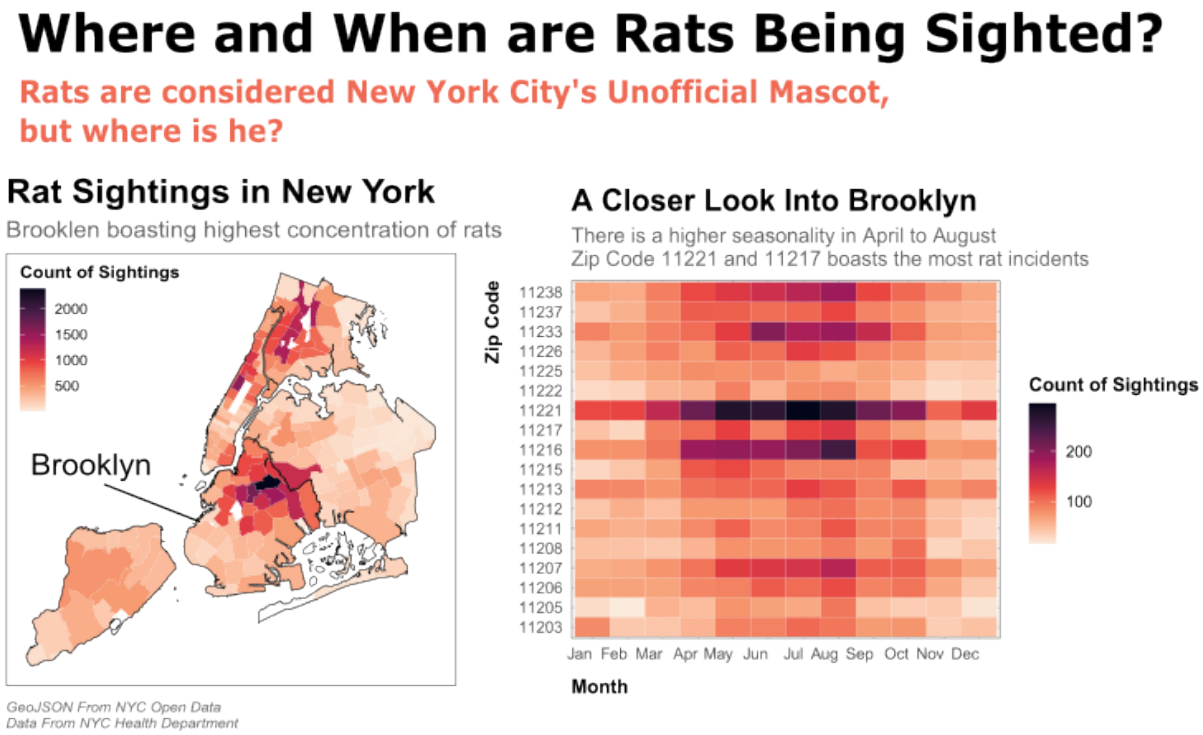
A1: Sightings

Joo Han

2024-02-17



Executive Summary

The New York City Rat Sightings

The story I am trying to tell is where and when people can expect more rat sightings which can be an important insight for those scouting for a new business location, those looking into possible neighborhoods they may want to move to, or private and government owned rodent control services as they can expect an increase in reports or calls at a specific location or time of year. The first chart, "Rat Sightings in New York" delves into where rodents have been reported. The count has been separated by zip code which gives the reader a more fine insight into rat sighting locations instead of a broader location such as borough. However, I did include the border by borough as most rodent control services usually have more than one zip code they work under and is more likely to work over the whole borough. Additionally, I added an annotation to point at the borough with the highest rat sightings which was Brooklyn.

The pointing out of Brooklyn in the first plot flows into the second plot which takes a closer look at this trend. Rather than a line graph that would look like the ones I created in my EDA, utilizing the heat map could better show the discrepancy between 11221 and the other zip codes. Furthermore, the color gradient allows us to see the seasonality of the rat sightings as well which happens to be from April to September. This can prepare business owners, home owners, and pest control workers alike for yearly upcoming influx of rats.

CRAP Principles The largest contrast I make is based on the color gradient I use for rat sighting: going from a light red to show that there are only a few rats to a very dark, almost black, red for the highest concentration of rat sightings. I specifically chose red as it is typically indicates warning and had it span to such a dark color as darker colors can specify illness and disease which has been coincided with rats in the past. The repetition can also be seen by the utilizing of the same color choice for the same KPIs, titles, subtitles, and more. Alignment is shown by having all the elements in the charts go on the left and having a specific order to the charts as most people read from left to right. Finally, proximity can be seen by the utilization of white space in both the graph especially in the map. For example, originally the map has a legend on the right, but was moved to inside the map in the final visualization for white space utilization. This also makes it more apparent which legend goes with what chart as they have different ranges despite the same color scheme.

Alberto Cairo's Five Qualities For truthfulness, I made sure to accurately represent the data on rat sightings in New York, ensuring that the map elements correspond precisely to the data from NYC Open Data and Health Department reports. The map and heatmap accurately reflect the quantity and distribution of rat sightings without distortion, providing an honest visual account of the situation.

Functionality was a key consideration in selecting a map and a heatmap to display the data. These formats are not only familiar to the audience (mostly those who live in NY) but also convey geographical and temporal patterns in data. The choice of color gradients serves a functional purpose, allowing for quick visual differentiation between areas and times of high and low sighting frequencies.

In terms of beauty, I aimed for a clean and aesthetically pleasing design that invites viewers to engage with the content. The visualizations employ a harmonious color palette that still creates an alarming feeling in those with high rat sightings. The layout is balanced, with clear, legible fonts that facilitates easy reading and comprehension of the data.

Insightfulness is achieved through the juxtaposition of spatial and temporal data, which reveals patterns that may not be immediately apparent from raw data alone. For instance, the heatmap uncovers seasonal trends in rat sightings, offering new insights into when interventions might be most needed.

Finally, the visualization serves to enlighten by providing not just data, but also context and potential for understanding. It suggests a story of urban ecology and public health, prompting viewers to consider the implications of the data and to ask further questions about the causes and effects of the rat sightings in New York City, particularly in Brooklyn.

```
library(dplyr)
library(tidyverse)
library(lubridate)
library(scales)
library(countrycode)
library(plotly)
library(sf)
library(extrafont)
```

```
sightings <- read_csv("data/A1_sightings.csv")

sight_dim <- dim(sightings)
col_names <- colnames(sightings)
```

```r
print(sight_dim)
```

```
## [1] 101914     52
```

```r
print(col_names)
```

```
##  [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"
```

```r
# Replace "na" and "n/a" with NA
sightings[sightings == "na" | sightings == "n/a" |
            sightings == "NA" | sightings == "N/A"] <- NA

# threshold if 80% of rows in a column is NA, remove the column
threshold <- nrow(sightings) * 0.8

for (col in names(sightings)) {
  if (sum(is.na(sightings[[col]])) > threshold) {
    sightings[[col]] <- NULL
  }
}

print(colSums(is.na(sightings)))
```

```
##                    Unique Key                  Created Date
##                             0                             0
##                   Closed Date                        Agency
```

```
##                         10931                             0
##                   Agency Name                Complaint Type
##                             0                             0
##                    Descriptor                 Location Type
##                             0                             6
##                  Incident Zip               Incident Address
##                           336                          9074
##                   Street Name                 Cross Street 1
##                          9075                         16657
##                Cross Street 2                   Address Type
##                         16690                           346
##                          City                        Status
##                           342                             0
##                     Due Date Resolution Action Updated Date
##                           117                             3
##               Community Board                       Borough
##                             0                             0
##    X Coordinate (State Plane)    Y Coordinate (State Plane)
##                           706                           706
##            Park Facility Name                  Park Borough
##                             0                             0
##                   School Name                 School Number
##                             0                             0
##                 School Region                  School Code
##                             0                             0
##           School Phone Number                School Address
##                             0                             0
##                   School City                  School State
##                             0                             0
##                    School Zip               School Not Found
##                             0                           917
##                      Latitude                     Longitude
##                           706                           706
##                      Location
##                           706
```

```
print(dim(sightings))
```

```
## [1] 101914     37
```

```
num_empty_rows <- sum(!complete.cases(sightings))

print(num_empty_rows)
```

```
## [1] 29036
```

```
sightings <- sightings[complete.cases(sightings), ]

head(sightings)
```

```
## # A tibble: 6 x 37
##   `Unique Key` `Created Date`         `Closed Date`         Agency `Agency Name`
```

```
##          <dbl> <chr>                  <chr>                  <chr>  <chr>
## 1     31464026 09/04/2015 12:00:00 AM 09/14/2015 12:00:00 ~ DOHMH  Department o~
## 2     31464027 09/04/2015 12:00:00 AM 09/22/2015 12:00:00 ~ DOHMH  Department o~
## 3     31464188 09/04/2015 12:00:00 AM 09/22/2015 12:00:00 ~ DOHMH  Department o~
## 4     31464195 09/04/2015 12:00:00 AM 09/22/2015 04:26:36 ~ DOHMH  Department o~
## 5     31464802 09/04/2015 12:00:00 AM 09/25/2015 12:00:00 ~ DOHMH  Department o~
## 6     31464803 09/04/2015 12:00:00 AM 07/30/2015 12:00:00 ~ DOHMH  Department o~
## # i 32 more variables: 'Complaint Type' <chr>, Descriptor <chr>,
## #   'Location Type' <chr>, 'Incident Zip' <dbl>, 'Incident Address' <chr>,
## #   'Street Name' <chr>, 'Cross Street 1' <chr>, 'Cross Street 2' <chr>,
## #   'Address Type' <chr>, City <chr>, Status <chr>, 'Due Date' <chr>,
## #   'Resolution Action Updated Date' <chr>, 'Community Board' <chr>,
## #   Borough <chr>, 'X Coordinate (State Plane)' <dbl>,
## #   'Y Coordinate (State Plane)' <dbl>, 'Park Facility Name' <chr>, ...
```

```r
# first checking school related columns
school_cols <- c("School Name", "School Number","School Region","School Code",
                 "School Phone Number","School Address","School City",
                 "School State","School Zip","School Not Found")

# Loop over each column
for (col in school_cols) {
  unique_values <- unique(sightings[[col]])
  cat("Unique values for column", col, ":", unique_values, "\n")
}
```

```
## Unique values for column School Name : Unspecified
## Unique values for column School Number : Unspecified
## Unique values for column School Region : Unspecified
## Unique values for column School Code : Unspecified
## Unique values for column School Phone Number : Unspecified
## Unique values for column School Address : Unspecified
## Unique values for column School City : Unspecified
## Unique values for column School State : Unspecified
## Unique values for column School Zip : Unspecified
## Unique values for column School Not Found : N
```

```r
# no interesting insights from school columns
sightings <- sightings[, -which(names(sightings) %in% school_cols)]

head(sightings)
```

```
## # A tibble: 6 x 27
##   'Unique Key' 'Created Date'         'Closed Date'          Agency 'Agency Name'
##          <dbl> <chr>                  <chr>                  <chr>  <chr>
## 1     31464026 09/04/2015 12:00:00 AM 09/14/2015 12:00:00 ~ DOHMH  Department o~
## 2     31464027 09/04/2015 12:00:00 AM 09/22/2015 12:00:00 ~ DOHMH  Department o~
## 3     31464188 09/04/2015 12:00:00 AM 09/22/2015 12:00:00 ~ DOHMH  Department o~
## 4     31464195 09/04/2015 12:00:00 AM 09/22/2015 04:26:36 ~ DOHMH  Department o~
## 5     31464802 09/04/2015 12:00:00 AM 09/25/2015 12:00:00 ~ DOHMH  Department o~
## 6     31464803 09/04/2015 12:00:00 AM 07/30/2015 12:00:00 ~ DOHMH  Department o~
## # i 22 more variables: 'Complaint Type' <chr>, Descriptor <chr>,
## #   'Location Type' <chr>, 'Incident Zip' <dbl>, 'Incident Address' <chr>,
```

```
## #   `Street Name` <chr>, `Cross Street 1` <chr>, `Cross Street 2` <chr>,
## #   `Address Type` <chr>, City <chr>, Status <chr>, `Due Date` <chr>,
## #   `Resolution Action Updated Date` <chr>, `Community Board` <chr>,
## #   Borough <chr>, `X Coordinate (State Plane)` <dbl>,
## #   `Y Coordinate (State Plane)` <dbl>, `Park Facility Name` <chr>, ...
```

```
colnames(sightings)
```

```
##  [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"                "Address Type"
## [15] "City"                          "Status"
## [17] "Due Date"                      "Resolution Action Updated Date"
## [19] "Community Board"               "Borough"
## [21] "X Coordinate (State Plane)"    "Y Coordinate (State Plane)"
## [23] "Park Facility Name"            "Park Borough"
## [25] "Latitude"                      "Longitude"
## [27] "Location"
```

```
key_cols <- c("Unique Key", "Created Date", "Agency",
              "Complaint Type", "Descriptor", "Location Type",
              "Incident Zip", "City", "Borough",
              "Latitude", "Longitude")

sightings <- sightings[, key_cols]
colnames(sightings) <- tolower(gsub(" ", "_", colnames(sightings)))
head(sightings)
```

```
## # A tibble: 6 x 11
##   unique_key created_date         agency complaint_type descriptor location_type
##        <dbl> <chr>                <chr>  <chr>          <chr>      <chr>
## 1   31464026 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Ap~
## 2   31464027 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Mi~
## 3   31464188 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Ap~
## 4   31464195 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Ap~
## 5   31464802 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 1-2 Family D~
## 6   31464803 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ Commercial B~
## # i 5 more variables: incident_zip <dbl>, city <chr>, borough <chr>,
## #   latitude <dbl>, longitude <dbl>
```

```
sightings <- sightings %>%
  mutate(
    created_year = year(mdy_hms(created_date)),
    created_month = month(mdy_hms(created_date)),
    created_day = day(mdy_hms(created_date)),
    created_hour = hour(mdy_hms(created_date))
  )

head(sightings)
```

```
## # A tibble: 6 x 15
##   unique_key created_date         agency complaint_type descriptor location_type
##        <dbl> <chr>                <chr>  <chr>          <chr>      <chr>
## 1   31464026 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Ap~
## 2   31464027 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Mi~
## 3   31464188 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Ap~
## 4   31464195 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 3+ Family Ap~
## 5   31464802 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ 1-2 Family D~
## 6   31464803 09/04/2015 12:00:00~ DOHMH  Rodent         Rat Sight~ Commercial B~
## # i 9 more variables: incident_zip <dbl>, city <chr>, borough <chr>,
## #   latitude <dbl>, longitude <dbl>, created_year <dbl>, created_month <dbl>,
## #   created_day <int>, created_hour <int>
```

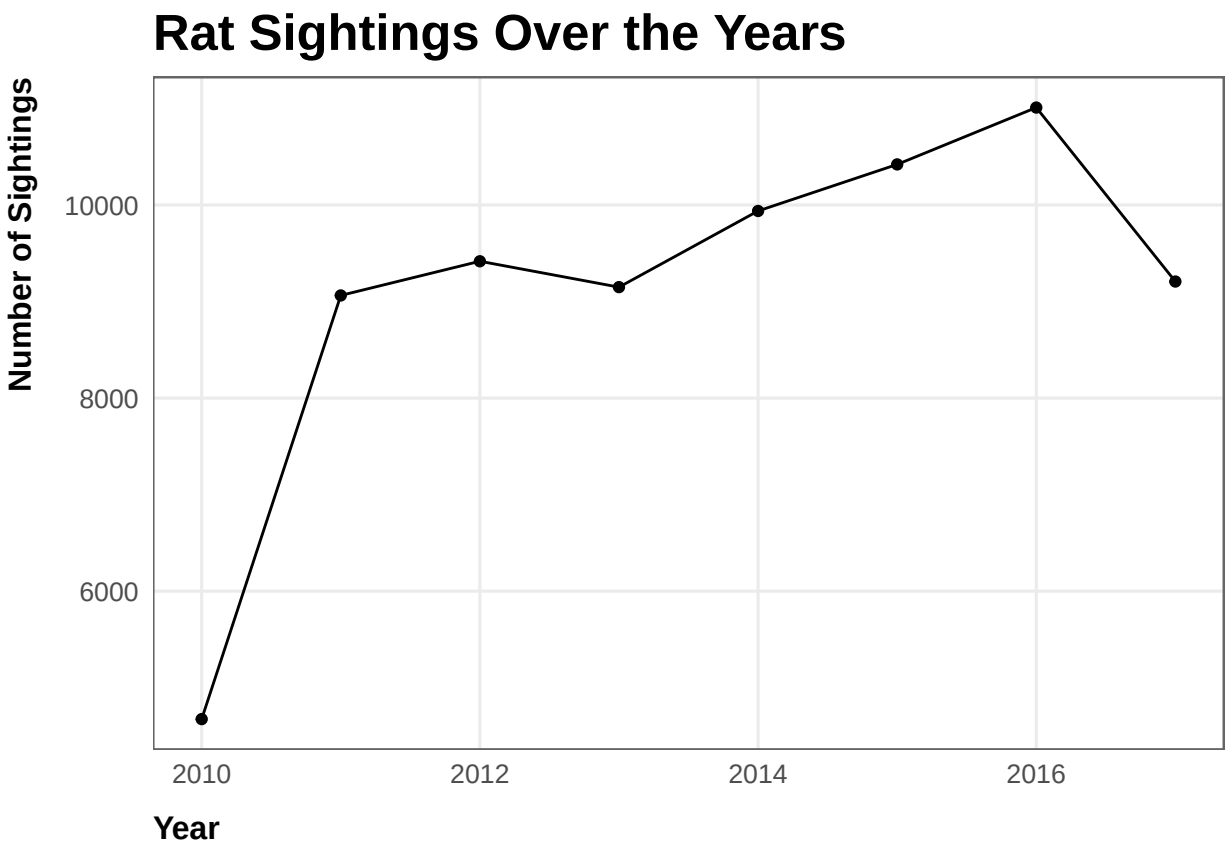
```
unique_created_year <- unique(sightings["created_year"])

print(unique_created_year)
```

```
## # A tibble: 8 x 1
##   created_year
##          <dbl>
## 1         2015
## 2         2017
## 3         2016
## 4         2010
## 5         2011
## 6         2014
## 7         2012
## 8         2013
```

```
my_theme <- theme_minimal(base_family = "Helvetica", base_size = 12) +
  theme(panel.grid.minor = element_blank(),
        plot.title = element_text(face = "bold", size = rel(1.6)),
        plot.subtitle = element_text(face = "plain", size = rel(1.1),
                                     color = "grey40"),
        plot.caption = element_text(face = "italic", size = rel(0.7),
                                    color = "grey40", hjust = 0),
        legend.title = element_text(face = "bold"),
        strip.text = element_text(face = "bold", size = rel(1.2), hjust = 0),
        axis.title = element_text(face = "bold"),
        axis.title.x = element_text(margin = margin(t = 10), hjust = 0),
        axis.title.y = element_text(margin = margin(r = 10), hjust = 1),
        strip.background = element_rect(fill = "grey40", color = NA),
        panel.border = element_rect(color = "grey40", fill = NA))
```

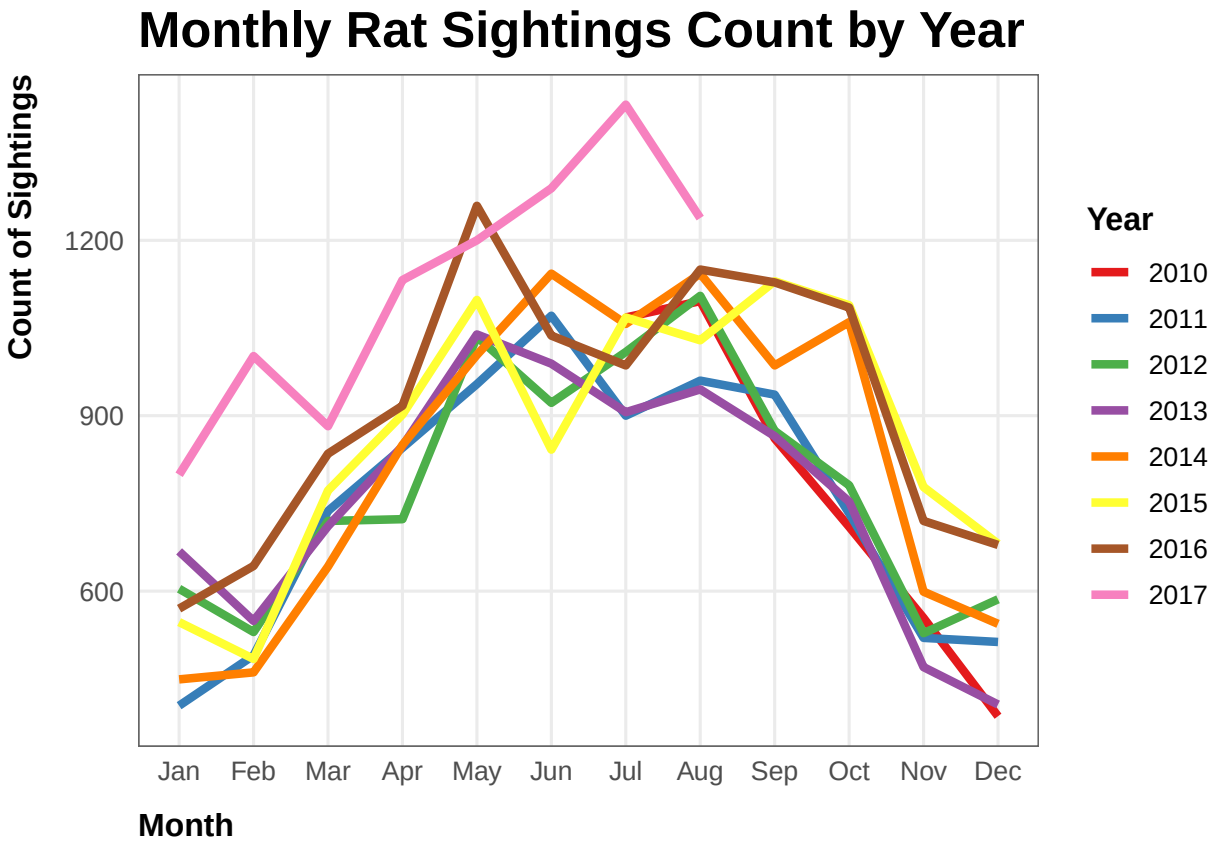
```
rat_sightings_yearly <- sightings %>%
  filter(descriptor == "Rat Sighting") %>%
  group_by(created_year) %>%
  summarise(count = n())

# Generate the line plot
ggplot(rat_sightings_yearly, aes(x = created_year, y = count)) +
  geom_line() +
  geom_point() + # Optional: adds points to each year-count pair
```

```
  theme_minimal() +
  labs(title = "Rat Sightings Over the Years",
       x = "Year",
       y = "Number of Sightings") +
  my_theme
```




```
# Filter out months after August for the year 2017
sightings <- sightings %>%
  mutate(created_year_month = as.Date(paste(created_year, created_month,
                                            "01", sep = "-"))) %>%
  filter(!(created_year == 2017 & created_month > 8))

# Aggregating the data
monthly_counts <- sightings %>%
  filter(descriptor == "Rat Sighting") %>%
  group_by(created_year, created_month) %>%
  summarise(count = n(), .groups = 'drop') # updated summarise to drop groups

# Define a color palette that has distinct colors for each year
distinct_colors <- c("2010" = "#E41A1C", "2011" = "#377EB8", "2012" = "#4DAF4A",
                     "2013" = "#984EA3", "2014" = "#FF7F00", "2015" = "#FFFF33",
                     "2016" = "#A65628", "2017" = "#F781BF")

# Plotting with distinct colors and having the line for 2017 stop at August
ggplot(monthly_counts, aes(x = created_month, y = count, group = created_year,
```

```
                          color = as.factor(created_year))) +
  geom_line(size = 1.5) +
  scale_color_manual(values = distinct_colors) + # Using the manual color scale
  labs(title = "Monthly Rat Sightings Count by Year",
       x = "Month",
       y = "Count of Sightings",
       color = "Year") +
  scale_x_continuous(breaks = 1:12, labels = month.abb) +
  my_theme
```

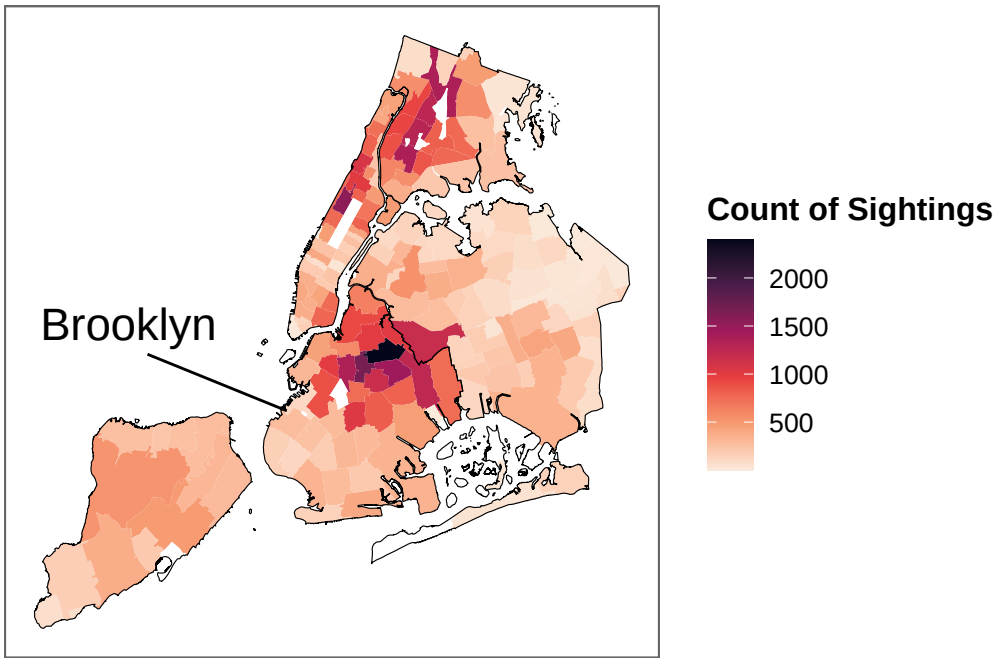



```
ny_zip_shapefile <- read_sf("data/ny_zipcode_gjson.geojson")

# Aggregating the data to get the count of rat sightings per ZIP code
sightings_by_zip <- sightings %>%
  filter(descriptor == "Rat Sighting") %>%
  group_by(incident_zip) %>%
  summarise(count = n()) %>%
  ungroup()

# Merge the sightings data with the spatial data
ny_zip_shapefile$modzcta <- as.character(ny_zip_shapefile$modzcta)
sightings_by_zip$incident_zip <- as.character(sightings_by_zip$incident_zip)

ny_zip_data <- merge(ny_zip_shapefile, sightings_by_zip, by.x = "modzcta",
                     by.y = "incident_zip", all.x = TRUE)
```

```r
nyc_borough_boundaries <- st_read('data/ny_borough.geojson')
```

```
## Reading layer ‘ny_borough’ from data source
##   ‘C:\Users\justi\OneDrive\Desktop\School\Hult Masters\Spring 2024\Visualizing & Analyzing Data with
##   using driver ‘GeoJSON’
## Simple feature collection with 5 features and 4 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -74.25559 ymin: 40.49613 xmax: -73.70001 ymax: 40.91553
## Geodetic CRS:  WGS 84
```

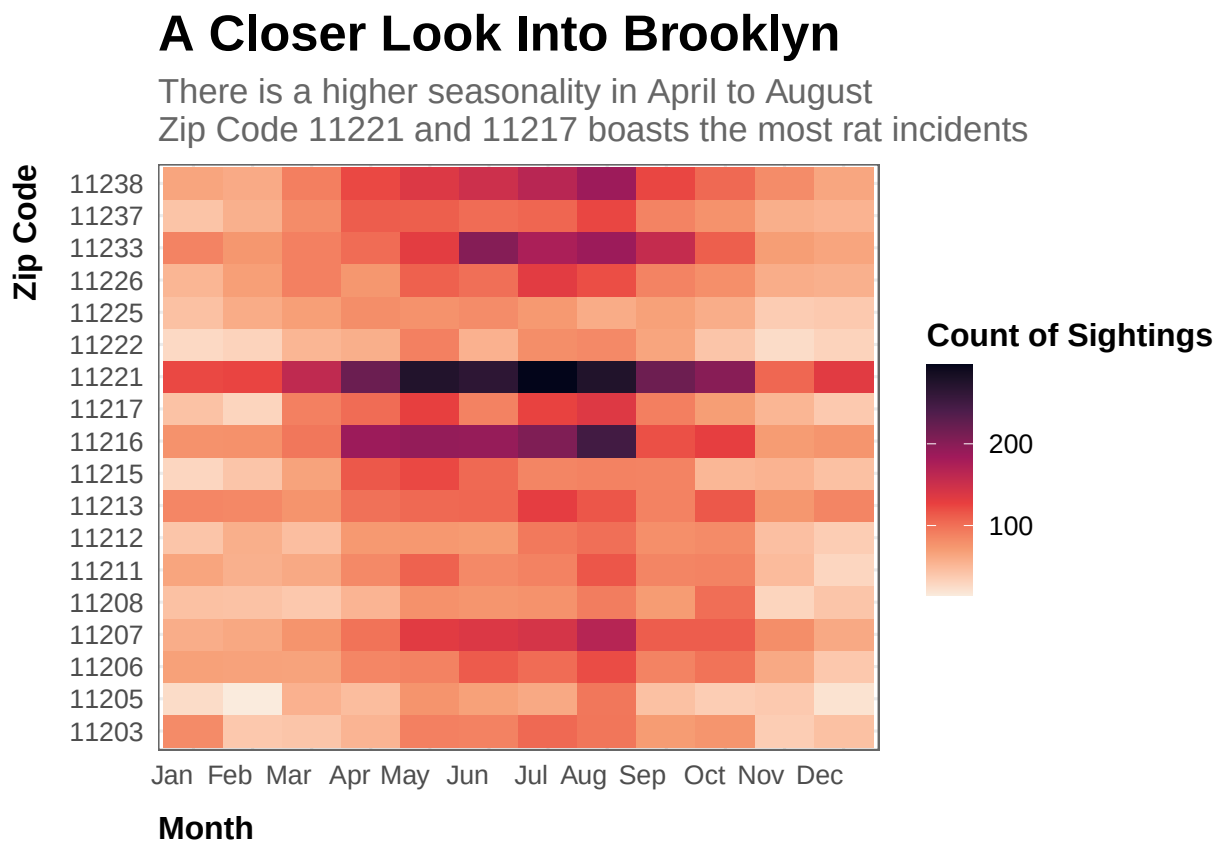
```r
# Plotting the map
my_map <- ggplot(data = ny_zip_data) +
  geom_sf(aes(fill = count), color = NA) +
  geom_sf(data = nyc_borough_boundaries, fill = NA,
          color = "black", size = 0.5) +
  scale_fill_viridis_c(option = "rocket",
                       direction = -1, na.value = "white") +
  labs(title = "Rat Sightings in New York",
       fill = "Count of Sightings",
       subtitle = "Brooklen boasting highest concentration of rats",
       caption = "GeoJSON From NYC Open Data\nData From NYC Health Department") +
  annotate("text", x = -74.25, y = 40.7, label = "Brooklyn", size = 6,
           hjust = 0, vjust = 0, color = "black") +
  annotate("segment", x = -74.15, xend = -74.02, y = 40.69,
           yend = 40.65, colour = "black") +
  my_theme +
  theme(
    axis.text.x = element_blank(),
    axis.text.y = element_blank(),
    axis.title.x = element_blank(),
    axis.title.y = element_blank(),
    axis.ticks = element_blank(),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()
  )

print(my_map)
```




```
ggsave(my_map, filename = "A1_fig1.png", width = 5, height = 5)
ggsave(my_map, filename = "A1_fig1.pdf", width = 5, height = 5)
```

```
brooklyn_sightings <- sightings %>%
  filter(borough == "BROOKLYN") %>%
  group_by(incident_zip, created_month) %>%
  summarise(count = n()) %>%
  ungroup()

brooklyn_sightings$incident_zip <- factor(brooklyn_sightings$incident_zip,
                                          levels = unique(brooklyn_sightings$incident_zip))
brooklyn_sightings$created_month <- factor(brooklyn_sightings$created_month,
                                           levels = 1:12, labels = month.abb)

# Filter ZIP codes with sighting counts greater than 200
high_sightings_zip <- brooklyn_sightings %>%
  group_by(incident_zip) %>%
  summarise(total_count = sum(count)) %>%
  filter(total_count > 500) %>%
  select(incident_zip)

# Filter the aggregated data to only include these ZIP codes
final_brooklyn_sightings <- brooklyn_sightings %>%
  filter(incident_zip %in% high_sightings_zip$incident_zip)
```

```
my_hm <- ggplot(final_brooklyn_sightings, aes(x = created_month,
                                              y = incident_zip, fill = count)) +
  geom_tile() +
  scale_fill_viridis_c(option = "rocket", direction = -1) +
  labs(title = "A Closer Look Into Brooklyn",
       x = "Month",
       y = "Zip Code",  # Adjusted to represent month
       fill = "Count of Sightings",
       subtitle = "There is a higher seasonality in April to August\nZip Code 11221 and 11217 boasts the
  my_theme +
  theme(axis.text.x = element_text(angle = 0, hjust = 1),  # Rotate x-axis text for readability
        axis.text.y = element_text(angle = 0))   # Rotate x-axis text for readability

print(my_hm)
```




```
ggsave(my_hm, filename = "A1_fig2.png", width = 7, height = 5)
ggsave(my_hm, filename = "A1_fig2.pdf", width = 7, height = 5)
```