GR5702 EDAV Homework 3

Po-Chieh Liu (pl2441)

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/ (https://project.wnyc.org/dogs-of-nyc/)

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 (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/ (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.

```
# import necessary libraries insta
library(tidyverse)
library(extracat)
library(vcd)
library(grid)
library(choroplethr)
library(choroplethrZip)
library(tidyquant)
Sys.setlocale("LC_TIME", "C")
```

```
## [1] "C"
```

```
# import given data file
NYCdog <- read_csv('NYCdogs.csv')
```

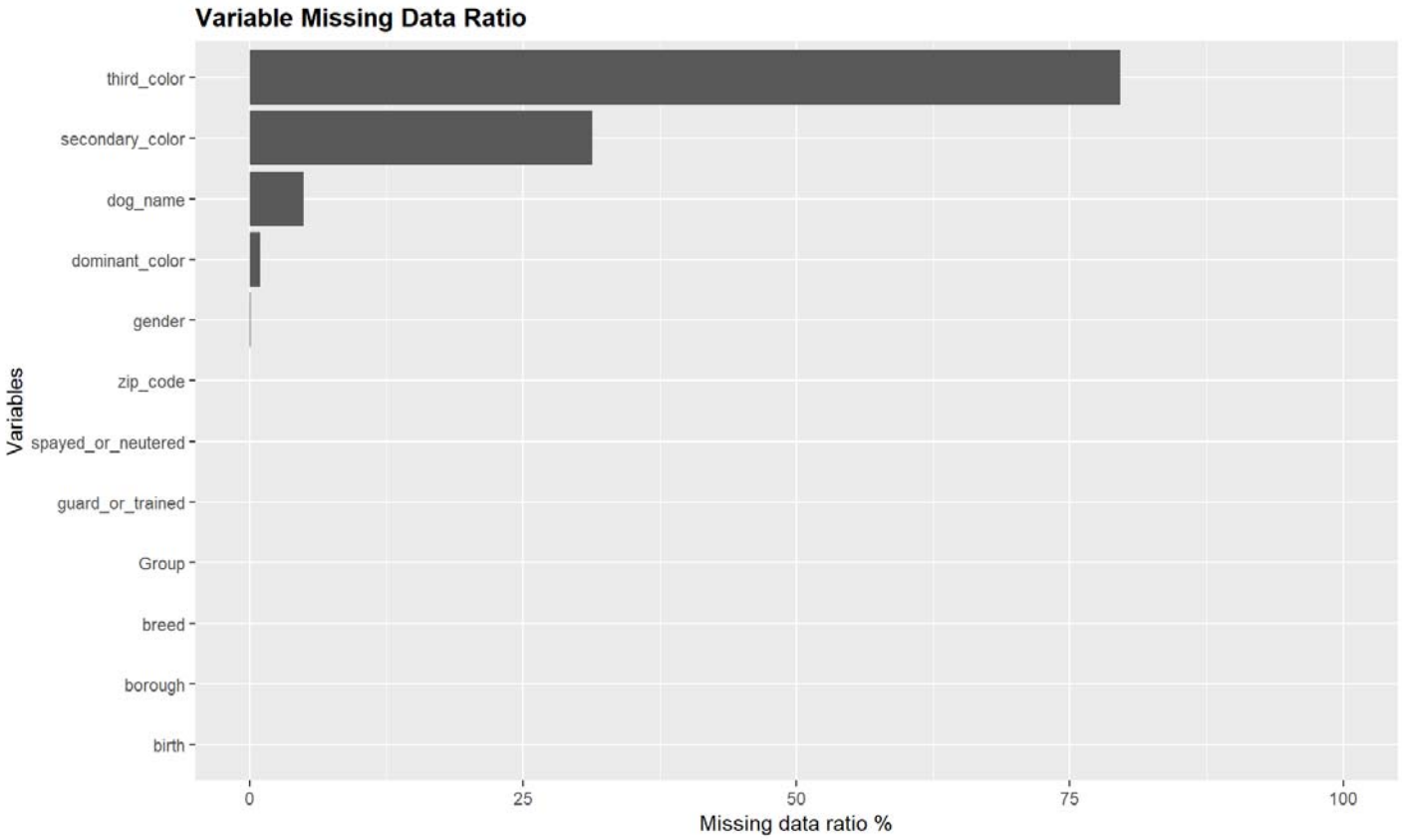
1. Missing Data

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

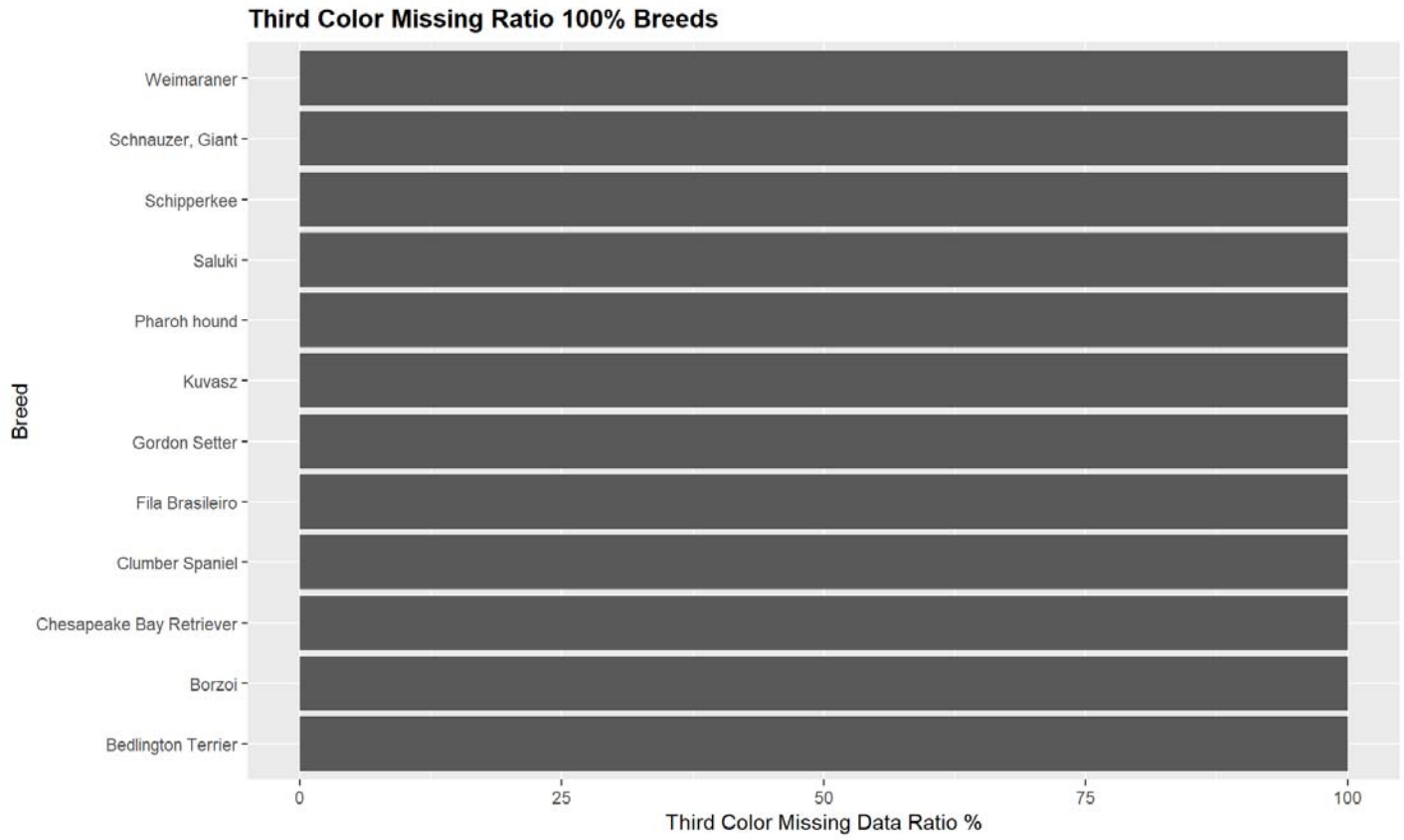
Observing the data after loaded into R, several data points are showing "n/a" for missing data. The following R codes first count the number of "n/a" of each column then convert the counts to variable missing ratios. Then the R codes plot missing data bar chart of each variable with descending order. The plot shows third and secondary color are the two variables with highest missing data ratios. I think the reason might be lots of breed dog are usually not triple color dog. Thus there is no information for filling third color. Dominant color is fourth highest missing data variable, those missing records might be caused by some types of dog don't have dominant color, for example, Border Collies with black and white color are hard to determine their dominant color. The second, third and fourth plots are for exploratory purpose. Second and third plots show the breeds with 100% and lower than 30% of third color missing rates. The fourth plot shows the breeds have more than 2% dominant color missing rate. I did some internet search on dog breed and its color. I think the image results are agreeing with my thought. For example, Bernese Mountain Dog has third color but Weimaraner usually has one color. However, the actual

reason for missing data on some data points might be simpler like just not recording properly. The third highest missing ratio variable is dog name. Maybe name is not required for licensing, owner can choose to finish the processes first and name their pet later. Note that gender variable also has missing ratio around 0.07%, and the rest of the variables don't have missing data.

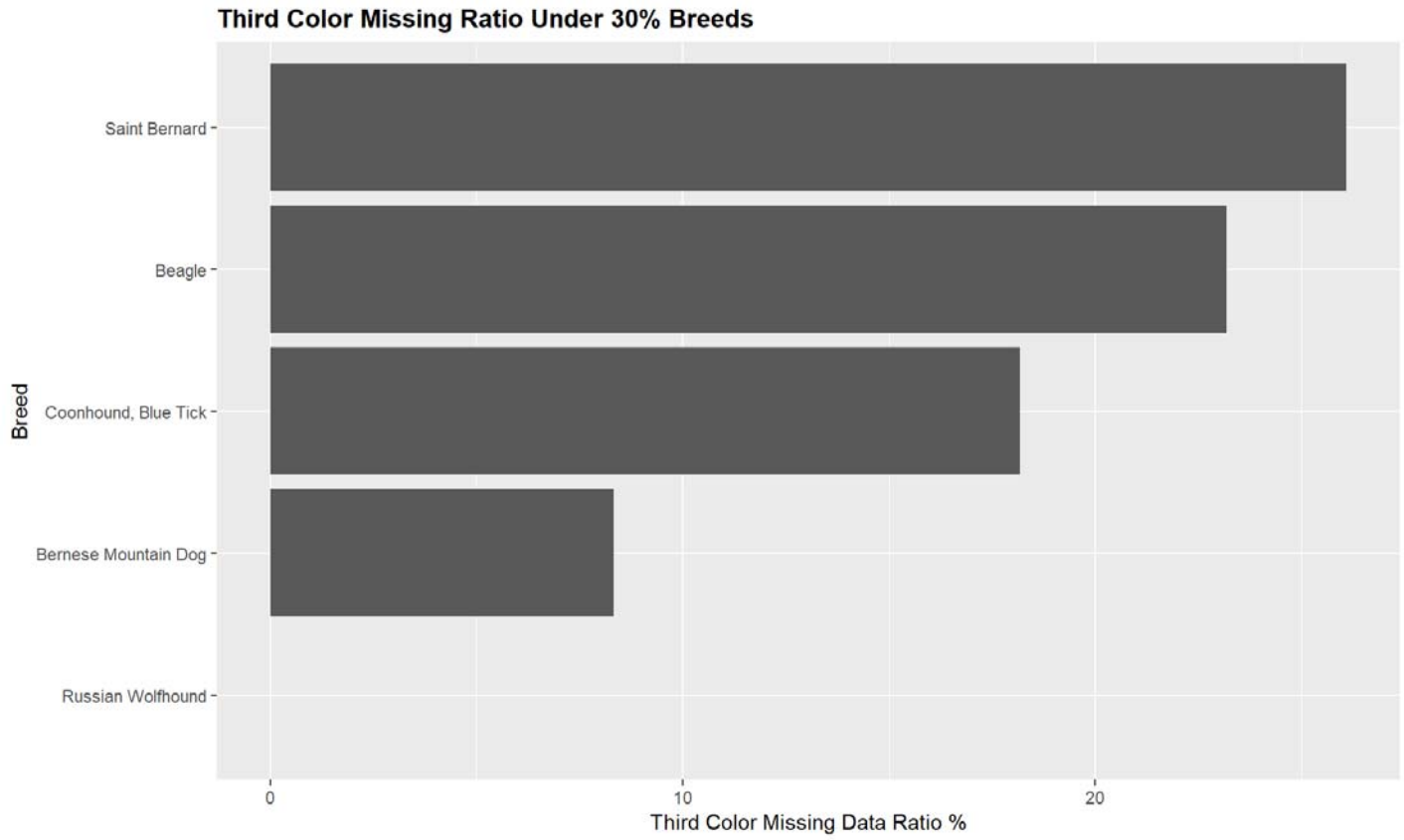
```
# count the missing data
missing_data_ratio <- NYCdog %>% select(everything()) %>% summarise_all(funs(sum(.=='n/a')/n()))
 %>% gather()
# generate bar chart
ggplot(missing_data_ratio, aes(x = fct_reorder(key,value), value*100)) +
  geom_col() + coord_flip() + ylim(0,100) +
  labs(x= "Variables", y = "Missing data ratio %") +
  ggtitle("Variable Missing Data Ratio") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```



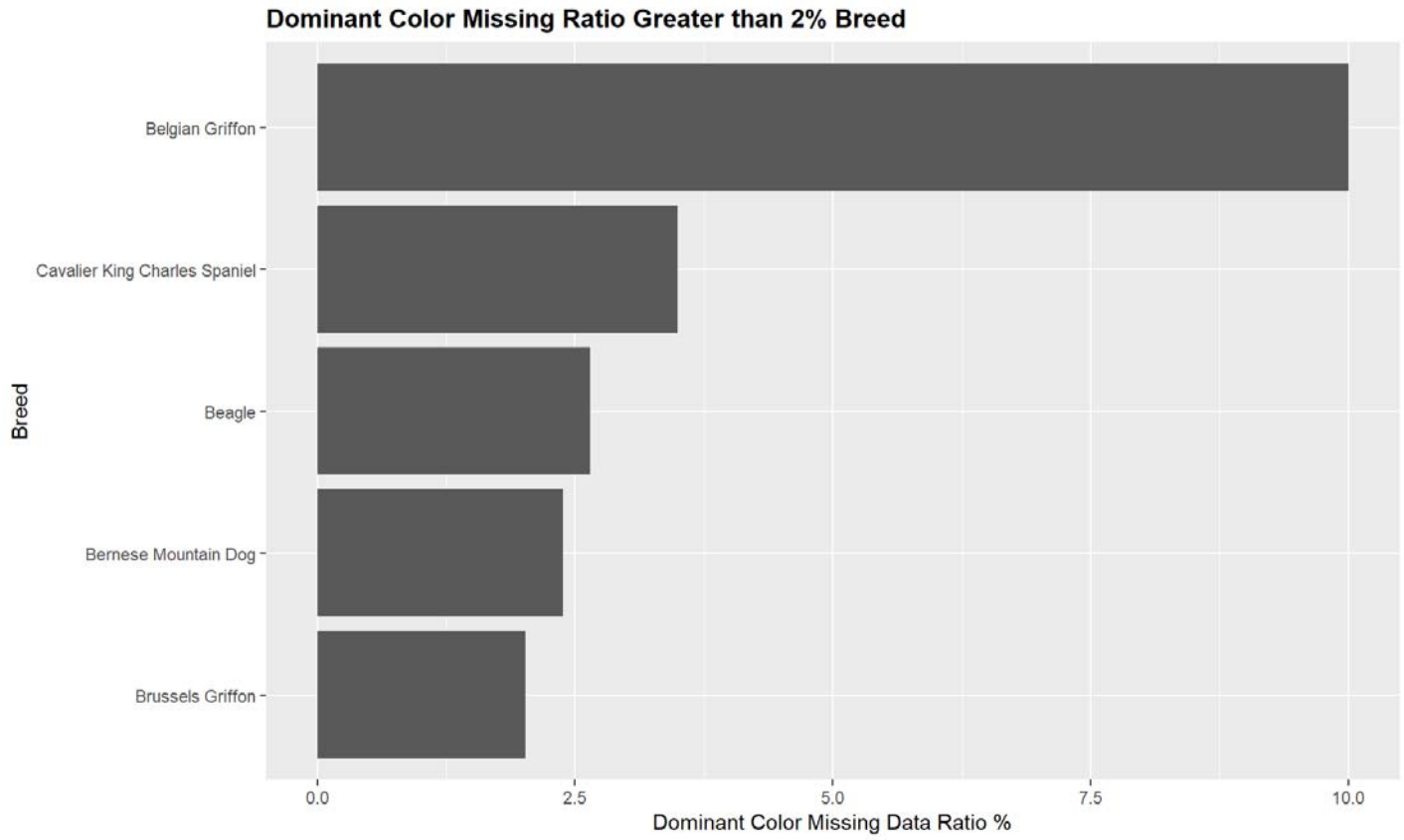

```
df <- NYCdog %>% select(third_color, breed) %>% group_by(breed) %>% summarise_all(funs(sum(.==
'n/a')/n())) %>% filter(third_color ==1)
ggplot(df, aes(breed, third_color*100)) +
  geom_col() + coord_flip() +
  labs(x= "Breed", y = "Third Color Missing Data Ratio %") +
  ggtitle("Third Color Missing Ratio 100% Breeds") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```




```
df <- NYCdog %>% select(third_color, breed) %>% group_by(breed) %>% summarise_all(funs(sum(.==
'n/a')/n())) %>% filter(third_color <0.3)
ggplot(df, aes(x= fct_reorder(breed,third_color), third_color*100)) +
  geom_col() + coord_flip() +
  labs(x= "Breed", y = "Third Color Missing Data Ratio %") +
  ggtitle("Third Color Missing Ratio Under 30% Breeds") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```



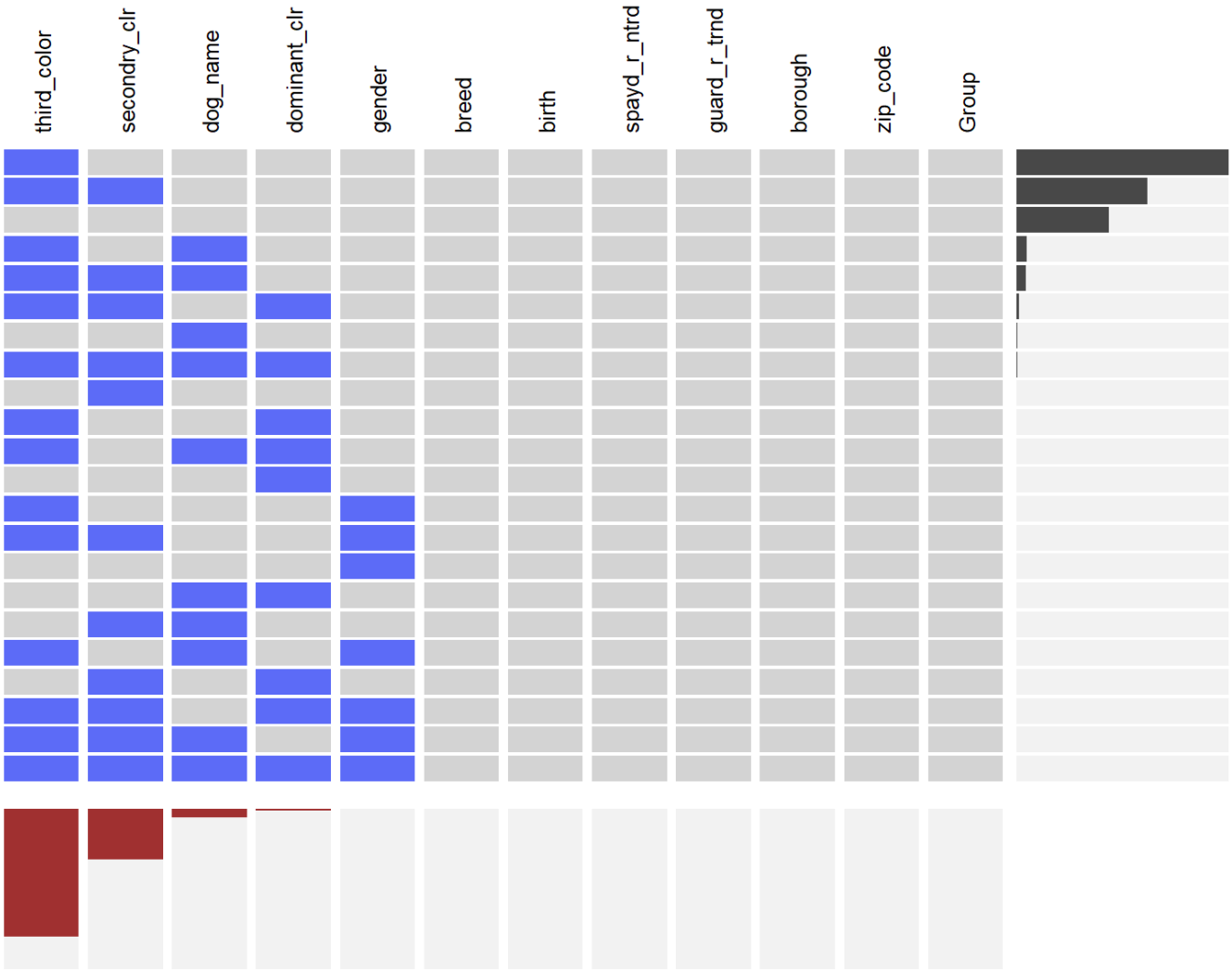

```
df <- NYCdog %>% select(dominant_color, breed) %>% group_by(breed) %>% summarise_all(funs(sum(.=
='n/a')/n())) %>% filter(dominant_color >0.02)
ggplot(df, aes(x=fct_reorder(breed,dominant_color), dominant_color*100)) +
  geom_col() + coord_flip() +
  labs(x= "Breed", y = "Dominant Color Missing Data Ratio %") +
  ggtitle("Dominant Color Missing Ratio Greater than 2% Breed") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```



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

The following R codes first convert the “n/a” into R’s N/A value, then apply visna function to generate missing data pattern. On the bottom of the plot, the missing data bar chart is visually the same as previous plotted bar chart. Based on the grids, there are 22 missing variable patterns of this data set. From the pattern frequency plot on the right, we can find that the dominant patterns are missing third color, missing both third and secondary color, and no missing data.

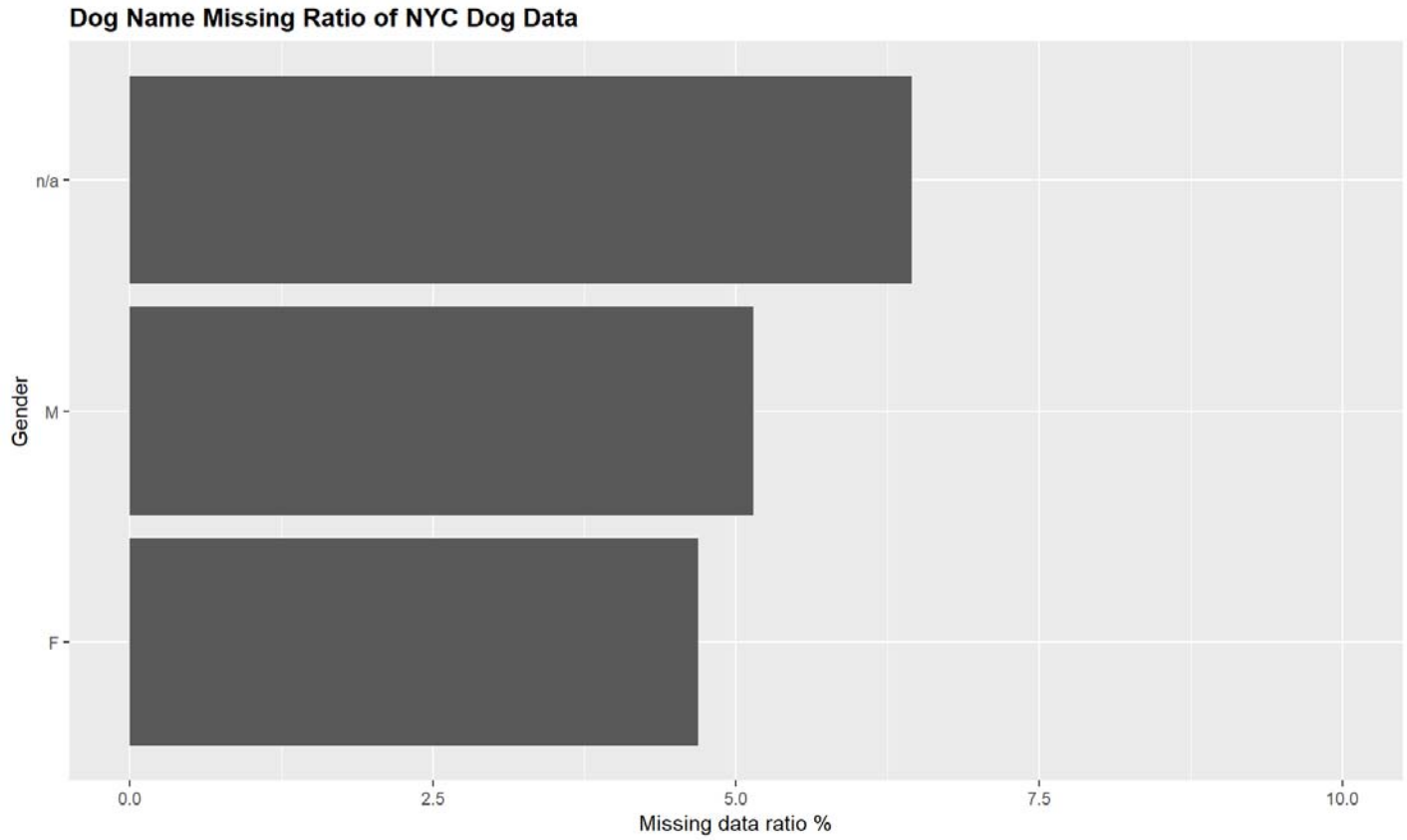
```
df <- NYCdog %>% na_if('n/a')
visna(df, sort = "b")
```



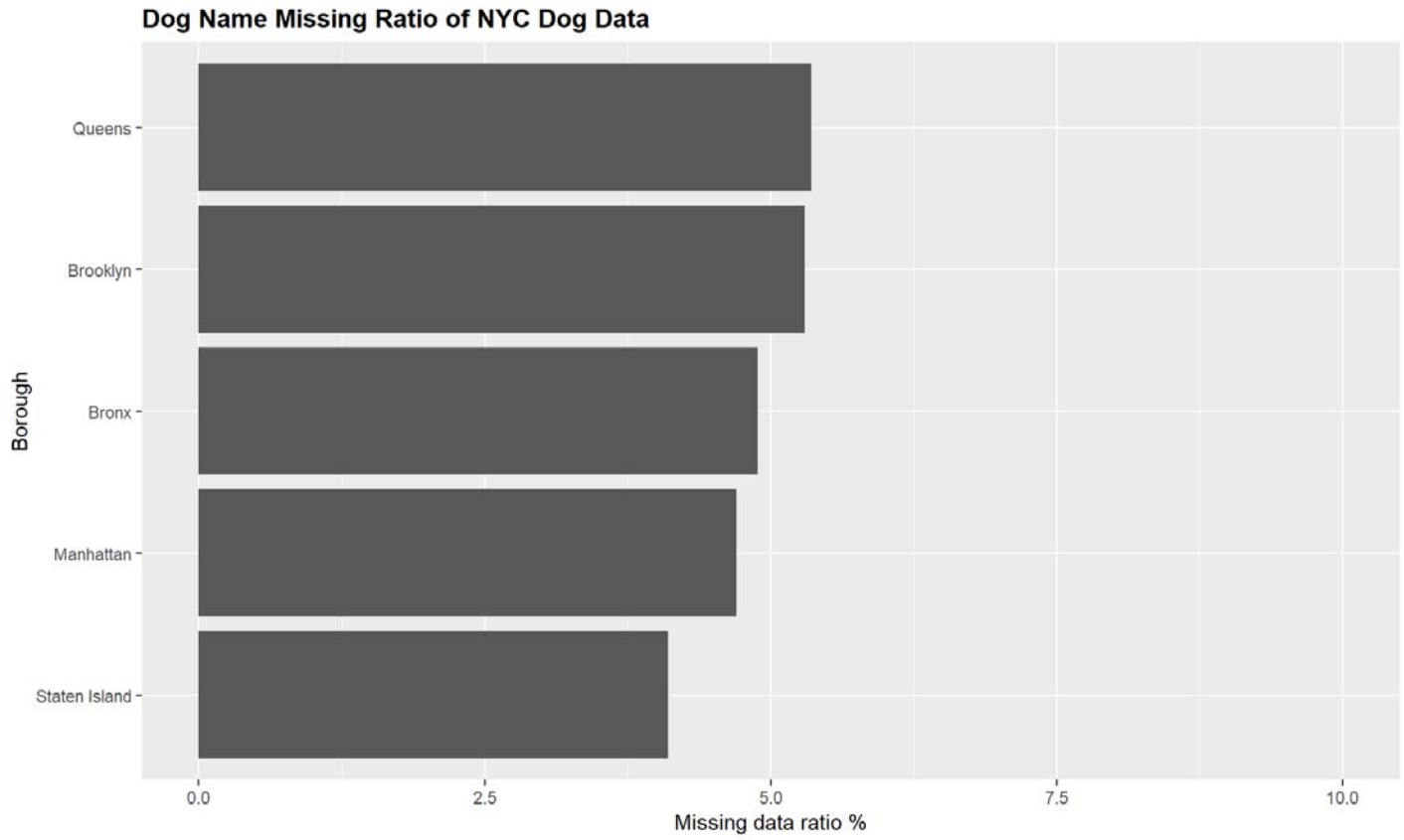
c. Do `dog_name` missing patterns appear to be associated with the *value* of `gender`, `Group` *or* `borough`?

The following R codes extract and generate the missing data ratio under different grouping conditions. From the first, we can observe the rate of both missing dog name and gender is higher than only missing dog name. Second, the male group has higher dog name missing rate than female group, however, the difference is relatively small in compared with n/a gender group. All groups have missing rate under 7.5%. The second plot shows the dog name missing ratio across five boroughs. We can see that Queens and Brooklyn have the highest dog name missing ratio, followed by Bronx, Manhattan and then Staten Island. All groups have missing rate under 6.25%. The dog name missing ratios are different under different genders and boroughs, however, the differences are small. The third plot shows the dog name missing rate under different groups. The differences between groups are more obvious than previous two plots. We can observe that non-sporting, toy, and mutt groups have relatively higher missing rates than other groups, and all above 5%. Hound group shows middle missing rate in this plot, and rest groups have missing rates around 2.5%. From the plot, we can found dog name missing patterns appear to be associated with Group.

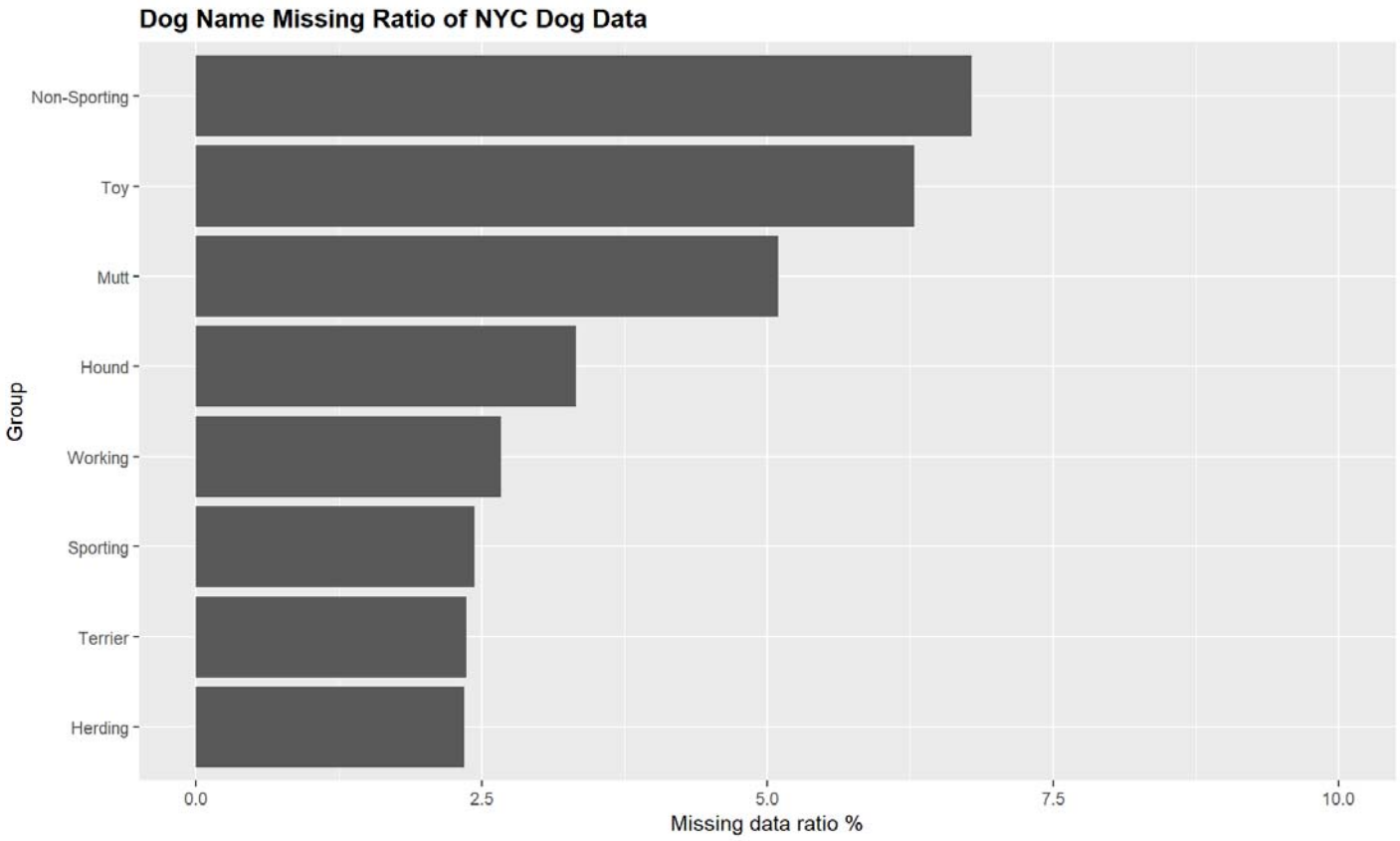
```
# extract dog_name, gender, Group, and borough
df <- NYCdog %>% select(dog_name, gender, Group, borough)
# calculate the dog name missing ratio under different gender
df_temp <- df %>% select(dog_name, gender) %>% group_by(gender) %>% summarise_all(funs(sum(.==
'n/a')/n()))
ggplot(df_temp, aes(gender, dog_name*100)) +
  geom_col() + coord_flip() + ylim(0,10) +
  labs(x= "Gender", y = "Missing data ratio %") +
  ggtitle("Dog Name Missing Ratio of NYC Dog Data") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```




```
# calculate the dog name missing ratio under different borough
df_temp <- df %>% select(dog_name, borough) %>% group_by(borough) %>% summarise_all(funs(sum(.==
'n/a')/n()))
ggplot(df_temp, aes(x = fct_reorder(borough,dog_name), dog_name*100)) +
  geom_col() + coord_flip() + ylim(0,10) +
  labs(x= "Borough", y = "Missing data ratio %") +
  ggtitle("Dog Name Missing Ratio of NYC Dog Data") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```

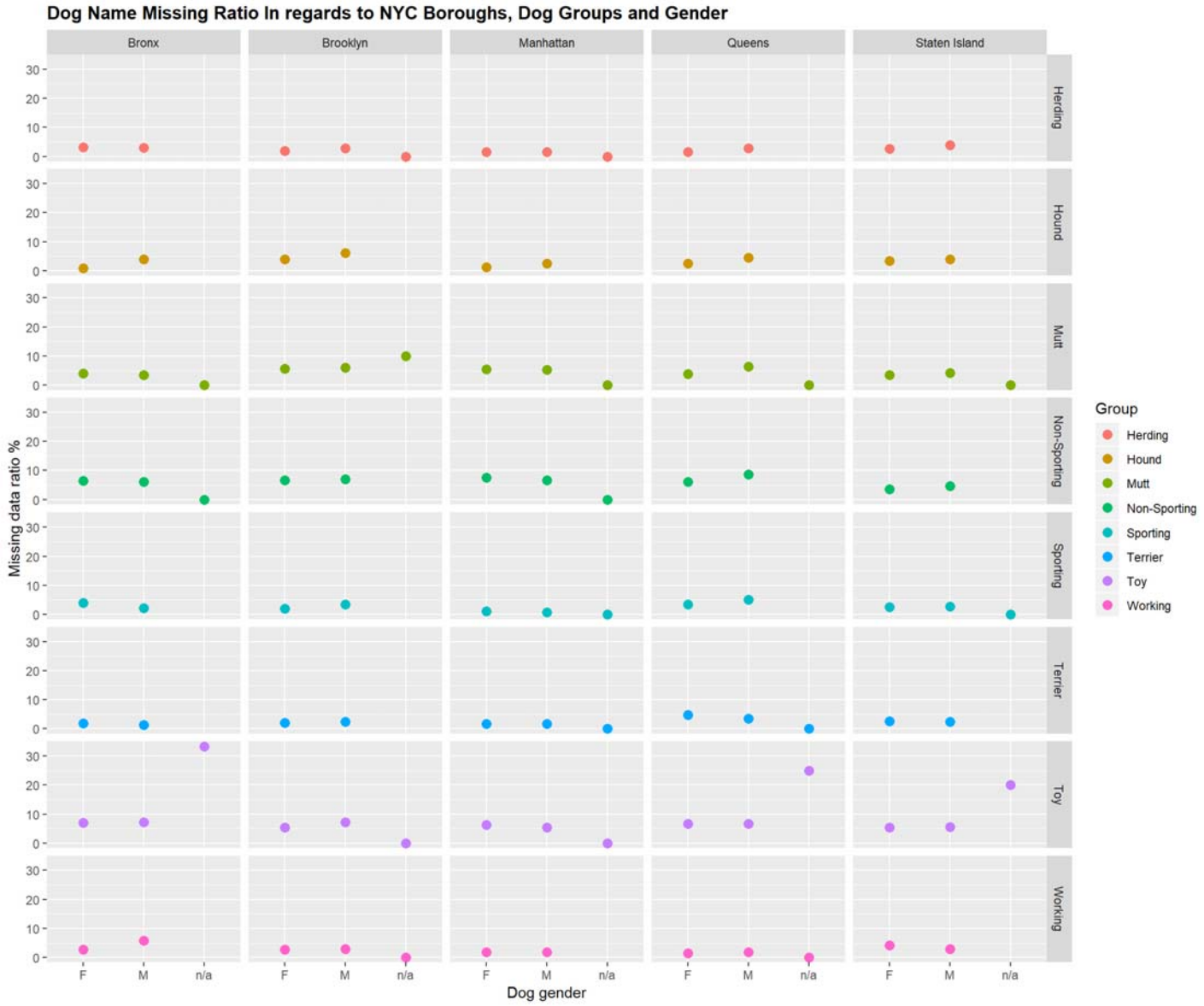



```
# calculate the dog name missing ratio under different Group
df_temp <- df %>% select(dog_name, Group) %>% group_by(Group) %>% summarise_all(funs(sum(.=='n/
a')/n()))
ggplot(df_temp, aes(x = fct_reorder(Group,dog_name), dog_name*100)) +
  geom_col() + coord_flip() + ylim(0,10) +
  labs(x= "Group", y = "Missing data ratio %") +
  ggtitle("Dog Name Missing Ratio of NYC Dog Data") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```



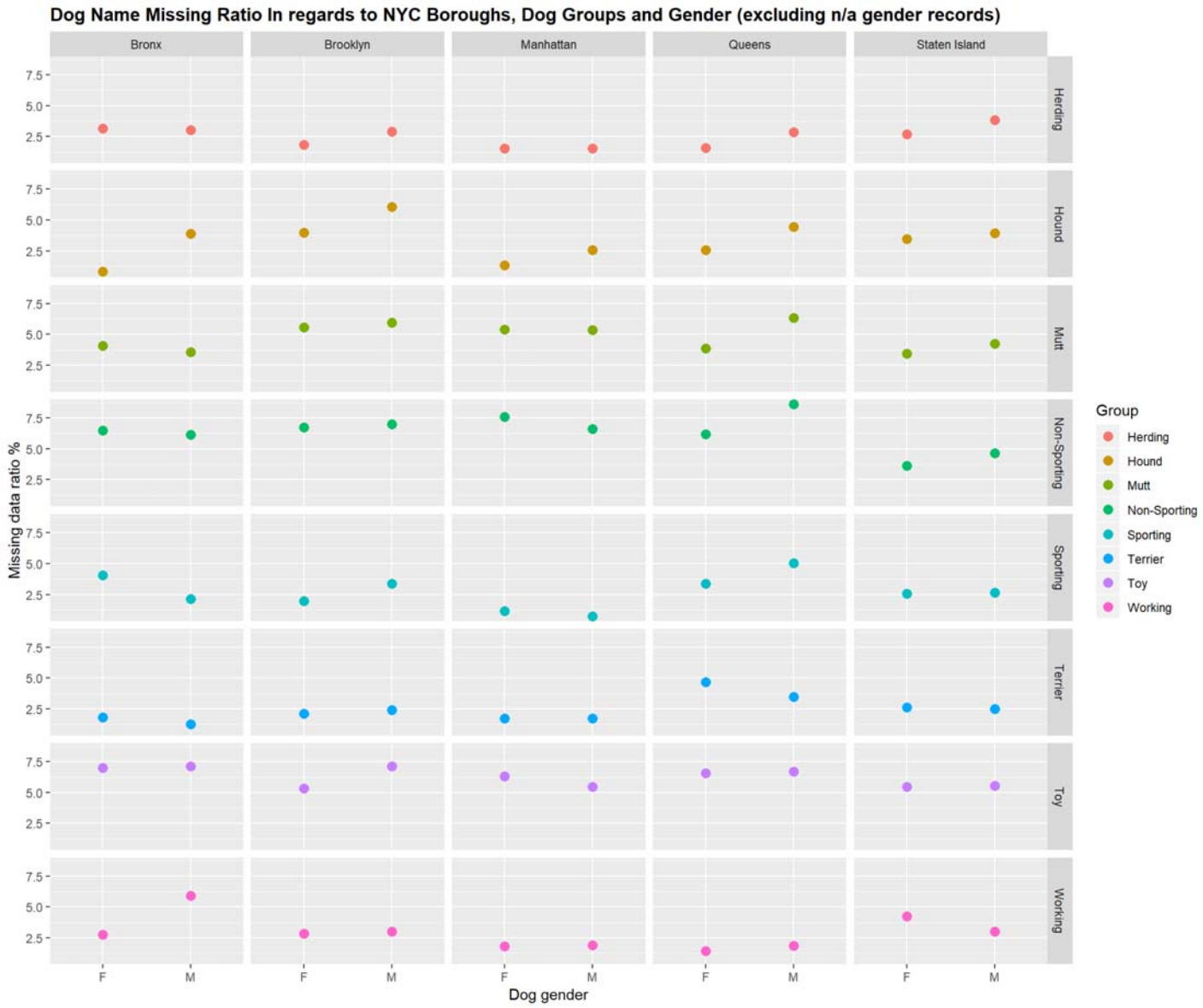
The following R codes generate two dot plots for missing dog name ratio in regards to dog gender, group and borough variables. The purpose of generating this plot is to put all variables together to see if there exist some patterns. The first plot shows that toy group in Staten Island, Queens and Bronx have relatively high rates of missing both dog name and gender. I checked the raw data, the high missing ratios of n/a gender group is due to rare data points. There are 21 dogs in toy group missing gender, and 3 of 21 are also missing name. All other group/borough/gender combination are all under 10%.

```
df <- NYCdog %>% select(dog_name, gender, Group, borough) %>% group_by(gender, Group, borough) %
>% summarise_all(funs(sum(.=='n/a')/n())) %>% ungroup()

ggplot(df, aes(x=gender, y=dog_name*100, color= Group )) +
  geom_point(size = 3, position="dodge") +
  facet_grid(Group~borough) +
  labs(x= "Dog gender", y = "Missing data ratio %") +
  ggtitle("Dog Name Missing Ratio In regards to NYC Boroughs, Dog Groups and Gender") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```



The second plot drops the records with n/a value in gender. From the second plot, mutt groups have high missing rates across 5 boroughs for both male and female. For hound group, the male dog name missing rates are higher than female across 5 boroughs. Male working dog group in Bronx and Female working dog group in Staten Island have relatively higher missing rate than other working dogs.

```
df <- NYCdog %>% select(dog_name, gender, Group, borough) %>% group_by(gender, Group, borough) %
>% filter(!(gender=='n/a'))%>% summarise_all(funs(sum(.=='n/a')/n())) %>% ungroup()

ggplot(df, aes(x=gender, y=dog_name*100, color= Group )) +
  geom_point(size = 3, position="dodge") +
  facet_grid(Group~borough) +
  labs(x= "Dog gender", y = "Missing data ratio %") +
  ggtitle("Dog Name Missing Ratio In regards to NYC Boroughs, Dog Groups and Gender (excluding
 n/a gender records)") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```

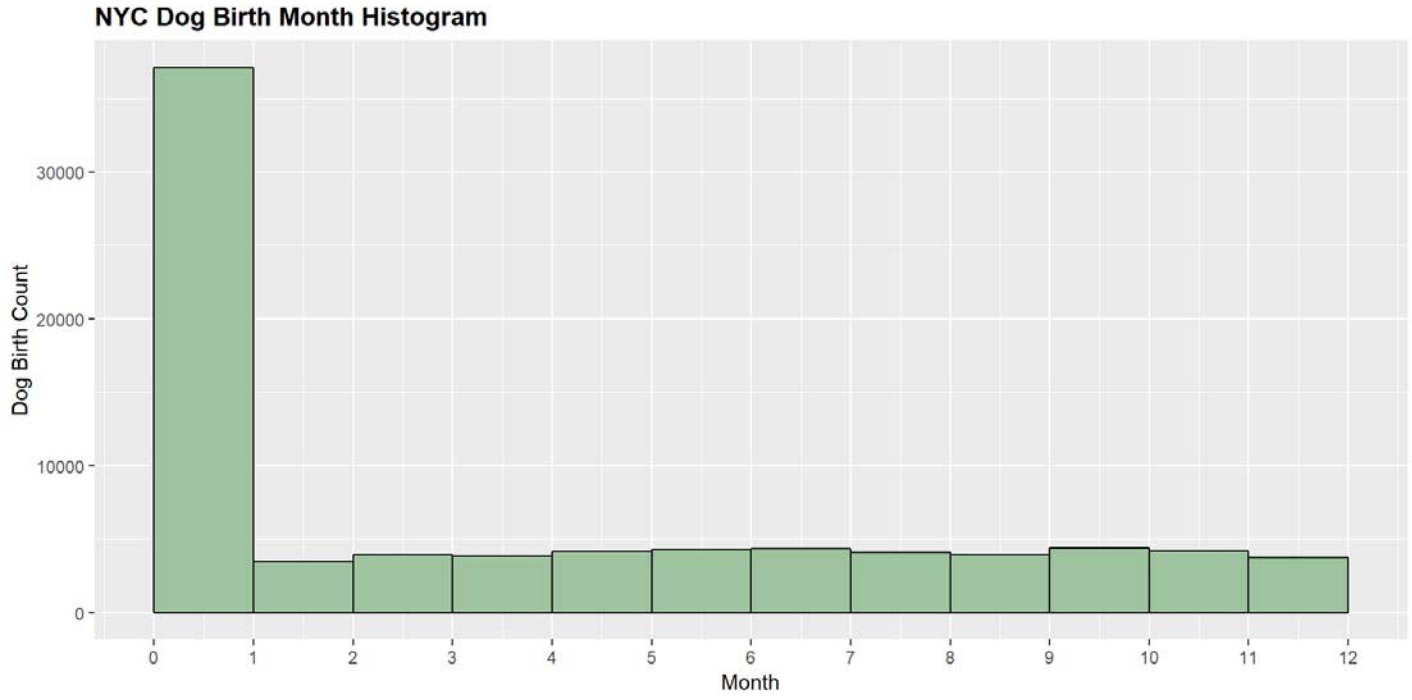


2. Dates

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.

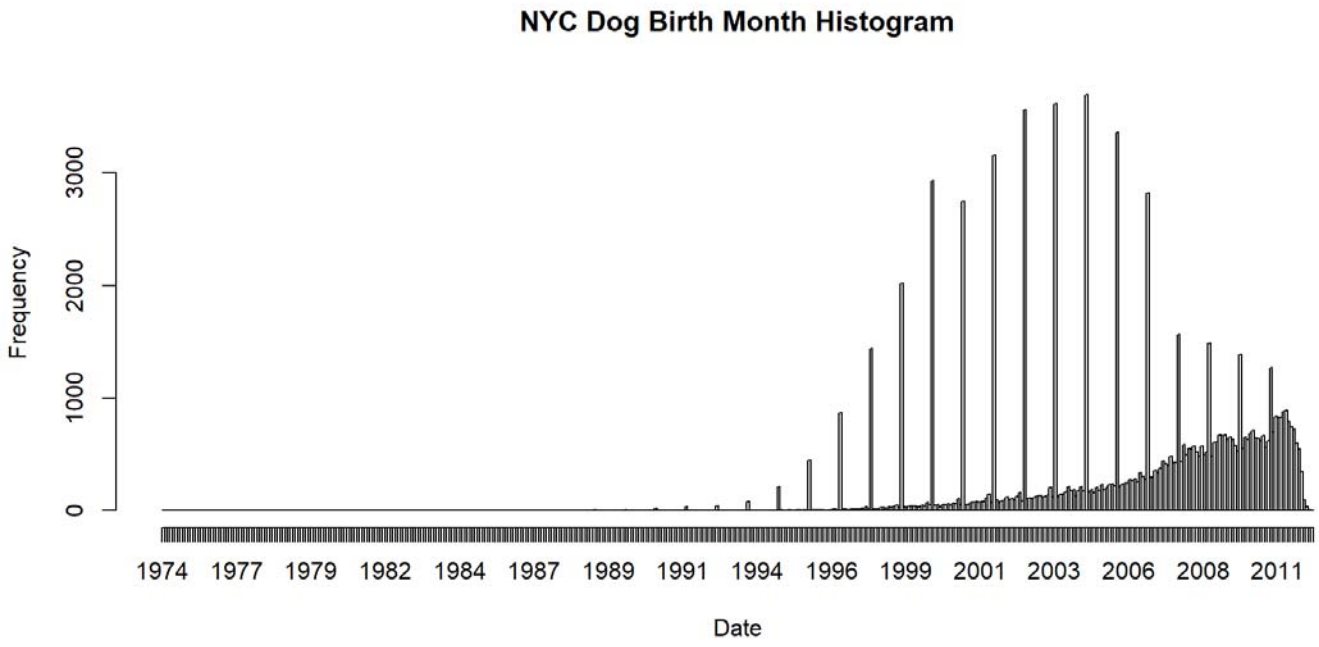
Observing the data, there are two different types of birth data. The following R codes extract the month word and convert the word into numerical month for generating histogram. From the plot, we can observe that more than 35,000 dogs were born in January, and the amount is way much higher than other months. Intuitively, I think the dog birth month should be uniformly distributed from January to December. From the plot, I guess January might be a default option for filling the license form, or owners might fill January if they don't know the actual birth month.

```
rule1 <- c("1-"="", "2-"="","3-"="","4-"="", "5-"="", "6-"="", "7-"="", "8-"="", "9-"="", "10-"=
"", "11-"="", "12-"="", "1"="")
rule2 <- c("Jan"="01", "Feb"="02", "Mar"="03", "Apr"="04", "May"="05", "Jun"="06", "Jul"="07",
"Aug"="08" , "Sep"="09", "Oct"="10","Nov"="11", "Dec"="12")

birth_1 <- NYCdog %>% select(birth) %>% filter(str_detect(birth,"^\\d")) %>% mutate(birth = str_
replace_all(birth,rule1)) %>% mutate(birth = str_replace_all(birth,rule2)) %>% mutate(birth = a
s.numeric(birth)) %>% filter(birth<=12) %>% filter(birth>0)

birth_2 <- NYCdog %>% select(birth) %>% filter( str_detect(birth,"^\\D")) %>% mutate(birth = sub
str(birth,0,3)) %>% mutate(birth = str_replace_all(birth,rule2)) %>% mutate(birth = as.numeric(b
irth))

df <- rbind(birth_1,birth_2)

ggplot(df, aes(x=birth)) +
  geom_histogram(bins=12, colour="black", fill = "#9FC29F", position = position_nudge(-0.5))+
  scale_x_continuous(labels = c("0","1","2","3","4","5","6","7","8","9","10","11","12"),breaks =
 seq(0, 12, len = 13)) +
  labs(x= "Month", y = "Dog Birth Count") +
  ggtitle("NYC Dog Birth Month Histogram") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```

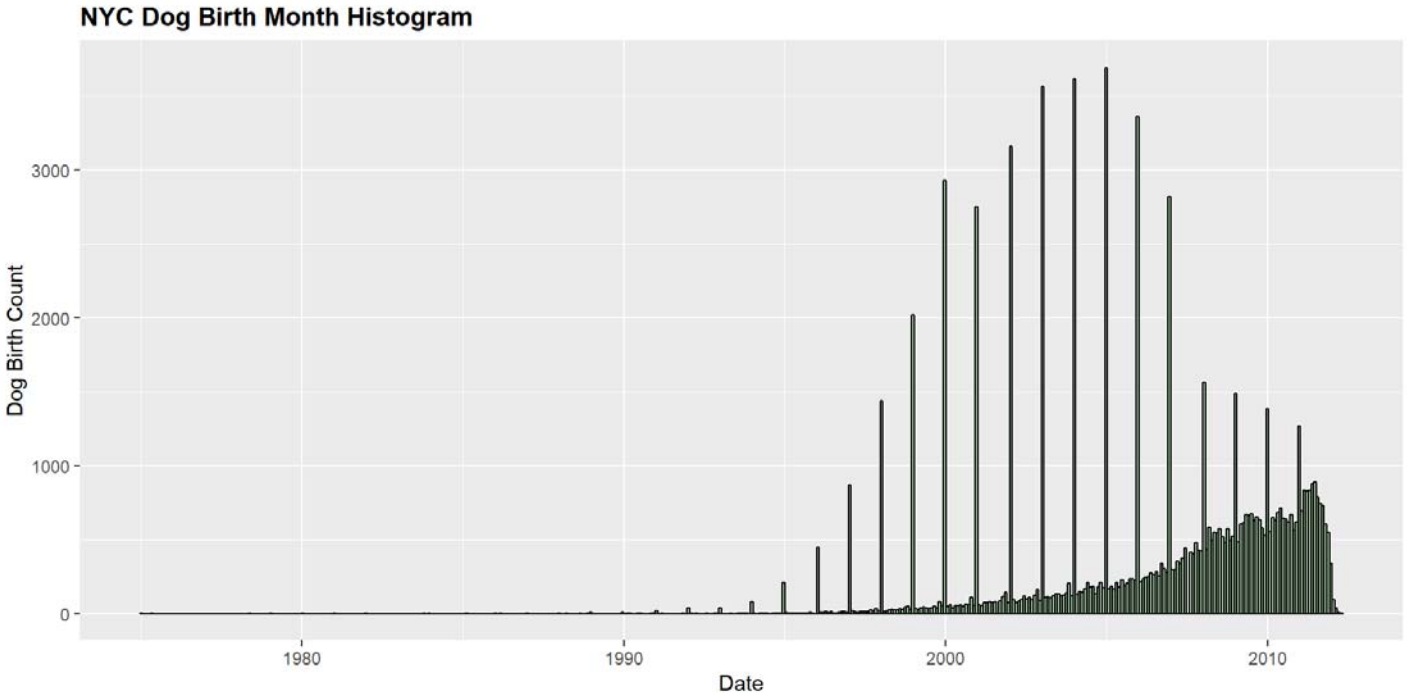


Before start working on next question, the following R codes generate the dog birth histogram based on both month and year for observing the pattern. Note the dog birth data with year greater than 2012 were dropped due to impossible birth year. We can observe the relatively high spikes exist on the beginning of every year since 1995. I think my hypothesis is consistent with this plot that lots of birth month might be filled in January due to default option or unknown information. And before 2008, the ratios of Jan are much higher than other months. The situation was improved since 2008. The second plot was trying to make same histogram using ggplot2. After some research I still can't figure out how to modify the binwith with month, thus I used 30 days as binwith to generate the plot. Visually the plot is similar to first plot generated by hist function.

```
birth_1 <- NYCdog %>% select(birth) %>% filter(str_detect(birth,"^\\d")) %>% mutate(birth = str_
replace_all(birth,rule2)) %>% mutate(birth = paste(birth,"-1",sep = ""))  %>% mutate(birth = as.
Date(birth, format = '%y-%m-%d')) %>% filter(birth<"2013-01-01")

birth_2 <- NYCdog %>% select(birth) %>% filter( str_detect(birth,"^\\D")) %>% mutate(birth = str
_replace_all(birth,rule2)) %>% mutate(birth = paste("1-",birth,sep = "")) %>% mutate(birth = as.
Date(birth, format = '%d-%m-%y')) %>% filter(birth<"2013-01-01")

df <- rbind(birth_1,birth_2)

hist(x = df$birth, breaks = 'months', plot = TRUE, format = "%Y",xlab = "Date", freq = TRUE, mai
n = "NYC Dog Birth Month Histogram")
```



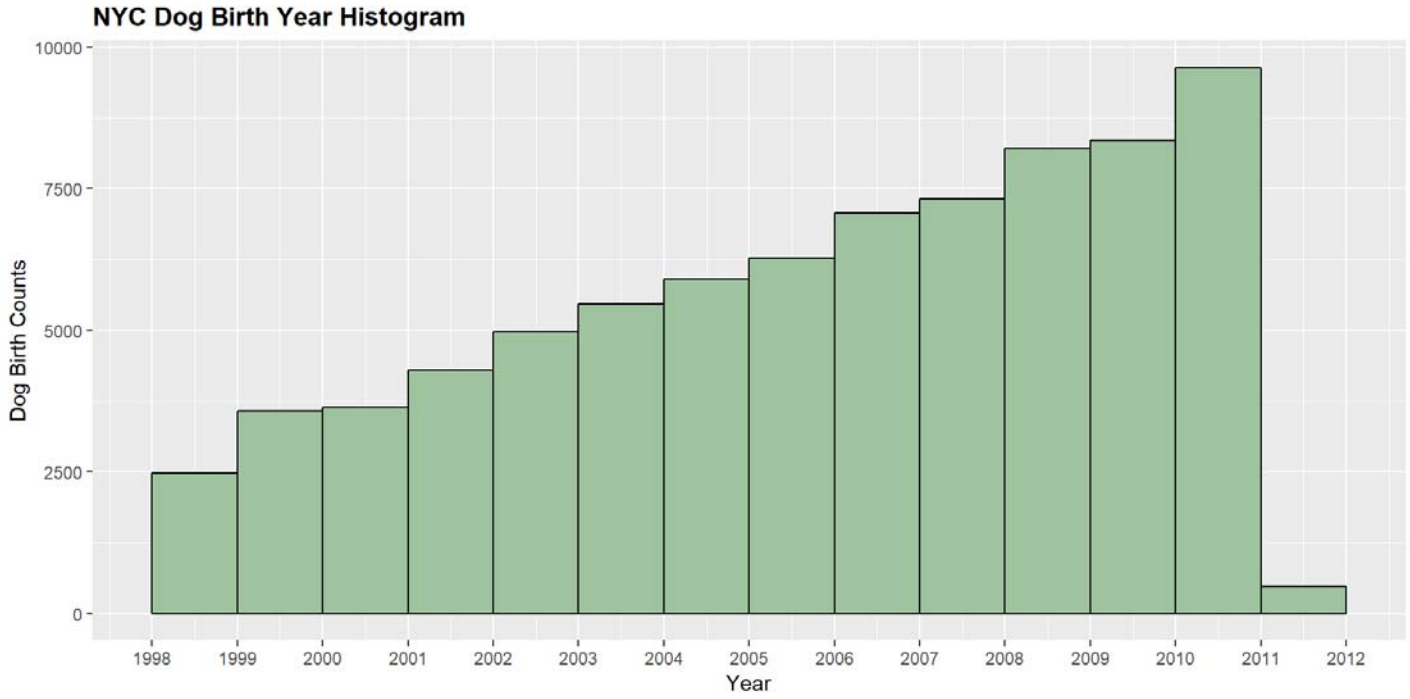

```
ggplot(df, aes(x=birth)) +
  geom_histogram(binwidth = 30,colour="black", fill = "#9FC29F", position = position_nudge(-0.5
))+
  labs(x= "Date", y = "Dog Birth Count") +
  ggtitle("NYC Dog Birth Month Histogram") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```



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

The following R codes generate dog birth histogram under year scale. The bin width is set as 1 year. The average dog life is around 10 to 13 years and this is the dataset as of 2012. Thus for data points above year 2012 or before 1999 are dropped. Also, there are 8 data points only include numbers like 0 to 99 are dropped due to lack information to decode. We can observe that we have fewer data points in 2012, which might be caused by the data collection time. The 2012 data set might not be complete when published. For the rest of the years, the dog counts are increased from birth year 1998 to 2011. This plot makes more sense to me because the quantities of elder dog groups are less than younger dog groups.

```
birth <- NYCdog %>% select(birth)
rule <- c("Jan"="1", "Feb"="2", "Mar"="3", "Apr"="4", "May"="5", "Jun"="6", "Jul"="7", "Aug"="8"
 , "Sep"="9", "Oct"="10","Nov"="11", "Dec"="12", "-"="/")
birth_1 <- birth %>% filter(!str_length(birth)<5) %>% filter(str_detect(birth,"^\\d")) %>% mutat
e(birth = str_replace_all(birth,rule)) %>% mutate(birth = paste(birth,"/1",sep = "")) %>% mutate
(birth = as.Date(birth, format = '%y/%m/%d'))

birth_2 <- birth %>% filter(!str_length(birth)<5) %>% filter( str_detect(birth,"^\\D")) %>% muta
te(birth = str_replace_all(birth,rule)) %>% mutate(birth = paste("1/",birth,sep = "")) %>% mutat
e(birth = as.Date(birth, format = '%d/%m/%y'))

df <- rbind(birth_1, birth_2)
df<- df %>% mutate(birth=year(birth)) %>% filter(!(birth>2012)) %>% filter(!(birth<1999))

ggplot(df, aes(x=birth)) +
  geom_histogram(binwidth = 1, colour="black", fill = "#9FC29F",position = position_nudge(-0.5))
+
  scale_x_continuous(breaks = 1998:2012) +
  labs(x= "Year", y = "Dog Birth Counts") +
  ggtitle("NYC Dog Birth Year Histogram") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))
```

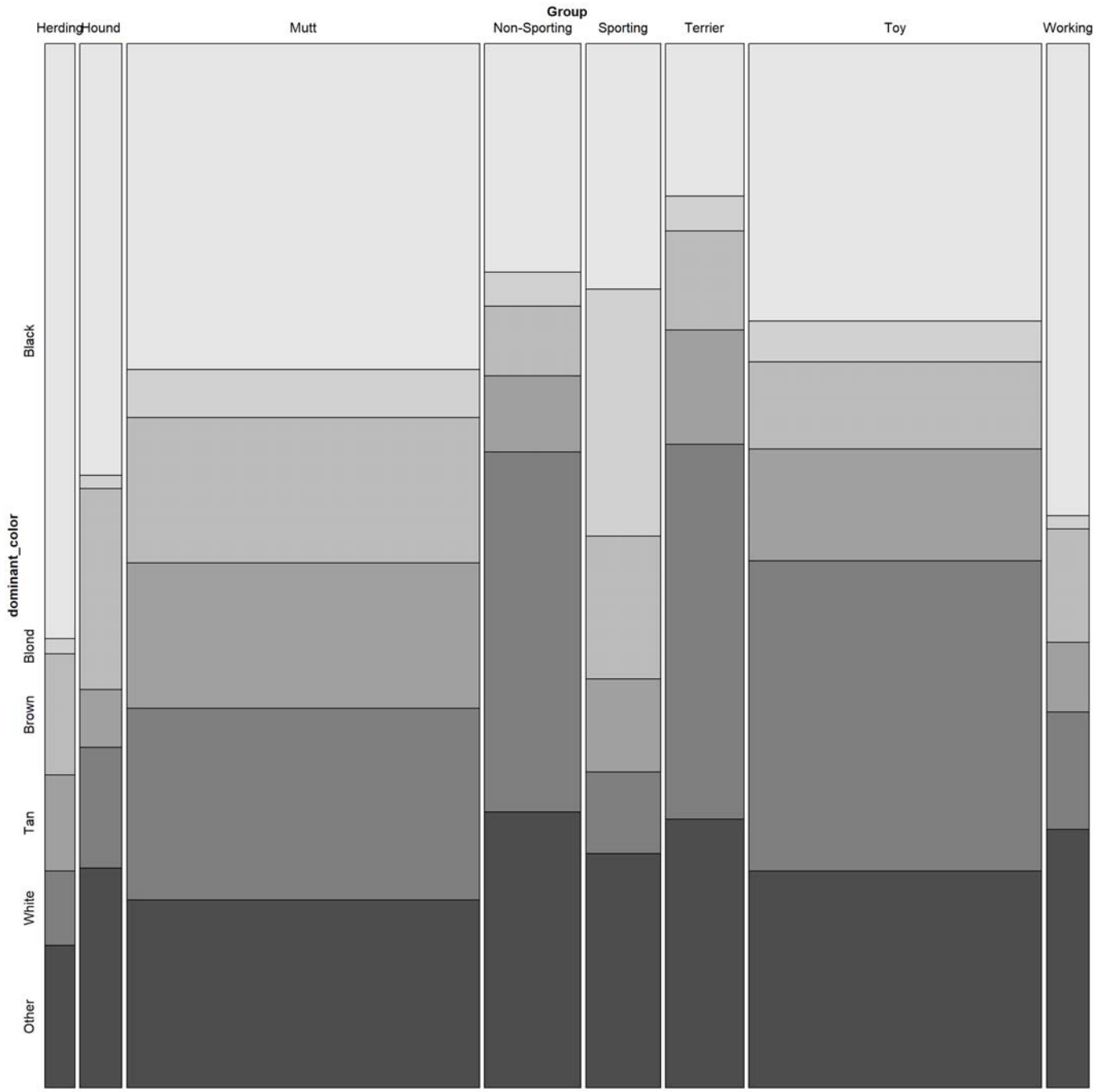


3. Mosaic plots

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.

Following R codes generate the mosaic plot of dog dominant color and group. From the plot, I think the color is associated with group. We can observe that the black color has greater portions than others in herding, hound, mutt, and working groups. White color has largest portions in toy, non-sporting and terrier groups. Except sporting group, both black and white color are the majority color. Blond color has a large portion in sporting group, but very few in others. For black and white dominant groups, I think the color might be an important factor for owners. For sporting group, the physical factors such as speed and agility are more important than color.

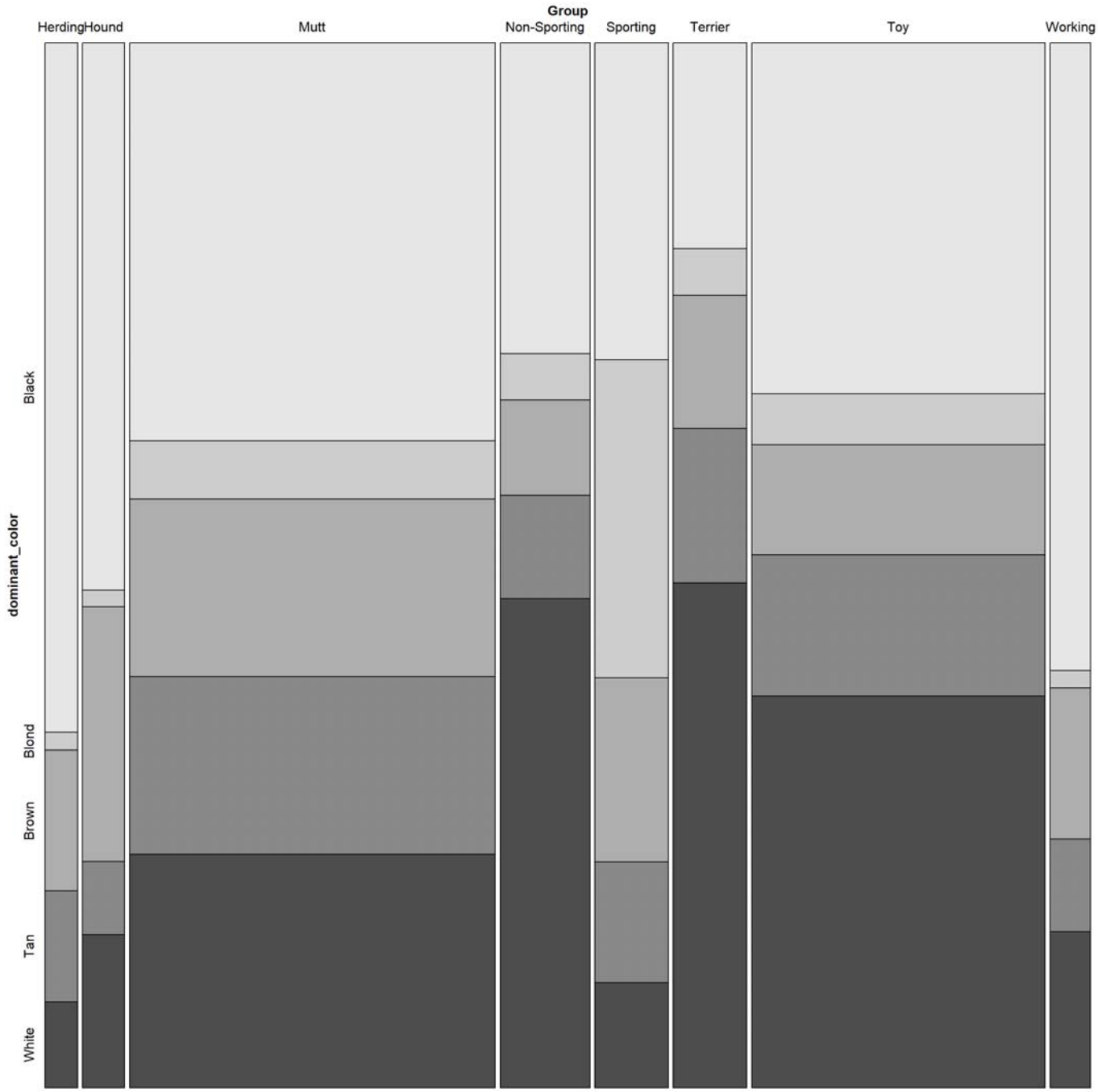
```
rule <- c("BLACK", "BLOND", "BROWN", "TAN", "WHITE")
df <- NYCdog %>% select(dominant_color, Group) %>% mutate(dominant_color = if_else(dominant_colo
r %in% rule, dominant_color, "OTHER" )) %>% group_by(dominant_color, Group)
df$dominant_color <- as.factor(df$dominant_color)
df$dominant_color <- factor(df$dominant_color, levels = c("BLACK", "BLOND", "BROWN", "TAN", "WHI
TE","OTHER"))
mosaic(dominant_color~Group, df, direction = c("v","h"), set_labels=list(dominant_color = c("Bla
ck", "Blond", "Brown", "Tan", "White","Other")))
```



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?

After removing OTHER category, we can observe that black or white dominate trend is more clear due to less number of color group. Similarly, black color has largest portions in heading, hound, mutt and working groups. For toy, non-sporting and terrier groups, white has largest portions. Sporting group also has similar portions like before. Those observed results are remained after removing other category. However, I prefer to keep other category to show that there are still other types of color exist. Also, if high percentage is observed in the other category, one may want to isolate a new color group (Ex. GRAY, BRINDLE).

```
df <- NYCdog %>% select(dominant_color, Group) %>% filter(dominant_color %in% rule) %>% group_by
(dominant_color, Group)
df$dominant_color <- as.factor(df$dominant_color)
df$dominant_color <- factor(df$dominant_color, levels = c("BLACK", "BLOND", "BROWN", "TAN", "WHI
TE"))
mosaic(dominant_color~Group, df, direction = c("v","h"), set_labels=list(dominant_color = c("Bla
ck", "Blond", "Brown", "Tan", "White")))
```



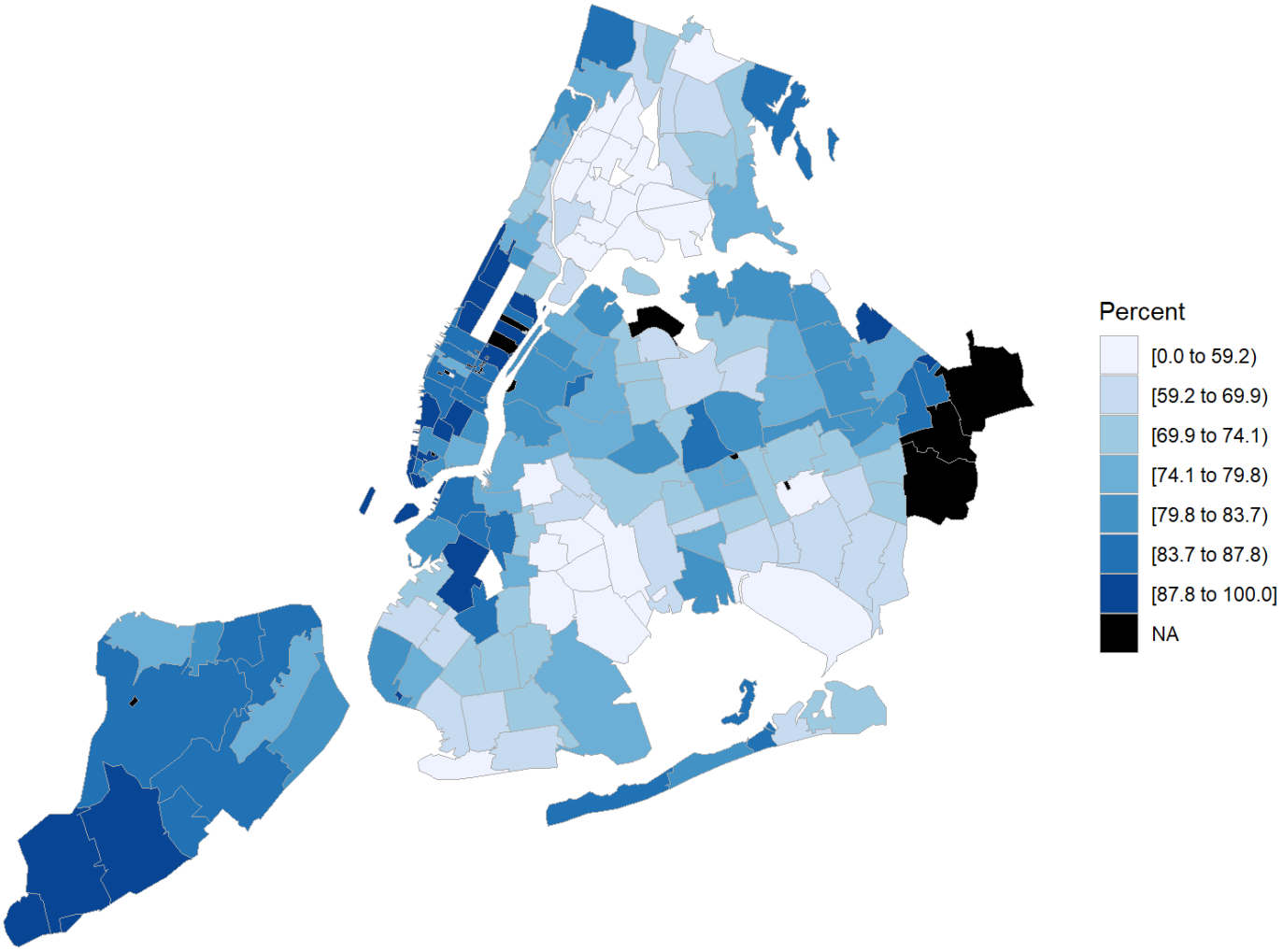
4. Maps

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

The following R codes generate the heat map of spayed or neutered dog rate in NYC. There are some zip code areas are N/A due to no data in those regions, for example zip code 11001. There are also some package mapping issues for mapping certain zip codes in this map, for example zip code 10008. From the plot, first we can find center Bronx has relatively low neutered dogs. Low neutered rates in Queens are observed around JFK airport. Staten Island and Manhattan have overall neutered rates greater than 50% and higher other three boroughs.

```
df <- NYCdog %>% select(zip_code, spayed_or_neutered) %>% mutate(spayed_or_neutered = if_else(sp
ayed_or_neutered == "Yes", 1, 0)) %>% group_by(zip_code) %>% summarise_all(funs(sum(.=='1')/n()*
100)) %>% mutate(zip_code = as.character(zip_code))

names(df) <- c("region","value")
nyc_county_fips = c(36005, 36047, 36061, 36081, 36085)
zip_choropleth(df, county_zoom = nyc_county_fips, title = "Spatial Heat Map of The Percent Spaye
d or Neutered Dogs",legend = "Percent")
```

Spatial Heat Map of The Percent Spayed or Neutered Dogs



5. Time Series

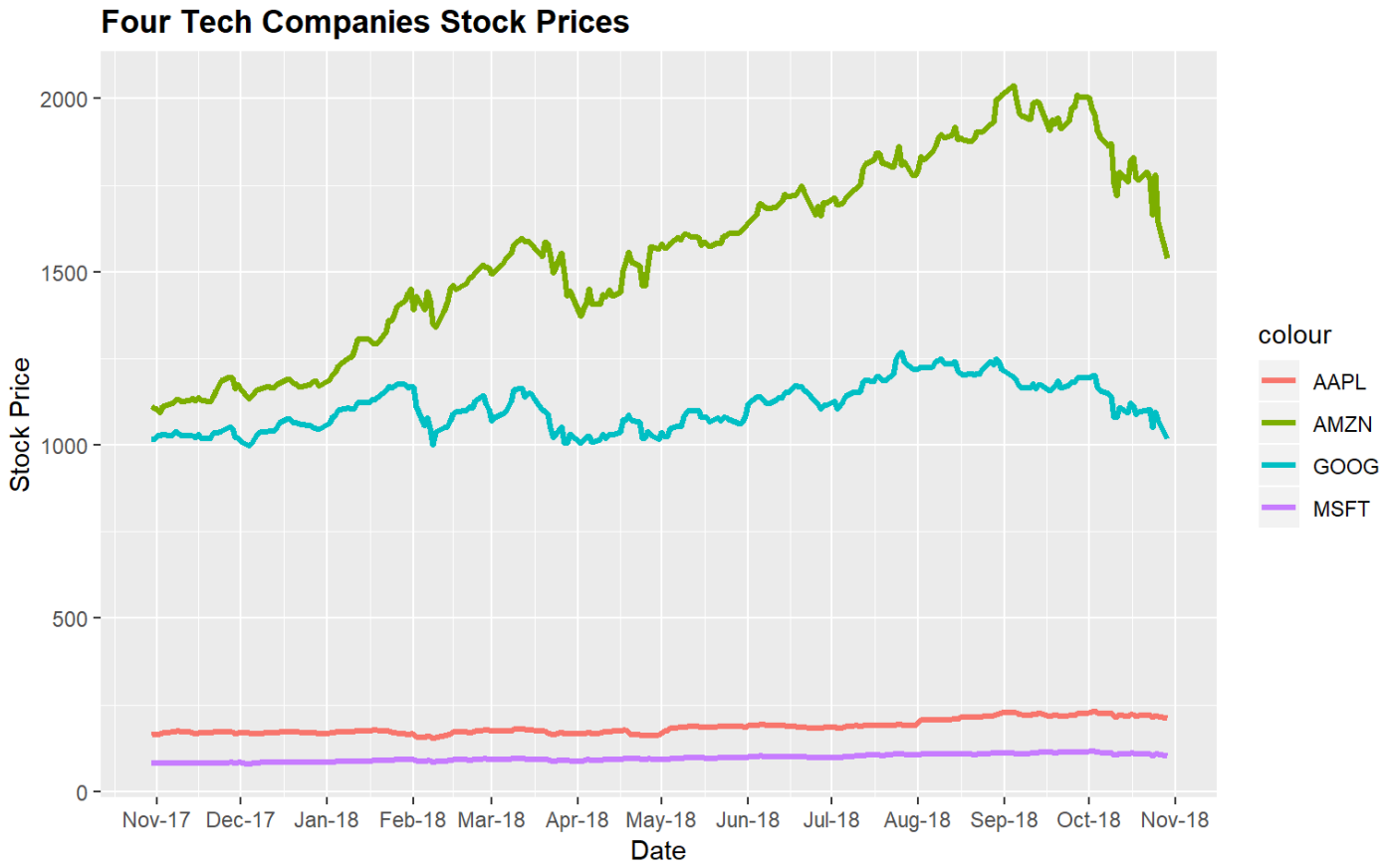
a. Use the `tidyquant` package to collect information on four tech stocks of your choosing. Create a multiple line chart of the closing prices of the four stocks on the same graph, showing each stock in a different color.

The following R codes pull stock data of Apple (AAPL), Amazon (AMZN), Google (GOOG) and Microsoft (MSFT) with 1-year period prior to the plot generating date. Adjusted price data was selected for generating plots. Other types of stock price such as open/close price are also available when pulling data using tq_get function.

```
from <- today() - years(1)
AAPL <- tq_get("AAPL", get = "stock.prices", from = from) %>% select(date, adjusted)
AMZN <-  tq_get("AMZN", get = "stock.prices", from = from) %>% select(date, adjusted)
GOOG <-  tq_get("GOOG", get = "stock.prices", from = from) %>% select(date, adjusted)
MSFT <-  tq_get("MSFT", get = "stock.prices", from = from) %>% select(date, adjusted)
```

The following R codes generate the stock price plot of four different tech companies. In the absolute price scale, we can observe that Amazon has an obvious price increasing trend for the past year. The top price is around $2,000, but recently down to $1,700. Google shows fluctuations and mildly increasing from April to October 2018. The price of the rest of the two companies are relatively stable in the axis scale.

```
ggplot() +
  geom_line(data = AAPL, aes(x=date, y= adjusted, colour = "AAPL"), size = 1.2) +
  geom_line(data = AMZN, aes(x=date, y= adjusted, colour = "AMZN"), size = 1.2) +
  geom_line(data = GOOG, aes(x=date, y= adjusted, colour = "GOOG"), size = 1.2) +
  geom_line(data = MSFT, aes(x=date, y= adjusted, colour = "MSFT"), size = 1.2) +
  labs(x= "Date", y = "Stock Price") +
  ggtitle("Four Tech Companies Stock Prices") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))+
  scale_x_date(date_breaks = "1 month",date_labels = "%b-%y")
```



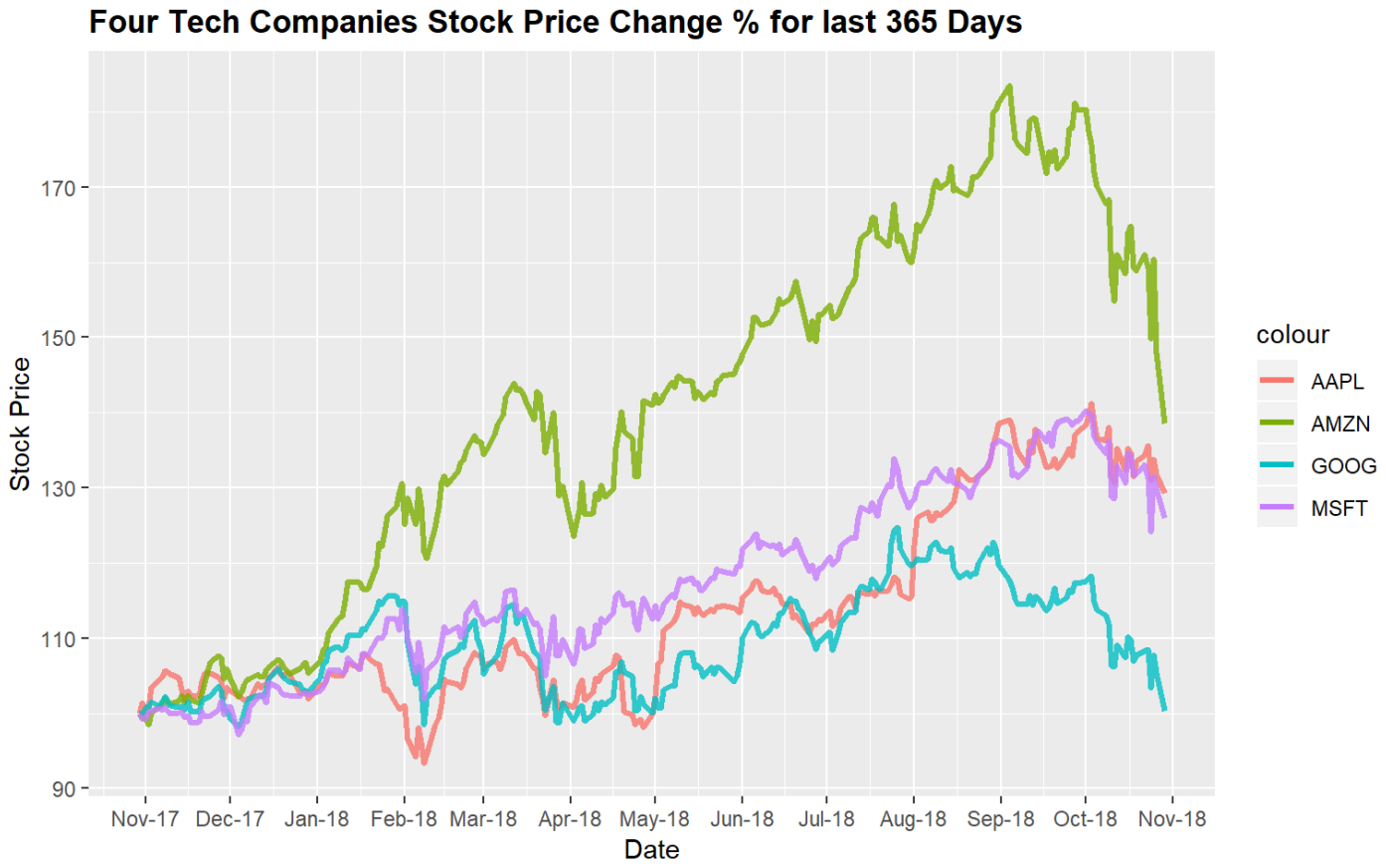
b. Transform the data so each stock begins at 100 and replot. Choose a starting date for which you have data on all of the stocks. Do you learn anything new that wasn't visible in (a)?

The following R codes transform the previous price data based on the first record of each company.

```
AAPL$adjusted = AAPL$adjusted / AAPL$adjusted[1] * 100
AMZN$adjusted = AMZN$adjusted / AMZN$adjusted[1] * 100
GOOG$adjusted = GOOG$adjusted / GOOG$adjusted[1] * 100
MSFT$adjusted = MSFT$adjusted / MSFT$adjusted[1] * 100
```

The following R codes generate the transformed stock price plot. Visually, the plot shows strongly agreement with previous plot of Amazon performance. Amazon price has increased 50-70% from 1 year ago. We can also observe that Apple and Microsoft perform well after transforming the data. In contrast, the price trends are visually very stable in previous plot scale. Both companies' prices have increased around 30-40%. In contrast, Google relatively performed worse than other companies recently. Google only increased 10% from 1 year ago.

```
ggplot() +
  geom_line(data = AAPL, aes(x=date, y= adjusted, colour = "AAPL"), size = 1.2, alpha=0.8) +
  geom_line(data = AMZN, aes(x=date, y= adjusted, colour = "AMZN"), size = 1.2, alpha=0.8) +
  geom_line(data = GOOG, aes(x=date, y= adjusted, colour = "GOOG"), size = 1.2, alpha=0.8) +
  geom_line(data = MSFT, aes(x=date, y= adjusted, colour = "MSFT"), size = 1.2, alpha=0.8) +
  labs(x= "Date", y = "Stock Price") +
  ggtitle("Four Tech Companies Stock Price Change % for last 365 Days") +
  theme(plot.title = element_text(face = "bold")) +
  theme(plot.subtitle = element_text(face = "bold", color = "grey35"))+
  scale_x_date(date_breaks = "1 month",date_labels = "%b-%y")
```



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…)

The audience might be Mayor DeBlasio or officers from NYC Department of Health and Mental Hygiene.

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

Since I came US, I keep hearing news about Pit Bull attacking people. I think it's a good idea that we can track how many Pit Bull in NYC. Based on the dog data, I filter out Pit Bull not spayed or neutered and make a heat map. The reason for filtering out non neutered dog is because a myth that dog might be more clam after neutering. Also I think a responsible owner will neuter their dog for helping prevent uterine infections and breast tumors. Then the Mayor can base on the map to help them to make policy such as owner education and free neuter/spay events in order to minimize the pit bull attacking tragedy. Based on the map, we can see that central Bronx, and border of Queens and Brooklyn highest density with not-neutered or not spayed Pit Bull.

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.

```
df <- NYCdog %>% select(breed, zip_code, spayed_or_neutered) %>% filter(str_detect(breed,"Pit"))
 %>%filter(str_detect(spayed_or_neutered,"No")) %>% group_by(zip_code) %>% summarise(n=n()) %>%
 mutate(zip_code = as.character(zip_code))
names(df) <- c("region","value")
nyc_county_fips = c(36005, 36047, 36061, 36081, 36085)
zip_choropleth(df, county_zoom = nyc_county_fips, title = "Registered Non-spayed or Non-neutered
 Pit Bulls in NYC", legend = "Dog Dounts")
```

Registered Non-spayed or Non-neutered Pit Bulls in NYC