Homework #3

Xinyuan Cao (xc2461)

For questions 1-4 in this problem set, we will work with a dataset on dogs of New York City, found here: https://project.wnyc.org/dogs-of-nyc/

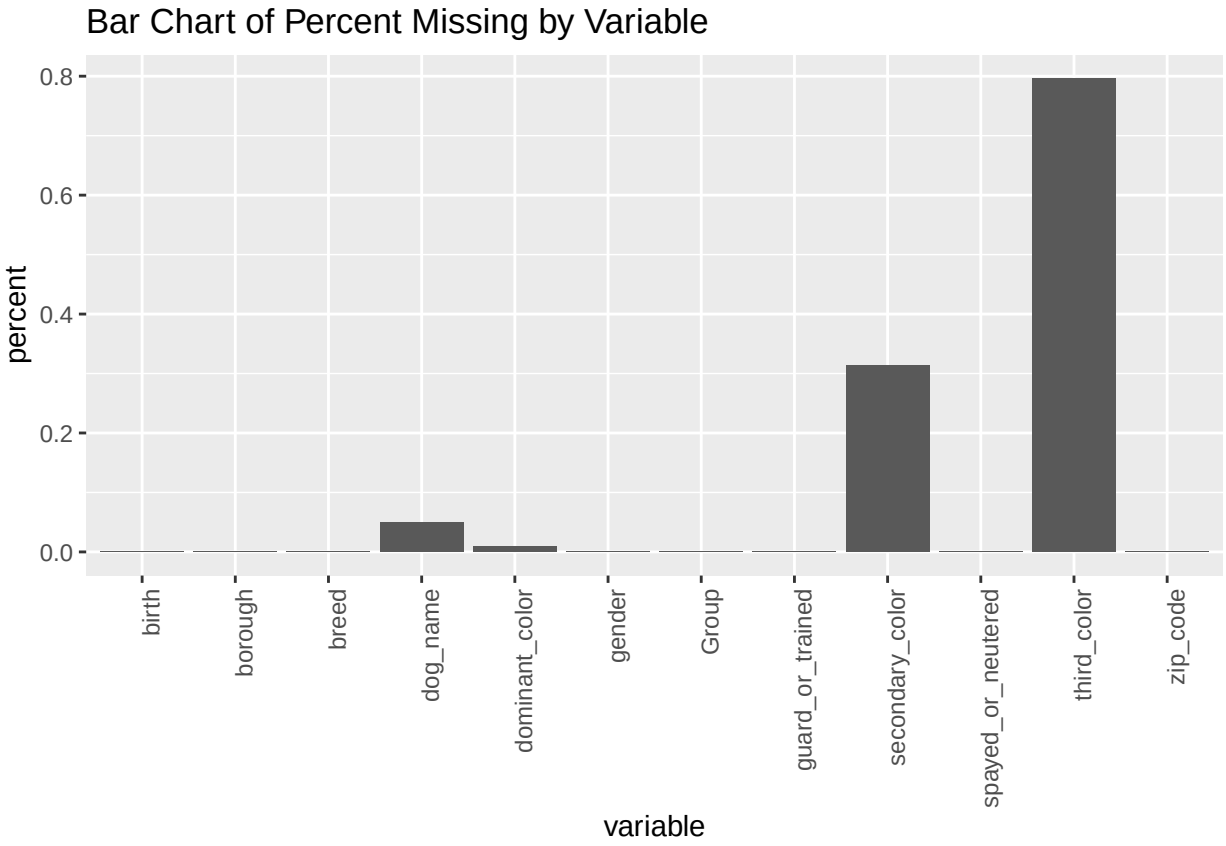
Please use the "NYCdogs.csv" version found in Files/Data folder on CourseWorks, which includes a Group column. If you already did some of the questions that didn't require the Group column, you do not have to redo them.

Background: The dataset is dated June 26, 2012. Although the data were originally produced by the NYC Department of Mental Health and Hygiene, it no longer seems to be available on any official NYC web site. (There is a 2016 dataset on dog licenses with different variables available here: https://data.cityofnewyork.us/Health/NYC-Dog-Licensing-Dataset/nu7n-tubp). Also of note is the fact that this dataset has 81,542 observations. The same summer, the New York City Economic Development Corporation estimated that there were 600,000 dogs in New York City (source: https://blog.nycpooch.com/2012/08/28/how-many-dogs-live-in-new-york-city/) Quite a difference! How many dogs were there really in 2012?!? Might be an interesting question to pursue for a final project, but for now we'll work with what we've got.

1. Missing Data

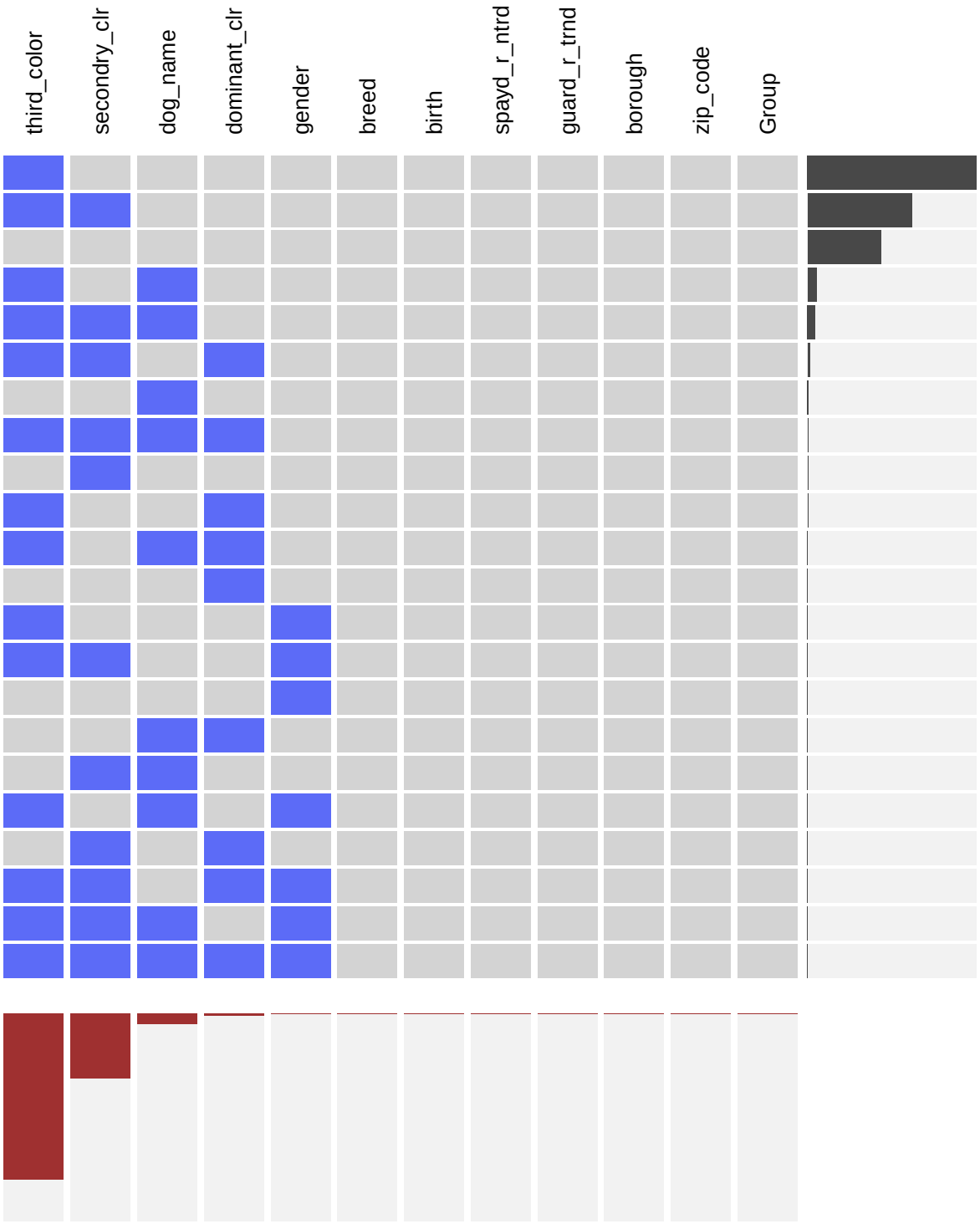
(a) Create a bar chart showing percent missing by variable.

```
library(tidyverse)
nycdogs <- read.csv("NYCdogs.csv")

# change n/a to NA
nycdogs[nycdogs == "n/a"] <- NA

# missing values by column (variable)
percent.missing <- data.frame(colSums(is.na(nycdogs))/nrow(nycdogs))
percent.missing <- cbind(rownames(percent.missing), percent.missing)
colnames(percent.missing) <- c("variable", "percent")
rownames(percent.missing) <- 1:nrow(percent.missing)

# draw bar chart of missing percentage
p1a <- ggplot(percent.missing, aes(x = variable, y = percent)) +
  geom_col() +
  ggtitle("Bar Chart of Percent Missing by Variable") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
p1a
```



(b) Use the `extracat::visna()` to graph missing patterns. Interpret the graph.

```
extracat::visna(nycdogs, sort = "b")
```



- If we see the data row by row we can see the relative frequencies of the missing pattern by the bar chart on the right side, from the top is the most frequent pattern of the data, that is missing exactly "third_color" variable, and the second most frequent pattern is missing "secondary_color" and "third_color". The third most frequent pattern is not missing any variables. The remains's frequencies are dramatically lower than the previous three.
- And if we see the data column by column, the bar chart below the table shows the proportions of the missings by variables. We can get the same result as the bar chart in question (a), and we find the top

two frequent missing values are "third_color" and "secondary_color". And it is also corresponding to missing pattern stated above.

- Also see the graph in an overall view, we find each data misses at most five variables, and there are seven variables that are not missing in any data point.

(c) Do `dog_name` missing patterns appear to be associated with the *value* of `gender`, `Group` *or* `borough`?

```
# find associating of missing patterns of 'dog_name' with the value of 'gender'
percent_missing <- nycdogs %>%
  group_by(gender) %>%
  summarize(num_dog_name=n(), num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na/num_dog_name, 2)) %>%
  arrange(-percent_na)
percent_missing
```

```
## # A tibble: 3 x 4
##   gender num_dog_name num_na percent_na
##   <fct>         <int>  <int>      <dbl>
## 1 <NA>             62      4       0.06
## 2 F             37156   1742       0.05
## 3 M             44324   2279       0.05
```

```
# find associating of missing patterns of 'dog_name' with the value of 'Group'
percent_missing <- nycdogs %>%
  group_by(Group) %>%
  summarize(num_dog_name=n(), num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na/num_dog_name, 2)) %>%
  arrange(-percent_na)
percent_missing
```

```
## # A tibble: 8 x 4
##   Group        num_dog_name num_na percent_na
##   <fct>               <int>  <int>      <dbl>
## 1 Non-Sporting         7796    529       0.07
## 2 Toy                 23640   1486       0.06
## 3 Mutt                28472   1451       0.05
## 4 Hound                3395    113       0.03
## 5 Working              3448     92       0.03
## 6 Herding              2431     57       0.02
## 7 Sporting             6027    147       0.02
## 8 Terrier              6333    150       0.02
```

```
# find associating of missing patterns of 'dog_name' with the value of 'borough'
percent_missing <- nycdogs %>%
  group_by(borough) %>%
  summarize(num_dog_name=n(), num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na/num_dog_name, 2)) %>%
  arrange(-percent_na)
percent_missing
```

```
## # A tibble: 5 x 4
##   borough       num_dog_name num_na percent_na
##   <fct>                <int>  <int>      <dbl>
## 1 Bronx                 9293    454       0.05
## 2 Brooklyn             19333   1024       0.05
## 3 Manhattan            26029   1224       0.05
```

```
## 4 Queens               17506    938       0.05
## 5 Staten Island         9381    385       0.04
```

- According to the tables above, we may find that missing patterns of 'dog_name' does not have relationship with the value of 'gender' and 'borough'.
- But when it comes to 'Group', we may find that 'non-sporting' group has relatively higher missing rate, and the 'herding', 'sporting' and 'terrier' groups have relatively lower missing rate. But they are all very low (0.02 - 0.07), so it still does not have strong associations with 'Group'. There is certain associations with the missing pattern of 'dog_name' and 'Group'.

2. Dates

(a) Convert the `birth` column of the NYC dogs dataset to `Date` class (use "01" for the day since it's not provided). Create a frequency histogram of birthdates with a one-month binwidth. (Hint: don't forget about base R.) What do you observe? Provide a reasonable hypothesis for the prominent pattern in the graph.

```
library(lubridate)
# convert the birth column of the dataset to Date class
nycdogs$birth <- as.Date(parse_date_time(nycdogs$birth,
                         order = c('ymd', 'myd'),
                         truncated = 1))

# omit invalid values
invalid_index = which(is.na(nycdogs$birth))
nycdogs = nycdogs[-invalid_index, ]
```

```
# process wrong parsed values
error_index = which(nycdogs$birth>"2012-06-26")
error_index
```

```
## [1]  1222 35419
```

```
print(nycdogs$birth[error_index])
```

```
## [1] "2066-01-01" "2067-08-01"
```

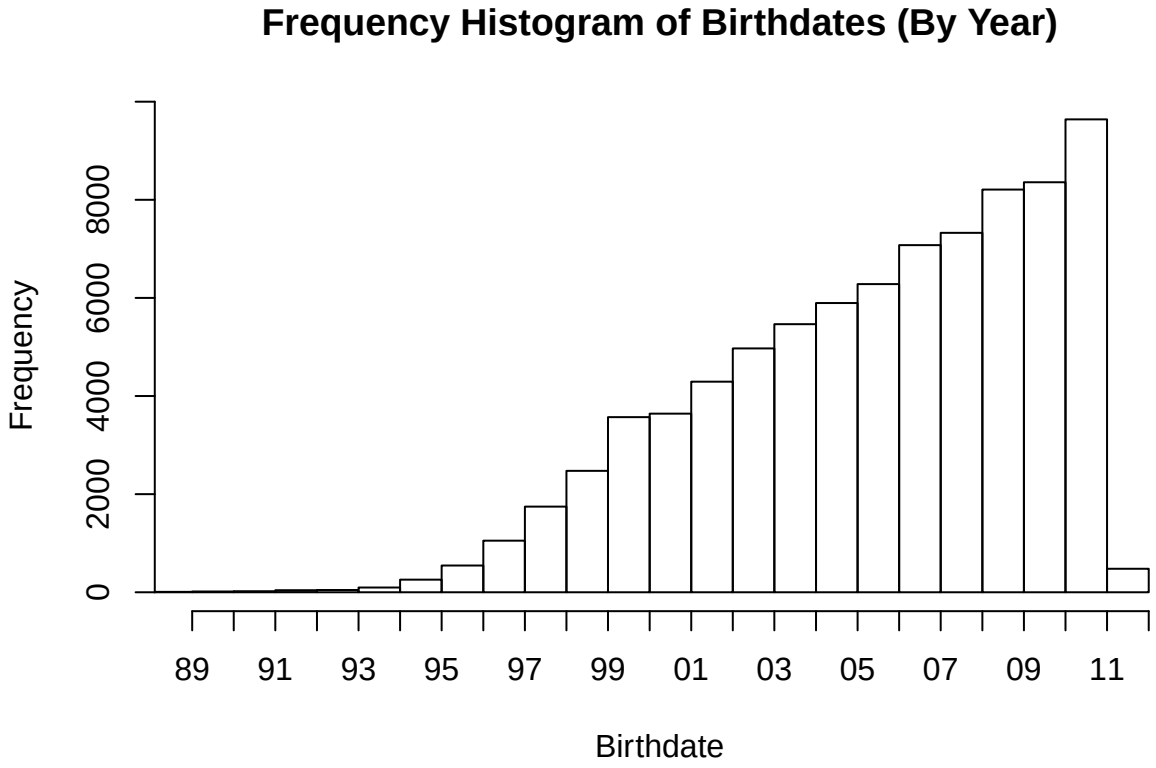
```
year(nycdogs$birth[error_index]) = year(nycdogs$birth[error_index]) - 100
```

```
# draw the frequency histogram
hist(nycdogs$birth,
     breaks = "months",
     freq = TRUE,
     xlab = "Birthdate",
     format= "%y-%m",
     main="Frequency Histogram of Birthdates (By Month)"
     )
```

- From the histogram we find there is a fixed month whose birth rate is quite larger compared to others. This may because many dogs' birthdate data are not collected, so they give a default month number (Jan) to these missing birth month.
- Also in an overrall view, we can find that the dogs' birth frequency is increasing rapidly from the end of last century to around 2004, and then it begins to decrease and there is a dramatic drop in around 2006, and then it continues to decreases slowly.

(b) Redraw the frequency histogram with impossible values removed and a more reasonable binwidth.

```
# draw the frequency histogram
hist(nycdogs$birth,
     breaks = "years",
     xlim = c(as.Date("1989-12-31"), as.Date("2012-06-26")),
     freq = TRUE,
     xlab = "Birthdate",
     format= "%y",
     main="Frequency Histogram of Birthdates (By Year)"
     )
```



- Here "Year" is more reasonable binwidth. Because the month-frequency data is very high in January and much lower in other months, the data is not that accurate in month-level. So drawing by year can show the general trend of dogs' birth in long term.
- Also remove the data before 1989 because the number is much lower.
- Lastly because the data is collect on June 26 2012, so the data after this day are invalid, and should be removed.

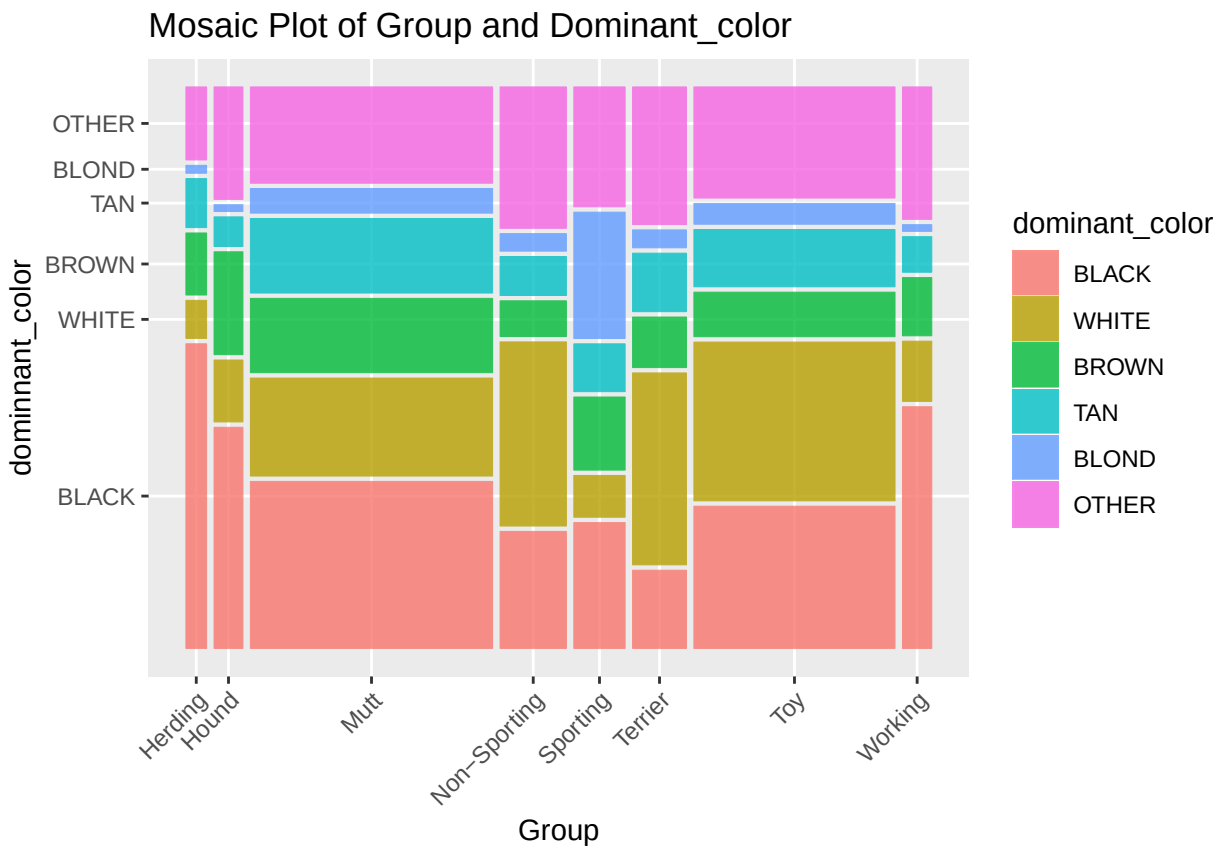
3. Mosaic plots

(a) Create a mosaic plot to see if `dominant_color` depends on `Group`. Use only the top 5 dominant colors; group the rest into an "OTHER" category. The last split should be the dependent variable and it should be horizontal. Sort each variable by frequency, with the exception of "OTHER", which should be the last category for dominant color. The labeling should be clear enough to identify what's what; it doesn't have to be perfect. Do the variables appear to be associated? Briefly describe.

```
library(vcd)
library(grid)
# use only the top 5 dominant colors, and group the rest into an "OTHER" category.
top.colors <- sort(table(nycdogs$dominant_color), decreasing = TRUE)[1:5]
nycdogs$dominant_color <- ifelse(nycdogs$dominant_color %in% names(top.colors),
                                 as.character(nycdogs$dominant_color),
                                 'OTHER')

# put "OTHER" in the last category for dominant_color
top.colors['OTHER'] = 0
nycdogs$dominant_color <- factor(nycdogs$dominant_color, levels = names(top.colors))
```

```
library(ggmosaic)
p3a <- ggplot(nycdogs) +
```

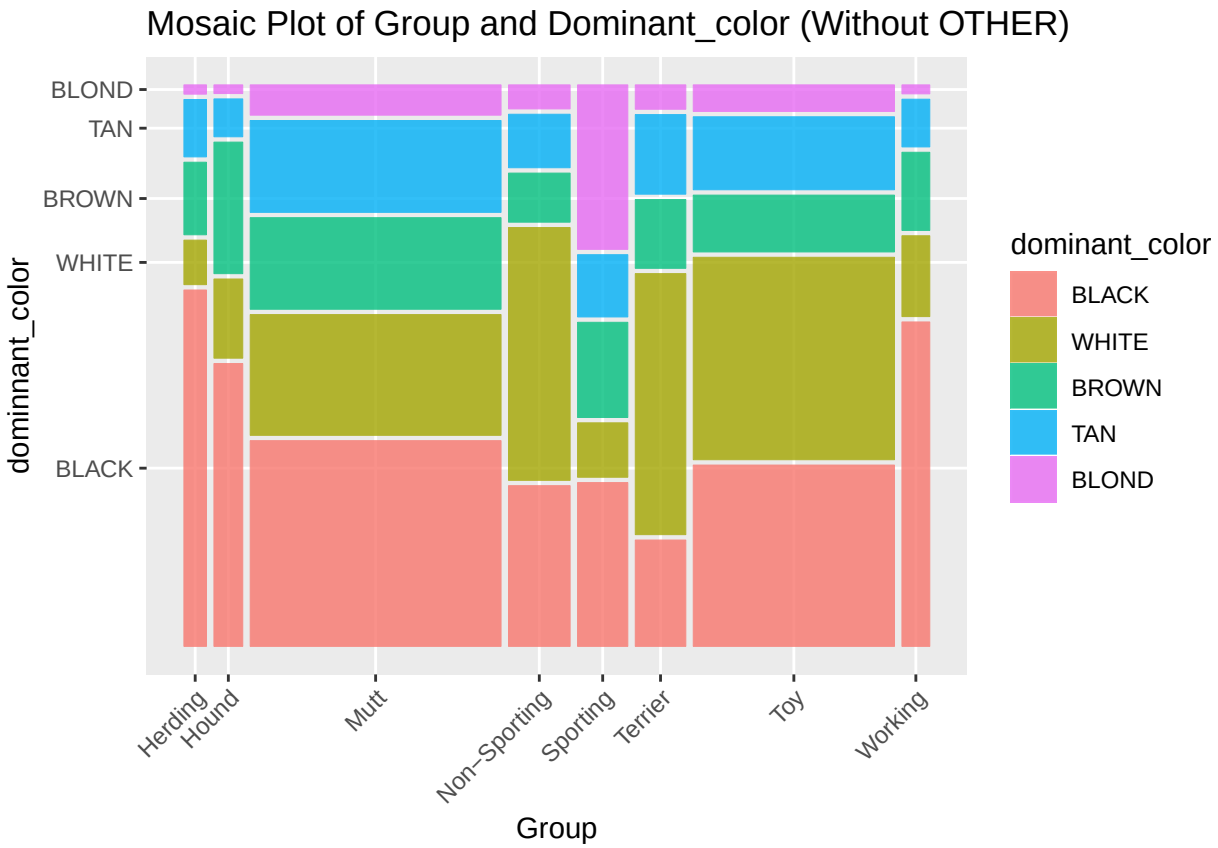
```
  geom_mosaic(aes(x = product(dominant_color, Group), fill = dominant_color)) +
  labs(x = "Group", y = "dominnant_color", fill = "dominant_color") +
  ggtitle("Mosaic Plot of Group and Dominant_color") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
p3a
```



- There are some association between group and dominant_color, because the last cut of dominant_color is not proportional to the whole colors in different groups. Also the association is not very strong because for each group there are all kinds of dominant_color with certain proportion.

(b) Redraw with the "OTHER" category filtered out. Do the results change? How should one decide whether it's necessary or not to include an "OTHER" category?

```
# filtered nycdogs of "OTHER" category
filtered.nycdogs <- subset(nycdogs, dominant_color != "OTHER")
name_level <- names(top.colors)
name_level <- name_level[-length(name_level)]
filtered.nycdogs$dominant_color <- factor(filtered.nycdogs$dominant_color, levels = name_level)

# draw the mosaic plot
p3b <- ggplot(filtered.nycdogs) +
  geom_mosaic(aes(x = product(dominant_color, Group), fill = dominant_color)) +
  labs(x = "Group", y = "dominnant_color", fill = "dominant_color") +
  ggtitle("Mosaic Plot of Group and Dominant_color (Without OTHER)") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
p3b
```



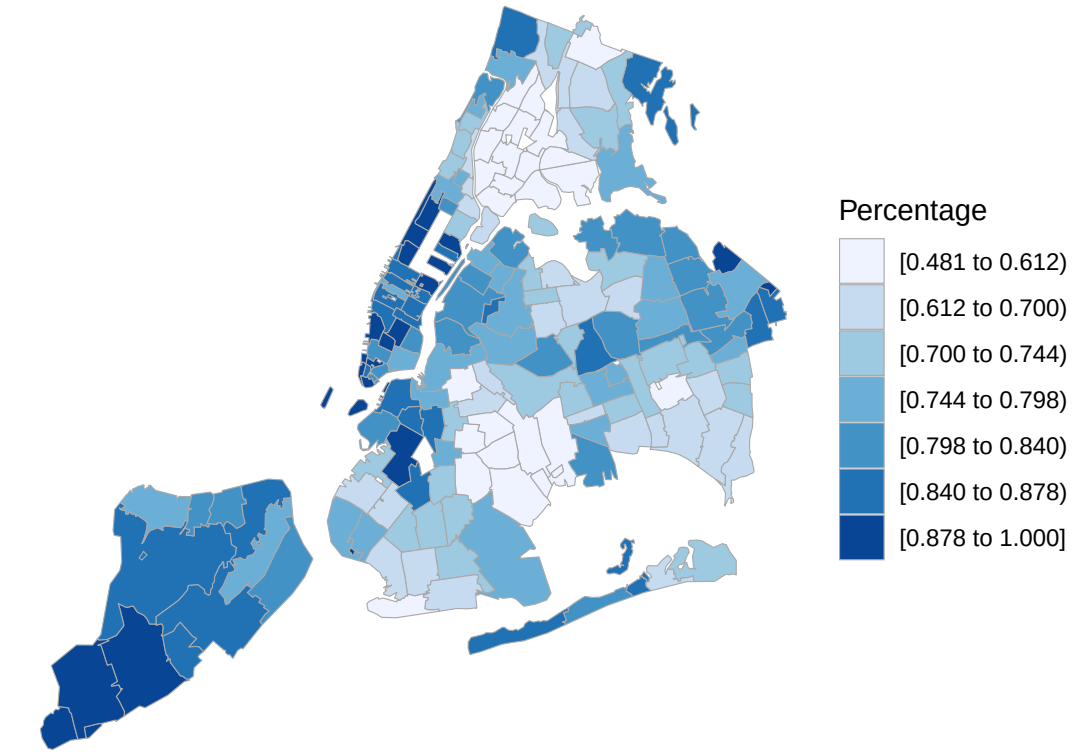
- The results do not change much actually. Dominant_color still has certain but not strong association with group.
- I think one may consider whether "OTHER" varies a lot among each group, and whether "OTHER" occupies a relative large proportion of all data.
- If "OTHER" varies a lot among each group, then we may remove "OTHER" may have a better view of seeing the association of the top five colors because it may interfere with seeing the relationship of the top five colors and group.
- If "OTHER" is quite similar in each group, then we should keep "OTHER" category because there can be some findings on association with "OTHER" color and group.
- Also, if "OTHER" is quite small in all color categories, it's fine remove it because it does not matter much, we can only study the top colors.

4. Maps

Draw a spatial heat map of the percent spayed or neutered dogs by zip code. What patterns do you notice?

```
library(choroplethrZip)
dogs.region <- nycdogs %>%
  group_by(zip_code, spayed_or_neutered) %>%
  summarise(num = n()) %>%
  mutate(freq = num / sum(num))
dogs.percent = dogs.region[dogs.region$spayed_or_neutered == "Yes",] %>%
  transmute(region = as.character(zip_code), value = freq)

data(zip.regions)
```

```
dogs.percent = dogs.percent[dogs.percent$region %in% zip.regions$region, ]
zip_choropleth(dogs.percent,
               zip_zoom = dogs.percent$region,
               state_zoom = "new york",
               title = "Spatial Heat Map of the Percentage of Spayed or Neutered Dogs in New York",
               legend = "Percentage")
```



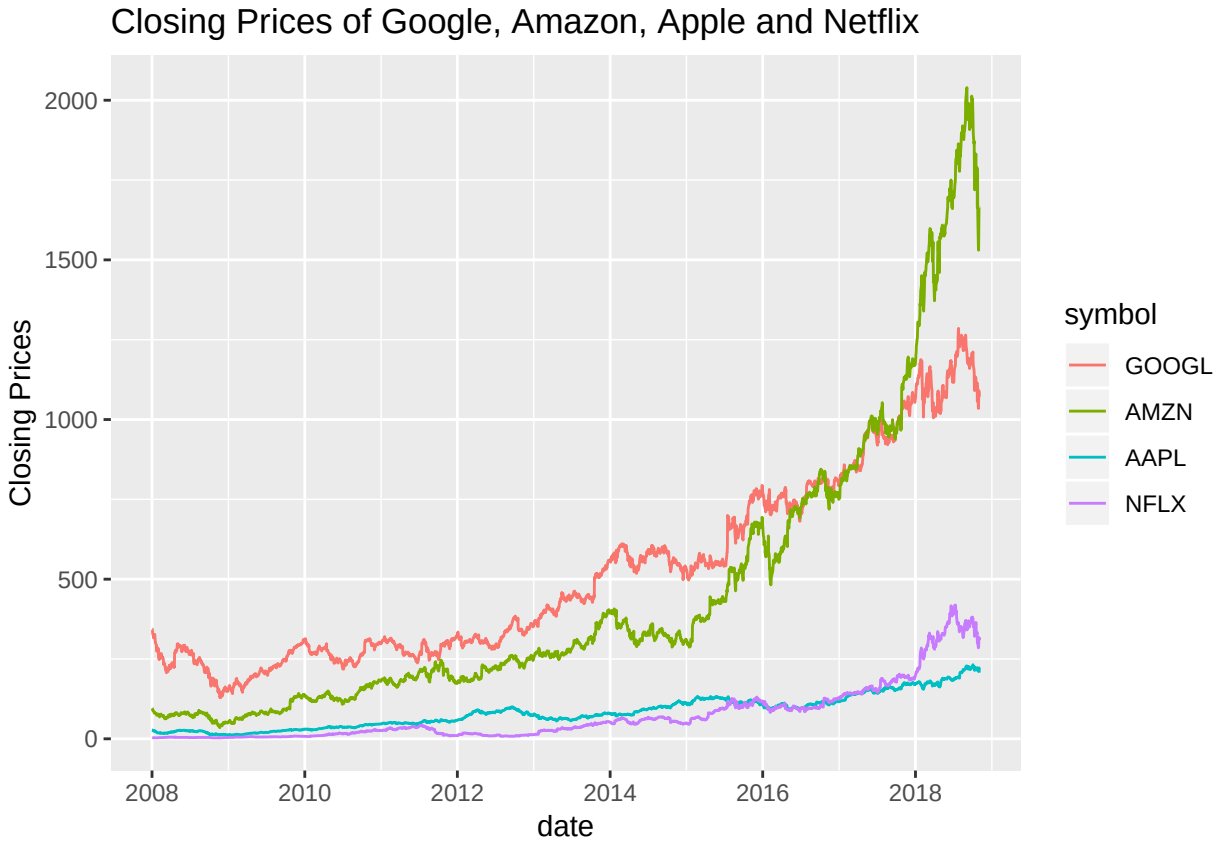
- From the spacial heap map we find that in the areas of south west corner of Stalen Island, Manhattan downtown and midtown, and mid Brooklyn, the percentages of spayed or neutered dogs are very high. In the areas of east Brooklyn and Bronx, the percentages of spayed or neutered dogs are very low. And we can have the conclusion that the in general the richer the place is, the higher the percentage of spayed or neutered dogs is. But in Stalen Island I am not quite familiar with whether it is rich or not, but it shows that people there love to spay or neuter dogs.

5. Time Series

(a) Use the `tidyquant` package to collect information on four tech stocks of your choosing. Create a multiple line chart of the closing prices of the four stocks on the same graph, showing each stock in a different color.

```
library(tidyquant)
library(dplyr)
stock.data <- tq_get(c("GOOGL", "AMZN", "AAPL", "NFLX" ))
stock.data$symbol <- factor(stock.data$symbol, levels = c("GOOGL", "AMZN", "AAPL", "NFLX"))

p5a <- ggplot(stock.data, aes(x = date, y = close)) +
  geom_line(aes(color = symbol)) +
```

```
  ylab("Closing Prices") +
  ggtitle("Closing Prices of Google, Amazon, Apple and Netflix")
p5a
```

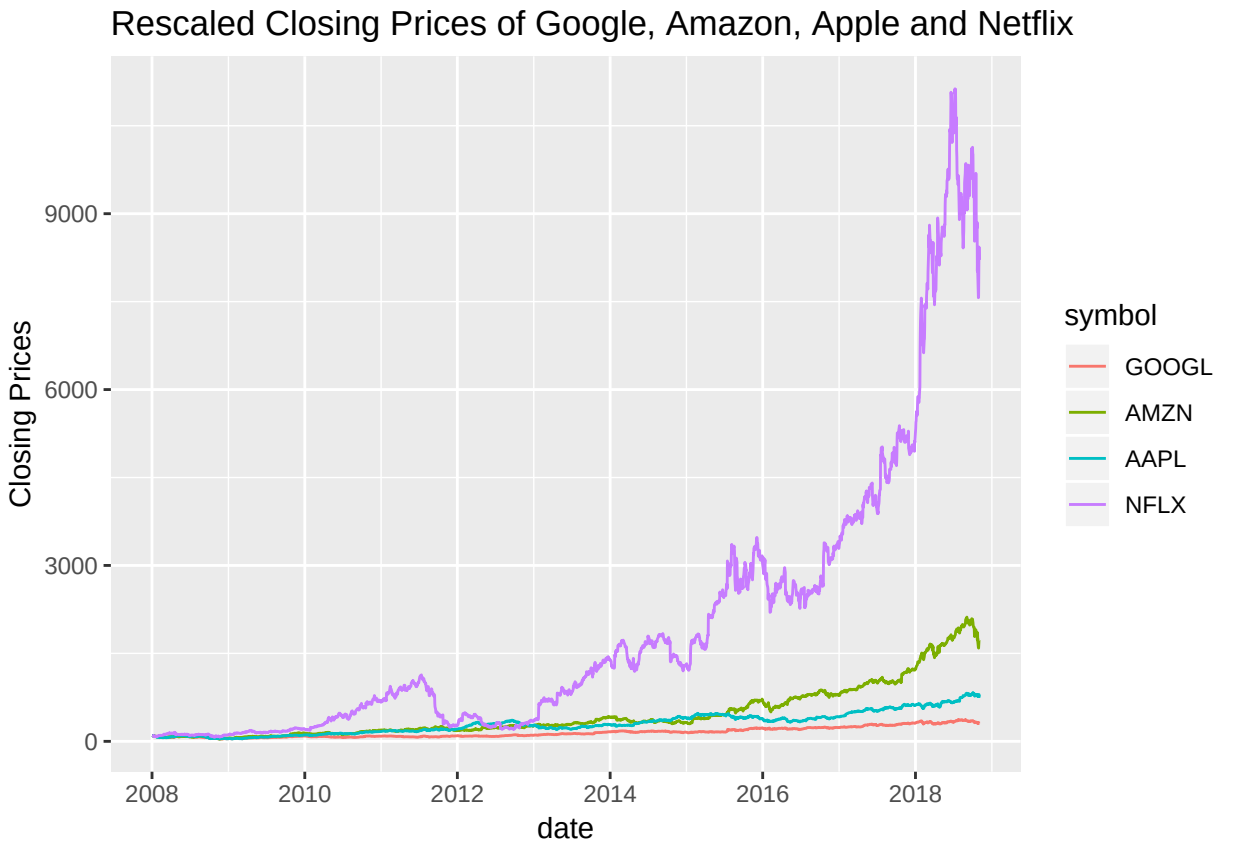


- From the graph we find Amazon's closing price is growing more repaidly than other three, especially in recent two years.
- All four companies' closing prices are growing, and in 2008 the closing prices from high to low is: Google, Amazon, Apple and Netflix. But in recent days, the closing prices from high to low is: Amazon, Google, Netflix and Apple.

(b) Transform the data so each stock begins at 100 and replot. Choose a starting date for which you have data on all of the stocks. Do you learn anything new that wasn't visible in (a)?

```
# rescale the data
rescale.data <- stock.data
rescale.data$close[rescale.data$symbol == "GOOGL"] <-
  rescale.data$close[rescale.data$symbol == "GOOGL"] /
  rescale.data$close[rescale.data$symbol == "GOOGL"][1] * 100
rescale.data$close[rescale.data$symbol == "AMZN"] <-
  rescale.data$close[rescale.data$symbol == "AMZN"] /
  rescale.data$close[rescale.data$symbol == "AMZN"][1] * 100
rescale.data$close[rescale.data$symbol == "AAPL"] <-
  rescale.data$close[rescale.data$symbol == "AAPL"] /
  rescale.data$close[rescale.data$symbol == "AAPL"][1] * 100
rescale.data$close[rescale.data$symbol == "NFLX"] <-
  rescale.data$close[rescale.data$symbol == "NFLX"] /
  rescale.data$close[rescale.data$symbol == "NFLX"][1] * 100
```

```
p5b <- ggplot(rescale.data, aes(x = date, y = close)) +
  geom_line(aes(color = symbol)) +
  ylab("Closing Prices") +
  ggtitle("Rescaled Closing Prices of Google, Amazon, Apple and Netflix")
p5b
```



- In graph (a) we only can only compare the overal prices in a time cross-section, but after we rescale the data, we can find the relative change and growing trend of four stocks. Of which, we find Netflix grows the most rapidly, especially in the first half year of 2018. But this is also partly because the original price of Netflix is extremely low (3.76).
- Google grows most stably of the four stocks.That is, the perentage growth of its stock is the lowest. And Amazon (whose closing price is the highest according to graph(a))'s relative growing speed ranks the second.
- But overall all these four stocks are in growing trend from 2018 till now.

6. Presentation

Imagine that you have been asked to create a graph from the Dogs of NYC dataset that will be presented to a very important person (or people). The stakes are high.

(a) Who is the audience? (Mayor DeBlasio, a real estate developer, the voters, the City Council, the CEO of Purina. . .)

(b) What is the main point you hope someone will take away from the graph?

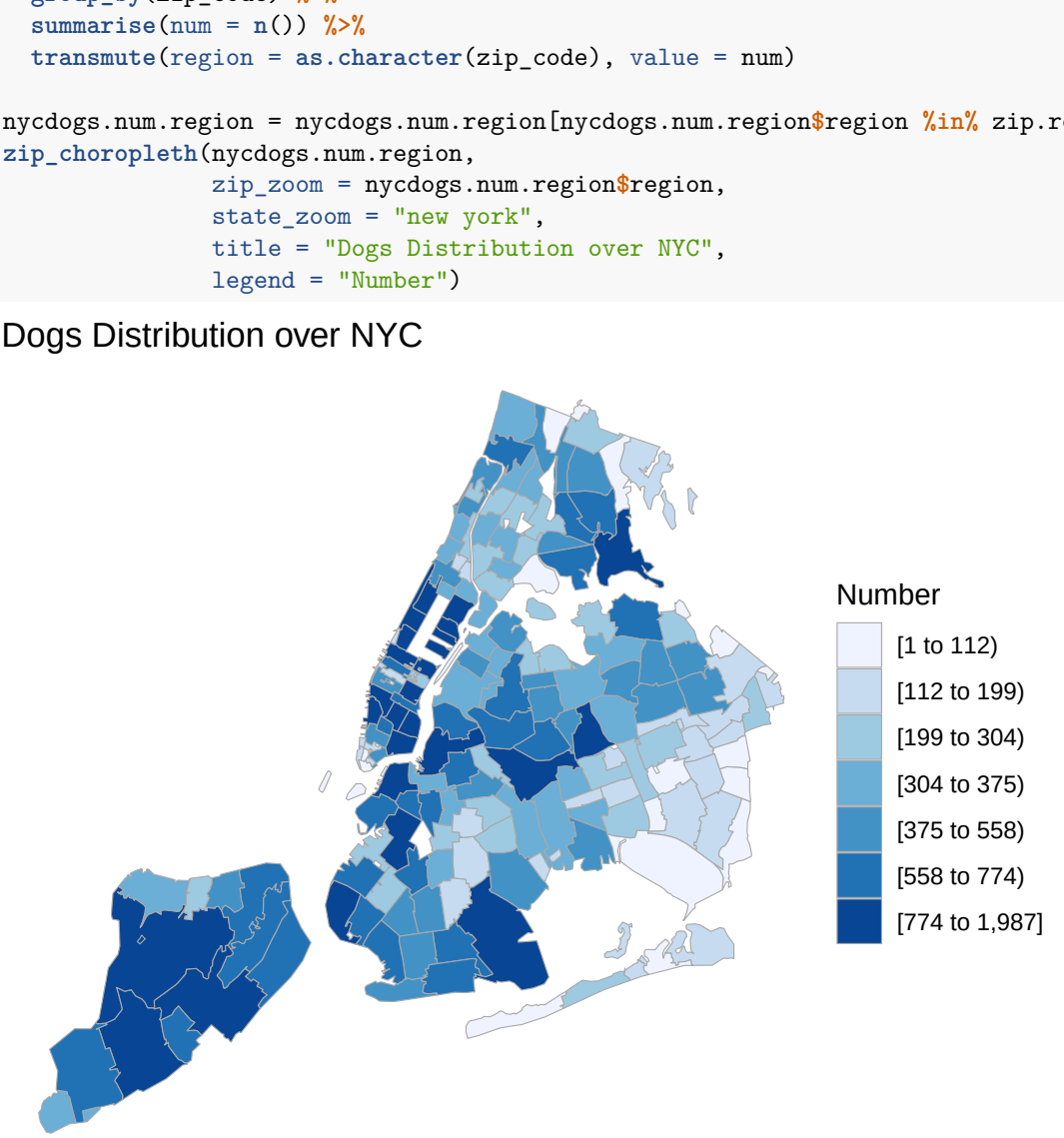
(c) Present the graph, cleaned up to the standards of "presentation style." Pay attention to choice of graph

type, if and how the data will be summarized, if and how the data will be subsetted, title, axis labels, axis breaks, axis tick mark labels, color, gridlines, and any other relevant features.

Answers: (a) The audience is the CEO of Purina.

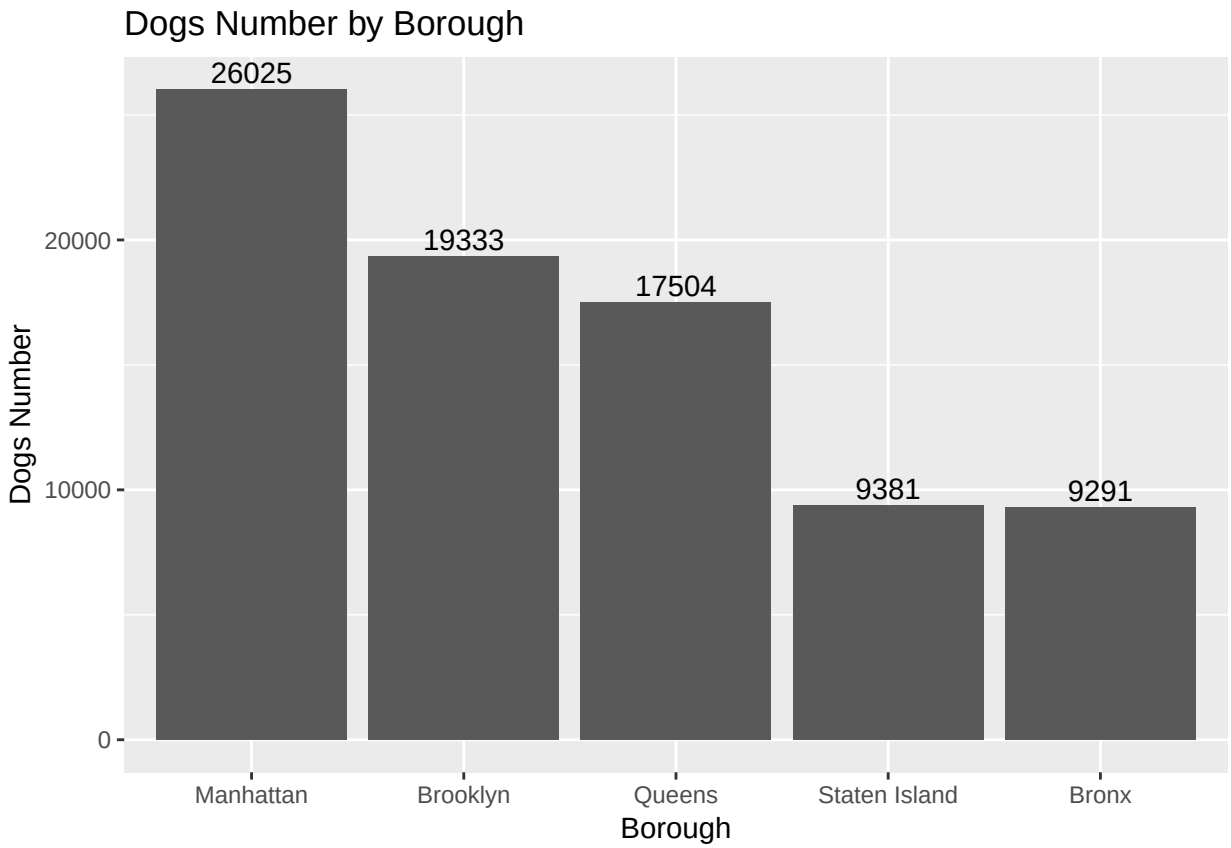
(b) I'll draw a spatial heat map on the number of dogs by zipcode, and to show the distribution of dogs over New York City. This will indicate where on earth has more dogs. I'll also draw a bar chart by borough, which quantitively tells the number of dogs in five boroughs. So from the graphs, the CEO of Purina can better arrange their pet shelters and pet stores according to the density and number of dogs so as to service more dogs in relative low cost.

```
# draw the spatial heat map on the number of dogs by zipcode
nycdogs.num.region <- nycdogs %>%
  group_by(zip_code) %>%
  summarise(num = n()) %>%
  transmute(region = as.character(zip_code), value = num)

nycdogs.num.region = nycdogs.num.region[nycdogs.num.region$region %in% zip.regions$region, ]
zip_choropleth(nycdogs.num.region,
               zip_zoom = nycdogs.num.region$region,
               state_zoom = "new york",
               title = "Dogs Distribution over NYC",
               legend = "Number")
```




```
# draw the bar chart of number of dogs in different boroughs
dogs.borough <- nycdogs %>%
  group_by(borough) %>%
  summarise(num = n())

p6c <- ggplot(dogs.borough, aes(x = reorder(borough, -num), y = num)) +
```

```
  geom_bar(stat = "identity") +
  ggtitle("Dogs Number by Borough") +
  xlab("Borough") + ylab("Dogs Number") +
  geom_text(aes(label = num), vjust = -0.3)
p6c
```



- So from these graphs, we know we should set more shelters and pet shops in Manhattan, Brooklyn and Queens. We can also find some areas by zipcode (like mid Staten Island) with great numbers of dogs, which will be a good place to set shelters and pet shops.