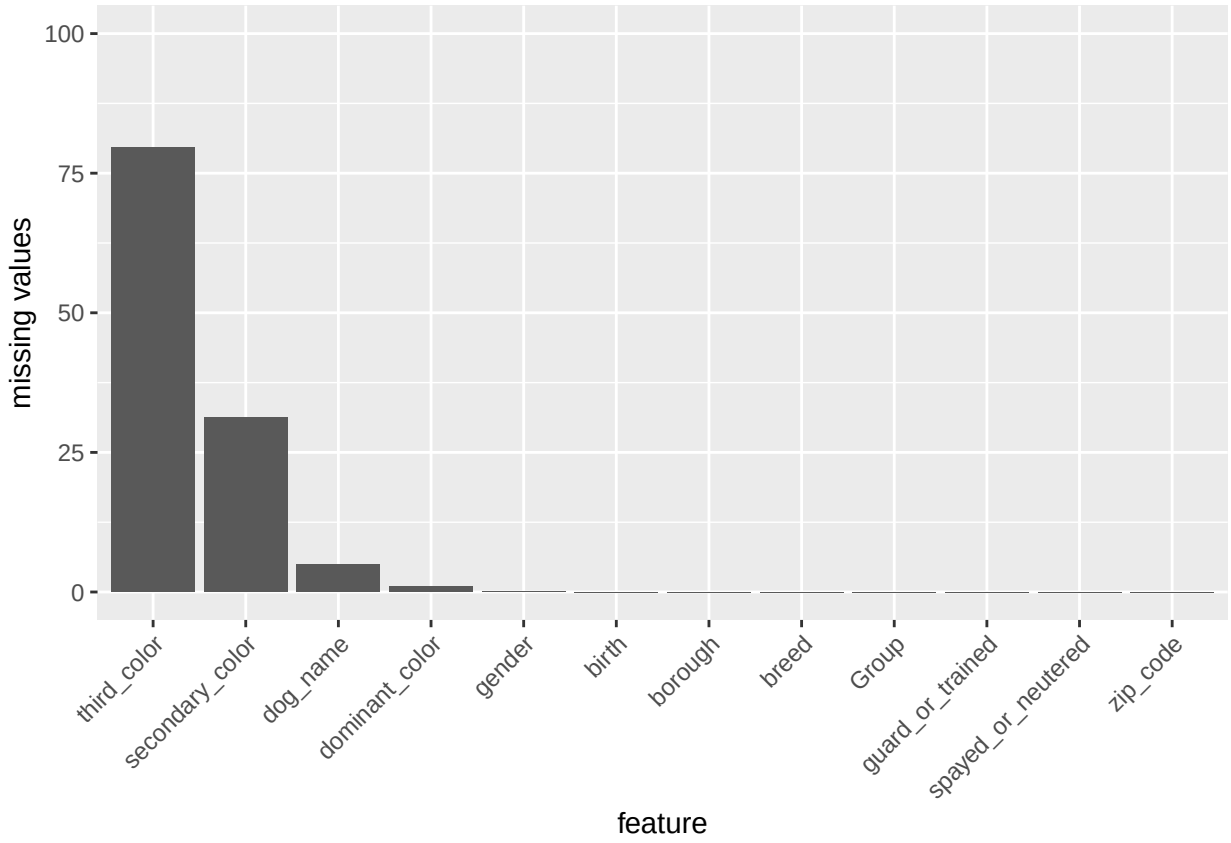
Homework #3

Eric Boxer ecb2198

1. Missing Data

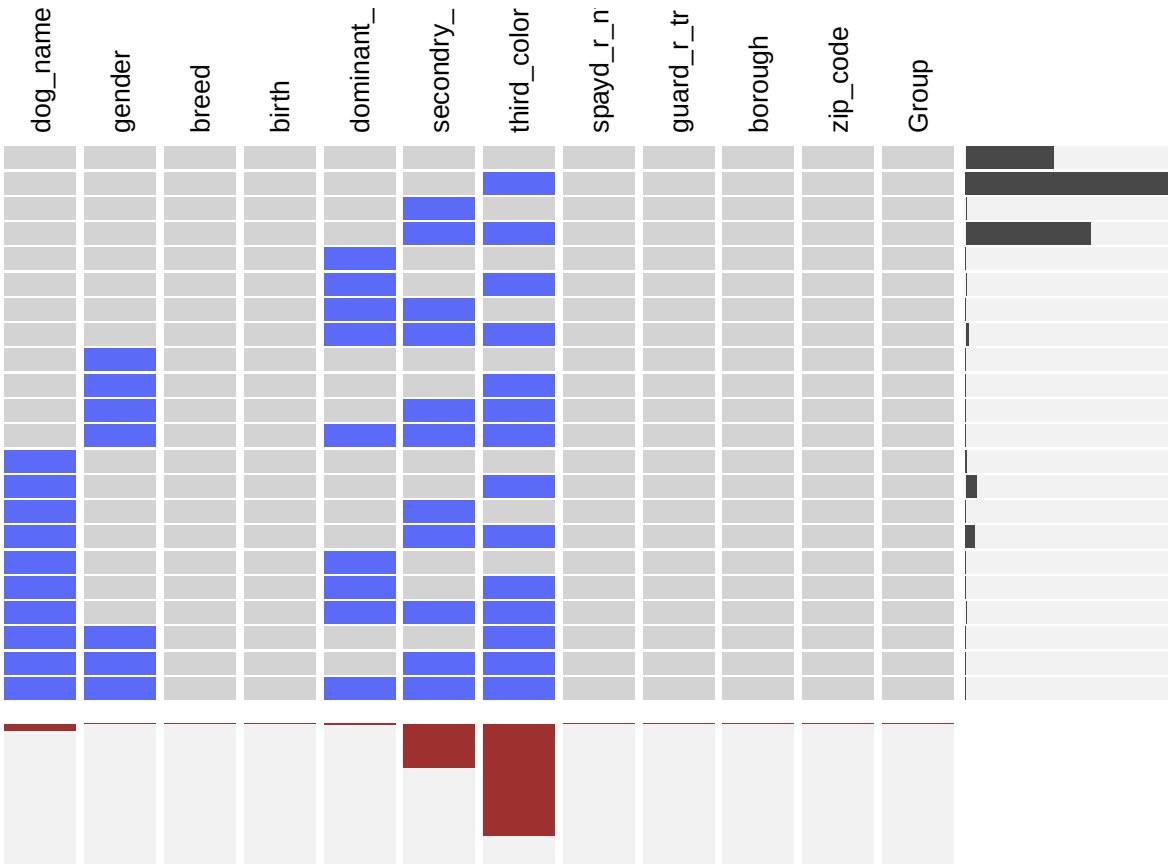
(a) Create a bar chart showing percent missing by variable.

```
NYCdogs <- read_csv("resources/NYCdogs.csv")
NYCdogs[NYCdogs == 'n/a'] <- NA
N <- nrow(NYCdogs)
dfMissing <- colSums(is.na(NYCdogs)) %>%
  sort(decreasing = T) %>%
  as_data_frame() %>%
  mutate(feature = c('third_color', 'secondary_color',
                     'dog_name', 'dominant_color', 'gender',
                     'breed', 'birth', 'spayed_or_neutered',
                     'guard_or_trained', 'borough', 'zip_code',
                     'Group'),
         perc_missing = round(value / N, 4) * 100) #add percent!
dfMissing %>% ggplot(aes(reorder(feature, -perc_missing, 'identity'),
                         perc_missing)) +
  geom_bar(stat = 'identity') +
  xlab('feature') + ylab('missing values') +
  ylim(c(0, 100)) +
  theme(axis.text.x = element_text(angle=45, hjust=1))
```



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

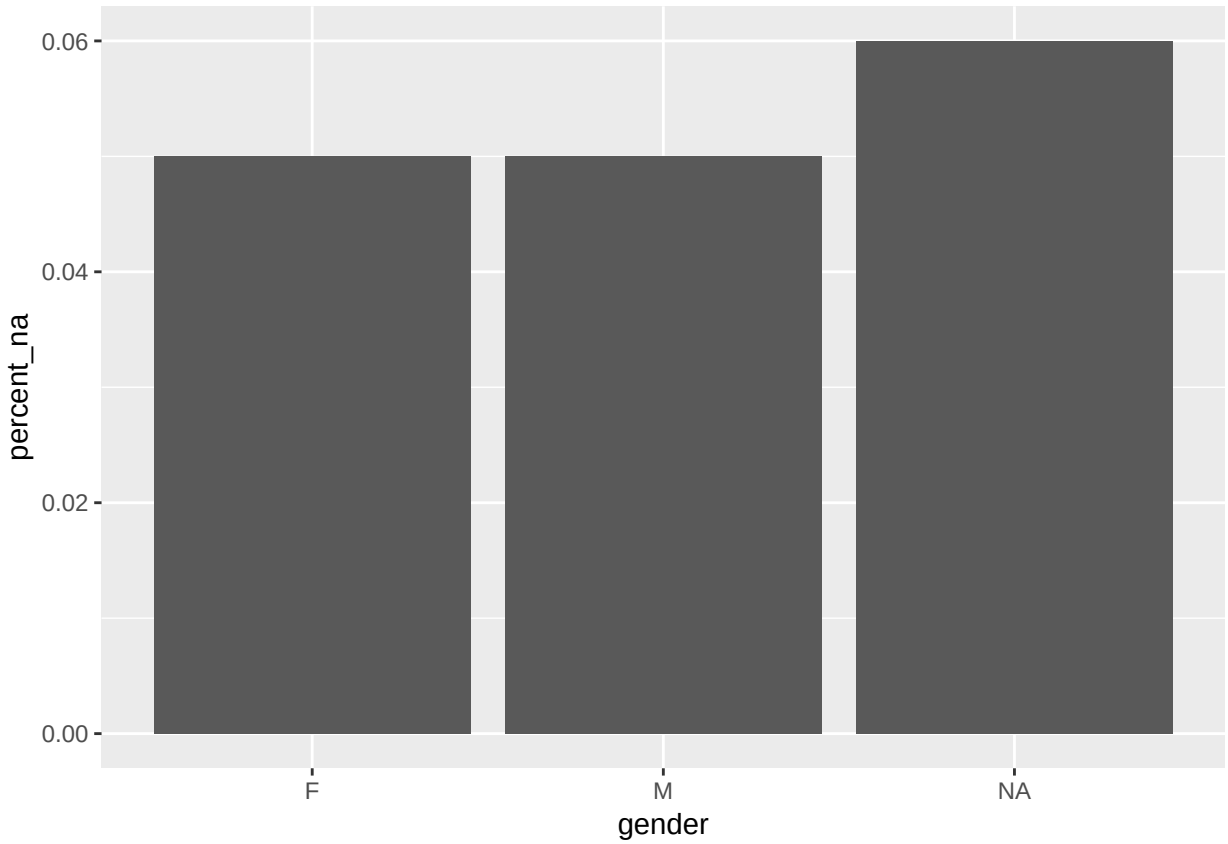
```
NYCdogs %>% extracat::visna()
```



The feature missing the most values is third_color, with more than twice as many missing values as secondary_color, which in turn has about four times as many missing values as dog_name. The most common missing value pattern for observations is to be missing only third_color. Appearing in a bit more than half as many observations is the second most common pattern, secondary_color and third_color. The third most common pattern is no missing values. From the visna chart we can see that more than three-quarters of the observations are missing third_color or secondary_color.
There are some observations which are missing secondary_color and not third_color, which we can tell from the patterns secondary_color and dominant_color + secondary_color. There are also observations with dominant_color missing but with secondary_color or third_color present, so this could be a case of incorrect data entry. We could address this by moving whichever feature is present to primary_color. There could be a logical explanantion for the missing colors. For example, a dog without a 'dominant' color is a patchwork of black and white and has its coloration classed as secondary and tertiary, but without a dominant color.

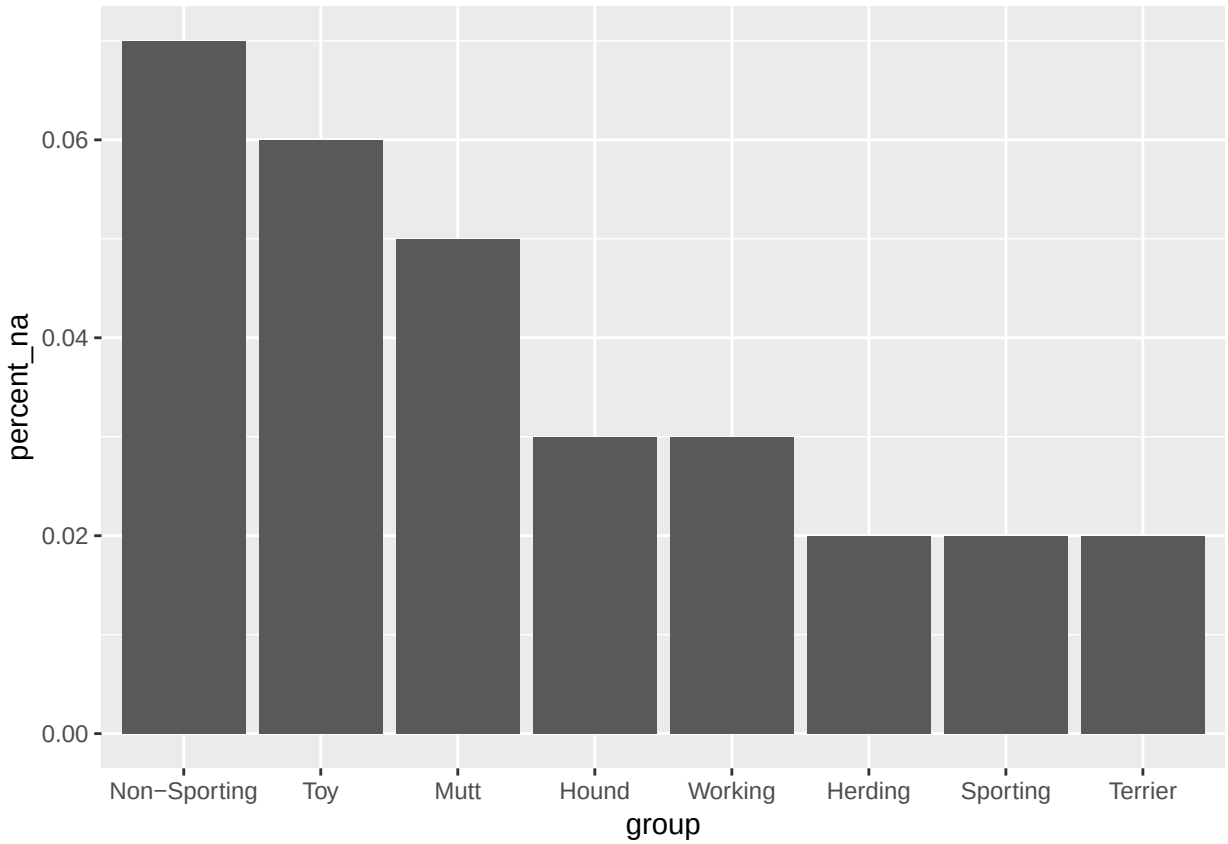
(c) Do `dog_name` missing patterns appear to be associated with the *value* of `gender`, `Group` *or* `borough`?

```
NYCdogs %>% group_by(gender) %>%
  summarize(num_gender = n(),
            num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na / num_gender, 2)) %>%
  arrange(-percent_na) %>%
  ggplot(aes(gender, percent_na)) +
  geom_bar(stat = 'identity')
```



There does not appear to be an association between missing dog_name and gender.

```
NYCdogs %>% group_by(Group) %>%
  summarize(num_group = n(),
            num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na / num_group, 2)) %>%
  arrange(-percent_na) %>%
  ggplot(aes(reorder(Group, -percent_na), percent_na)) +
  geom_bar(stat = 'identity') +
  xlab('group')
```



More non-sporting toy and mutt dogs have missing names. With half as many missing names are the herding, sporting and terrier groups. I worked on the breed feature too, before the announcement pointing to the correct csv.

```
NYCdogs %>% group_by(breed) %>%
  summarize(num_breed = n(),
            num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na / num_breed, 2)) %>%
  arrange(-percent_na) %>% head(10)
```

```
## # A tibble: 10 x 4
##    breed               num_breed num_na percent_na
##    <chr>                   <int>  <int>      <dbl>
##  1 Shiba Inu                 590     93       0.16
##  2 Skye Terrier                7      1       0.14
##  3 Bull Dog, French          872     98       0.11
##  4 Pomeranian               1403    148       0.11
##  5 Silky Terrier             288     31       0.11
##  6 Schnauzer, Standard       164     17       0.1
##  7 Cotton De Tulear          120     11       0.09
##  8 Havanese                  952     87       0.09
##  9 Poodle, Standard         1281    121       0.09
## 10 Puggle                    606     53       0.09
```

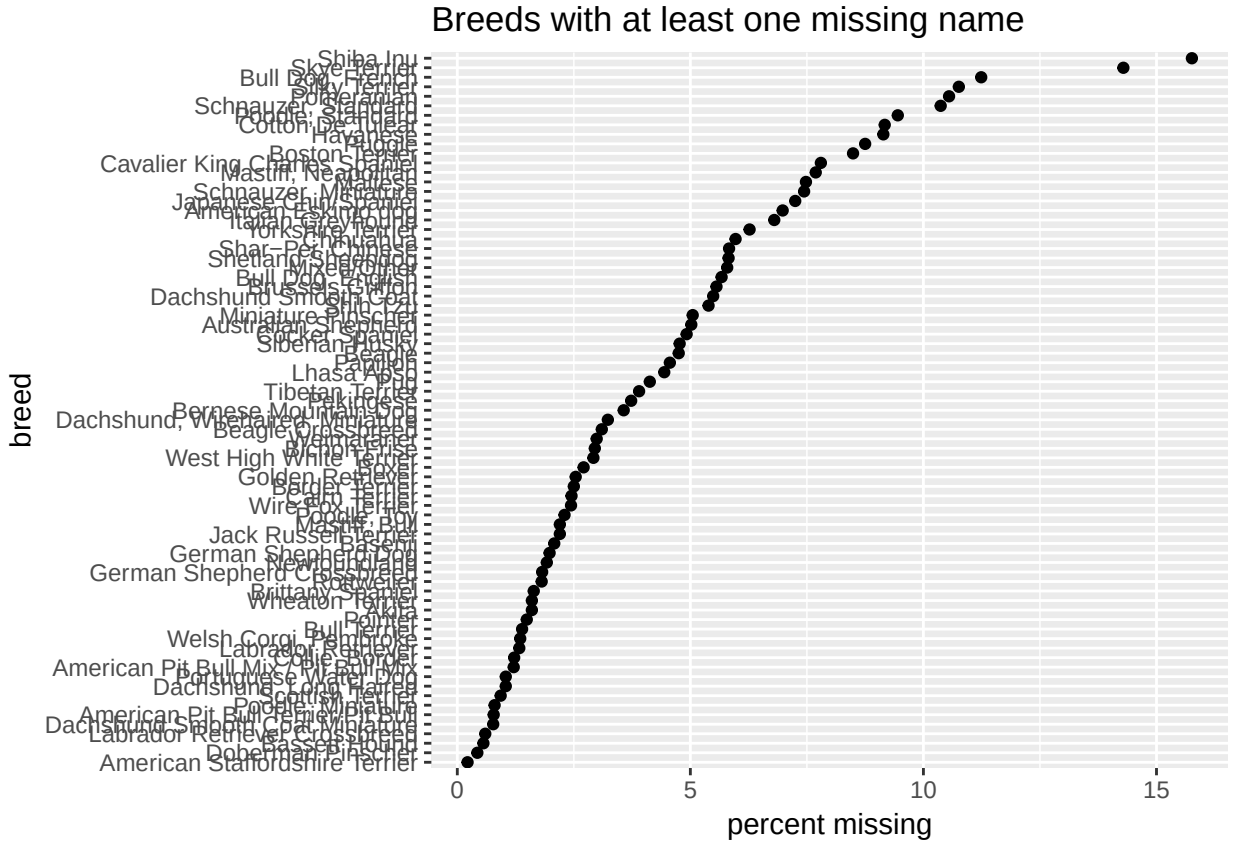
```
NYCdogs %>% group_by(breed) %>%
  summarize(num_breed = n(),
            num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na / num_breed, 2)) %>%
```

```
  arrange(-percent_na) %>%
  filter(percent_na == 0) %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    65
```

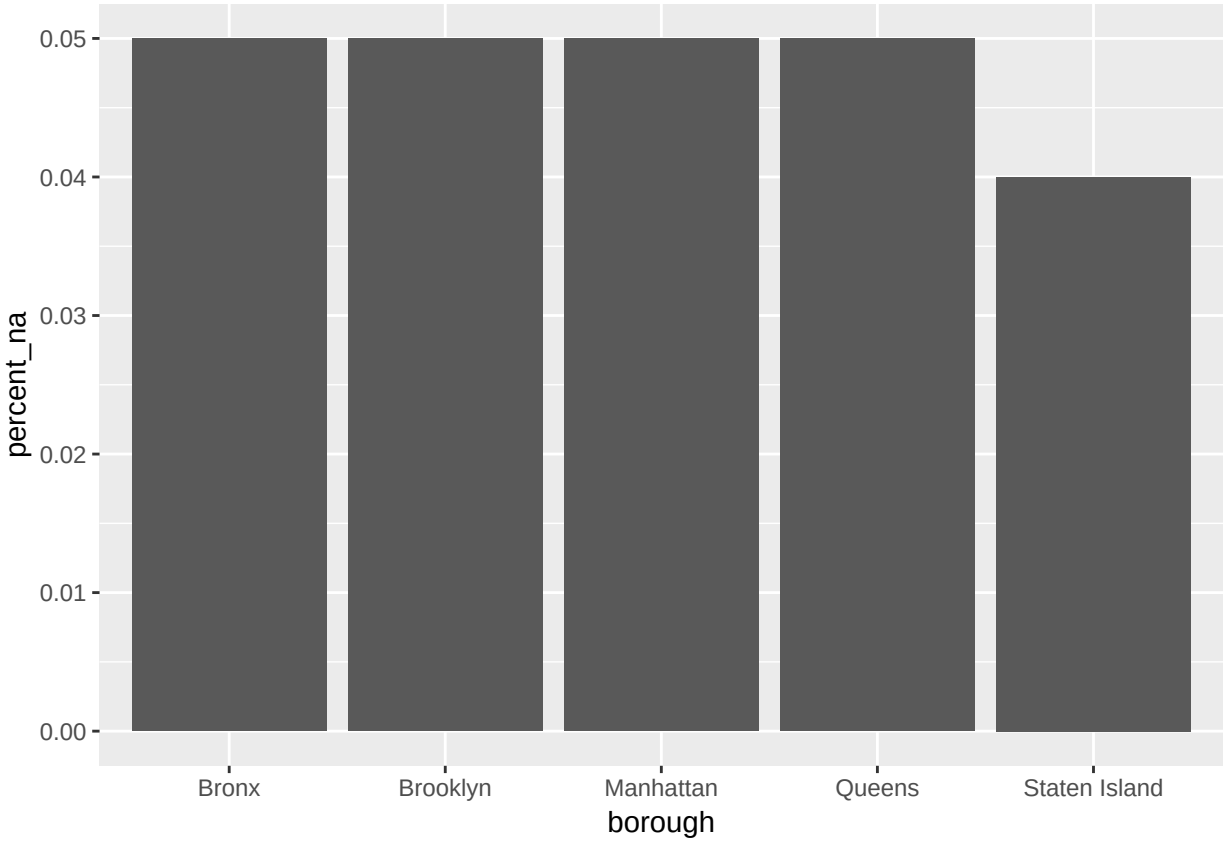
There does appear to be an association between breed and missing dog_name. Shiba Inu, French Bull dogs and Pomeranians are popular dog breeds with more than 10% missing names, while 65 breeds are entirely without missing dog_name values.

```
NYCdogs %>% group_by(breed) %>%
  summarize(num_breed = n(),
            num_na = sum(is.na(dog_name))) %>%
  filter(num_na > 0) %>%
  mutate(percent_na = round(num_na / num_breed, 4) * 100) %>%
  ggplot(aes(reorder(breed, percent_na), percent_na)) +
  geom_point() +
  coord_flip() + xlab('breed') + ylab('percent missing') +
  ggtitle('Breeds with at least one missing name')
```




```
NYCdogs %>% group_by(borough) %>%
  summarize(num_borough = n(),
            num_na = sum(is.na(dog_name))) %>%
  mutate(percent_na = round(num_na / num_borough, 2)) %>%
  arrange(-percent_na) %>%
  ggplot(aes(borough, percent_na)) +
```

```
  geom_bar(stat = 'identity')
```

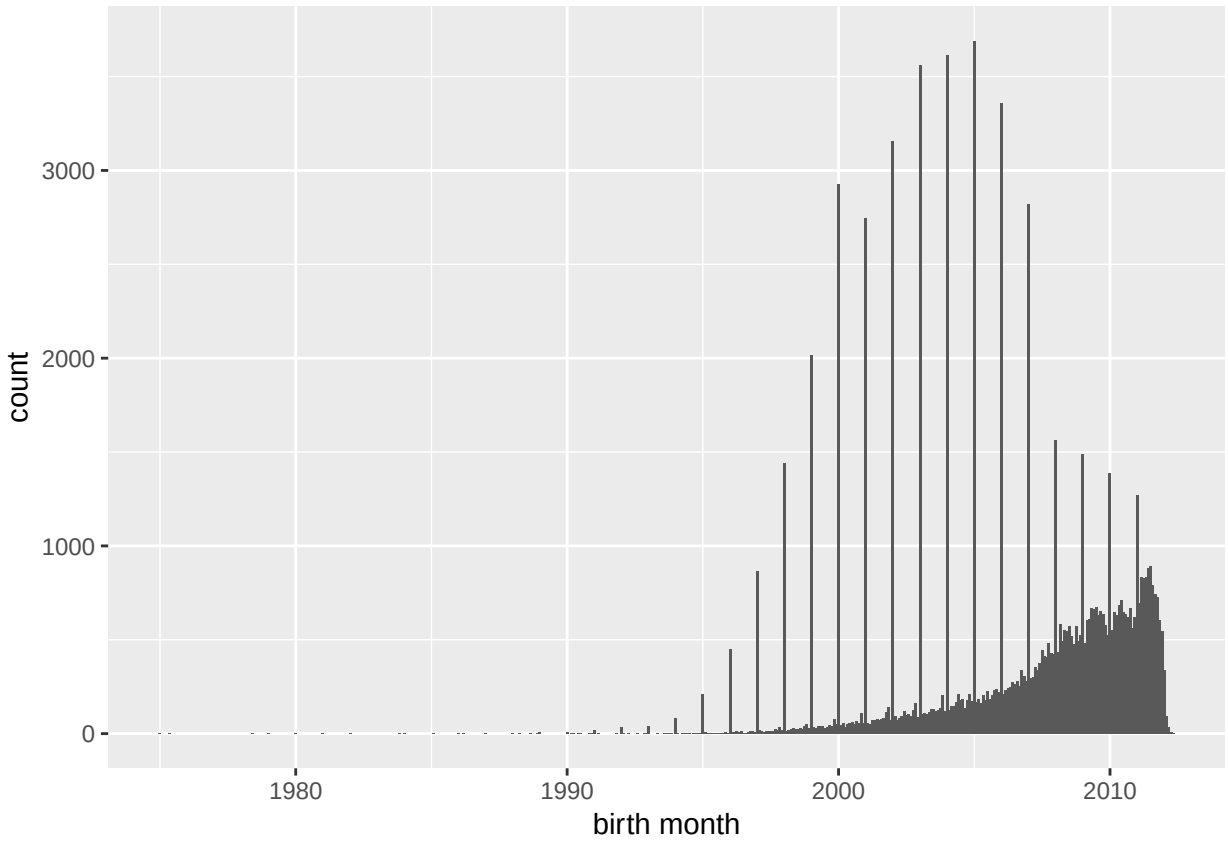


There does not appear to be an association between missing dog_name and borough.

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.

```
DogsofNYC <- read_csv("resources/DogsofNYC.csv")
DogsofNYC[DogsofNYC == 'n/a'] <- NA
DogsofNYC$birth <- DogsofNYC$birth %>%
  lubridate::parse_date_time2(orders = 'my')
DogsofNYC$birth[DogsofNYC$birth > '2018-12-01 UTC'] <- NA
DogsofNYC %>%
  mutate(yearmon = zoo::as.yearmon(birth)) %>%
  group_by(yearmon) %>%
  tally() %>%
  ggplot(aes(yearmon, n)) + geom_bar(stat = 'identity') +
  xlab('birth month') + ylab('count')
```



It looks like most people said that their dogs were born in the same month.

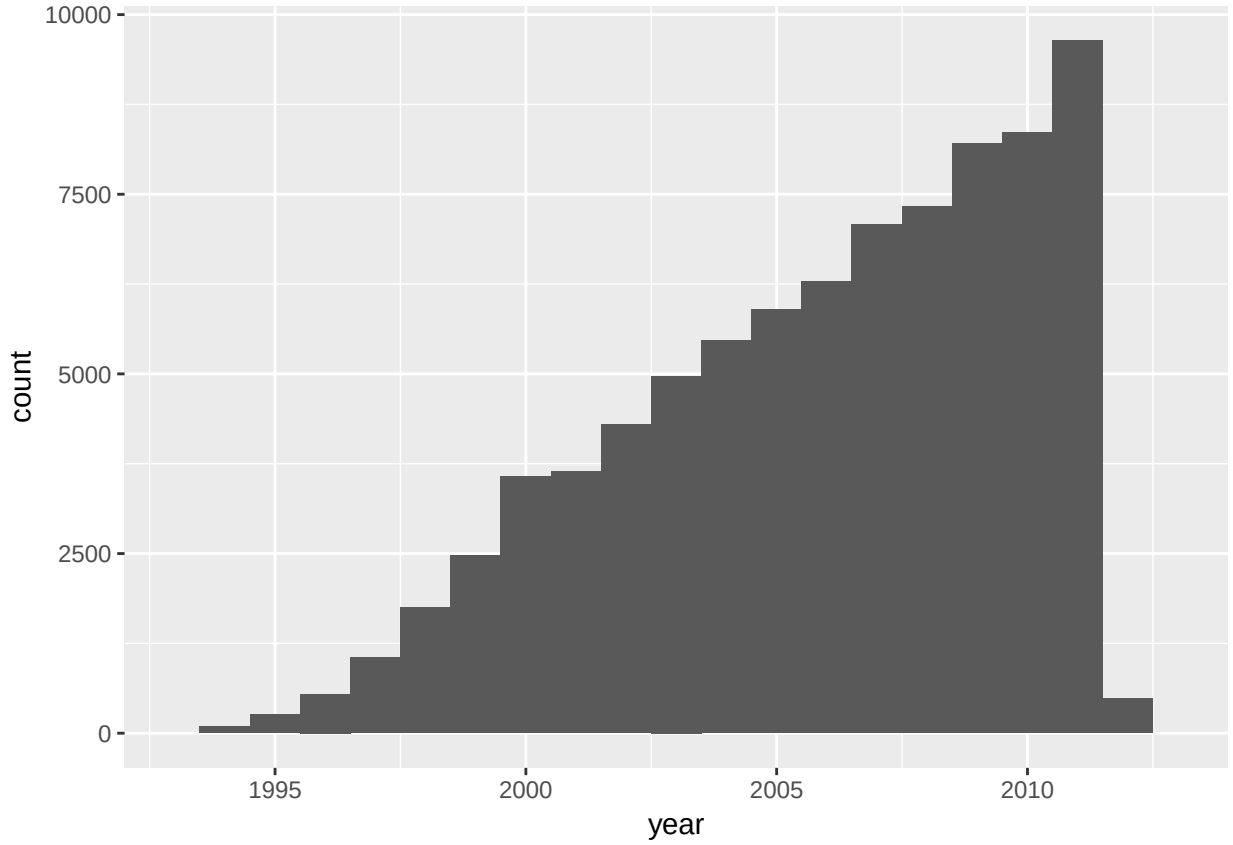
```
DogsofNYC %>%
  mutate(yearmon = zoo::as.yearmon(birth)) %>%
  group_by(yearmon) %>%
  tally() %>%
  arrange(desc(n)) %>% head(10)
```

```
## # A tibble: 10 x 2
##    yearmon           n
##    <S3: yearmon> <int>
##  1 Jan 2005       3690
##  2 Jan 2004       3612
##  3 Jan 2003       3563
##  4 Jan 2006       3359
##  5 Jan 2002       3157
##  6 Jan 2000       2929
##  7 Jan 2007       2818
##  8 Jan 2001       2747
##  9 Jan 1999       2017
## 10 Jan 2008       1564
```

People seem to be writing down that their dogs were born in January, because they don't know the true birth month. It could also be the case that many people adopt dogs in January as a New Year event and consider their date of adoption to be the 'birth date'. However, this explanation would not justify the massive gap in birth dates between January and for example December, when we celebrate Christmas.

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

```r
DogsofNYC %>%
  mutate(year = as.numeric(substr(birth, 0, 4))) %>%
  ggplot(aes(year)) +
  geom_histogram(binwidth = 1) +
  xlim(c(1993, 2013))
```



3. Mosaic plots

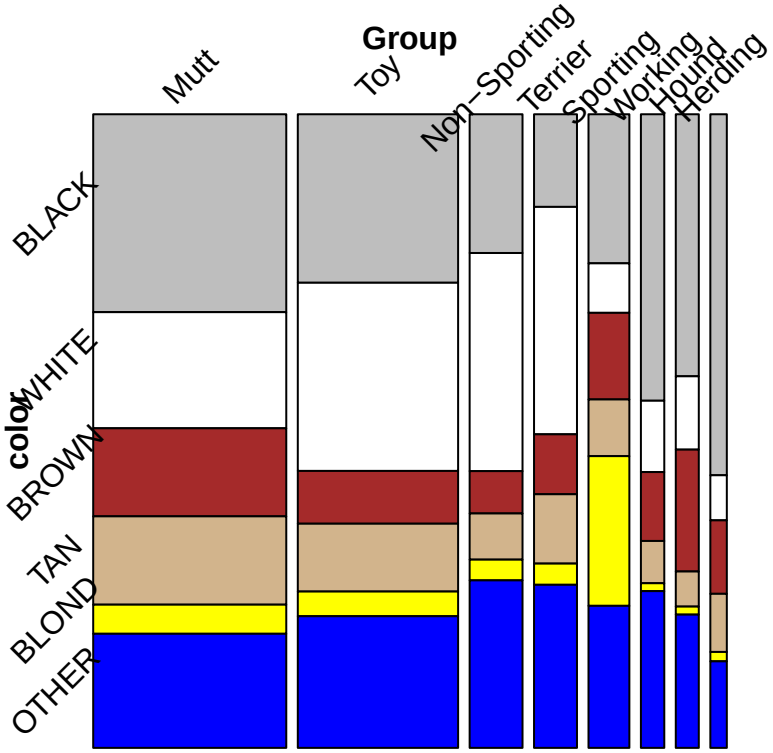
(a) Create a mosaic plot to see if `dominant_color` depends on `Group`. Use only the top 5 dominant colors; group the rest into an "OTHER" category. The last split should be the dependent variable and it should be horizontal. Sort each variable by frequency, with the exception of "OTHER", which should 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.

```r
NYCdogs %>%
  group_by(dominant_color) %>%
  count() %>% arrange(desc(n)) %>% head(5) %>%
  select(dominant_color)
```

```
## # A tibble: 5 x 1
## # Groups:   dominant_color [5]
##   dominant_color
##   <chr>
## 1 BLACK
## 2 WHITE
## 3 BROWN
```

```
## 4 TAN
## 5 BLOND
```

```
top5 <- c('BLACK', 'WHITE', 'BROWN', 'TAN', 'BLOND')
color <- function (x) {
  ifelse(x %in% top5,
         x,
         'OTHER')
}
NYCdogs$color <- lapply(NYCdogs$dominant_color, color)
NYCdogs <- transform(NYCdogs, color=unlist(color))
dfcounts <- NYCdogs %>% group_by(color, Group) %>%
  count()
order <- c(top5, 'OTHER')
orderGroup <- c('Mutt', 'Toy', 'Non-Sporting',
                'Terrier', 'Sporting', 'Working',
                'Hound', 'Herding')
NYCdogs$color <- factor(NYCdogs$color,
                        levels=order)
NYCdogs$Group <- factor(NYCdogs$Group,
                        levels=orderGroup)
color1 = c('grey', 'white', 'brown', 'tan', 'yellow', 'blue')
vcd::mosaic(color~Group,
            direction=c('v', 'h'),
            NYCdogs,
            gp = gpar(fill=color1),
            rot_labels=c(45, 45),
            sort=order)
```

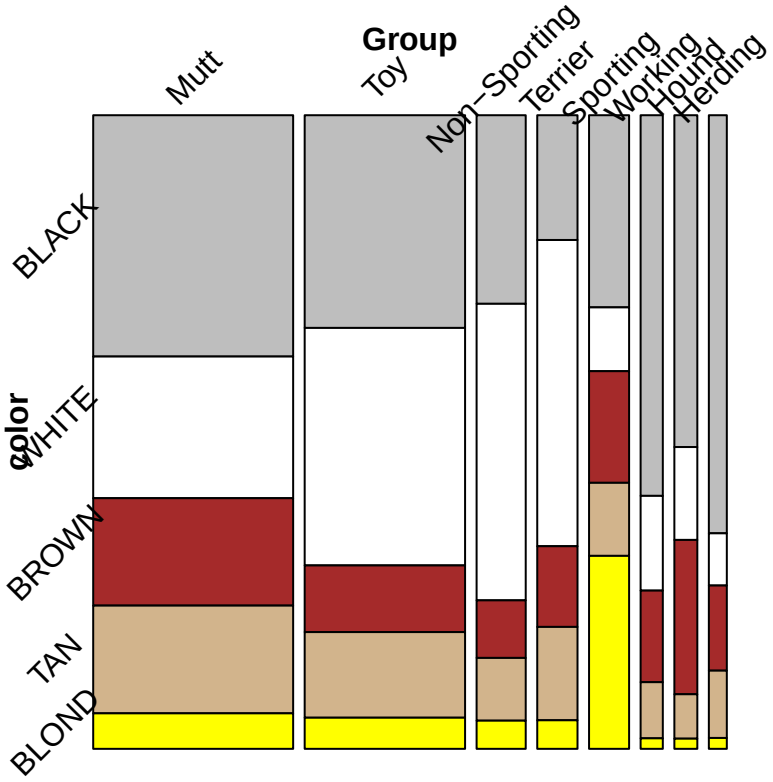


There is a huge increase in the proportion of blond dogs in the sporting group. The black, white and brown colors seem to have the strongest relationship to the group feature. Black is the predominant color for herding

dogs, and popular for hound and working dogs, but much less common among terrier, non-sporting and sporting dogs. Something similar is true for brown dogs which are most common among hounds. White dogs more common among terriers, non-sporting and toy groups, and uncommon among sporting, herding and working. I would hypothesize that white colored dogs are more desirable as pets than they are as companions for hunters or workers. This would be more clearcut if I understood the distinction between the groups more clearly, because I believed that terriers were bred as fox-hunting dogs, but maybe the white color has been bred back into them and they are no longer considered to be sporting. The mutt group is one of the most evenly distributed among the colors, which makes sense since if color is dependent on species the interspecies category should represent a blending and dilution of the dependencies.

(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?

```
NYCdogs1 <- NYCdogs %>%
  filter(color != 'OTHER')
color2 = c('grey', 'white', 'brown', 'tan', 'yellow')
NYCdogs1$color <- factor(NYCdogs1$color,
                         levels=top5)
vcd::mosaic(color~Group,
            direction=c('v', 'h'),
            NYCdogs1,
            gp = gpar(fill=color2),
            rot_labels=c(45, 45),
            sort=top5)
```



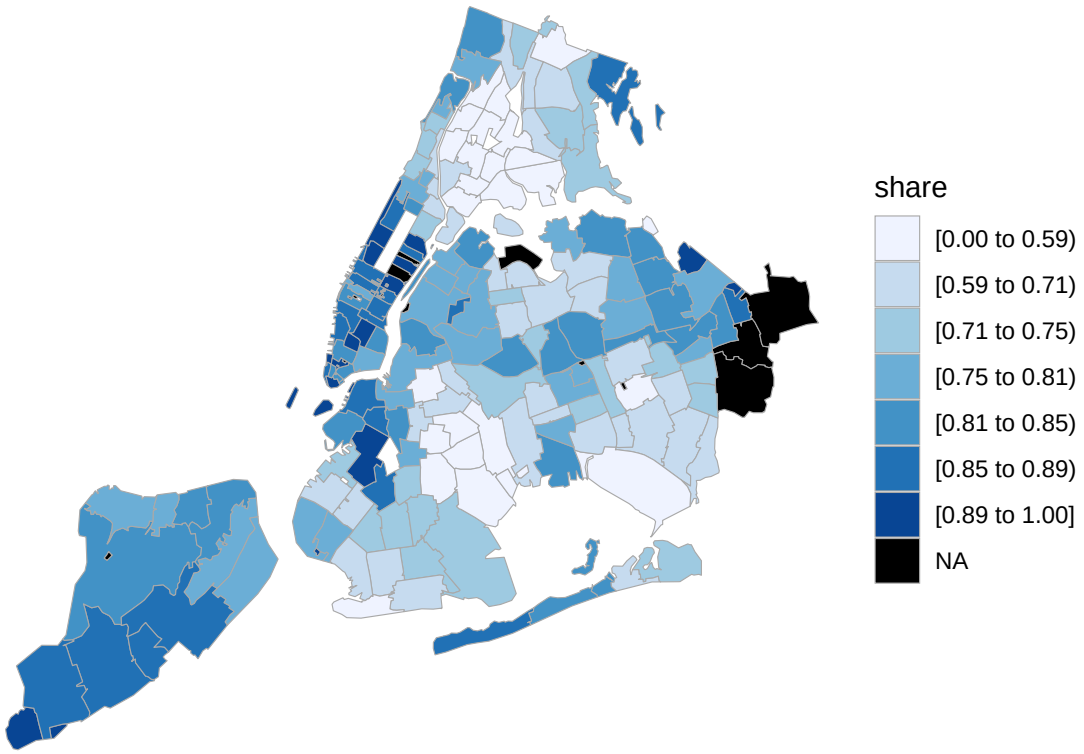
The results do not change. A rough heuristic could be to exclude the 'OTHER' category if it is not greater than one-quarter of the total data. Maybe you could formalize this as exclude if count(OTHER) < (1 / n) * observations, where n is the number of values excluded from OTHER. The idea would be that if OTHER is small enough whether or not you include it would not make a big difference to the exploratory analysis. It would also be important to take into account the number of categories *included* in OTHER, if it is just

one then exclusion could seem arbitrary and deceptive. It is probably good practice to try visualizing with and without the OTHER category and looking at the composition of the data before grouping. In our case, OTHER is composed of 14 colors so I feel okay about excluding it on that criterion, too.

4. Maps

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

```
zips <- NYCdogs %>% group_by(zip_code) %>%
  count() %>% arrange(zip_code)
NYCdogs$spaybin <- ifelse(NYCdogs$spayed_or_neutered == 'Yes', 1, 0)
dfspay <- NYCdogs %>% group_by(zip_code) %>%
  summarise(n = n(),
            num_spay = sum(spaybin)) %>%
  mutate(percent_spay = round(num_spay / n, 2))
colnames(dfspay)[1] <- 'region'
colnames(dfspay)[4] <- 'value'
dfzip <- dfspay %>% select('region', 'value')
dfzip$region <- dfzip$region %>% as.character()
nyc_fips = c(36085,36005, 36047, 36061, 36081)
data("zip.regions")
nyc_zips<-data.frame(county.fips.numeric=nyc_fips)%>%inner_join(zip.regions)%>%select(region)%>%t
zip_choropleth(dfzip, zip_zoom=nyc_zips,
               title='Share of Spaying/Neutering',
               legend='share')
```



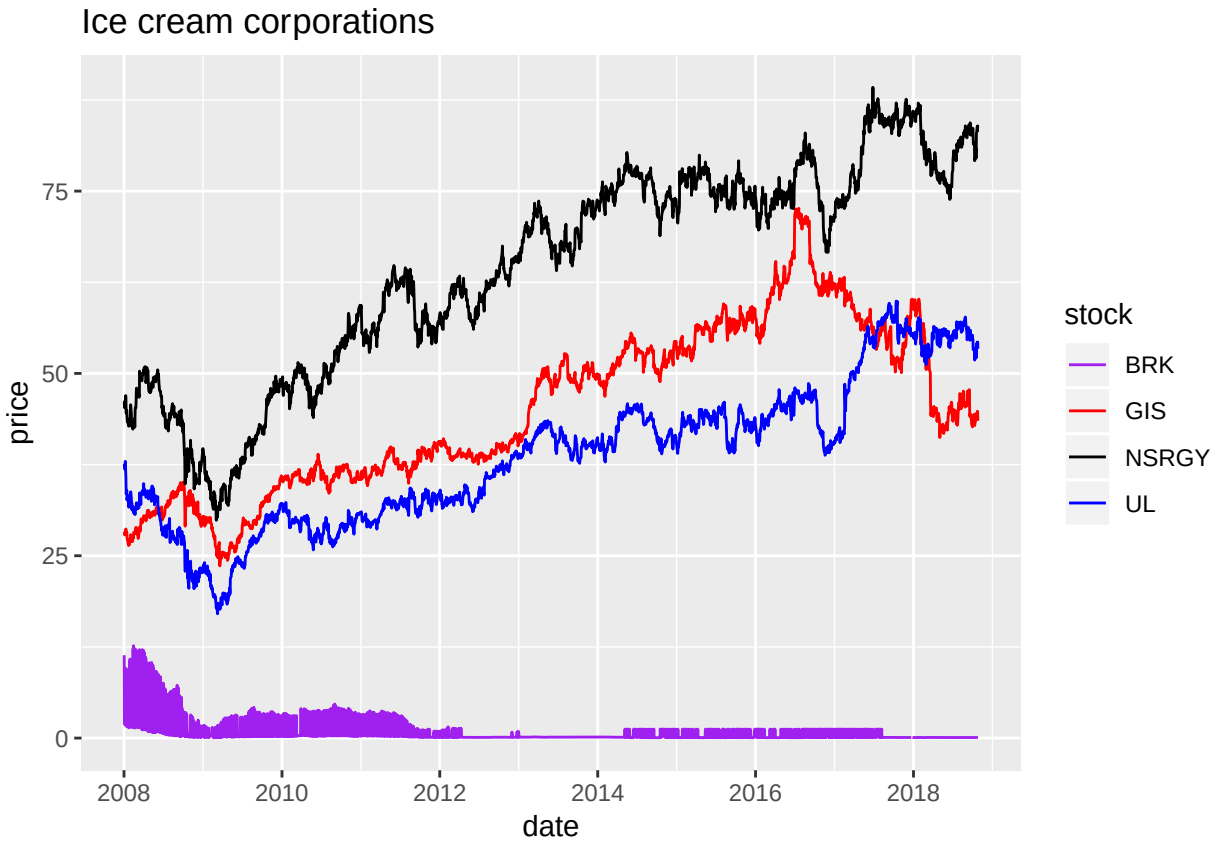
The heatmap accords with my hypothesis that more wealthy neighborhoods, downtown Brooklyn and downtown and midtown Manhattan, correspond to high rates of spaying/neutering. Unexpectedly, Staten

Island seems like the second most spayed borough, after Manhattan. Less urban areas such as the BRonx bordering on Westchester, Eastern Queens and the Rockaways also have a higher proportion of spaying than their more urban neighbors. Greater rates of spaying seems to be a function which increases with wealth and with 'suburb-ness'.

5. Time Series

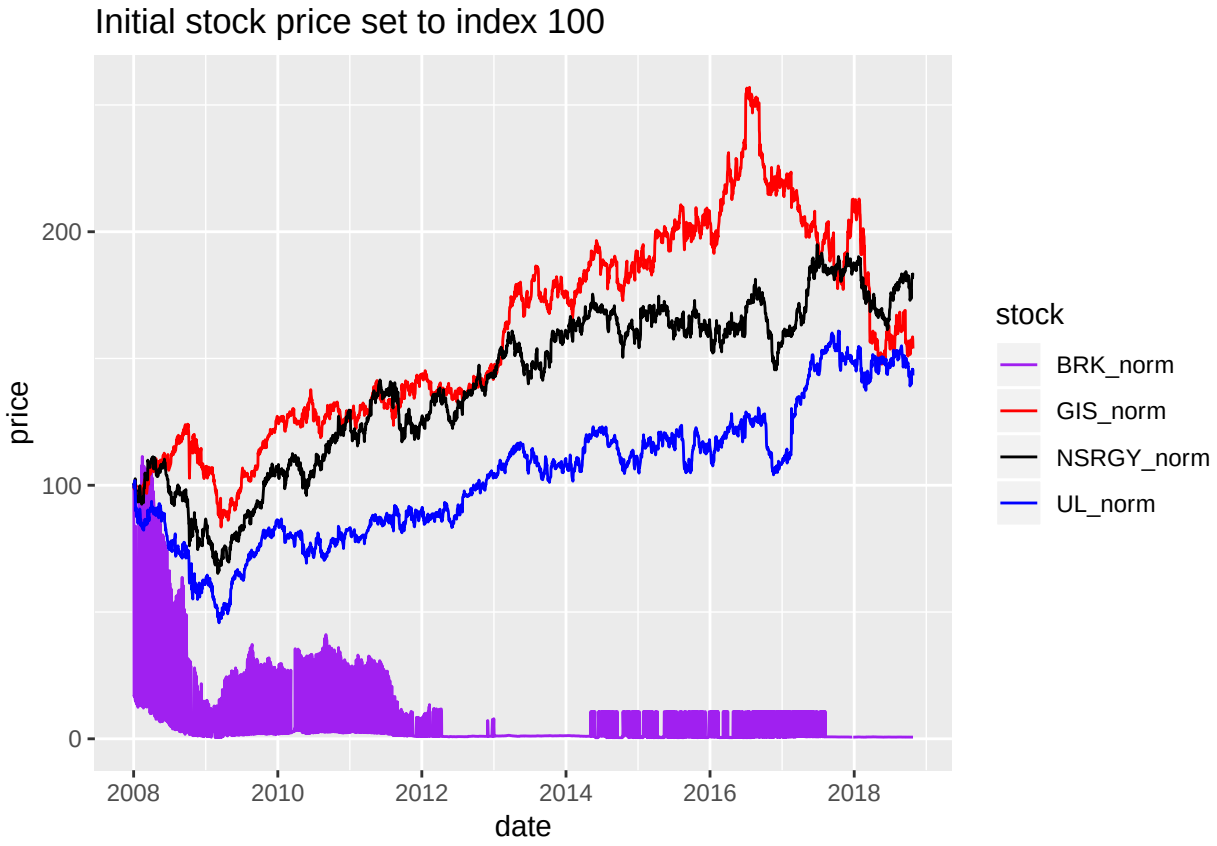
(a) Use the `tidyquant` package to collect information on four tech stocks of your choosing. Create a multiple line chart of the closing prices of the four stocks on the same graph, showing each stock in a different color.

```
ul <- tq_get('UL', get='stock.prices') #Unilever
gis <- tq_get('GIS', get='stock.prices') #General Mills
ns <- tq_get('NSRGY', get='stock.prices') #Nestle
brk <- tq_get('BRK', get='stock.prices') #Berkshire Hathaway
cols <- c('UL' = 'blue', 'GIS' = 'red',
          'NSRGY' = 'black', 'BRK' = 'purple')
df_ul <- ul %>% select(date, close)
colnames(df_ul)[2] <- 'UL'
df_gis <- gis %>% select(close)
colnames(df_gis)[1] <- 'GIS'
df_ns <- ns %>% select(close)
colnames(df_ns)[1] <- 'NSRGY'
df_brk <- brk %>% select(close)
colnames(df_brk)[1] <- 'BRK'
df_tot <- data.frame(df_ul, df_gis, df_ns, df_brk)
df_tot <- df_tot %>% gather(stock, price, -date)
df_tot %>% ggplot(aes(x=date, y=price, color=stock)) +
  geom_line() +
  scale_color_manual(values=cols) +
  ggtitle('Ice cream corporations')
```



(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)?

```
df_ul <- df_ul %>%
  mutate(UL_norm = UL / df_ul$UL[1] * 100) %>%
  select(date, UL_norm)
df_gis <- df_gis %>%
  mutate(GIS_norm = GIS / df_gis$GIS[1] * 100) %>%
  select(GIS_norm)
df_ns <- df_ns %>%
  mutate(NSRGY_norm = NSRGY / df_ns$NSRGY[1] * 100) %>%
  select(NSRGY_norm)
df_brk <- df_brk %>%
  mutate(BRK_norm = BRK / df_brk$BRK[1] * 100) %>%
  select(BRK_norm)
df_tot <- data.frame(df_ul, df_gis, df_ns, df_brk)
df_tot <- df_tot %>% gather(stock, price, -date)
cols <- c('UL_norm' = 'blue', 'GIS_norm' = 'red',
          'NSRGY_norm' = 'black', 'BRK_norm' = 'purple')
df_tot %>% ggplot(aes(x=date, y=price, color=stock)) +
  geom_line() +
  scale_color_manual(values=cols) +
  ggtitle('Initial stock price set to index 100')
```



All of the stocks start on 2008-01-02. With the normalized plot Berkshire Hathaway is still the underperformer of the bunch and Unilever lags behind the top two stocks. However, the difference between General Mills and Nestle has narrowed, and General Mills actually groes faster from 2013 to 2017. We can also see that the three top companies have gathered at an index of about 150 to 180 since 2018, and are closer to one anotherthan they have been since 2010.

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. . .)
A gathering of the NYS Landlord Association

(b) What is the main point you hope someone will take away from the graph?
The proportion of male dogs is lowest in the most populous areas of NYC. This corresponds to the idea that female dogs are better behaved and overall are a more appropriate choice for apartment living. In this *article* it is found that neutering male dogs increases the likelihood of behavioral issues. Landlords may want to consider informing their tenants of potential downsides to keeping particular dogs in cities and the importance of giving any pet enough time outdoors.

(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.

```
NYCdogs$male[NYCdogs$gender == 'M'] <- 1
NYCdogs$male[NYCdogs$gender == 'F'] <- 0
NYCdogs$male[is.na(NYCdogs$gender)] <- 0
NYCdogs$male <- NYCdogs$male %>% as.numeric()
dfmale <- NYCdogs %>% group_by(zip_code) %>%
  summarise(n = n(),
            num_male = sum(male)) %>%
  mutate(percent_male = round(num_male / n, 4))
colnames(dfmale)[1] <- 'region'
colnames(dfmale)[4] <- 'value'
dfzipmale <- dfmale %>% select('region', 'value')
dfzipmale$region <- dfzipmale$region %>% as.character()
nyc_fips = c(36085,36005, 36047, 36061, 36081)
data("zip.regions")
nyc_zips<-data.frame(county.fips.numeric=nyc_fips)%>%inner_join(zip.regions)%>%select(region)%>%t
dfzipmale$valuecat[dfzipmale$value < .425] <- 'Less than .425'
dfzipmale$valuecat[dfzipmale$value >= .425 & dfzipmale$value < .45] <- '[.425, .45)'
dfzipmale$valuecat[dfzipmale$value >= .45 & dfzipmale$value < .475] <- '[.45, .475)'
dfzipmale$valuecat[dfzipmale$value >= .475 & dfzipmale$value < .5] <- '[.475, .5)'
dfzipmale$valuecat[dfzipmale$value >= .5 & dfzipmale$value < .525] <- '[.5, .525)'
dfzipmale$valuecat[dfzipmale$value >= .525 & dfzipmale$value < .55] <- '[.525, .55)'
dfzipmale$valuecat[dfzipmale$value >= .55 & dfzipmale$value < .575] <- '[.55, .575)'
dfzipmale$valuecat[dfzipmale$value >= .575 & dfzipmale$value < .6] <- '[.575, .6)'
dfzipmale$valuecat[dfzipmale$value >= .6] <- 'Greater than .6'
dfzipM <- dfzipmale %>% select(region, valuecat)
colnames(dfzipM)[2] <- 'value'
order <- c('Less than .425',
           '[.425, .45)', '[.45, .475)',
           '[.475, .5)', '[.5, .525)',
           '[.525, .55)', '[.55, .575)',
           '[.575, .6)', 'Greater than .6')
dfzipM$value <- factor(dfzipM$value,
                       levels=order)
choro <- zip_choropleth(dfzipM, zip_zoom=nyc_zips,
              title='Proportion of Male Dogs',
              legend='Proportion',
              num_colors = 9)
choro <- ZipChoropleth$new(dfzipM)
choro$title <- 'Proportion of Male Dogs by Zip Code'
choro$set_zoom_zip(state_zoom=NULL,
                   zip_zoom=nyc_zips,
                   county_zoom=NULL,
                   msa_zoom=NULL)
scale <- c('#ffffe5', '#fff7bc',
           '#fee391', '#fec44f',
           '#fe9929', '#ec7014',
           '#cc4c02', '#8c2d04')
choro$ggplot_scale = scale_fill_manual(name='Proportion Male',
                                       values=scale)
choro$render()
```

Proportion of Male Dogs by Zip Code