2             989                      3091       0       133
## 3             223                       607       2       111
## 4             369                      1391       3       113
## 5               4                        45       2        26
##   Taxi Complaint Taxi Compliment Taxi Report Teaching/Learning/Instruction
## 1           2246               0          20                          128
## 2          12729               0         114                          127
## 3          68990               2         739                           70
## 4          12387               0         170                          121
## 5            146               0           1                           25
##   Traffic Traffic Signal Condition Trans Fat
## 1    1447                   10974         1
## 2    3515                   62380         4
## 3    6363                    8293         7
## 4    3203                   50274         8
## 5     901                    9653         2
##   Transportation Provider Complaint Trapping Pigeon Unleashed Dog
## 1                                 6               0          618
## 2                                20               0          988
## 3                                20               1          512
## 4                                14               0         1443
## 5                                 1               0          668
##   Unlicensed Dog Unsanitary Animal Facility Unsanitary Animal Pvt Property
## 1              0                         74                          2176
## 2              1                        131                          3479
## 3              0                         94                          1642
## 4              0                        150                          3190
## 5              0                         30                          1420
##   UNSANITARY CONDITION Unsanitary Pigeon Condition Urinating in Public
## 1                23963                         370                 199
## 2                29605                         871                 567
## 3                14768                         898                 701
## 4                11649                         935                 428
## 5                 1953                         213                  64
##   VACANT APARTMENT Vacant Lot Vending Violation of Park Rules
## 1                0       1042    1777                   1512
## 2                0       3189    3604                   1461
## 3                2        459   10655                   3838
## 4                0       2237    2900                   1923
## 5                0       1639     203                    386
##   Water Conservation WATER LEAK Water Quality Water System Window Guard
## 1               2438      12061           642        53644          677
## 2               5207      15549          1448        78053         1222
## 3               1412       7564          1129        53159          259
## 4               6930       4768          1636        82590          308
```

```
## 5               3020        690           526        27494           83
##   WLWP WNW X-Ray Machine/Equipment
## 1  103 182                      7
## 2  126 275                     16
## 3   64 200                     24
## 4  259 440                     19
## 5   56 125                      4
```

Here we are trying to show the most famous complaints that we noticed on word cloud and their frequency for every Borough. It can be seen Heating complaint is most prevalent in BRONX whereas QUEENS has lot of complaints regarding street conditions, Water system, blocked driveway, street light condition. Looks like QUEENS has a lot of problems ! If you dont like potholes, choppy roads and flat tires you may want to steer clear of Queens! If you don't like winters you may want to avoid Brooklyn and Bronx and if you hate Traffic Jams, Traffic Signal Problems definitely avoid Brooklyn and Queens! Manhattan surprisingly does not have much problems, it is not suffering like Brooklyn, Queens and Bronx. Maybe because Manhattan is not as big as Brooklyn, Queens and Bronx? Maybe Manhattan has a better law and order? Lastly if you want to have peace of mind, you can get a house in Staten Island and click picturesque photos of New York skyline everyday. Until you get bored. Go on a free ferry ride and see liberty island. Yes. Its Staten Island! The best Borough overall (thats what stats tell).

```
# building data frames to plot

heatdata <- data.frame(dat.new$`newdf$Borough`, dat.new$HEATING)
streetcond <- data.frame(dat.new$`newdf$Borough`, dat.new$`Street Condition`)
plumbdat <- data.frame(dat.new$`newdf$Borough`, dat.new$PLUMBING)
watersys <- data.frame(dat.new$`newdf$Borough`, dat.new$`Water System`)
streetlightcond <- data.frame(dat.new$`newdf$Borough`, dat.new$`Street Light Condition`)
blockeddriveway <- data.frame(dat.new$`newdf$Borough`, dat.new$`Blocked Driveway`)
trafficsignalcond <- data.frame(dat.new$`newdf$Borough`, dat.new$`Traffic Signal Condition`)
paintplaster <- data.frame(dat.new$`newdf$Borough`, dat.new$`PAINT - PLASTER`)

# building plots here

heatplot <- ggplot(data = heatdata, aes(x=heatdata$dat.new..newdf.Borough., y=heatdata$dat.new.HEATING)
  geom_bar(stat = "identity", fill="blue") +
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +x

streetcondplot <- ggplot(data = streetcond, aes(x=streetcond$dat.new..newdf.Borough., y=streetcond$dat.
  geom_bar(stat = "identity", fill="yellow") +
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +x

plumbplot <- ggplot(data = plumbdat, aes(x=plumbdat$dat.new..newdf.Borough., y=plumbdat$dat.new.PLUMBIN
  geom_bar(stat = "identity", fill="red") +
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +x

waterplot <- ggplot(data = watersys, aes(x=watersys$dat.new..newdf.Borough., y=watersys$dat.new..Water.
  geom_bar(stat = "identity", fill="green") +
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +x

streetlightcondplot <- ggplot(data = streetlightcond, aes(x=streetlightcond$dat.new..newdf.Borough., y=
  geom_bar(stat = "identity", fill="black") +
```

```
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +xl

blockeddrivewayplot <- ggplot(data = blockeddriveway, aes(x=blockeddriveway$dat.new..newdf.Borough., y=b
  geom_bar(stat = "identity", fill="purple") +
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +xl

trafficsignalcondplot <- ggplot(data = trafficsignalcond, aes(x=trafficsignalcond$dat.new..newdf.Boroug
  geom_bar(stat = "identity", fill="grey") +
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +xl

paintplasterplot <- ggplot(data = paintplaster, aes(x=paintplaster$dat.new..newdf.Borough., y=paintplas
  geom_bar(stat = "identity", fill="pink") +
  theme(axis.text.x = element_text(size=8)) +
  scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) + ylab("Count of Complaint") +xl

# plotting the plots

heatplot
```

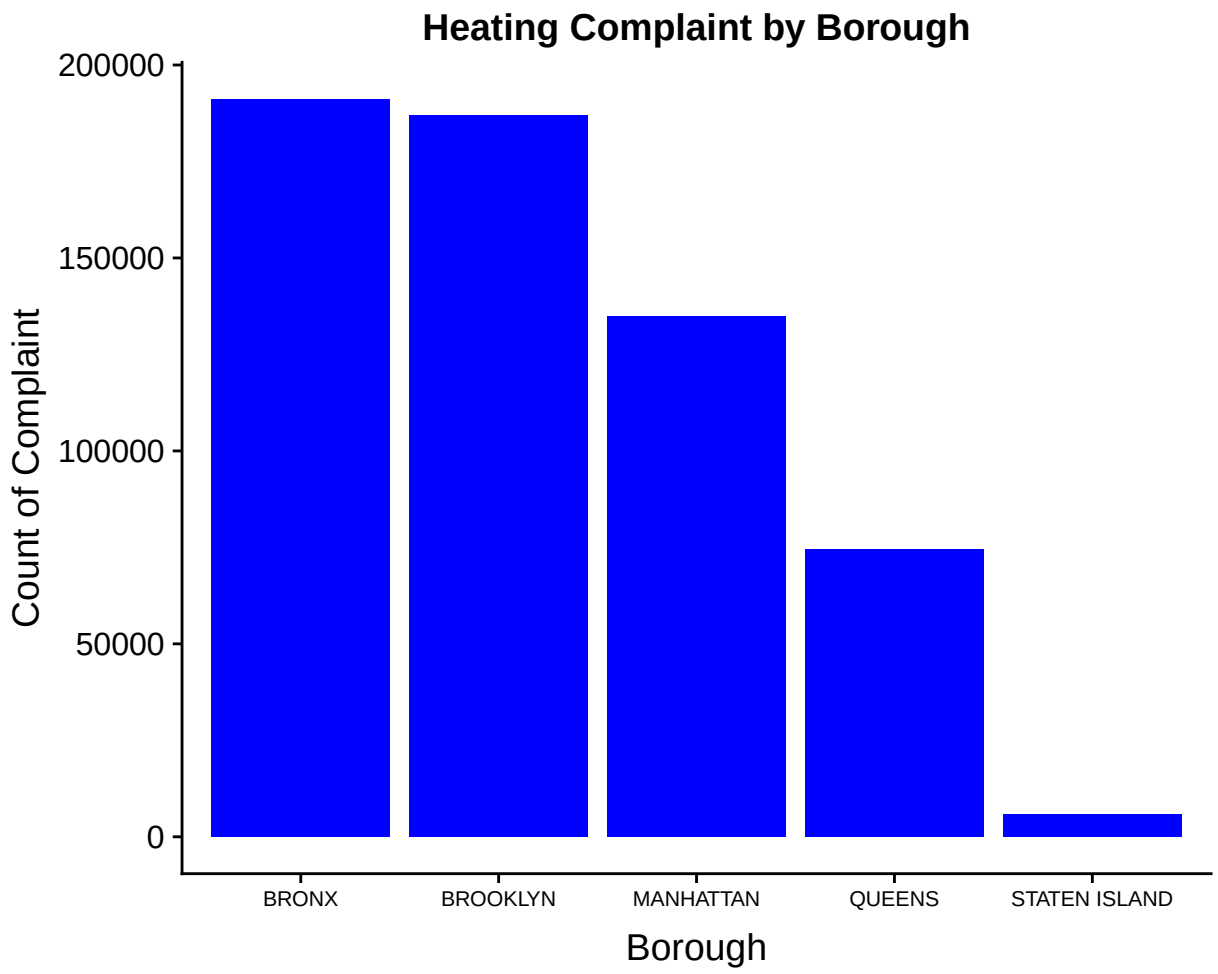


Figure 8: Different complaints, different Boroughs

```
streetcondplot
```

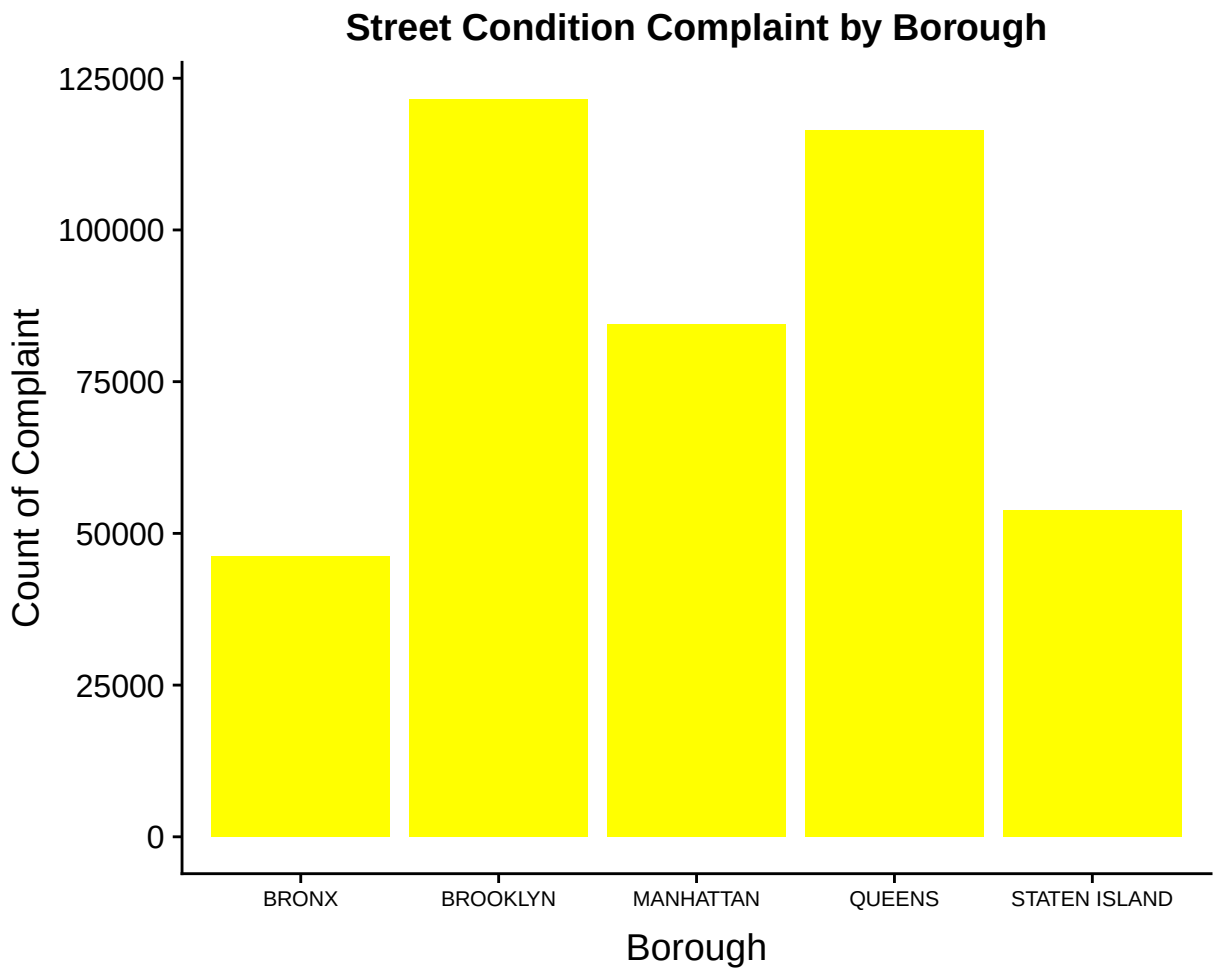


Figure 9: Different complaints, different Boroughs

```
plumbplot
```

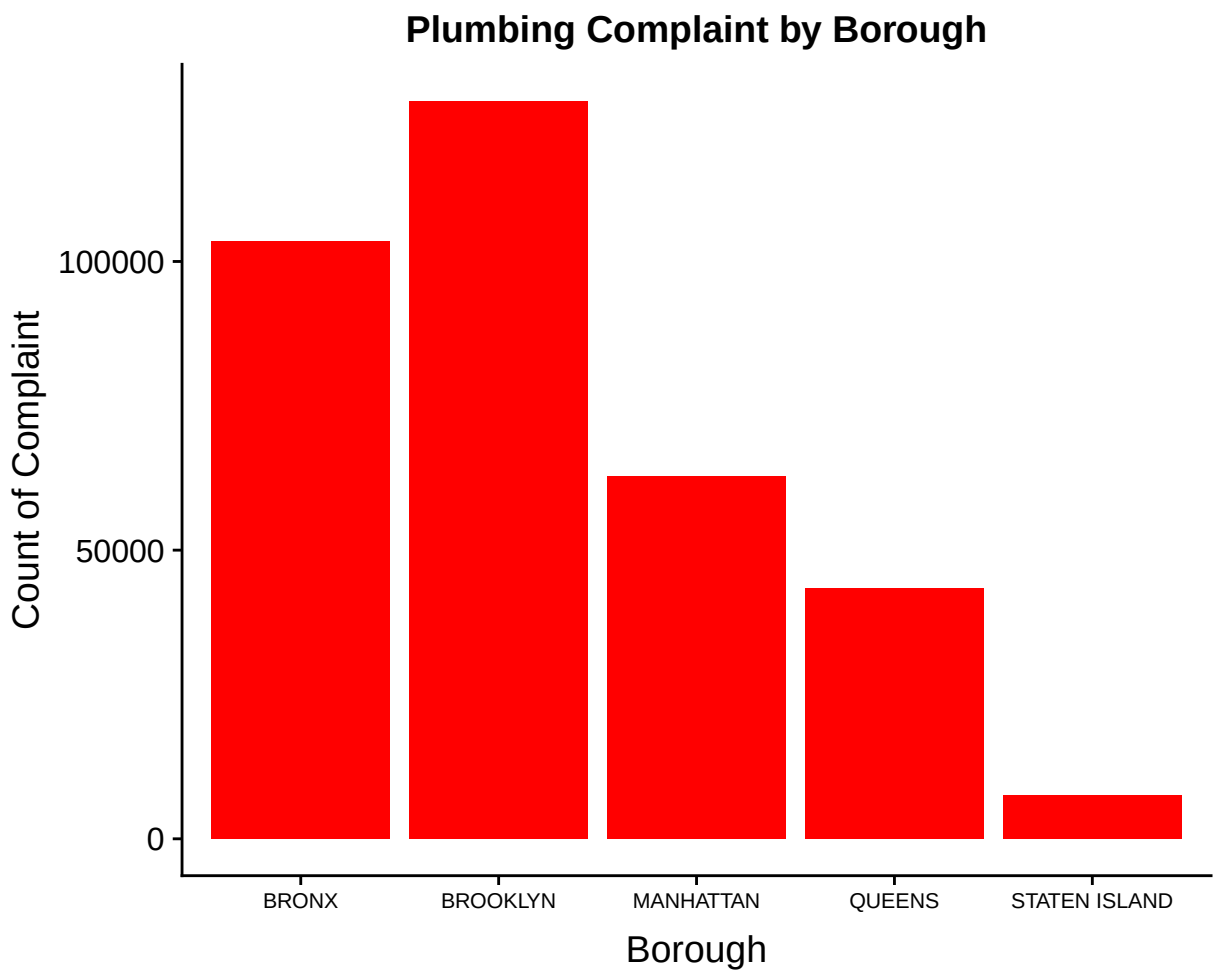


Figure 10: Different complaints, different Boroughs

```
waterplot
```

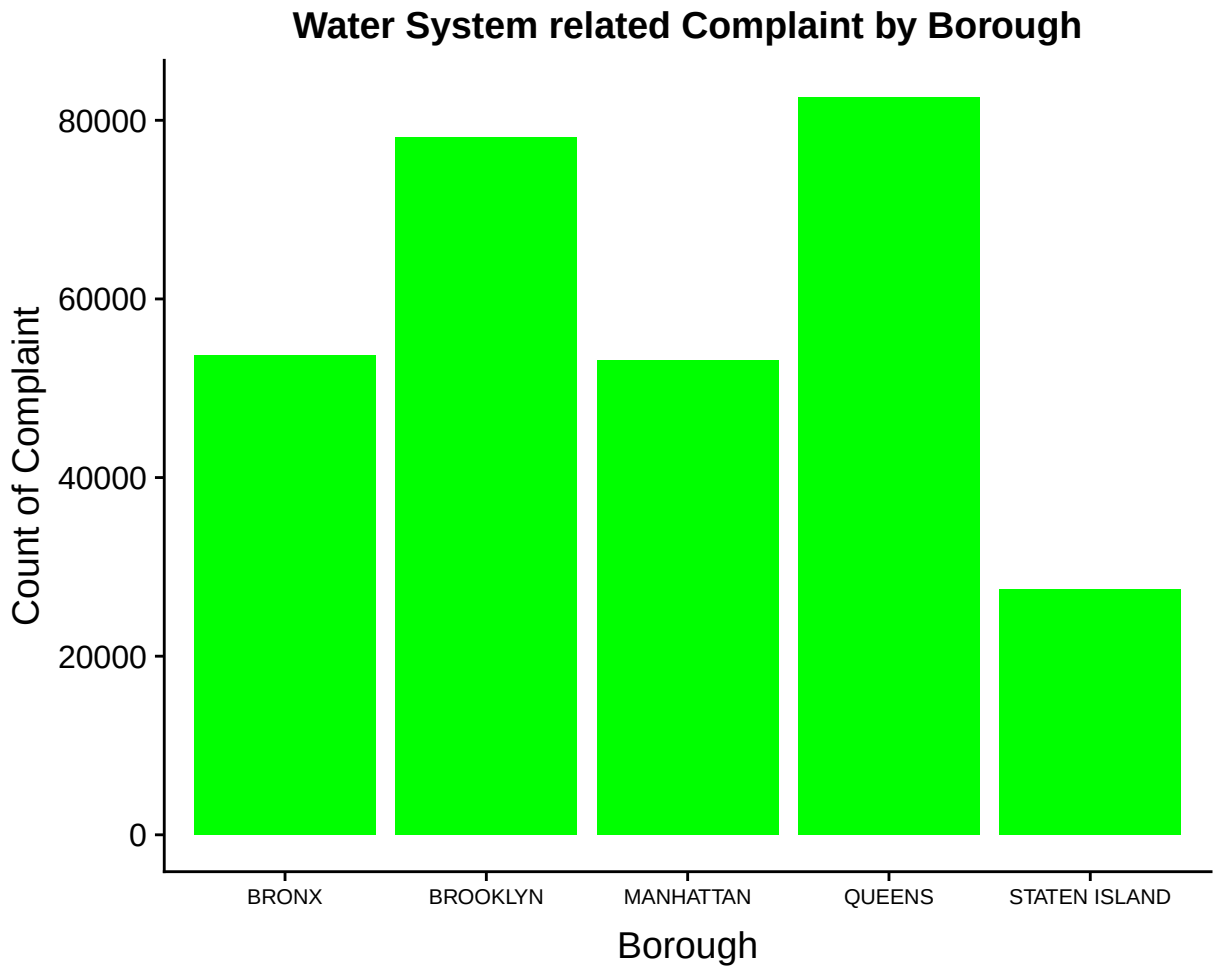


Figure 11: Different complaints, different Boroughs

```
streetlightcondplot
```

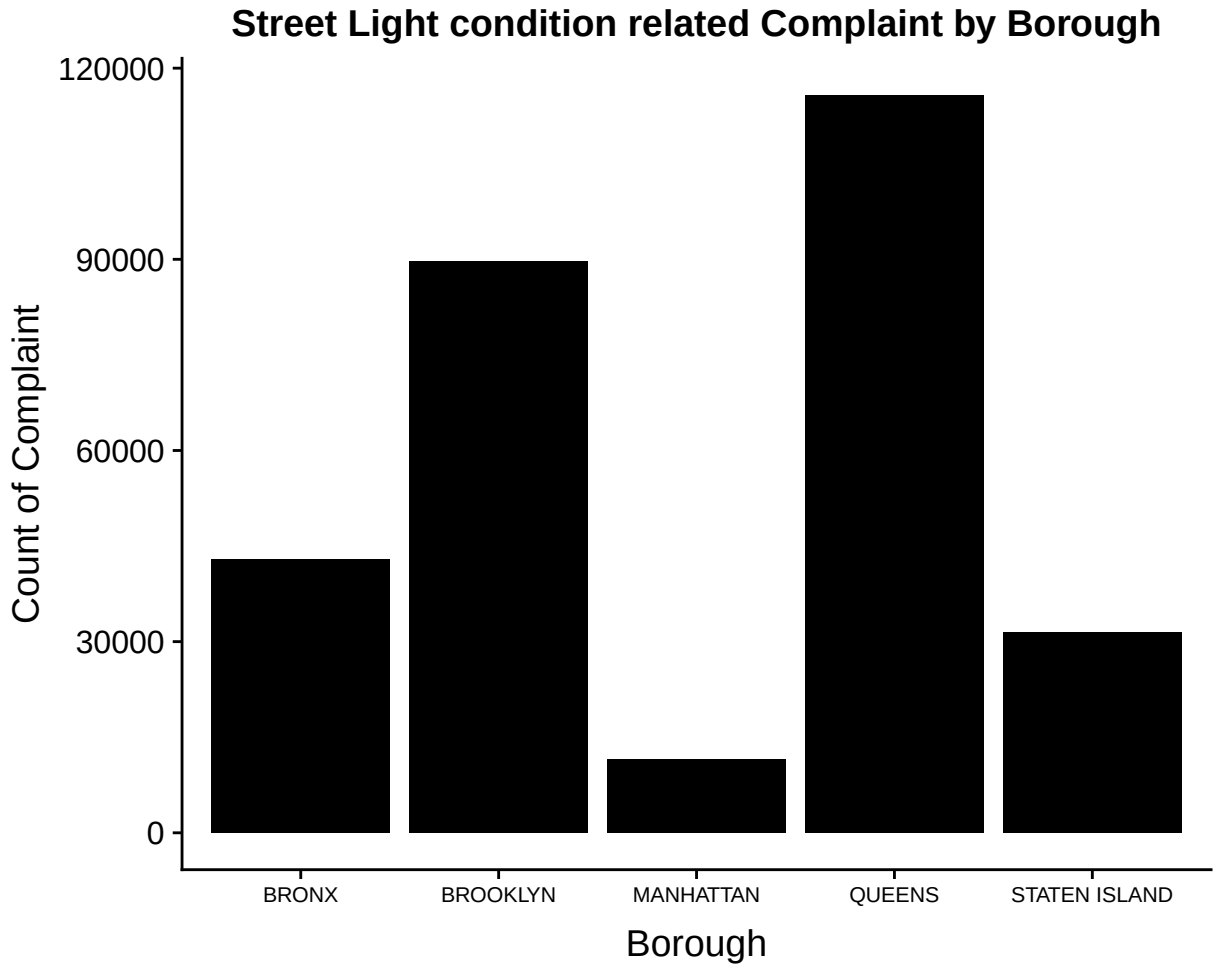


Figure 12: Different complaints, different Boroughs

```
blockeddrivewayplot
```

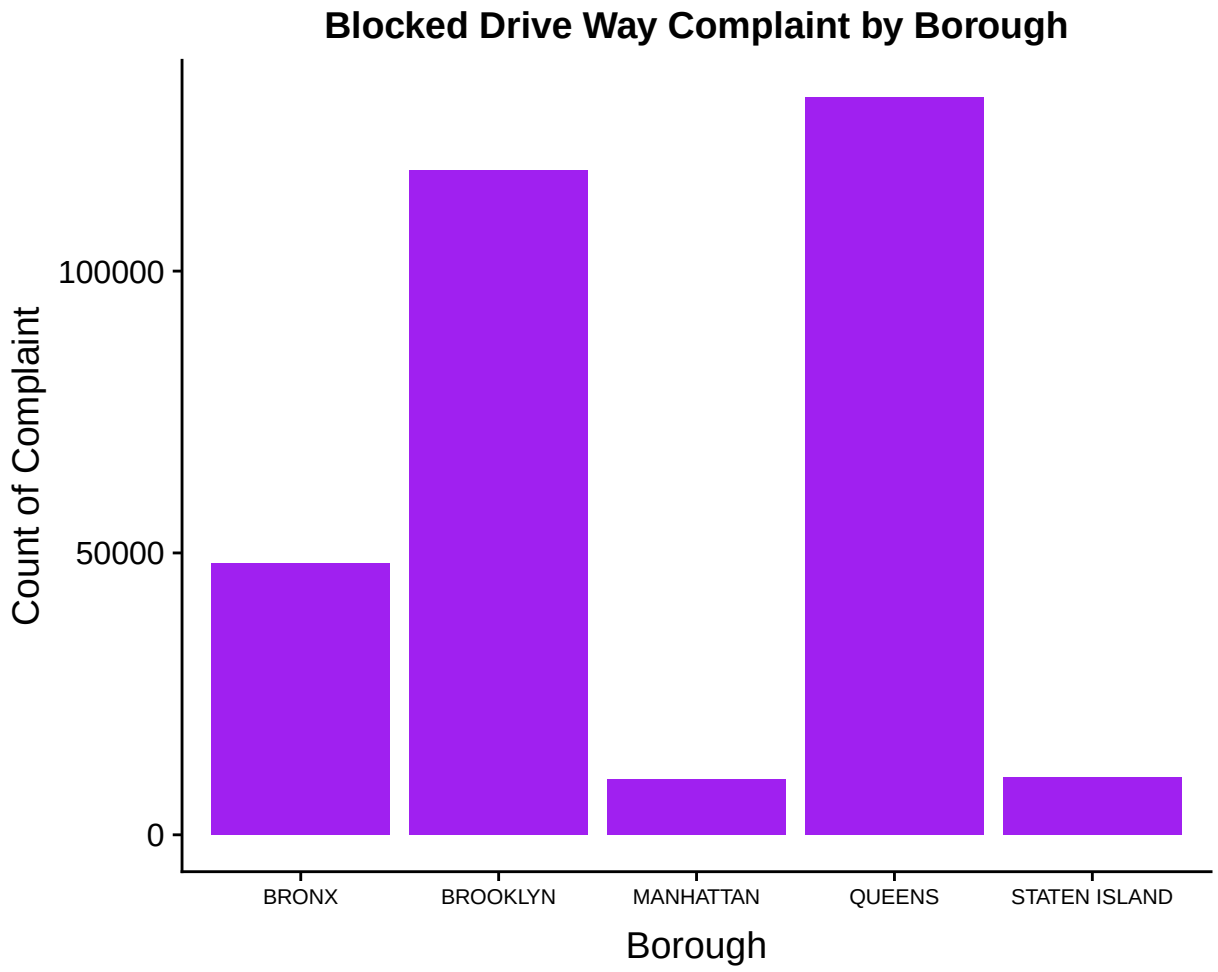


Figure 13: Different complaints, different Boroughs

```
trafficsignalcondplot
```

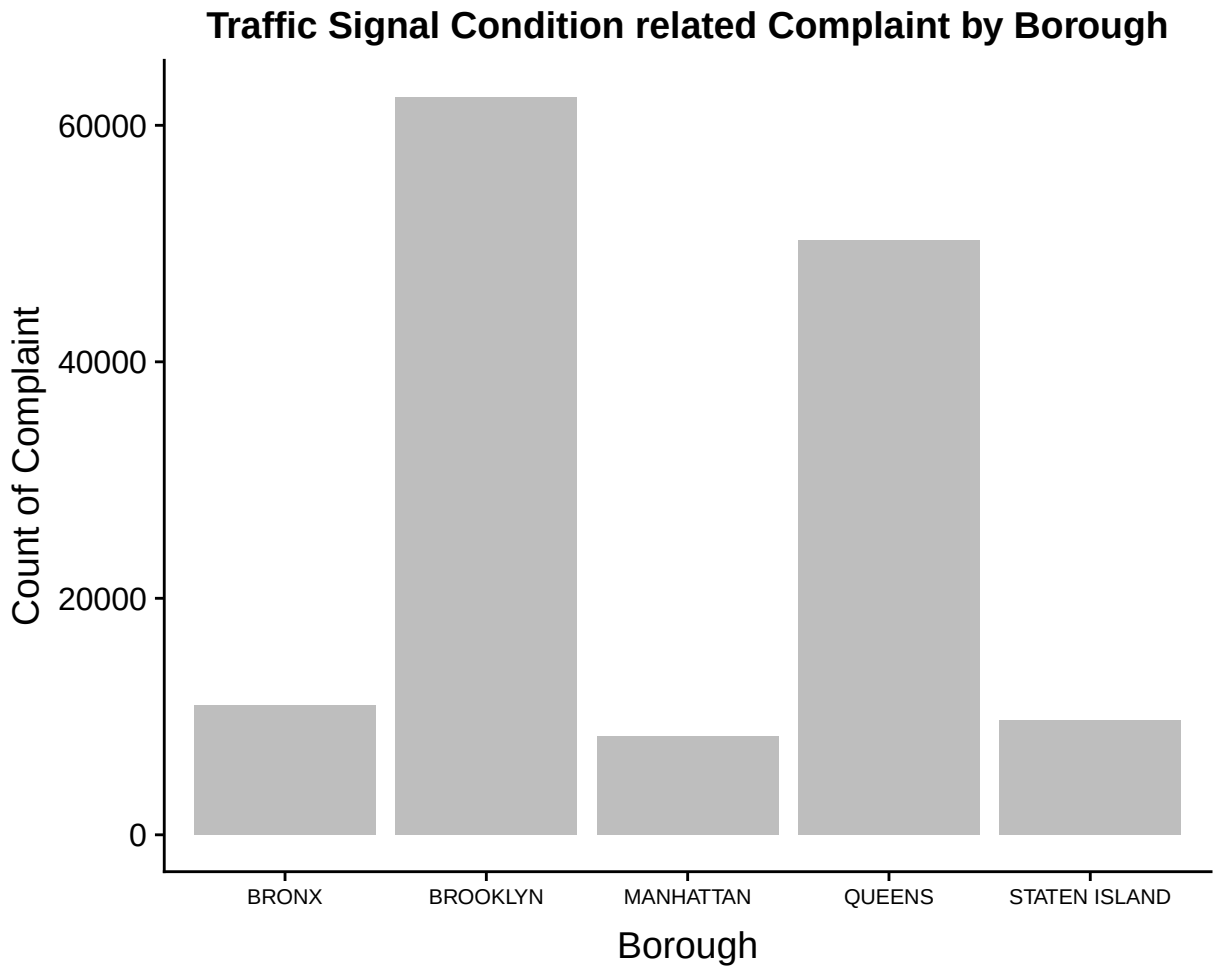


Figure 14: Different complaints, different Boroughs

```
paintplasterplot
```

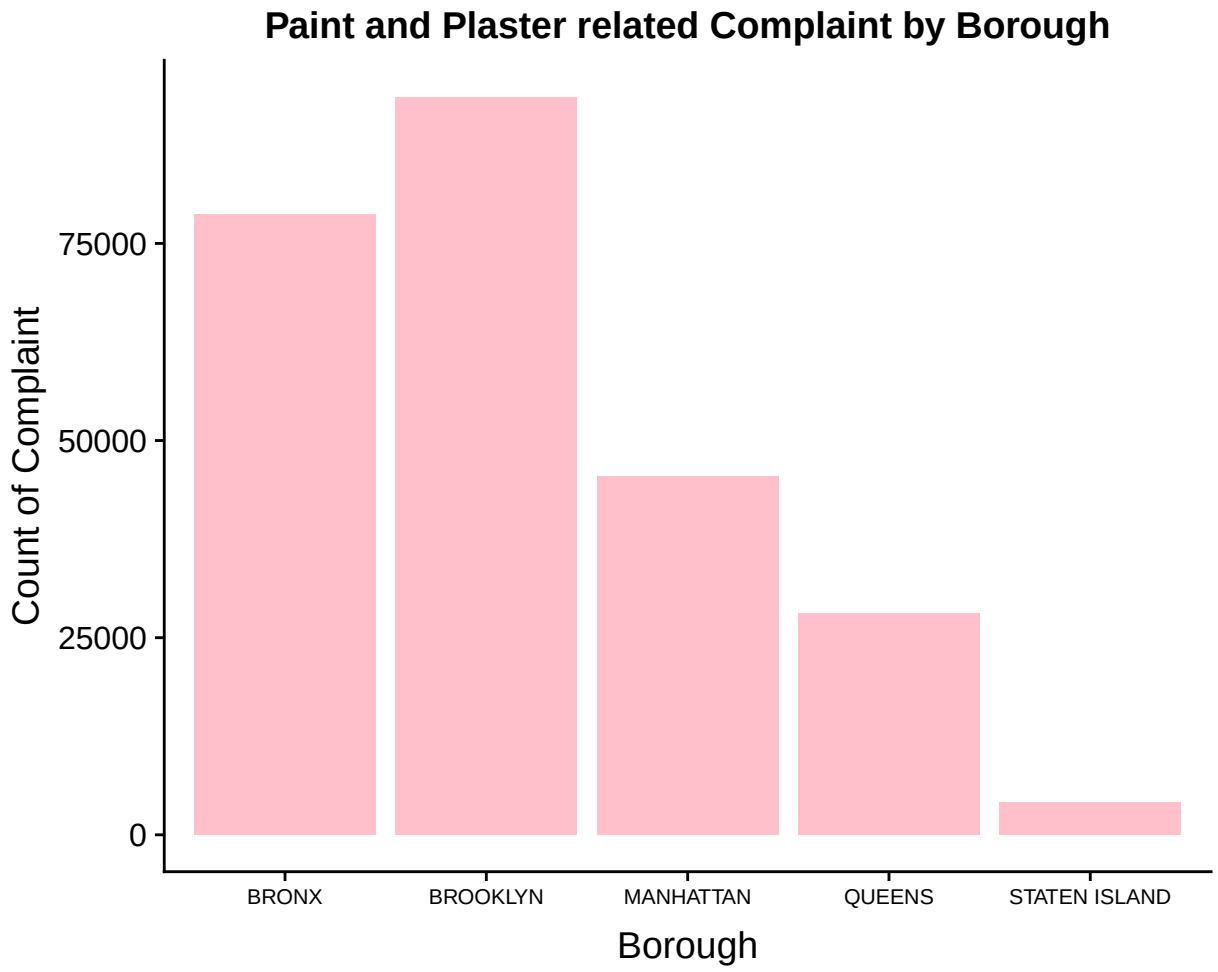


Figure 15: Different complaints, different Boroughs

Calculating closure time of complaints by Borough.

Removing objects from the Global Environment. This will keep the workspace clean, free the memory. This will allow efficient and faster processing to carry forward other tasks.

```
# Deleting Large Data Frames that are not required for analysis after this chunk so that our operations
rm(list=c("newdf", "watersys", "waterplot", "Freqs", "Freqs.complaints_by_city_top_20", "Freqs.complain
```

Getting the closure time by calculating the difference between the date on which complaint ticket was created and the date when complaint ticket was closed.

```
## we are calculating the differnce of 'Closed Date' and 'Created Date' of an incidents and storing it
nyc.df$Avg_Date = as.numeric(difftime(nyc.df$`Closed Date`, nyc.df$`Created Date`))
nyc.df$Avg_Date <- (nyc.df$Avg_Date/3600)
head(nyc.df)
```

```
##    Unique Key        Created Date         Closed Date Agency
## 1:   30387854 2015-04-14 02:14:40 2015-04-14 03:03:22   NYPD
## 2:   30388338 2015-04-14 02:10:12                <NA>   NYPD
## 3:   30395236 2015-04-14 02:03:01                <NA>   NYPD
## 4:   30394595 2015-04-14 02:02:40                <NA>   NYPD
## 5:   30390517 2015-04-14 02:00:04 2015-04-14 02:47:33   NYPD
```

```
## 6:   30389560 2015-04-14 01:52:15 2015-04-14 02:11:10   NYPD
##                        Agency Name          Complaint Type
## 1: New York City Police Department                 Vending
## 2: New York City Police Department        Blocked Driveway
## 3: New York City Police Department Noise - Street/Sidewalk
## 4: New York City Police Department Noise - Street/Sidewalk
## 5: New York City Police Department Noise - Street/Sidewalk
## 6: New York City Police Department Noise - Street/Sidewalk
##           Descriptor   Location Type Incident Zip
## 1: In Prohibited Area Street/Sidewalk        10465
## 2:          No Access Street/Sidewalk        11234
## 3:   Loud Music/Party Street/Sidewalk        11204
## 4:       Loud Talking Street/Sidewalk        11211
## 5:       Loud Talking Street/Sidewalk        10025
## 6:       Loud Talking Street/Sidewalk        11205
##            Incident Address         Street Name   Cross Street 1
## 1: 3775 EAST TREMONT AVENUE EAST TREMONT AVENUE   RANDALL AVENUE
## 2:        1524 RYDER STREET        RYDER STREET FLATLANDS AVENUE
## 3:                     <NA>                <NA>             <NA>
## 4:  361 METROPOLITAN AVENUE METROPOLITAN AVENUE HAVEMEYER STREET
## 5:                     <NA>                <NA>             <NA>
## 6:                     <NA>                <NA>             <NA>
##      Cross Street 2 Intersection Street 1 Intersection Street 2
## 1: ROOSEVELT AVENUE                  <NA>                  <NA>
## 2:         AVENUE P                  <NA>                  <NA>
## 3:             <NA>             71 STREET             16 AVENUE
## 4: HAVEMEYER STREET                  <NA>                  <NA>
## 5:             <NA>      WEST  104 STREET       COLUMBUS AVENUE
## 6:             <NA>        ST JAMES PLACE      LAFAYETTE AVENUE
##    Address Type     City Landmark Facility Type   Status
## 1:      ADDRESS    BRONX     <NA>      Precinct   Closed
## 2:      ADDRESS BROOKLYN     <NA>      Precinct     Open
## 3: INTERSECTION BROOKLYN     <NA>      Precinct     Open
## 4:      ADDRESS BROOKLYN     <NA>      Precinct Assigned
## 5: INTERSECTION NEW YORK     <NA>      Precinct   Closed
## 6: INTERSECTION BROOKLYN     <NA>      Precinct   Closed
##                  Due Date Resolution Action Updated Date Community Board
## 1: 04/14/2015 10:14:40 AM         04/14/2015 03:03:05 AM        10 BRONX
## 2: 04/14/2015 10:10:12 AM                           <NA>     18 BROOKLYN
## 3: 04/14/2015 10:03:01 AM                           <NA>     11 BROOKLYN
## 4: 04/14/2015 10:02:40 AM         04/14/2015 02:10:32 AM     01 BROOKLYN
## 5: 04/14/2015 10:00:04 AM         04/14/2015 02:04:59 AM    07 MANHATTAN
## 6: 04/14/2015 09:52:15 AM         04/14/2015 02:11:10 AM     02 BROOKLYN
##      Borough X Coordinate (State Plane) Y Coordinate (State Plane)
## 1:     BRONX                   1033758                       240162
## 2:  BROOKLYN                   1001544                       164726
## 3:  BROOKLYN                    984678                       164647
## 4:  BROOKLYN                    996477                       199445
## 5: MANHATTAN                    994260                       229982
## 6:  BROOKLYN                    994009                       190054
##    Park Facility Name Park Borough School Name School Number School Region
## 1:        Unspecified        BRONX Unspecified   Unspecified   Unspecified
## 2:        Unspecified     BROOKLYN Unspecified   Unspecified   Unspecified
## 3:        Unspecified     BROOKLYN Unspecified   Unspecified   Unspecified
```

```
## 4:        Unspecified     BROOKLYN Unspecified   Unspecified   Unspecified
## 5:        Unspecified    MANHATTAN Unspecified   Unspecified   Unspecified
## 6:        Unspecified     BROOKLYN Unspecified   Unspecified   Unspecified
##    School Code School Phone Number School Address School City School State
## 1: Unspecified         Unspecified    Unspecified Unspecified  Unspecified
## 2: Unspecified         Unspecified    Unspecified Unspecified  Unspecified
## 3: Unspecified         Unspecified    Unspecified Unspecified  Unspecified
## 4: Unspecified         Unspecified    Unspecified Unspecified  Unspecified
## 5: Unspecified         Unspecified    Unspecified Unspecified  Unspecified
## 6: Unspecified         Unspecified    Unspecified Unspecified  Unspecified
##     School Zip School Not Found School or Citywide Complaint Vehicle Type
## 1: Unspecified               N                         <NA>         <NA>
## 2: Unspecified               N                         <NA>         <NA>
## 3: Unspecified               N                         <NA>         <NA>
## 4: Unspecified               N                         <NA>         <NA>
## 5: Unspecified               N                         <NA>         <NA>
## 6: Unspecified               N                         <NA>         <NA>
##    Taxi Company Borough Taxi Pick Up Location Bridge Highway Name
## 1:                 <NA>                  <NA>                <NA>
## 2:                 <NA>                  <NA>                <NA>
## 3:                 <NA>                  <NA>                <NA>
## 4:                 <NA>                  <NA>                <NA>
## 5:                 <NA>                  <NA>                <NA>
## 6:                 <NA>                  <NA>                <NA>
##    Bridge Highway Direction Road Ramp Bridge Highway Segment
## 1:                     <NA>      <NA>                   <NA>
## 2:                     <NA>      <NA>                   <NA>
## 3:                     <NA>      <NA>                   <NA>
## 4:                     <NA>      <NA>                   <NA>
## 5:                     <NA>      <NA>                   <NA>
## 6:                     <NA>      <NA>                   <NA>
##    Garage Lot Name Ferry Direction Ferry Terminal Name Latitude Longitude
## 1:            <NA>            <NA>                <NA> 40.82573 -73.82111
## 2:            <NA>            <NA>                <NA> 40.61879 -73.93771
## 3:            <NA>            <NA>                <NA> 40.61859 -73.99846
## 4:            <NA>            <NA>                <NA> 40.71410 -73.95589
## 5:            <NA>            <NA>                <NA> 40.79792 -73.96385
## 6:            <NA>            <NA>                <NA> 40.68833 -73.96481
##                                    Location  Avg_Date
## 1:   (40.8257259931145, -73.82111429330192) 0.8116667
## 2: (40.618794391821936, -73.93770589155426)        NA
## 3:  (40.61859442131066, -73.99845832101916)        NA
## 4:  (40.71409874640673, -73.95589458206499)        NA
## 5:  (40.79791780509379, -73.96384631347463) 0.7913889
## 6:  (40.68832571866554, -73.96481079590191) 0.3152778
```

We observed that many rows in closed date or created date were empty and hence our Avg date was showing NA or sometimes coming out to be negative (in some cases created date was greater than the closed date), so we considered the data where the difference of closed and created date was positive or in short the closed date was greater than the created date, so we took Avg date which is greater than 0. This way we got rid of unnecessary data that was not useful for our exploratory analysis.

```
nyc.df <- nyc.df[nyc.df$Avg_Date>=0,]
```

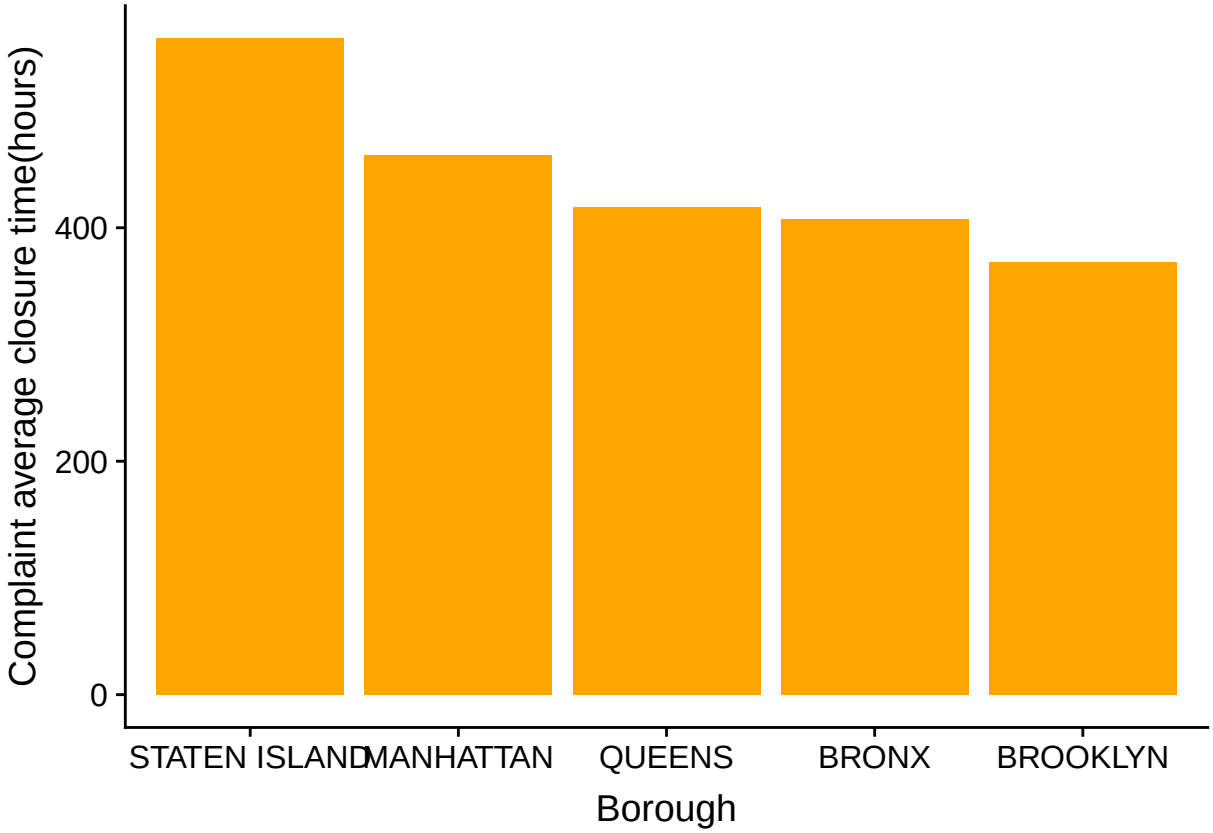
Here we calculated the mean average time it takes to close the complaint by Borough

```
df_time_to_closure <- aggregate(nyc.df$Avg_Date, list(nyc.df$Borough), mean, na.action = NULL, na.rm = T
df_time_to_closure
```

```
##         Group.1        x
## 1         BRONX 407.3171
## 2      BROOKLYN 370.3558
## 3     MANHATTAN 462.3753
## 4        QUEENS 417.5019
## 5 STATEN ISLAND 562.1765
```

It can be seen here that the mean time (in hours) of incident closure is maximum in Staten Island.

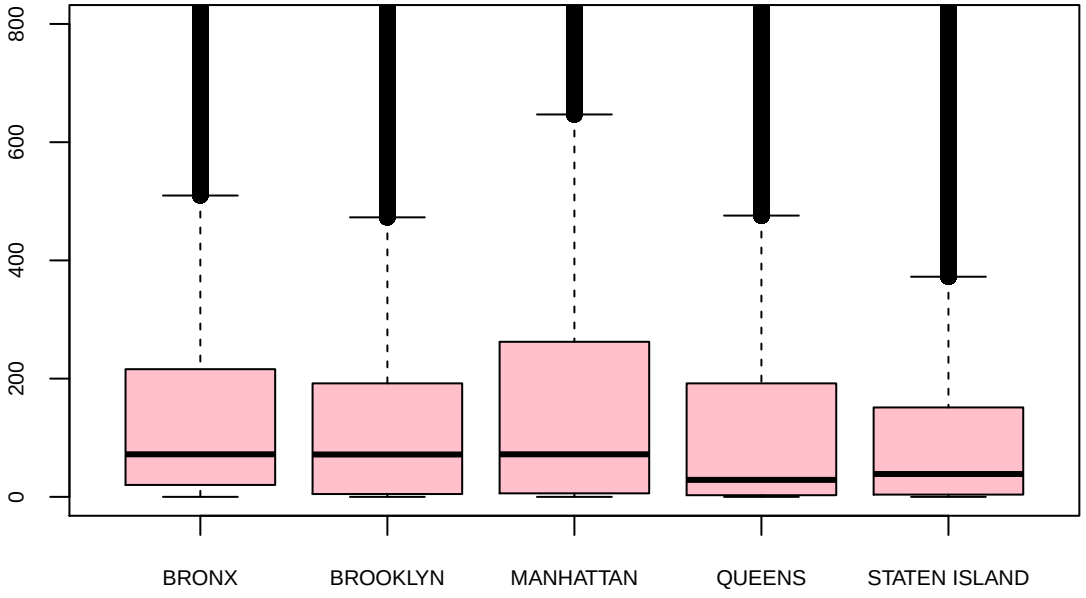
```
ggplot(df_time_to_closure, aes(x = reorder(df_time_to_closure$Group.1, -df_time_to_closure$x), y = df_ti
         geom_bar(stat = "identity", fill="orange") +
         xlab("Borough") + ylab("Complaint average closure time(hours)")
```



The below boxplot shows the minimum, first quartile, median, third quartile and maximum time (in hours) it is taken by every borough to solve the complaints. So it can be seen the median time of closure for QUEENS is minimum. We discussed about how QUEENS has a lot of problems. But looks like their problem resolution time is comparitively faster than other Boroughs. The number of complaints take lesser time to resolve and problems in Manhattan never gets resolved. So as we hypothesized earlier about Manhattans law and order being efficient, looks like we were wrong! And our second hypothesis about Manhattan being a smaller Borough can be well justified from this boxplot atleast. Staten Island still does a great job in this department.

For all the boroughs, Box-plot is rightly skewed which means that the mean is greater than the median or average time of closure of complaint for all the Boroughs are greater than the median of the complaints.

```
boxplot(nyc.df$Avg_Date ~ nyc.df$Borough, nyc.df, ylim=c(0,800), col = "pink", par(cex.axis=0.70))
```



NYPD collision dataset

Installing and Loading Important Packages

Data loading, cleaning and transformation for analysis

Loading the CSV file data-frame

```
nyccols.df <- fread(file="./NYPD_Motor_Vehicle_Collisions.csv", na.strings=c("","NA"), header=TRUE, sep
```

Finding missing values in the data

```
nyccols.df %>% select(everything()) %>% summarise_all(funs(sum(is.na(.))))
```

```
##   DATE TIME BOROUGH ZIP CODE LATITUDE LONGITUDE LOCATION ON STREET NAME
## 1    0    0  394402   394563   223550    223550   223550         296813
##   CROSS STREET NAME OFF STREET NAME NUMBER OF PERSONS INJURED
## 1            404326         1192608                        0
##   NUMBER OF PERSONS KILLED NUMBER OF PEDESTRIANS INJURED
## 1                        0                             0
##   NUMBER OF PEDESTRIANS KILLED NUMBER OF CYCLIST INJURED
## 1                            0                         0
##   NUMBER OF CYCLIST KILLED NUMBER OF MOTORIST INJURED
## 1                        0                         0
##   NUMBER OF MOTORIST KILLED CONTRIBUTING FACTOR VEHICLE 1
## 1                         0                          7300
```

```
##   CONTRIBUTING FACTOR VEHICLE 2 CONTRIBUTING FACTOR VEHICLE 3
## 1                        194663                        1264201
##   CONTRIBUTING FACTOR VEHICLE 4 CONTRIBUTING FACTOR VEHICLE 5 UNIQUE KEY
## 1                       1332559                       1346568          0
##   VEHICLE TYPE CODE 1 VEHICLE TYPE CODE 2 VEHICLE TYPE CODE 3
## 1               10816              214303             1234334
##   VEHICLE TYPE CODE 4 VEHICLE TYPE CODE 5
## 1             1305203             1341480
```

As we can see how an important attribute like BOROUGH has so many missing values. We have 2 options, either to impute the values to safe-guard loss of knowledge or just remove the rows. We will go by removing the rows in the subsequent sections.

Data Summary

```
kable(head(nyccols.df[, 1:9]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

DATE	TIME	BOROUGH	ZIP CODE	LATITUDE	LONGITUDE	LOCATION	ON STREET NAME	CROSS STREET NAME
09/24/2018	0:00	STATEN ISLAND	10306	40.57322	-74.10699	(40.57322, -74.106995)	HYLAN BOULEVARD	OTIS AVENUE
09/24/2018	0:00	NA	NA	40.61261	-74.07696	(40.612614, -74.07696)	FLETCHER STREET	NA
09/24/2018	0:00	NA	NA	NA	NA	NA	NA	NA
09/24/2018	0:00	NA	NA	40.83398	-73.82635	(40.83398, -73.82635)	BRUCKNER EXPRESSWAY	NA
09/24/2018	0:05	BROOKLYN	11249	40.72007	-73.95985	(40.720074, -73.95985)	WYTHE AVENUE	NORTH 8 STREET
09/24/2018	0:28	BROOKLYN	11210	40.63128	-73.94603	(40.631283, -73.94603)	NA	NA

```
kable(head(nyccols.df[, 10:18]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

OFF STREET NAME	NUMBER OF PERSONS INJURED	NUMBER OF PERSONS KILLED	NUMBER OF PEDESTRIANS INJURED	NUMBER OF PEDESTRIANS KILLED	NUMBER OF CYCLIST INJURED	NUMBER OF CYCLIST KILLED	NUMBER OF MOTORIST INJURED	NUMBER OF MOTORIST KILLED
NA	0	0	0	0	0	0	0	0
NA	0	0	0	0	0	0	0	0
161-18 140 street	0	0	0	0	0	0	0	0
NA	1	0	0	0	0	0	1	0
NA	0	0	0	0	0	0	0	0
1609 FLATBUSH AVENUE	0	0	0	0	0	0	0	0

```
kable(head(nyccols.df[, 19:29]), "latex", booktabs = T) %>%
  kable_styling(latex_options = c("striped", "scale_down", "HOLD_position"))
```

CONTRIBUTING FACTOR VEHICLE 1	CONTRIBUTING FACTOR VEHICLE 2	CONTRIBUTING FACTOR VEHICLE 3	CONTRIBUTING FACTOR VEHICLE 4	CONTRIBUTING FACTOR VEHICLE 5	UNIQUE KEY	VEHICLE TYPE CODE 1	VEHICLE TYPE CODE 2	VEHICLE TYPE CODE 3	VEHICLE TYPE CODE 4	VEHICLE TYPE CODE 5
Driver Inattention/Distraction	Unspecified	NA	NA	NA	3987045	Station Wagon/Sport Utility Vehicle	Sedan	NA	NA	NA
Driver Inattention/Distraction	Unspecified	NA	NA	NA	3987187	Station Wagon/Sport Utility Vehicle	Sedan	NA	NA	NA
Unspecified	NA	NA	NA	NA	3986792	Sedan	NA	NA	NA	NA
Unsafe Speed	Unspecified	NA	NA	NA	3986861	Sedan	Sedan	NA	NA	NA
Failure to Yield Right-of-Way	Unspecified	NA	NA	NA	3986532	Sedan	Sedan	NA	NA	NA
Driver Inattention/Distraction	Driver Inattention/Distraction	NA	NA	NA	3986656	Sedan	Sedan	NA	NA	NA

Creating proper headings of columns.

```
names(nyccols.df) <- str_replace_all(names(nyccols.df), c(" " = "." , "," = ""))
colnames(nyccols.df)
```

```
##  [1] "DATE"                          "TIME"
##  [3] "BOROUGH"                       "ZIP.CODE"
##  [5] "LATITUDE"                      "LONGITUDE"
##  [7] "LOCATION"                      "ON.STREET.NAME"
##  [9] "CROSS.STREET.NAME"             "OFF.STREET.NAME"
## [11] "NUMBER.OF.PERSONS.INJURED"     "NUMBER.OF.PERSONS.KILLED"
## [13] "NUMBER.OF.PEDESTRIANS.INJURED" "NUMBER.OF.PEDESTRIANS.KILLED"
## [15] "NUMBER.OF.CYCLIST.INJURED"     "NUMBER.OF.CYCLIST.KILLED"
## [17] "NUMBER.OF.MOTORIST.INJURED"    "NUMBER.OF.MOTORIST.KILLED"
```

```
## [19] "CONTRIBUTING.FACTOR.VEHICLE.1" "CONTRIBUTING.FACTOR.VEHICLE.2"
## [21] "CONTRIBUTING.FACTOR.VEHICLE.3" "CONTRIBUTING.FACTOR.VEHICLE.4"
## [23] "CONTRIBUTING.FACTOR.VEHICLE.5" "UNIQUE.KEY"
## [25] "VEHICLE.TYPE.CODE.1"           "VEHICLE.TYPE.CODE.2"
## [27] "VEHICLE.TYPE.CODE.3"           "VEHICLE.TYPE.CODE.4"
## [29] "VEHICLE.TYPE.CODE.5"
```

We can see how column names now have dots in place. This way columns are now more accessible. We can also see from the columns, either people are getting injured or killed as a result of the collisions. We need to consolidate them by some means. We will do that in the subsequent sections.

Lets first summarize column namely "BOROUGH"

```
summary(nyccols.df$BOROUGH)
```

```
##    Length     Class      Mode
##   1351214 character character
```

We see that "BOROUGH" has a class that is character, whereas a "BOROUGH" is supposed to be a factor. So converting "BOROUGH" to factor.

Converting Borough to Factors.

```
nyccols.df$BOROUGH <- as.factor(nyccols.df$BOROUGH)
unique(nyccols.df$BOROUGH)
```

```
## [1] STATEN ISLAND <NA>          BROOKLYN      QUEENS        MANHATTAN
## [6] BRONX
## Levels: BRONX BROOKLYN MANHATTAN QUEENS STATEN ISLAND
```

```
summary(nyccols.df$BOROUGH)
```

```
##         BRONX      BROOKLYN     MANHATTAN        QUEENS STATEN ISLAND
##        129619        295403        236959        251365         43466
##          NA's
##        394402
```

We can now see the "BOROUGH" has been converted to factor. Which has 6 values: BRONX, BROOKLYN, MANHATTAN, QUEENS, STATEN ISLAND, NA's. As we can see NA's This means we have to remove the NA value from the "BOROUGH" column. This is done here-under:

```
nyccols.df <- nyccols.df[!is.na(nyccols.df$BOROUGH), ]
unique(nyccols.df$BOROUGH)
```

```
## [1] STATEN ISLAND BROOKLYN      QUEENS        MANHATTAN     BRONX
## Levels: BRONX BROOKLYN MANHATTAN QUEENS STATEN ISLAND
```

We can now see there are only 5 unique BOROUGHS. The NA values have been removed.

Summarizing and improving "DATE" column

```
summary(nyccols.df$DATE)
```

```
##    Length     Class      Mode
##    956812 character character
```

We can see the "DATE" column is not in its proper format. It has a character class. Which is not correct for a date to have. If we want to do some kind of analysis that involves date we need to have it in the correct format. Therefore in the next line we are converting "DATE" into its proper format.

```
nyccols.df$DATE <- as.Date(nyccols.df$DATE, "%m/%d/%Y")
summary(nyccols.df$DATE)
```

```
##         Min.      1st Qu.       Median         Mean      3rd Qu.
## "2012-07-01" "2014-01-15" "2015-07-23" "2015-07-27" "2017-01-28"
##         Max.
## "2018-09-24"
```

Date is now in its proper format!

Lets re-arrange data, by columns properly.

Unique Key should be the first column. So lets do that:

```
colnames(nyccols.df)
```

```
##  [1] "DATE"                          "TIME"
##  [3] "BOROUGH"                       "ZIP.CODE"
##  [5] "LATITUDE"                      "LONGITUDE"
##  [7] "LOCATION"                      "ON.STREET.NAME"
##  [9] "CROSS.STREET.NAME"             "OFF.STREET.NAME"
## [11] "NUMBER.OF.PERSONS.INJURED"     "NUMBER.OF.PERSONS.KILLED"
## [13] "NUMBER.OF.PEDESTRIANS.INJURED" "NUMBER.OF.PEDESTRIANS.KILLED"
## [15] "NUMBER.OF.CYCLIST.INJURED"     "NUMBER.OF.CYCLIST.KILLED"
## [17] "NUMBER.OF.MOTORIST.INJURED"    "NUMBER.OF.MOTORIST.KILLED"
## [19] "CONTRIBUTING.FACTOR.VEHICLE.1" "CONTRIBUTING.FACTOR.VEHICLE.2"
## [21] "CONTRIBUTING.FACTOR.VEHICLE.3" "CONTRIBUTING.FACTOR.VEHICLE.4"
## [23] "CONTRIBUTING.FACTOR.VEHICLE.5" "UNIQUE.KEY"
## [25] "VEHICLE.TYPE.CODE.1"           "VEHICLE.TYPE.CODE.2"
## [27] "VEHICLE.TYPE.CODE.3"           "VEHICLE.TYPE.CODE.4"
## [29] "VEHICLE.TYPE.CODE.5"
```

We can see the column names from the above. Let arrange them using the dplyr package.

```
nyccols.df = nyccols.df %>% select("UNIQUE.KEY","BOROUGH","DATE","TIME","ZIP.CODE","LATITUDE","LONGITUDE
```

```
colnames(nyccols.df)
```

```
##  [1] "UNIQUE.KEY"                    "BOROUGH"
##  [3] "DATE"                          "TIME"
##  [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] "VEHICLE.TYPE.CODE.1"           "VEHICLE.TYPE.CODE.2"
## [27] "VEHICLE.TYPE.CODE.3"           "VEHICLE.TYPE.CODE.4"
## [29] "VEHICLE.TYPE.CODE.5"
```

We can see now the columnNames are in accordance to what we require.

Data Analysis

Making a location Data-frame

This will make it easy for exploratory data analysis. Aggregating BOROUGH, ZIP.CODE, LATITUDE, LONGITUDE, LOCATION.

```
loc_df <- nyccols.df[, c(2, 5, 6, 7, 8)]
kable(head(loc_df), "latex", booktabs = T, caption="Location data-frame") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 4: Location data-frame

BOROUGH	ZIP.CODE	LATITUDE	LONGITUDE	LOCATION
STATEN ISLAND	10306	40.57322	-74.10699	(40.57322, -74.106995)
BROOKLYN	11249	40.72007	-73.95985	(40.720074, -73.95985)
BROOKLYN	11210	40.63128	-73.94603	(40.631283, -73.94603)
QUEENS	11385	40.69385	-73.90118	(40.693848, -73.901184)
BROOKLYN	11201	40.70291	-73.98113	(40.702908, -73.98113)
BROOKLYN	11201	40.69469	-73.98174	(40.694687, -73.98174)

Some analysis. Lets Consolidate how many people got injured.

NUMBER.OF.PERSONS.INJURED, NUMBER.OF.PEDESTRIANS.INJURED, NUMBER.OF.CYCLIST.INJURED, NUMBER.OF.MOTORIST.INJURED aggregating these columns will be aggregated to form one data-frame for easy access and manipulation.

```
# Getting all the columns and saving it in a data frame namely injured
injured <- nyccols.df[,c(12, 14, 16, 18)]
kable(head(injured), "latex", booktabs = T, caption = "Injured data-frame") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 5: Injured data-frame

NUMBER.OF.PERSONS.INJURED	NUMBER.OF.PEDESTRIANS.INJURED	NUMBER.OF.CYCLIST.INJURED	NUMBER.OF.MOTORIST.INJURED
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Now tell me, does it matter whether it was the cyclist who got injured or a motorist? Maybe from specifc data analysis point of view it may be of a great importance, but in general, its a person only who got injured. Lets now add up the numbers in these columns for each row. So for every row number of injured people will be Injured = NUMBER.OF.PERSONS.INJURED + NUMBER.OF.PEDESTRIANS.INJURED + NUMBER.OF.CYCLIST.INJURED + NUMBER.OF.MOTORIST.INJURED and similarly carry out for Killed.

Finding total Injured and adding it to data frame

```
injured$total <- rowSums(injured)
loc_df$INJURED <- injured$total
kable(head(loc_df), "latex", booktabs = T, caption = "Adding total injured to location data-frame") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 6: Adding total injured to location data-frame

BOROUGH	ZIP.CODE	LATITUDE	LONGITUDE	LOCATION	INJURED
STATEN ISLAND	10306	40.57322	-74.10699	(40.57322, -74.106995)	0
BROOKLYN	11249	40.72007	-73.95985	(40.720074, -73.95985)	0
BROOKLYN	11210	40.63128	-73.94603	(40.631283, -73.94603)	0
QUEENS	11385	40.69385	-73.90118	(40.693848, -73.901184)	0
BROOKLYN	11201	40.70291	-73.98113	(40.702908, -73.98113)	0
BROOKLYN	11201	40.69469	-73.98174	(40.694687, -73.98174)	0

Lets Consolidate how many people got Killed Columns:

We will be working on these columns: NUMBER.OF.PERSONS.KILLED, NUMBER.OF.PEDESTRIANS.KILLED, NUMBER.OF.CYCLIST.KILLED, NUMBER.OF.MOTORIST.KILLED. So for every row number of killed people will be: killed = NUMBER.OF.PERSONS.KILLED + NUMBER.OF.PEDESTRIANS.KILLED + NUMBER.OF.CYCLIST.KILLED + NUMBER.OF.MOTORIST.KILLED and similarly carry out for Killed.

```
# Getting all the columns and saving it in a data frame namely killed
killed <- nyccols.df[,c(13, 15, 17, 19)]
kable(head(killed), "latex", booktabs = T, caption = "Killed data-frame") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 7: Killed data-frame

NUMBER.OF.PERSONS.KILLED	NUMBER.OF.PEDESTRIANS.KILLED	NUMBER.OF.CYCLIST.KILLED	NUMBER.OF.MOTORIST.KILLED
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Finding total Killed and adding it to data frame

```
killed$total <- rowSums(killed)
loc_df$KILLED <- killed$total
```

```
kable(head(loc_df), "latex", booktabs = T, caption = "Adding total killed to location data-frame") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 8: Adding total killed to location data-frame

BOROUGH	ZIP.CODE	LATITUDE	LONGITUDE	LOCATION	INJURED	KILLED
STATEN ISLAND	10306	40.57322	-74.10699	(40.57322, -74.106995)	0	0
BROOKLYN	11249	40.72007	-73.95985	(40.720074, -73.95985)	0	0
BROOKLYN	11210	40.63128	-73.94603	(40.631283, -73.94603)	0	0
QUEENS	11385	40.69385	-73.90118	(40.693848, -73.901184)	0	0
BROOKLYN	11201	40.70291	-73.98113	(40.702908, -73.98113)	0	0
BROOKLYN	11201	40.69469	-73.98174	(40.694687, -73.98174)	0	0

Finding injuries by hour due to collisions

```
injured$TIME <- nyccols.df$TIME
injured$HOUR <- hour(as.POSIXct(injured$TIME, format = "%H:%M"))
injured$BOROUGH <- nyccols.df$BOROUGH
kable(head(injured), "latex", booktabs = T, caption = "Injuries with respect to time in hours") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 9: Injuries with respect to time in hours

NUMBER.OF.PERSONS.INJURED	NUMBER.OF.PEDESTRIANS.INJURED	NUMBER.OF.CYCLIST.INJURED	NUMBER.OF.MOTORIST.INJURED	total	TIME	HOUR	BOROUGH
0	0	0	0	0	0:00	0	STATEN ISLAND
0	0	0	0	0	0:05	0	BROOKLYN
0	0	0	0	0	0:28	0	BROOKLYN
0	0	0	0	0	0:30	0	QUEENS
0	0	0	0	0	10:00	10	BROOKLYN
0	0	0	0	0	10:00	10	BROOKLYN

Getting to find accidents by hour

```
accident_by_hour <- injured %>% group_by(BOROUGH, HOUR) %>% summarise(Injured_people = sum(total))
accident_by_hour <- filter(accident_by_hour, BOROUGH != "")
kable(head(accident_by_hour), "latex", booktabs = T, caption = "Finding injury frequency over time") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Table 10: Finding injury frequency over time

BOROUGH	HOUR	Injured_people
BRONX	0	2116
BRONX	1	1217
BRONX	2	1009
BRONX	3	930
BRONX	4	1091
BRONX	5	964

```
ggplot(accident_by_hour, aes(HOUR, Injured_people, fill = BOROUGH)) +
  geom_bar(stat = "identity", show.legend = FALSE) + theme_calc() +
```

```
  ggtitle("People injured in collision according to time") +
  facet_grid(BOROUGH ~ .)
```

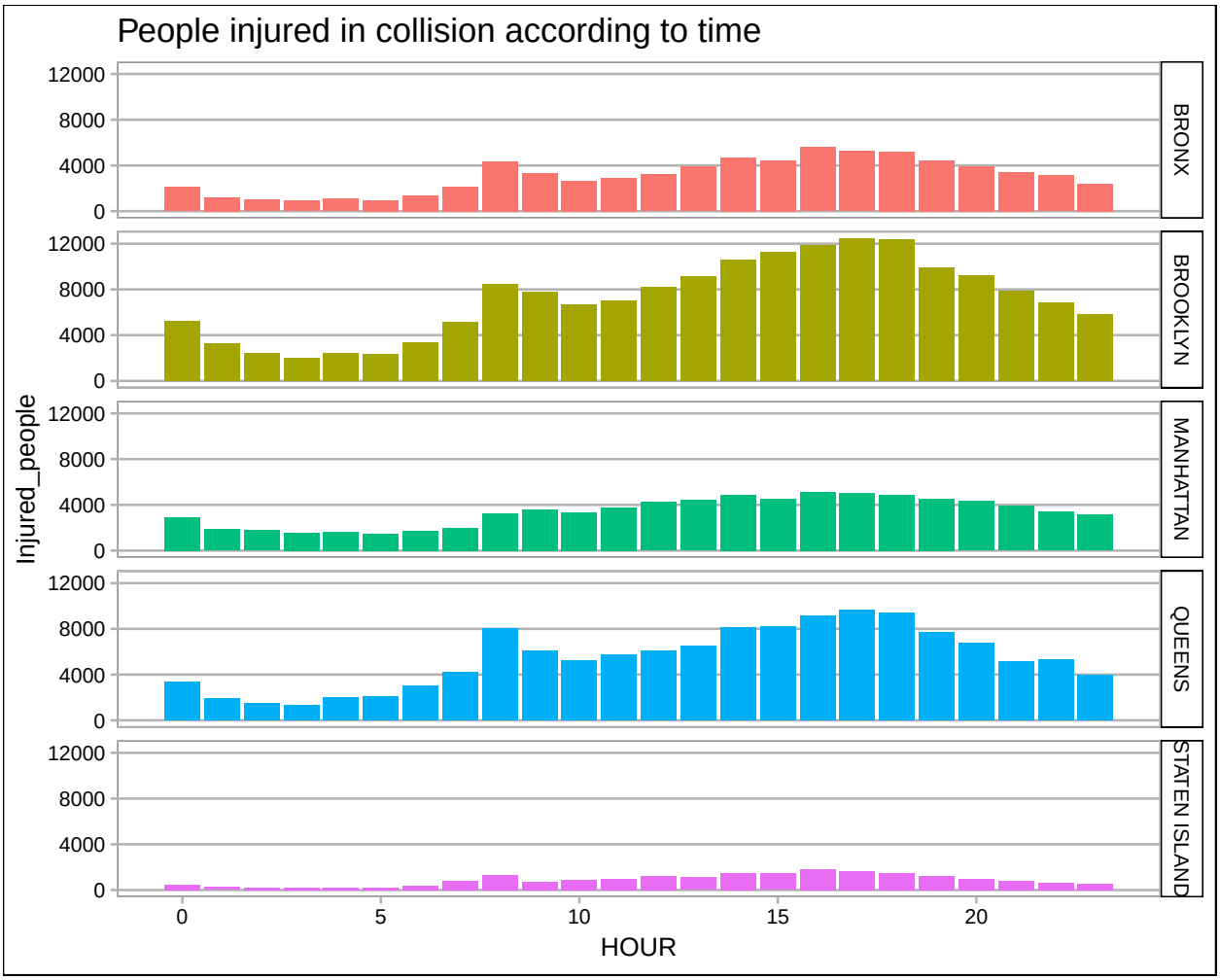


Figure 16: People injured due to collision with time

From above we can see a lot of injuries are happening to people due to collisions at around 8 AM till 8 PM (20:00 HRS). And there is a set pattern. All the BOROUGHS except BRONX show that the maxima is at 5 PM (17:00 HRS). Generally maxima is between 4 to 6 PM. This shows maximum injuries happen around this time. One speculation could be maybe because these are generally rush hour/office hours? Also, there is a spike at around 8 am as well. What does it tell? Generally people head to office during these hours. Why all this starts at around 2 PM? Looks like school hours! School timings in the US are generally 8:30 AM till 3:30 PM! So this is the speculation.

Finding how many people got killed by hour due to collisions

```
killed$TIME <- nyccols.df$TIME
killed$HOUR <- hour(as.POSIXct(killed$TIME, format = "%H:%M"))
killed$BOROUGH <- nyccols.df$BOROUGH
```

```
kable(head(killed), "latex", booktabs = T, caption = "People killed with respect to time in hours") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 11: People killed with respect to time in hours

NUMBER.OF.PERSONS.KILLED	NUMBER.OF.PEDESTRIANS.KILLED	NUMBER.OF.CYCLIST.KILLED	NUMBER.OF.MOTORIST.KILLED	total	TIME	HOUR	BOROUGH
0	0	0	0	0	0:00	0	STATEN ISLAND
0	0	0	0	0	0:05	0	BROOKLYN
0	0	0	0	0	0:28	0	BROOKLYN
0	0	0	0	0	0:30	0	QUEENS
0	0	0	0	0	10:00	10	BROOKLYN
0	0	0	0	0	10:00	10	BROOKLYN

Getting to find accidents by hour

```
accident_by_hour_killed <- killed %>% group_by(BOROUGH, HOUR) %>% summarise(killed_people = sum(total))
accident_by_hour_killed <- filter(accident_by_hour_killed, BOROUGH != "")
kable(head(accident_by_hour_killed), "latex", booktabs = T, caption = "Finding injury frequency over ti
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Table 12: Finding injury frequency over time

BOROUGH	HOUR	killed_people
BRONX	0	8
BRONX	1	12
BRONX	2	12
BRONX	3	8
BRONX	4	8
BRONX	5	10

```
ggplot(accident_by_hour_killed, aes(HOUR, killed_people, fill = BOROUGH)) +
  geom_bar(stat = "identity", show.legend = FALSE) + theme_calc() +
  ggtitle("People killed in collision according to time") +
  facet_grid(BOROUGH ~ .)
```

Figure 17: People killed in collision with time

There is not any set pattern as far a killings due to collisions are concerned. BROOKLYN is doing a really bad job! Maybe we can find out with further analysis of why is BROOKLYN doing so bad in this department.

Top reasons for collisions

```
reason_df <- loc_df
reason_df$Reason <- nyccols.df$CONTRIBUTING.FACTOR.VEHICLE.1
top10_reasons <- reason_df %>% group_by(Reason) %>% summarise(Total_accident = n()) %>% top_n(10, Total_
top10_reasons <- top10_reasons[order(top10_reasons$Total_accident, decreasing = TRUE),]
kable(top10_reasons, "latex", booktabs = T, caption = "Top 10 collision reasons") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Table 13: Top 10 collision reasons

Reason	Total_accident
Unspecified	464178
Driver Inattention/Distraction	143330
Failure to Yield Right-of-Way	51655
Backing Unsafely	35641
Other Vehicular	30177
Fatigued/Drowsy	26279
Turning Improperly	21327
Following Too Closely	19798
Lost Consciousness	15893
Passing or Lane Usage Improper	13317

Lets visualize it

```
ggplot(top10_reasons, aes(x = reorder(top10_reasons$Reason, -top10_reasons$Total_accident), y = top10_r
         geom_bar(stat = "identity", fill="blue") +
         scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
         coord_flip() +
         ylab("Frequency of Collision by Reason") +
         xlab("Reasons")+ ggtitle("Collision Reasons Frequency")
```

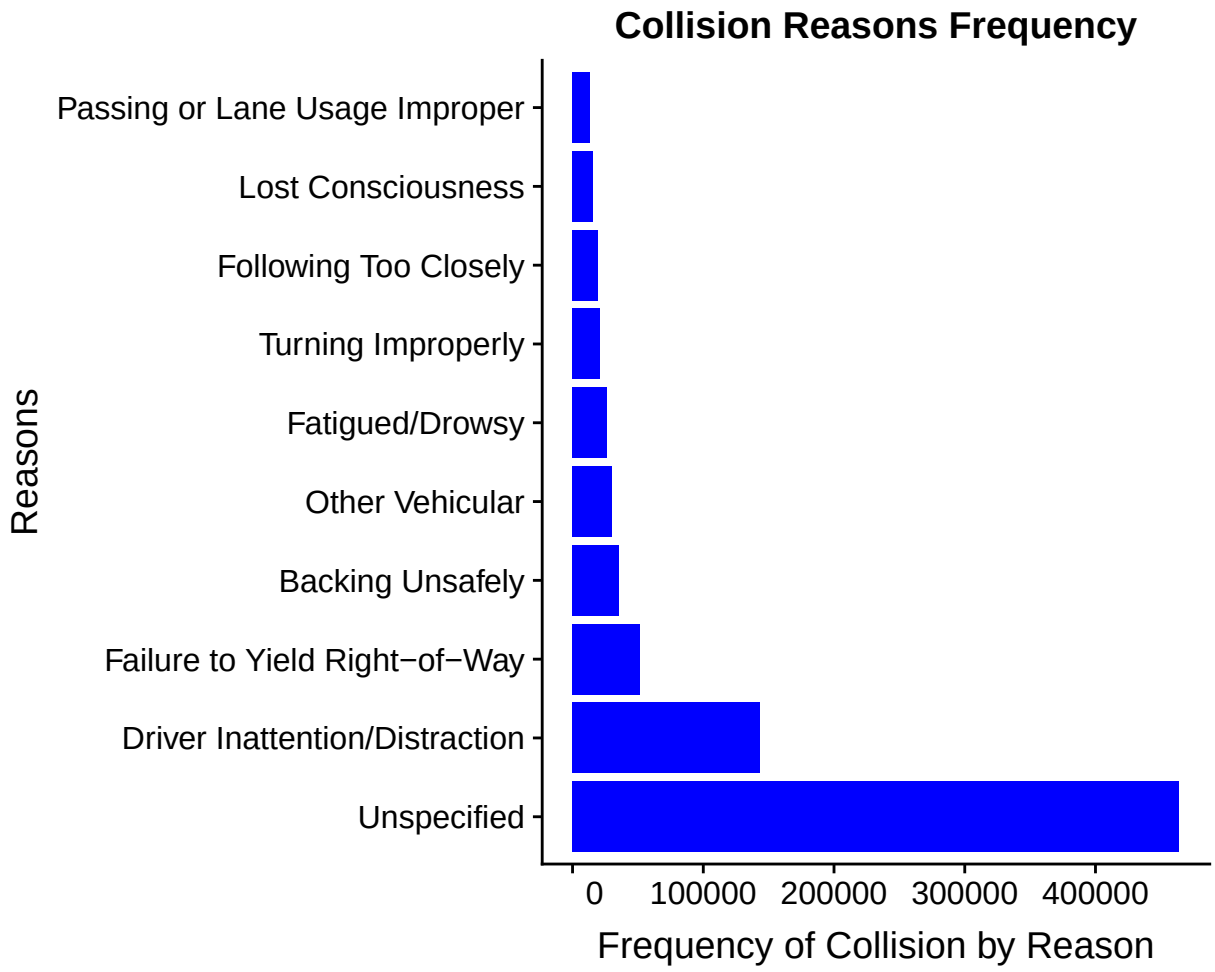


Figure 18: Collision Reasons and their frequencies

So, most of the times its "Driver Inattention/Distraction", that is leading to collisions. There are 2 shocking discoveries in our data-set. These are "Prescription Medication" scores higher than "Alcohol Involvement" and "Alcohol Involvement" surprisingly is not one of the top contender as always thought!

Contributing Vehicles to injuries

```
contri_vehicle <- loc_df[,c("BOROUGH", "INJURED")]
contri_vehicle$TYPE <- nyccols.df$VEHICLE.TYPE.CODE.1
top10_vehicle_types <- contri_vehicle %>% group_by(TYPE) %>%
  summarise(INJURED = sum(INJURED)) %>%
    top_n(10, INJURED)
top10_vehicle_types <- top10_vehicle_types[-c(1,4),]
top10_vehicle_types <- top10_vehicle_types[order(top10_vehicle_types$INJURED, decreasing = T),]
top10_vehicle_types <- top10_vehicle_types[-c(5, 10) , ]
top10_vehicle_types <- top10_vehicle_types[1:10, ]
kable(top10_vehicle_types, "latex", booktabs = T, caption = "Top 10 Vehicles Leading to Injuries") %>%
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Table 14: Top 10 Vehicles Leading to Injuries

TYPE	INJURED
PASSENGER VEHICLE	252735
SPORT UTILITY / STATION WAGON	106210
TAXI	20126
UNKNOWN	12748
VAN	8669
OTHER	7263
NA	5893
NA	NA
NA	NA
NA	NA

Lets visualize it

```
ggplot(top10_vehicle_types, aes(x = reorder(top10_vehicle_types$TYPE, -top10_vehicle_types$INJURED), y =
         geom_bar(stat = "identity", fill="blue") +
         scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
         coord_flip() + ylab("Injury count") +
         xlab("Contributing Vehicles") +
         ggtitle("Vehicles Causing Injuries")
```

```
## Warning: Removed 3 rows containing missing values (position_stack).
```

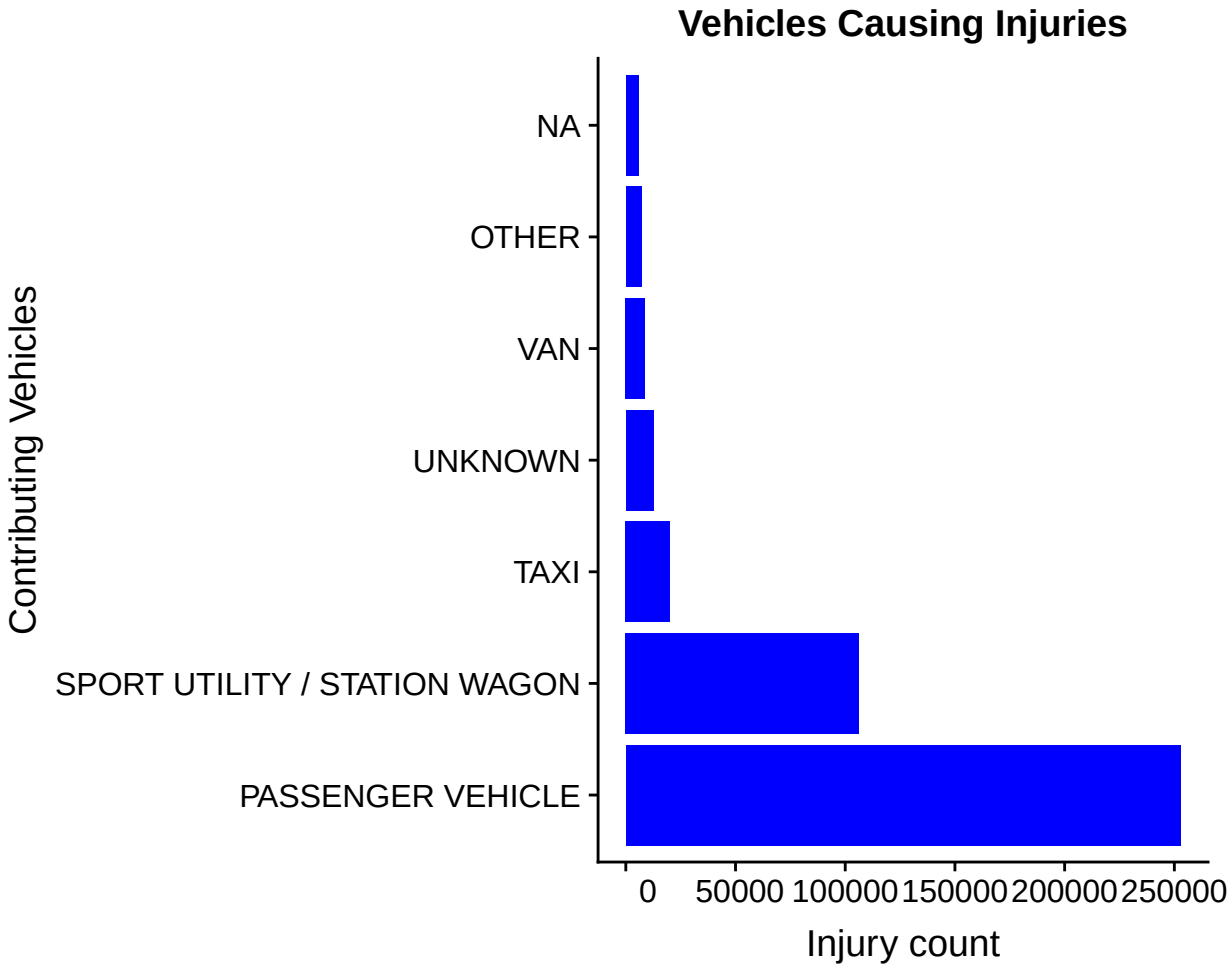


Figure 19: Top 10 Vehicles Causing Injuries

This shows that passenger vehicles are mostly involved in injuring people. Next SUVs, then TAXI and so on so forth!

Top 10 contributing vehicles to Killings

```
contri_vehicle_killed <- loc_df[,c("BOROUGH", "KILLED")]
contri_vehicle_killed$TYPE <- nyccols.df$VEHICLE.TYPE.CODE.1
top10_vehicle_types_killed <- contri_vehicle_killed %>%
  group_by(TYPE) %>%
  summarise(KILLED = sum(KILLED)) %>% top_n(20, KILLED)
top10_vehicle_types_killed <- top10_vehicle_types_killed[-c(1,4),]
top10_vehicle_types_killed <- top10_vehicle_types_killed[order(top10_vehicle_types_killed$KILLED, decrea
top10_vehicle_types_killed <- top10_vehicle_types_killed[-c(5, 9), ]
top10_vehicle_types_killed <- top10_vehicle_types_killed[1:10,]
kable(top10_vehicle_types_killed, "latex", booktabs = T, caption = "Top 10 Contributing Vehicles Leadin
  kable_styling(latex_options = c("striped", "HOLD_position"))
```

Table 15: Top 10 Contributing Vehicles Leading To Killings

TYPE	KILLED
PASSENGER VEHICLE	792
SPORT UTILITY / STATION WAGON	468
MOTORCYCLE	136
UNKNOWN	90
BUS	70
VAN	50
TAXI	48
OTHER	42
PICK-UP TRUCK	41
Station Wagon/Sport Utility Vehicle	38

Lets visualize it

```
ggplot(top10_vehicle_types_killed, aes(x = reorder(top10_vehicle_types_killed$TYPE, -top10_vehicle_types
         geom_bar(stat = "identity", fill="blue") +
         scale_y_continuous(labels=function(n){format(n, scientific = FALSE)}) +
         coord_flip() + ylab("Killings Count") +
         xlab("Contributing Vehicles") + ggtitle("Vehicles causing Killings") +
         theme(plot.title = element_text(hjust = 0.5))
```

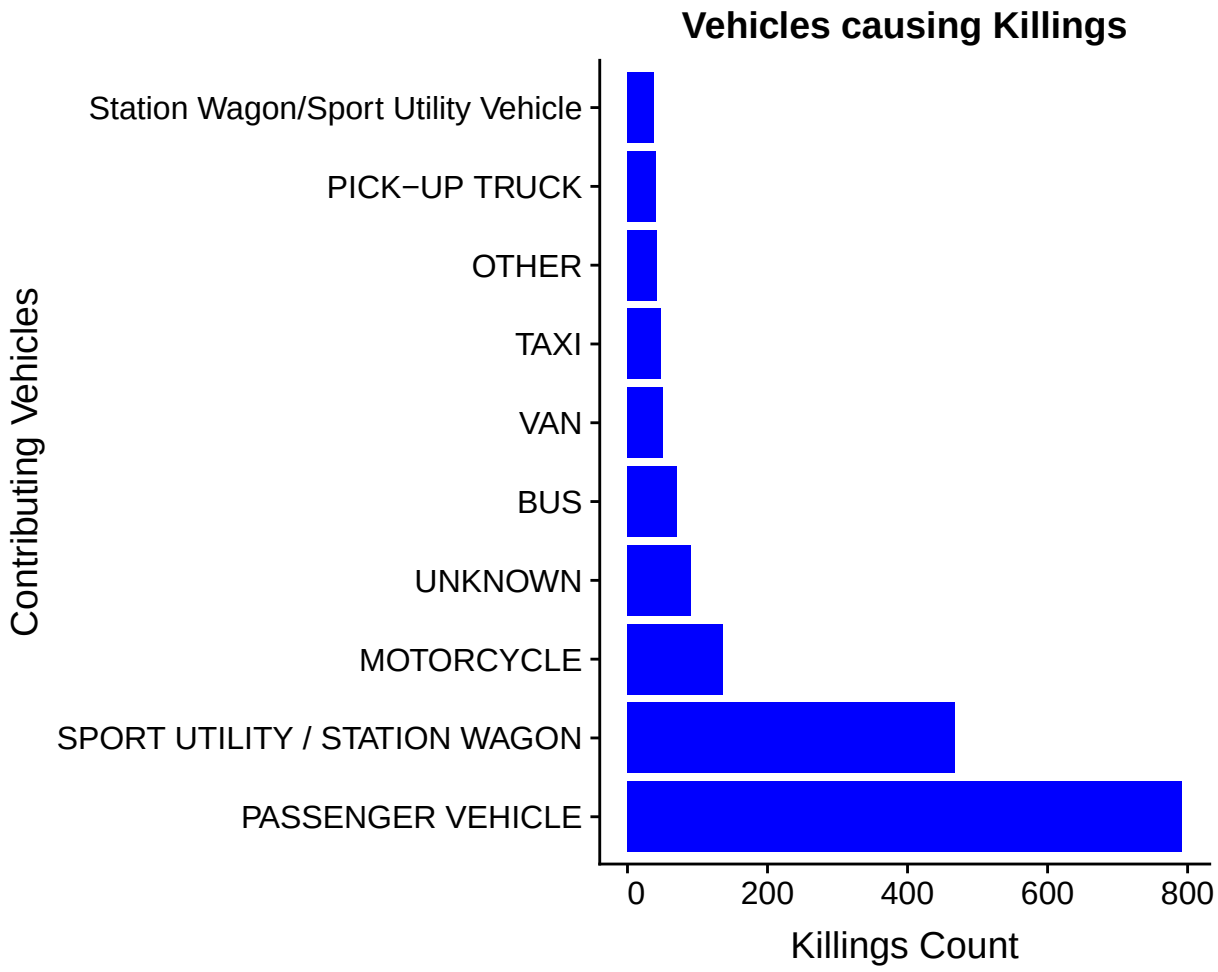


Figure 20: Top 10 Vehicles Causing Killings

Passenger Vehicle is as bad for killings as it is for injuries. So is SUV. Surprisingly Motorcycle? Is it the person riding Motorcycle got killed or did the motorcycle killed the person it collided with? My speculation is, its collision of a person on Motorcycle whether being hit by or did hit leads to a person being killed. Motorcycles are dangerous!

Data Dictionary

Creating Data dictionary for NYC 311

```
Row <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35
Variable.Names <- c('Address Type','Agency','Agency Name','Borough','Bridge Highway Direction','Bridge

Variable.Description <- c("Type of address, it includes just 5 entries Address, BlockFace, Intersection

Variable.Options <- c("ADDRESS,INTERSECTION,NA,LATLONG,BLOCKFACE,PLACENAME", "NYPD,TLC,DPR,DOT,CHALL,DOI
", "Mostly Empty", "Mostly Empty", "1 Missed Collection to Zoning - Non-Conforming/Illegal Vehicle Stora
", "Mostly Empty", "Precinct,N/A,NA,DSNY Garage,School", "Mostly Empty",  "Mostly Empty",  "Mostly Empty
", "Mostly Empty", "Mostly Empty", "110th Street and Lenox Avenue to Yellowstone Boulevard, 65th Road, 
```

```
", "ASTORIA to Unspecified", "Mostly Empty", "Alice Kornegay to Willow Lake Playground - PS 197", "Mostl

Variable.Type <- c("char","char","char","char","char","char","char","char","char","char","char","char",
dict_df <- data.frame(Row, Variable.Names, Variable.Description, Variable.Options, Variable.Type)

kable(dict_df , "latex", booktabs = T, caption = "NYC 311") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 16: NYC 311

Creating Data dictionary for NYPD Motor Vehicle Collision

```
# Data Dictionary
Row <- c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29)

Variable.Names <- c('Borough', 'Contributing Factor Vehicle 1', 'Contributing Factor Vehicle 2', 'Contri

Variable.Description <- c("Borough in which the incident occured", "There are 60 unique contributing fa

Variable.Options <- c("Borough contain 6 entries Staten Island, Brooklyn, Bronx, Queens, Manhattan and

Variable.Type <- c("char", "char", "char", "char", "char", "char", "char", "char", "dbl","char", "dbl",
dict_df <- data.frame(Row, Variable.Names, Variable.Description, Variable.Options, Variable.Type)

kable(dict_df , "latex", booktabs = T, caption = "Motor-vehicle Collision dataset") %>%
  kable_styling(latex_options = c("striped", "HOLD_position", "scale_down"))
```

Table 17: Motor-vehicle Collision dataset

Row	Variable.Names	Variable.Description	Variable.Options	Variable.Type
1	Borough	Borough in which the incident occured	Borough contain 6 entries Staten Island, Brooklyn, Bronx, Queens, Manhattan and NA	char
2	Contributing Factor Vehicle 1	There are 60 unique contributing factor causing the incident	Mostly contains Empty values	char
3	Contributing Factor Vehicle 2	This gaves the further description of the contributing factor, but is mostly empty	Mostly contains Empty value	char
4	Contributing Factor Vehicle 3	3rd contributing factor	Mostly contains Empty value	char
5	Contributing Factor Vehicle 4	4th contributing factor	Some rows are filled some are empty	char
6	Contributing Factor Vehicle 5	5th contributing factor	Mostly filled	char
7	Cross Street Name	Name of the cross street		char
8	Date	Date of the incident		char
9	Latitude	Latitude of the incident	Few entries contain NA in Latitude	dbl
10	Location	Location of the incident	Few entries contain NA in Location	char
11	Longitude	Longitude of the incident	Few entries contain NA in Longitude	dbl
12	Number of Cyclist Injured	Cyclist injured	Entries contain value from 0,1,2,3,4 and a weird value [1]	int
13	Number of Cyclist Killed	Exact number of the cyclist killed	Entries contain value from 0,1,2 and a weird value [1]	int
14	Number of Motorist Injured	Count of Motorist injured in an incident	Max value is 43	int
15	Number of Motorist Killed	Count of Motorist killed in an incident	Entries contain value 0,1,2,3,4,5	int
16	Number of Pedestrians Injured	Count of Pedestrian injured in an incident	Max value is 27	int
17	Number of Pedestrians Killed	Count of Pedestrian killed in an incident	Entries contain value 0,1,2,6	int
18	Number of Persons Injured	Count of Persons injured in an incident	Max value is 43	int
19	Number of Persons Killed	Count of Persons killed in an incident	Max value is 8	int
20	Off Street Name	Nameof the off street	Mostly Empty	char
21	On Street Name	Name of the street incident occured		char
22	Time	Time of the incident occurence		char
23	Unique Key	Key identifying each record uniquely	& digit unique number	int
24	Vehicle Type Code 1	Type of the vehicle	At some places the value is Contain NA,N/A or UNKNOWN	char
25	Vehicle Type Code 2	Further description of vehicle	Few NA	char
26	Vehicle Type Code 3	Description of vehicle 3	Mostly Empty	char
27	Vehicle Type Code 4	Description of vehicle 4	Mostly Empty	char
28	Vehicle Type Code 5	Decsription of vehicle 5	Mostly Empty	char
29	Zip Code	Zip code of the location where incident occured	Its a 5 digit number	int