Group 4 Writeup

Reid Jumper, Harsh Nagarkar, and Micheal Davis

12/4/2019

Analysis of Public Coffee Chains

```r
knitr::opts_chunk$set(echo = TRUE)
library(webshot)
library(tidyverse)
```

```
## -- Attaching packages ----------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts -------------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(socviz)
library(usmap)
library(cowplot)
```

```
##
## ********************************************************

## Note: As of version 1.0.0, cowplot does not change the

##   default ggplot2 theme anymore. To recover the previous

##   behavior, execute:
##   theme_set(theme_cowplot())

## ********************************************************
```

```r
library(maps)
```

```
##
## Attaching package: 'maps'

## The following object is masked from 'package:purrr':
##
##     map
```

```r
library(mapproj)
library(RColorBrewer)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:cowplot':
##
##     stamp

## The following object is masked from 'package:base':
##
##     date
```

```r
library(tinytex)
library(knitr)
library(htmltools)
library(leaflet)
library(zoo)
```

```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
#webshot::install_phantomjs
```

Where are the companies, Dunkin' Donuts and Starbucks, competing?

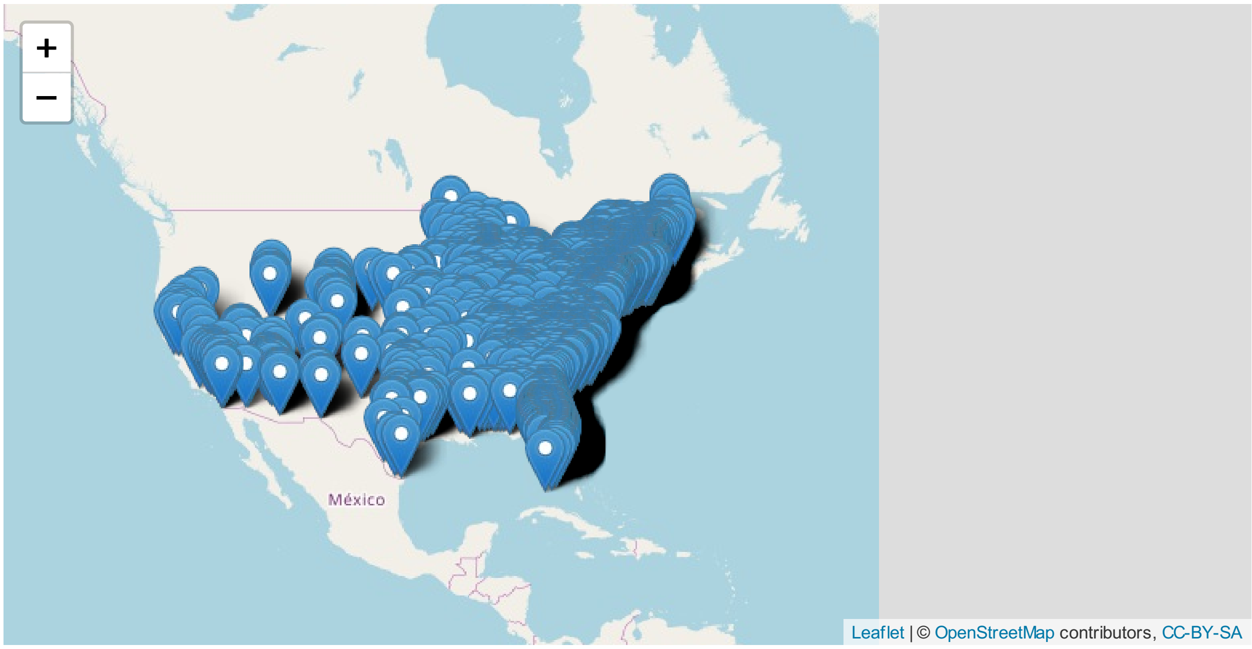
```r
sbuxloc <- read_csv("data/sbuxlocations.csv")
dnknloc <- read_csv("data/dnknlocations.csv", col_names = FALSE)
```

```r
dnknloc <- dnknloc %>%
  rename(Longitude = X1,
         Latitude = X2,
         Specs = X3,
         Location = X4)
sbuxloc <- sbuxloc %>%
  filter(Country %in% c("US"))

dnknloc <- dnknloc %>%
  mutate(store = "Dunkin")
sbuxloc <- sbuxloc %>%
  mutate(store = "Starbucks")
```

```r
dnknloc <- dnknloc %>%
  separate(Specs, c("Specs", "State"), sep = ",")

dnknloc <- dnknloc %>%
  mutate(State = dnknloc %>%
           pull(State) %>%
           str_replace_all("\\ ", ""))

dnknLabel <- sprintf("<b>Dunkin'</b><br>%s", dnknloc$State) %>%
  lapply(htmltools::HTML)
sbuxLabel <- sprintf("<b>Starbucks</b><br>%s", sbuxloc$`State/Province`) %>%
  lapply(htmltools::HTML)
```

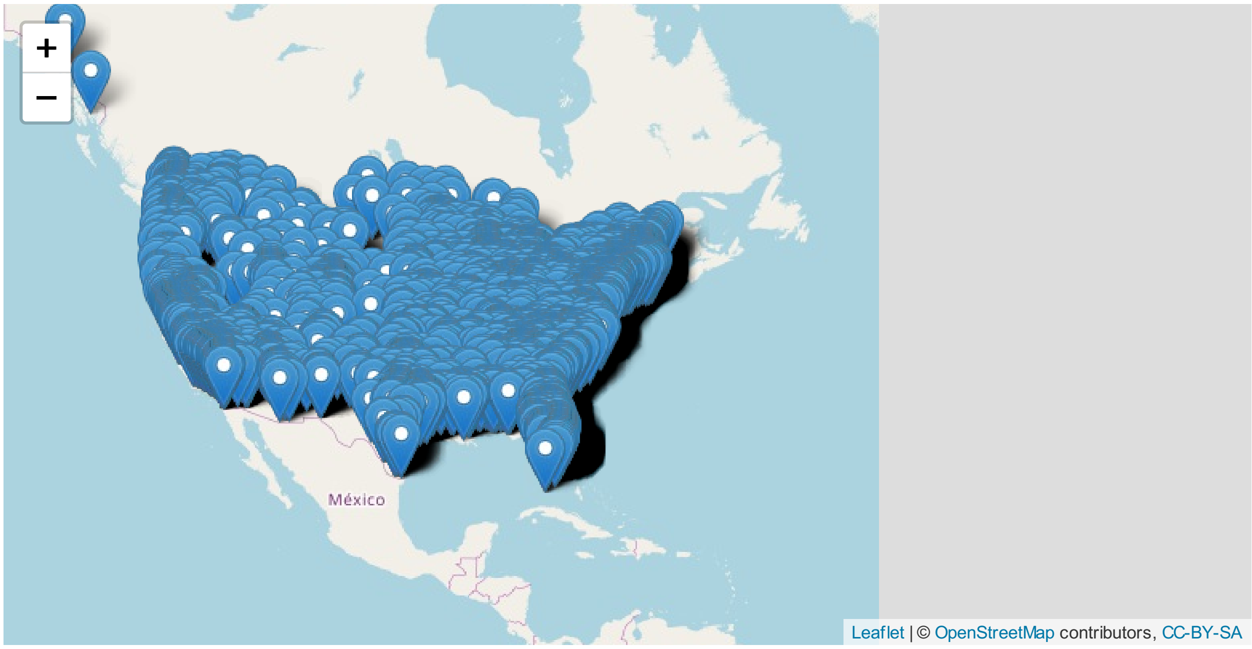
First, it is important to look at where these restaurants are located in the US. # Dunkin’ Leaflet

```r
dnknloc %>%
  leaflet(width = "100%", options = leafletOptions(zoomSnap = 0.1)) %>%
  setView(lng = -100, lat = 40, zoom = 3) %>%
  addTiles() %>%
  addMarkers(~Longitude, ~Latitude, popup = dnknLabel, label = dnknLabel)
```



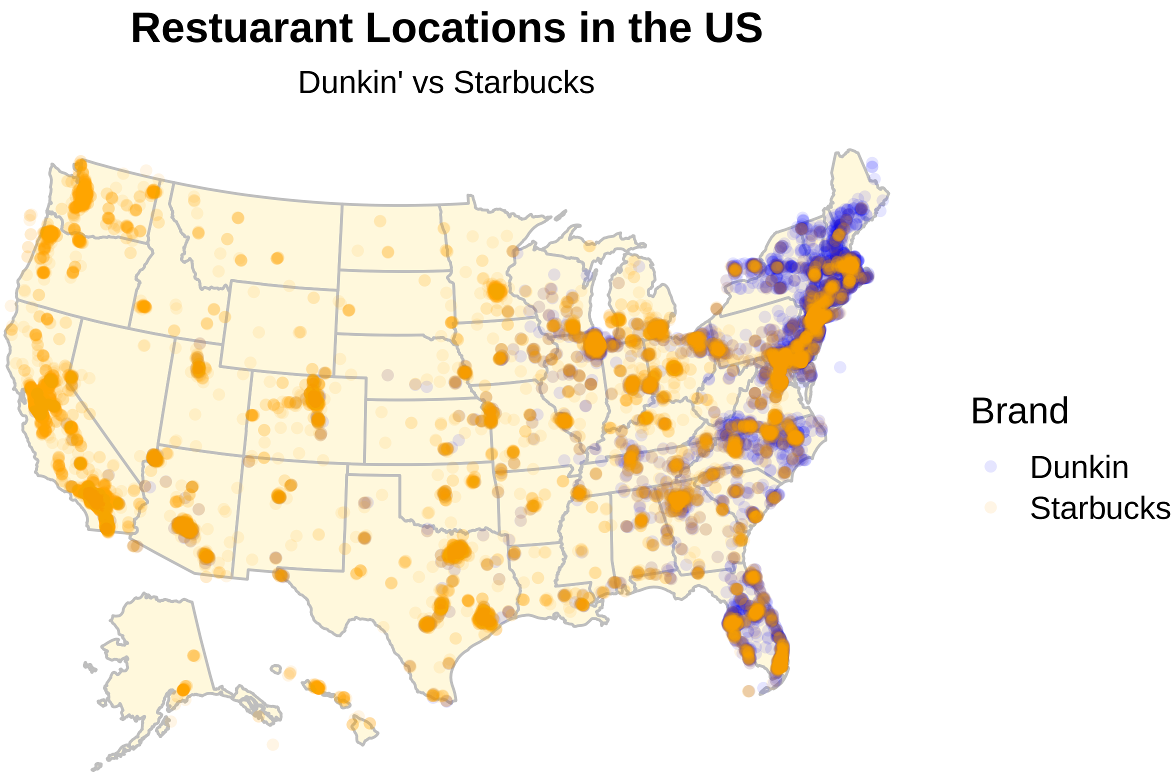
Starbucks Leaflet

```
sbuxloc %>%
  leaflet(width = "100%", options = leafletOptions(zoomSnap = 0.1)) %>%
  setView(lng = -100, lat = 40, zoom = 3) %>%
  addTiles() %>%
  addMarkers(~Longitude, ~Latitude, popup = sbuxLabel, label = sbuxLabel)
```



Locations Graphs

```
ggplot() +
  geom_polygon(data = us_map(regions = "states"), mapping = aes(x = x, y = y, group = group), color = "
  geom_point(data = modified_values, mapping = aes(x = long, y = lat, color = Store), alpha = 0.1) +
  coord_equal() +
  theme_map() +
  scale_color_manual(values = c("blue", "orange")) +
  labs(title = "Restuarant Locations in the US",
       subtitle = "Dunkin' vs Starbucks") +
  guides(color = guide_legend(title = "Brand")) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```

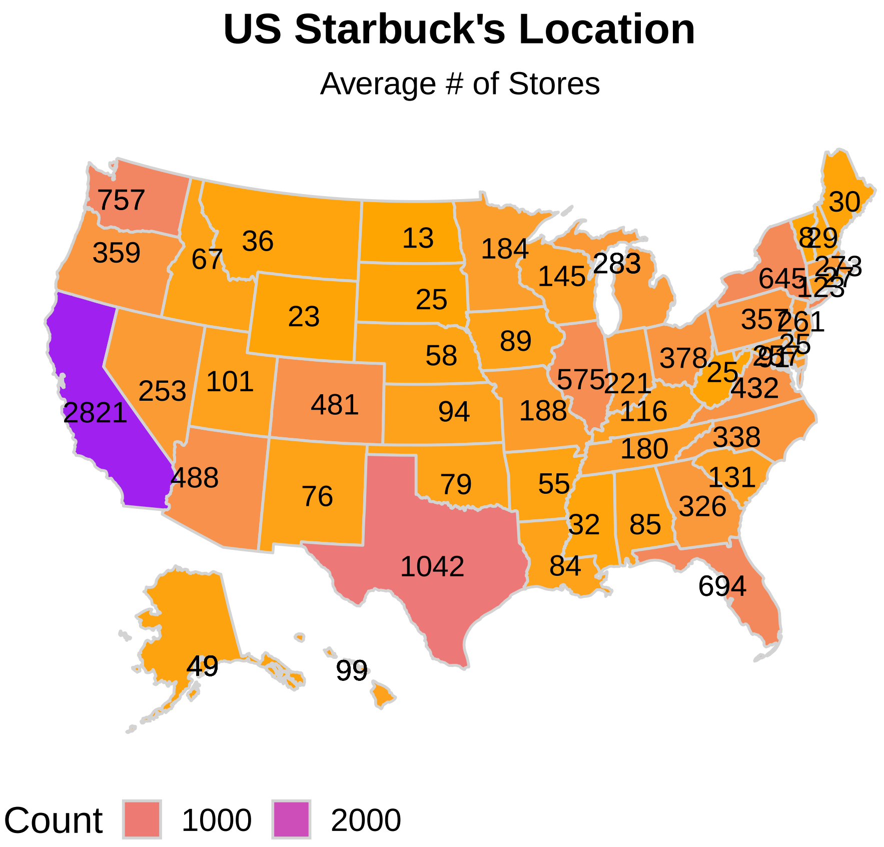


Which states have the highest number of stores for Dunkin Donuts and Starbucks?

```
#gather data from csv files
starbuck_local<- read_csv("data/directory.csv")
starbuck_zip<-read_csv("data/ZIP-COUNTY-FIPS-2018.csv")

#getting states dataset
us_states<-map_data("state")

#chaging dataset zip to numeric
starbuck_zip<-starbuck_zip %>%
  mutate(ZIP=as.numeric(ZIP)) %>%
  distinct(CITY, .keep_all = TRUE)

#filtering by country and renaming
starbuck_local <- starbuck_local %>%
  filter(Country == "US") %>%
  rename(
    CITY = City,
    STATE = `State/Province`
  )

#joining datasets by city
```

```
data<-left_join(starbuck_local,starbuck_zip,by=c("CITY","STATE"))

#grouping by state and removing na values
data<- data %>%
  group_by(STATE) %>%
  count() %>%
  na.omit()

#renaming states to match join function
us_states<-us_map("states") %>%
  rename(STATE=abbr)

#left joining and renaming variables
data<-left_join(us_states,data,by='STATE') %>%
  rename(count = n)



temp_data <- data %>%
  select(STATE, count) %>%
  group_by(STATE) %>%
  summarise(count = mean(count))

coord <- us_states %>%
  select(x, y, STATE) %>%
  group_by(STATE) %>%
  summarise(
    x_avg = mean(x),
    y_avg = mean(y))
```

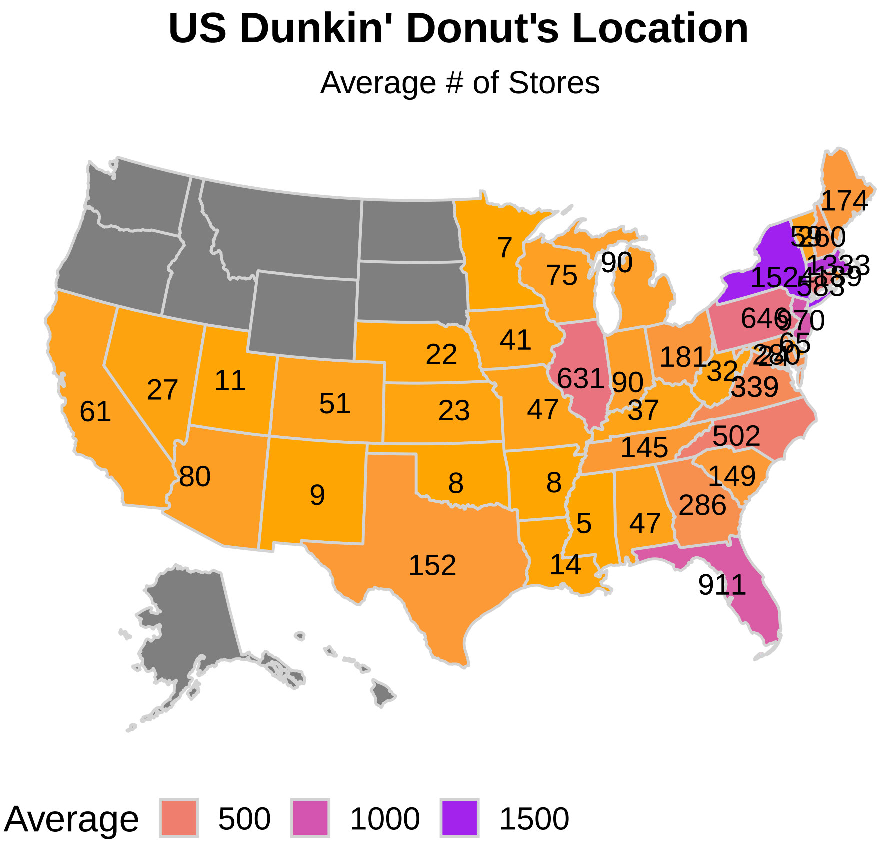
```
new_data <- merge(temp_data, coord, by = "STATE" )

final_data <- merge(new_data, us_states, by = "STATE" )

#plot data for locations
centroid <- aggregate(data=final_data, cbind(x_avg, y_avg) ~ count + group, FUN=mean)
data %>%
  ggplot() +
  geom_polygon(mapping = aes(x=x,y=y,group=group,fill=count), color = "lightgray", size = 0.5) +
  coord_equal()+
  theme_map()+
  theme(panel.background = element_blank()) + ggtitle("US Starbuck's Location", subtitle = "Average # o
  theme(legend.position="bottom", plot.title = element_text(hjust = 0.5),plot.subtitle =  element_text(
```



```
df<- read_csv("data/DD-US.csv",col_names = FALSE)
zip<-read_csv("data/ZIP-COUNTY-FIPS-2018.csv")
county_map <- county_map
us_states<-map_data("state")

zip<-zip %>%
  mutate(ZIP=as.numeric(ZIP))

df <- df %>%
  mutate(ZIP = sapply(strsplit(df$X4, split='|', fixed=TRUE), function(x) (x[2])))

df <- df %>%
  mutate(DATA = sapply(strsplit(df$ZIP, split=c(','), fixed=TRUE), function(x) (x[2])))

df <- df %>%
  mutate(ZIP = sapply(strsplit(df$DATA, split=c(' '), fixed=TRUE), function(x) (x[2])))
```

```
## Warning: NAs introduced by coercion
```

```
#plot data for locations
centroid <- aggregate(data=final_data, cbind(x_avg, y_avg) ~ n + group, FUN=mean)

data%>%
  ggplot() +
  geom_polygon(mapping = aes(x=x,y=y,group=group,fill=n),color = "lightgray", size = 0.5) +
  coord_equal()+
  theme_map()+
  theme(panel.background = element_blank()) + ggtitle("US Dunkin' Donut's Location", subtitle = "Averag
  scale_fill_gradient(low = "orange", high = "purple", guide = guide_legend()) + geom_text(data = centr
```

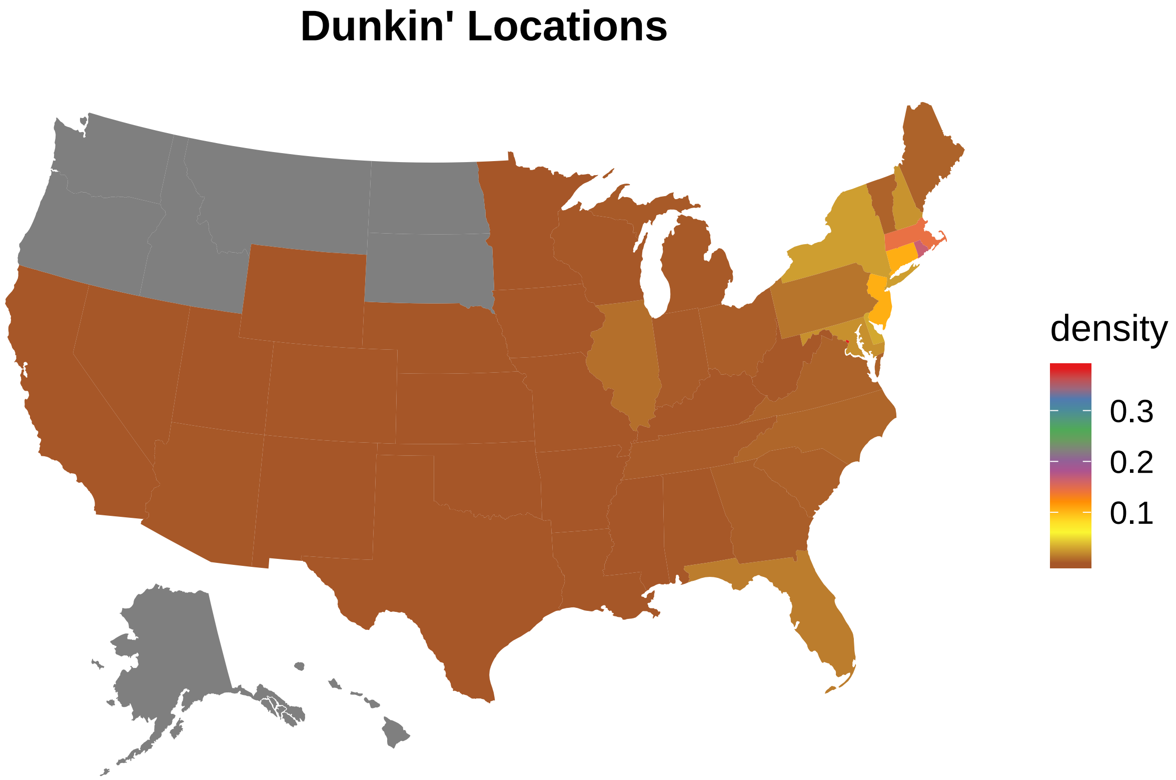


```
sq_mileage <- read_csv("data/mileage.csv")
DNKN_loc <- read_csv("data/DD-US.csv", col_names = FALSE)
SBUX_loc <- read_csv("data/directory.csv")

SBUX_loc <- SBUX_loc %>%
  filter(Country %in% c("US")) %>%
  rename(state = `State/Province`) %>%
  count(state)
DNKN_loc <- DNKN_loc %>%
  rename(specs = X3) %>%
  separate(specs, c("specs", "state"), sep = ",")
DNKN_loc <- DNKN_loc %>%
  mutate(state = DNKN_loc %>%
           pull(state) %>%
           str_replace_all("// ", "")) %>%
  count(state)

DNKN_loc <- left_join(us_map(region = "states") %>%
                        rename(state = abbr), DNKN_loc, by = "state")
SBUX_loc <- left_join(us_map(region = "states") %>%
                        rename(state = abbr), SBUX_loc, by = "state")

sq_mileage <- sq_mileage %>%
  rename(full = state,
         state_area = mileage)

DNKN_loc <- left_join(DNKN_loc, sq_mileage, by = "full")
SBUX_loc <- left_join(SBUX_loc, sq_mileage, by = "full")

DNKN_loc <- DNKN_loc %>%
  mutate(density = n/state_area)
SBUX_loc <- SBUX_loc %>%
  mutate(density = n/state_area)
```

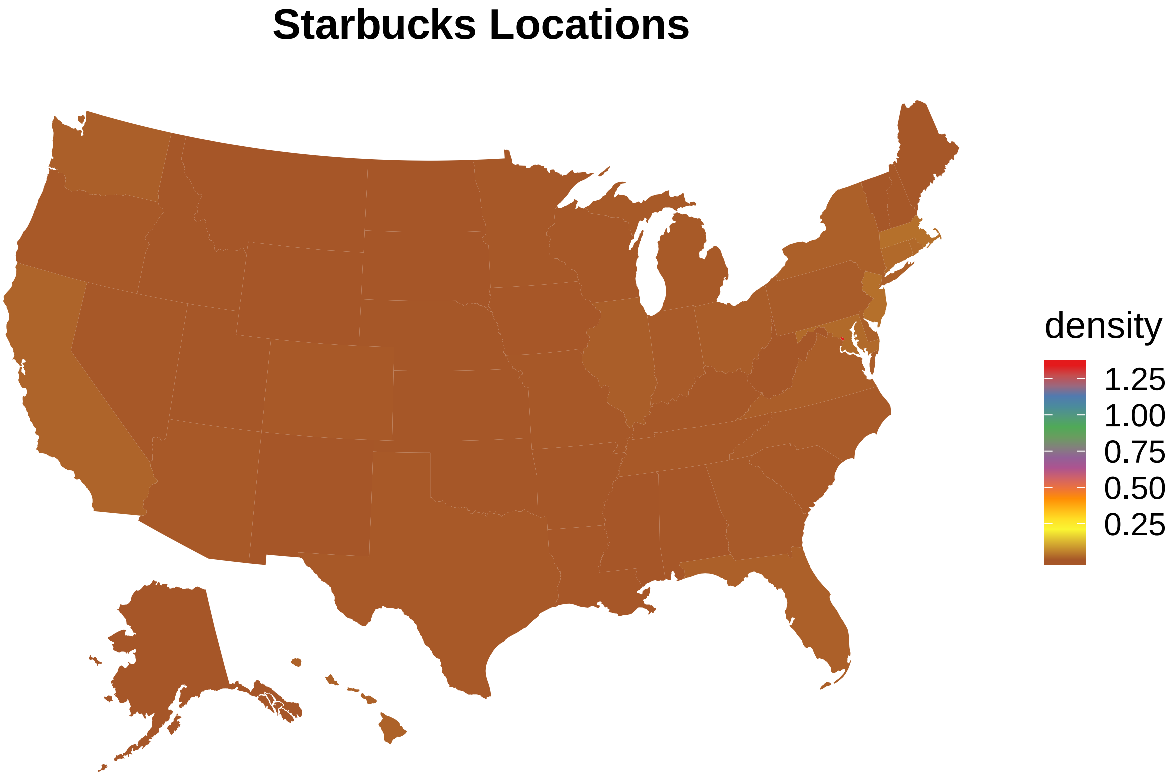
What states have the highest densities of Dunkin’ and Starbucks?

When Washington D.C. is included in the density plots, we noticed that the data is heavily skewed, so in order to get a proper look at the densities of these companies, Washington D.C. must be filtered out.

```
DNKN_loc %>%
  ggplot(mapping = aes(x = x, y = y, group = group, fill = density)) +
  geom_polygon() +
  coord_equal() +
  theme_map() +
  scale_fill_distiller(palette = "Set1") +
  labs(title = "Dunkin' Locations") +
  theme(plot.title = element_text(hjust = 0.5))
```

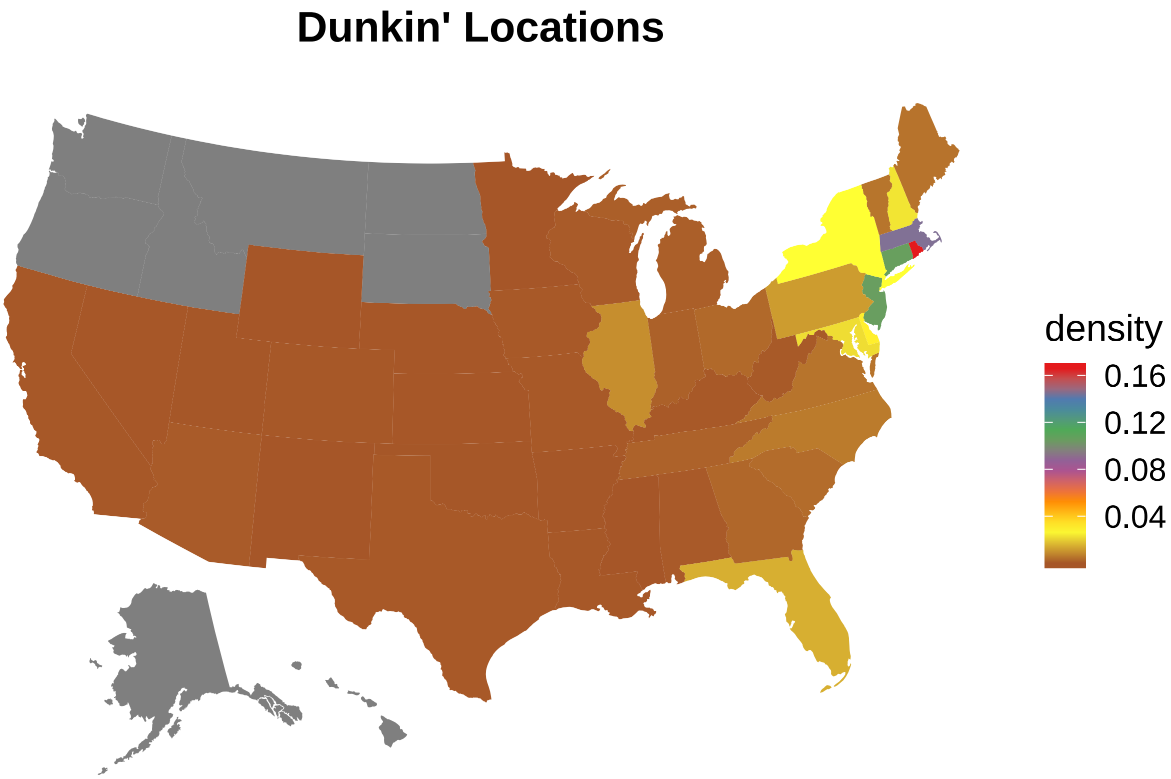


```
SBUX_loc %>%
  ggplot(mapping = aes(x = x, y = y, group = group, fill = density)) +
  geom_polygon() +
  coord_equal() +
  theme_map() +
  scale_fill_distiller(palette = "Set1") +
  labs(title = "Starbucks Locations") +
  theme(plot.title = element_text(hjust = 0.5))
```

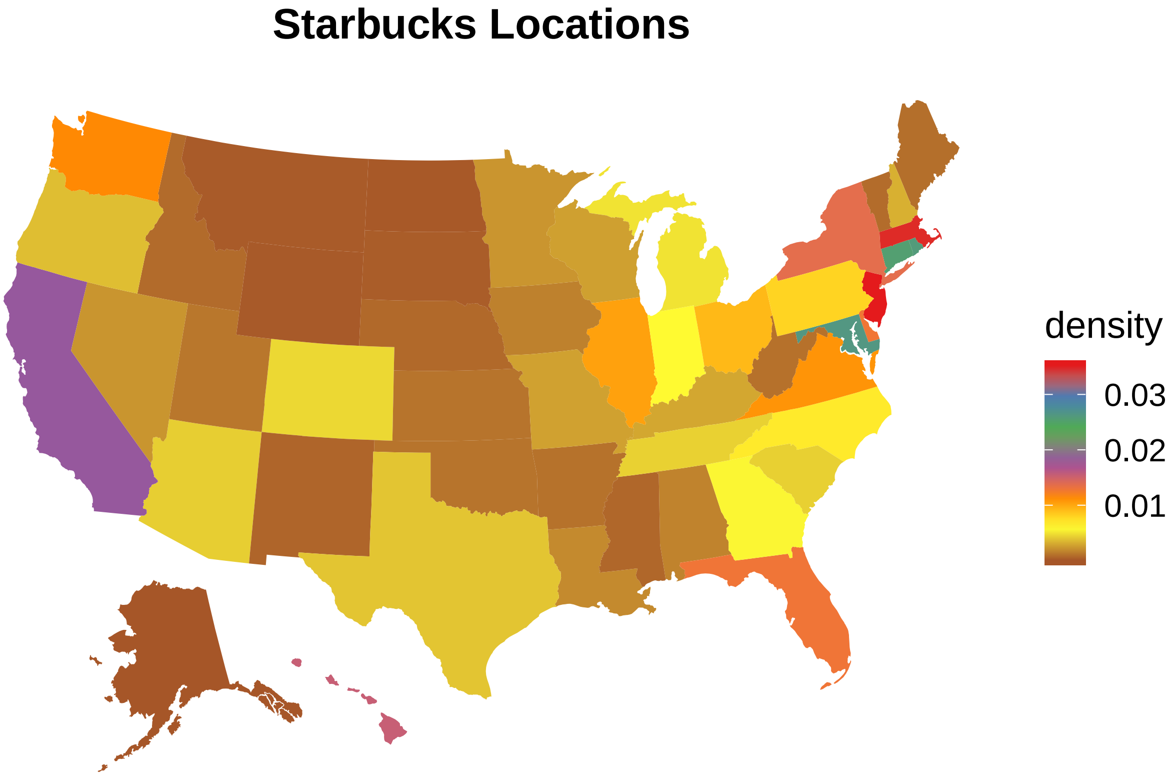


```
DNKN_loc <- DNKN_loc %>%
  filter(state %nin% c("DC"))
SBUX_loc <- SBUX_loc %>%
  filter(state %nin% c("DC"))
```

```
DNKN_loc %>%
  ggplot(mapping = aes(x = x, y = y, group = group, fill = density)) +
  geom_polygon() +
  coord_equal() +
  theme_map() +
  scale_fill_distiller(palette = "Set1") +
  labs(title = "Dunkin' Locations") +
  theme(plot.title = element_text(hjust = 0.5))
```



```
SBUX_loc %>%
  ggplot(mapping = aes(x = x, y = y, group = group, fill = density)) +
  geom_polygon() +
  coord_equal() +
  theme_map() +
  scale_fill_distiller(palette = "Set1") +
  labs(title = "Starbucks Locations") +
  theme(plot.title = element_text(hjust = 0.5))
```



```
DNKN_loc %>%
  distinct(state, density) %>%
  arrange(desc(density)) %>%
  top_n(5)
```

```
##   state    density
## 1    RI 0.16555024
## 2    MA 0.14413265
## 3    CT 0.10505676
## 4    NJ 0.10475934
## 5    DE 0.03121801
```

```
SBUX_loc %>%
  distinct(state, density) %>%
  arrange(desc(density)) %>%
  top_n(5)
```

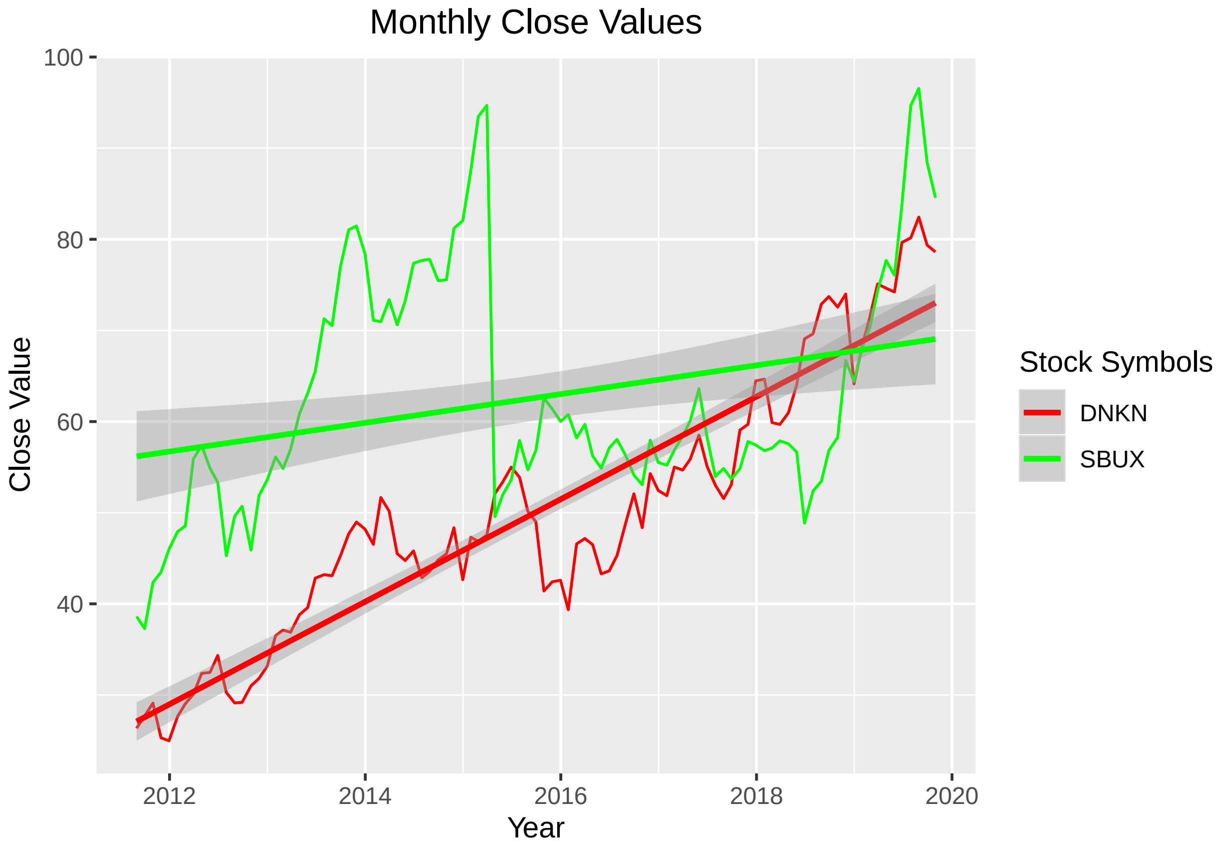
```
##   state    density
## 1    NJ 0.03518943
## 2    MA 0.03482143
## 3    MD 0.02629425
## 4    RI 0.02583732
## 5    CT 0.02538700
```

What are the growth rates of the stocks of Dunkin' and Starbucks?

```
DNKN <- read_csv("data/modified_DNKN.csv")
SBUX <- read_csv("data/modified_SBUX.csv")

DNKN <- DNKN %>%
  mutate(stock = "DNKN")
SBUX <- SBUX %>%
  mutate(stock = "SBUX")

D_S <- tibble(
  timestamp = c(DNKN %>%
                  pull(timestamp),
                SBUX %>%
                  pull(timestamp)),
  stock = c(DNKN %>%
              pull(stock),
            SBUX %>%
              pull(stock)),
  close = c(DNKN %>%
              pull(close),
            SBUX %>%
              pull(close))
)
```

Monthly Stock Close

```
D_S %>%
  group_by(stock) %>%
  ggplot() +
  geom_line(mapping = aes(x = timestamp, y = close, color = stock)) +
  scale_color_manual(values = c("red", "green")) +
  geom_smooth(mapping = aes(timestamp,close,color=stock),method = "lm") +
  labs(title = "Monthly Close Values",
       x = "Year",
       y = "Close Value") +
  guides(color = guide_legend(title = "Stock Symbols")) +
  theme(plot.title = element_text(hjust = 0.5))
```



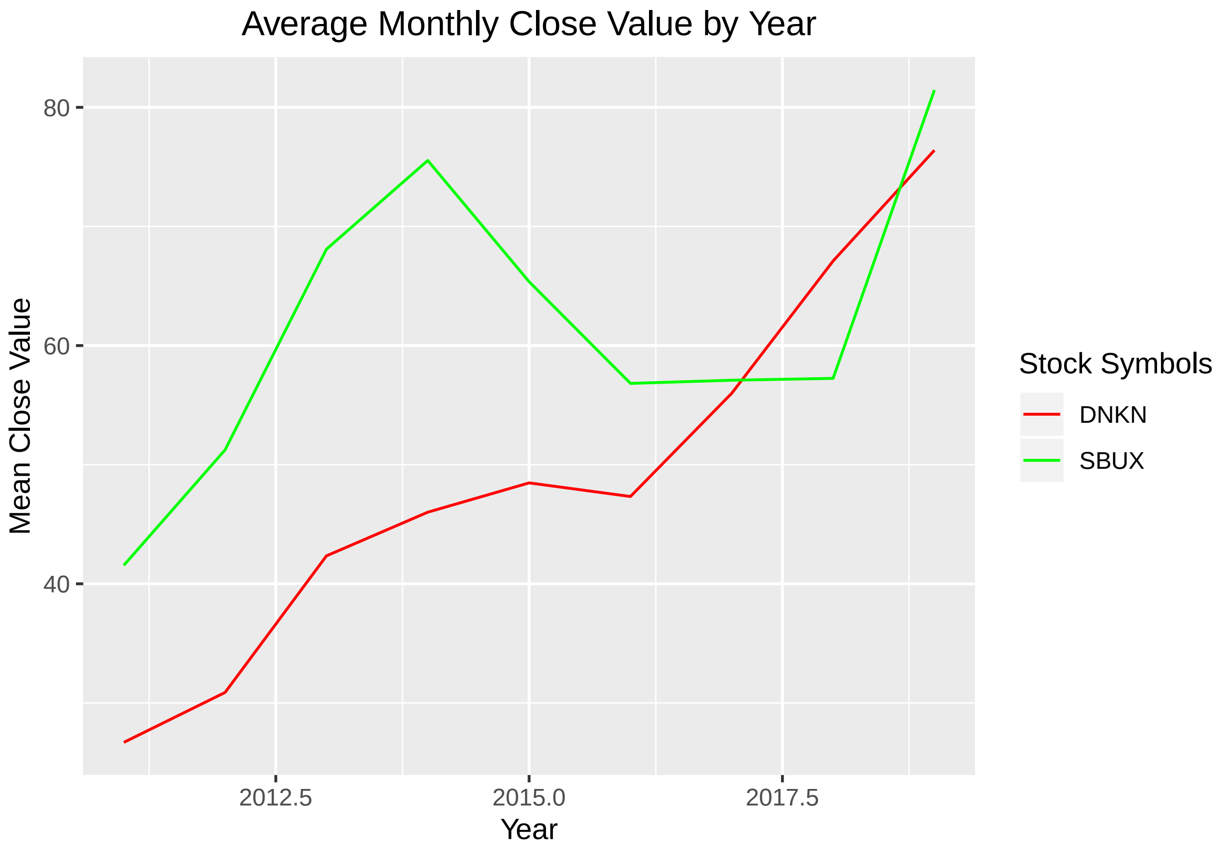
#Prediction Values

```
Starbucks_intercepts<-lm(SBUX$close~SBUX$timestamp)[1]$coefficient
Starbucks_monthly_average_growth_rate<-(-atan(Starbucks_intercepts[1]/Starbucks_intercepts[2]))

Dunkin_intercepts<-lm(DNKN$close~DNKN$timestamp)[1]$coefficient
Dunkin_monthly_average_growth_rate<-(-atan(Dunkin_intercepts[1]/Dunkin_intercepts[2]))
```

```
yrly <- D_S %>%
  mutate(year = year(timestamp)) %>%
  group_by(year, stock) %>%
  summarize(mean_close = mean(close))
```

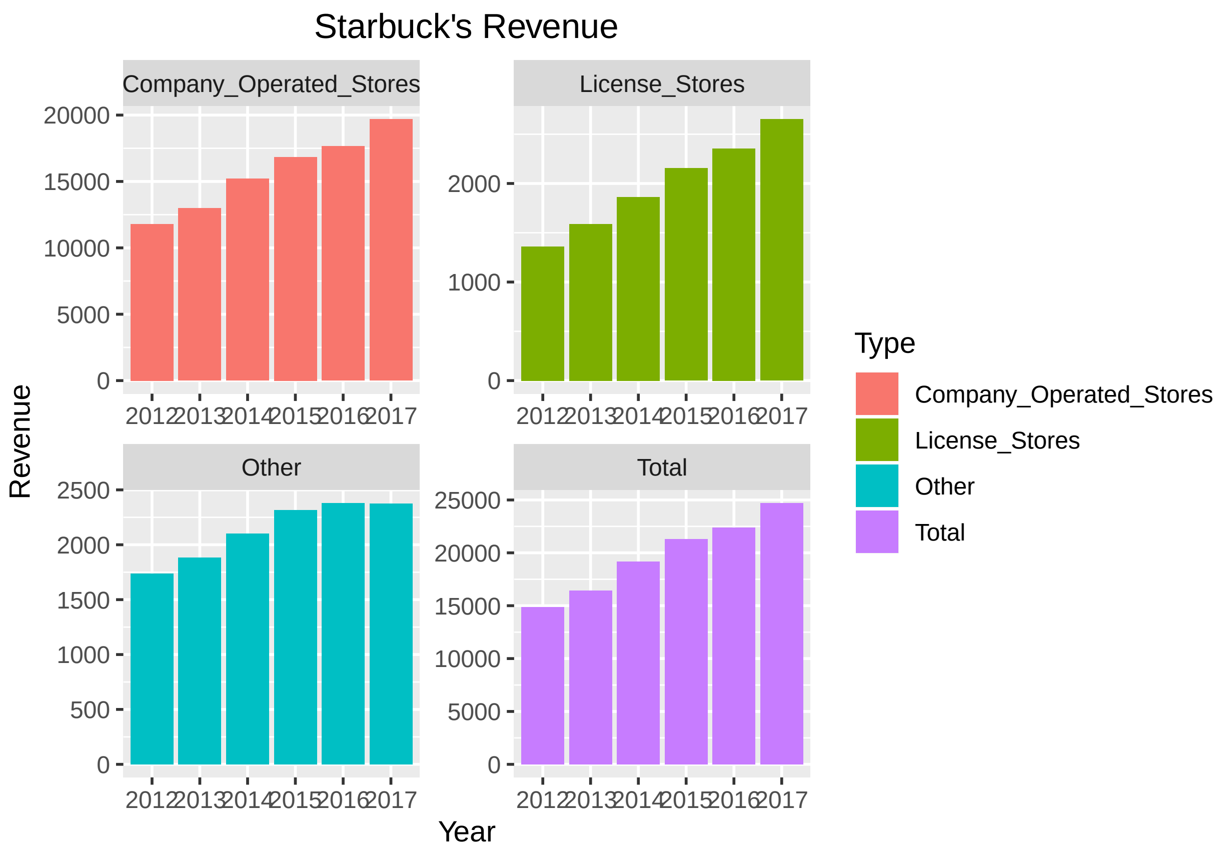
Yearly Mean Stock Close

```
yrly %>%
  group_by(stock) %>%
  ggplot() +
  geom_line(mapping = aes(x = year, y = mean_close, color = stock)) +
  scale_color_manual(values = c("red", "green")) +
  labs(title = "Average Monthly Close Value by Year",
       x = "Year",
       y = "Mean Close Value") +
  guides(color = guide_legend(title = "Stock Symbols")) +
  theme(plot.title = element_text(hjust = 0.5))
```

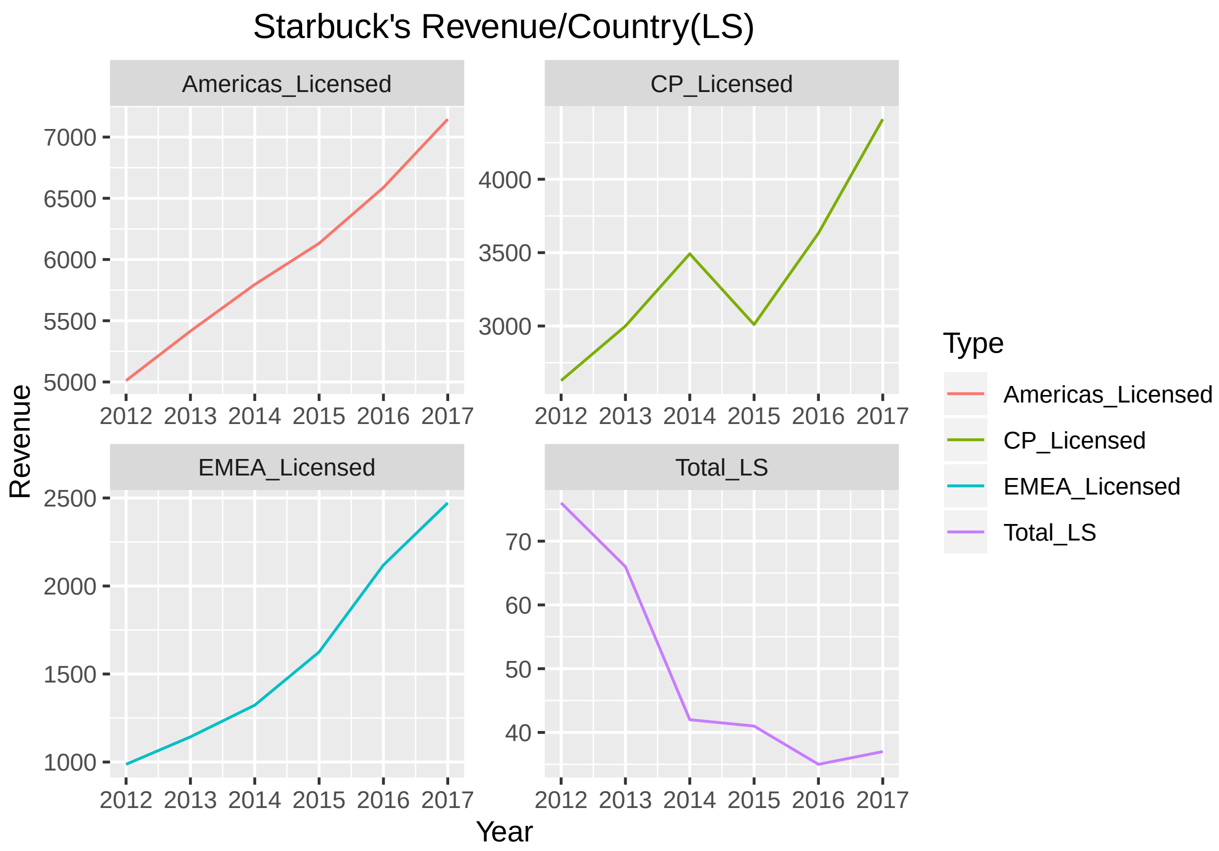


What are some patterns that show in the data of either Dunkin' or Starbucks?

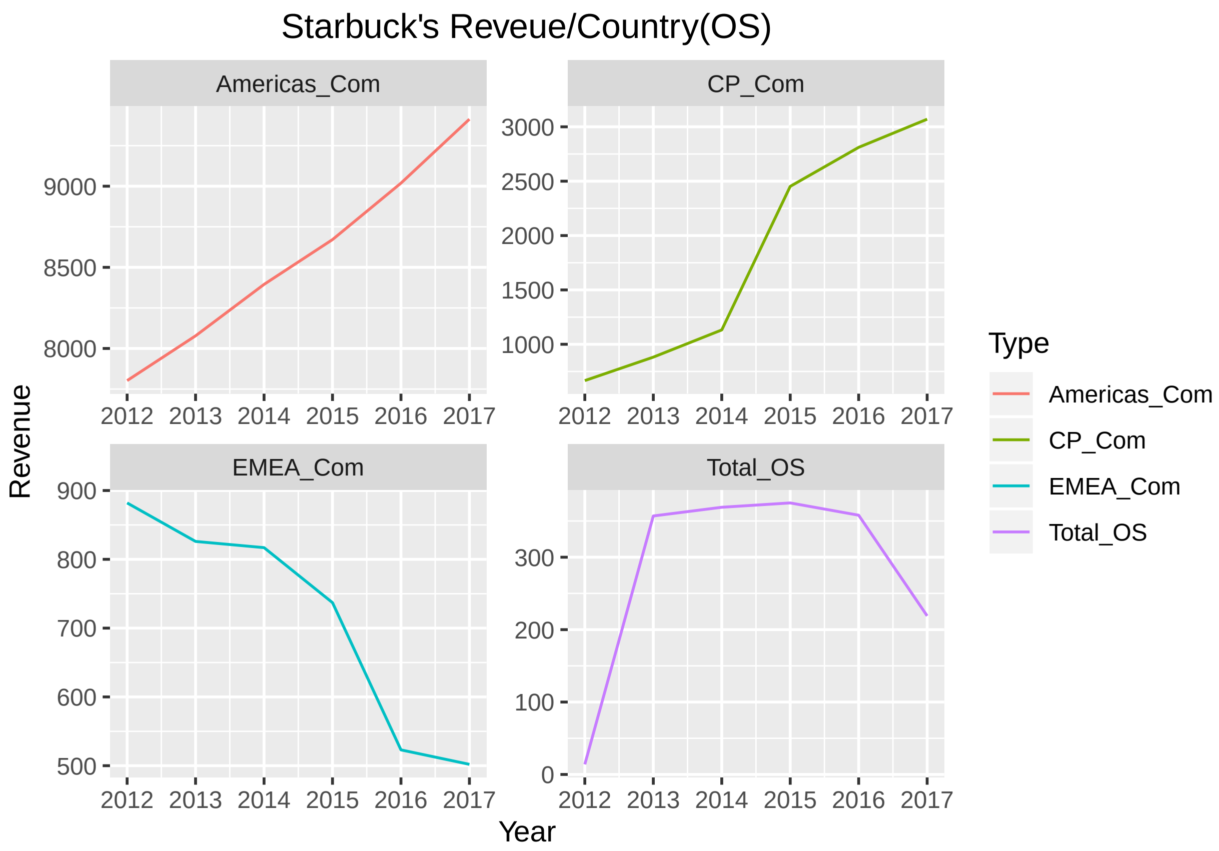
```
starbuck_rev %>%
  ggplot() +
  geom_col(mapping = aes(reorder(year, mean),
                         y = mean, fill=Type)) + labs(title = "Starbuck's Revenue")+
  theme(plot.title = element_text(hjust = 0.5)) + xlab("Year") + ylab("Revenue")  +  facet_wrap(~Type,
```



```
starbuck_rev %>%
  group_by(Type) %>%
  ggplot(mapping = aes(x = year, y = mean)) +
  geom_line(mapping = aes(color = Type)) + ylab("Revenue") + xlab("Year")+ facet_wrap(~Type, nrow = 3,s
theme(plot.title = element_text(hjust = 0.5))
```



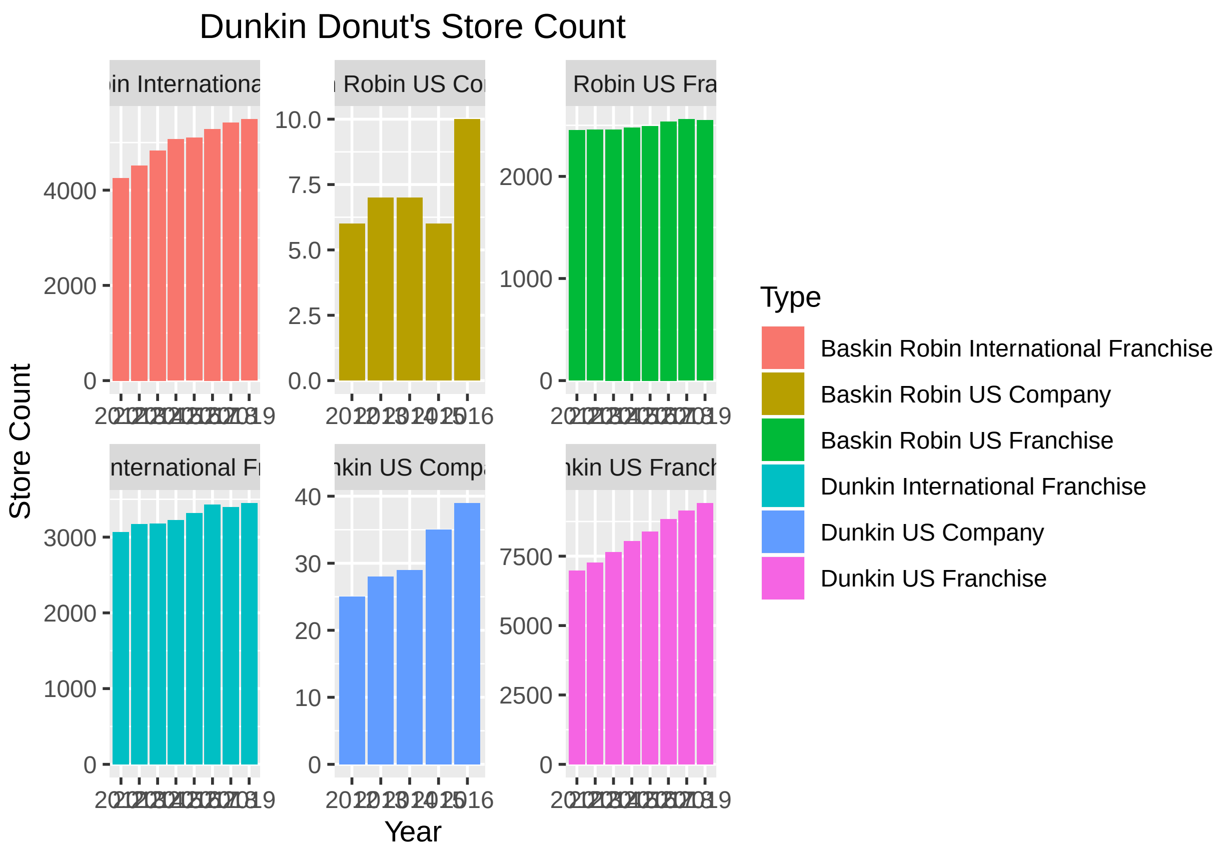
```
starbuck_rev %>%
  group_by(Type) %>%
  ggplot(mapping = aes(x = year, y = mean, )) +
  geom_line(mapping = aes(color = Type)) + ylab("Revenue") + xlab("Year")+ facet_wrap(~Type, nrow = 3,s
```



```
dunk_rev <- dunk_rev %>%
  select(`Dunkin US Franchise`, `Dunkin International Franchise`, `Baskin Robin US Franchise`,  `Baskin
  gather(Type, count,`Dunkin US Franchise`, `Dunkin International Franchise` , `Baskin Robin US Franchi
  na.omit()



dunk_rev <- dunk_rev%>%
  group_by(Type, year_of_publish) %>%
  summarise(mean = mean(count))
```

Dunkin Donuts Store Count

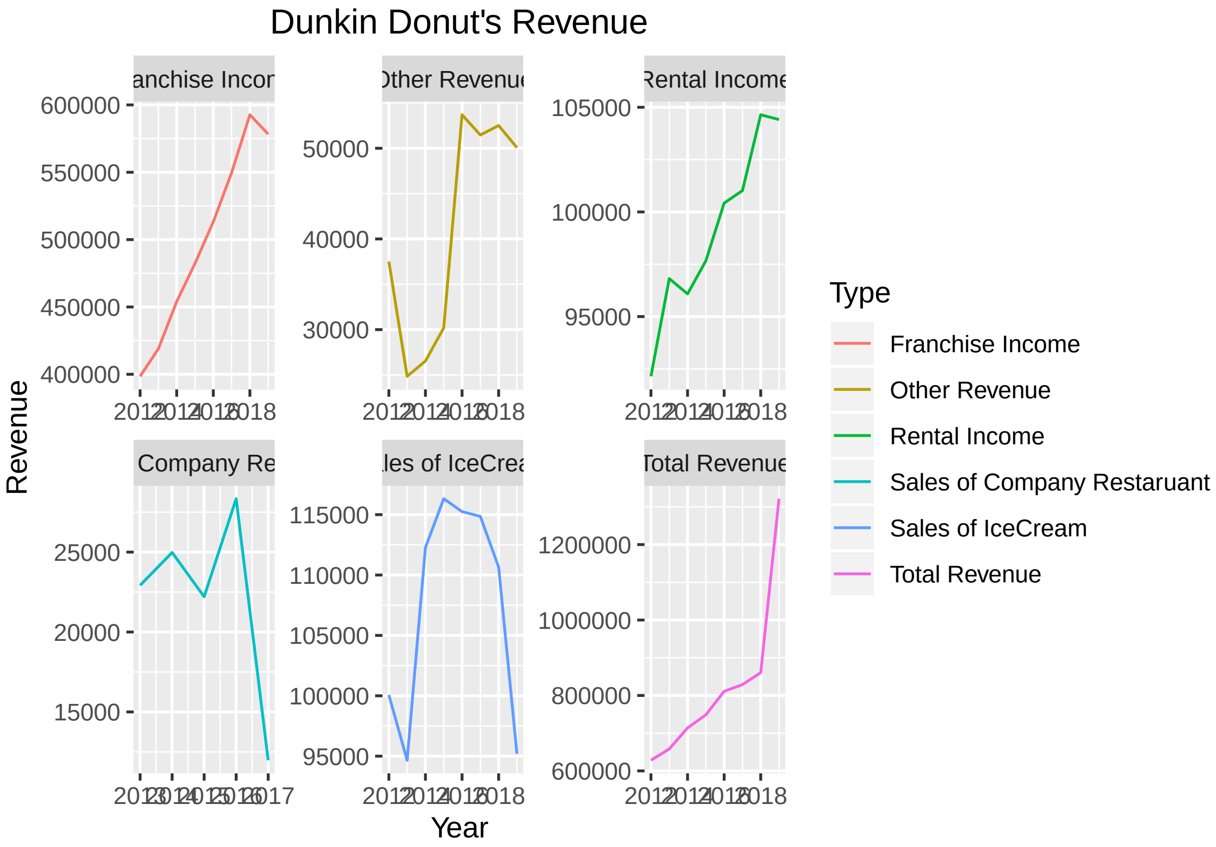
```
dunk_rev %>%
    ggplot() +
    geom_col(mapping = aes(reorder(year_of_publish, mean),
                           y = mean, fill=Type)) +
    theme(plot.title = element_text(hjust = 0.5)) + ylab("Store Count") + xlab("Year")+ labs(title = "Du
```



```
# dunk_rev %>%
#   group_by(Type) %>%
#   ggplot(mapping = aes(x = year_of_publish, y = mean)) +
#   geom_line(mapping = aes(color = Type)) + ylab("Store Count") + xlab("Year")+ facet_wrap(~Type, scal
```

Dunkin Donuts Revenue

```
dunk_rev %>%
  group_by(Type) %>%
  ggplot(mapping = aes(x = year_of_publish, y = mean)) +
  geom_line(mapping = aes(color = Type)) + ylab("Revenue") + xlab("Year")+ facet_wrap(~Type, scales = "
```


Dunkin Donut's Revenue
anchise Incor
600000
550000
500000
450000
400000
2012 2014 2016 2018
Other Revenue
50000
40000
30000
2012 2014 2016 2018
Rental Income
105000
100000
95000
2012 2014 2016 2018
Company Re
25000
20000
15000
2013 2014 2015 2016 2017
les of IceCrea
115000
110000
105000
100000
95000
2012 2014 2016 2018
Total Revenue
1200000
1000000
800000
600000
2012 2014 2016 2018
Revenue
Year
Type
Franchise Income
Other Revenue
Rental Income
Sales of Company Restaruant
Sales of IceCream
Total Revenue