# Project 1

Group 3: Reha Patel, Niraj Prasad, Sindhu Swaroop

10/31/2021

## Introduction

In this report, we will be investigating apartment data from the company called Equity Residential. Acquired from Kaggle, this data provides us with details such as price of the apartment, presence of specific features, and square footage. We will be analyzing relationships between the different variables found in the dataset to help understand which cities may be best for certain individuals as well as what features impact price.

```
#Read csv file
equity_apts_data <- read.csv ("Equity_Apartments_Data.csv", na.strings = "")
```

## Data Wrangling

### 1. Discovering

Prior to beginning our analysis of the data, it is important that we have an understanding of the dataset. Below we are importing the data and then looking at the first row of each column to understand how the dataset is set up as well as see the column names.

```
# View first six rows of each column
head(equity_apts_data, 1)
```

```
##   X Price Beds Baths sq.ft Floor Move_in_date building_id unit_id
## 1 1  2377    0     1   523     5   2021-09-02          01    0507
##                                                                              URL
## 1 https://www.equityapartments.com/washington-dc/northwest-dc/1210-mass-apartments
##   Day_Recorded
## 1   2021-07-17
##
## 1 Hard Surface Flooring Throughout\n\n\n                            \n\n\n\n
##         Apartment.Name                                           Address
## 1 1210 Mass Apartments 1210 Massachusetts Ave, NW\n\nWashington DC 20005
##            City Units Northern_Exposure Southern_Exposure Eastern_Exposure
## 1 Washington DC   144                 1                 0                0
##   Western_Exposure Balcony Walk_In_Closet Fireplace City_Skyline Kitchen_Island
## 1                1       0              0         0            0              0
##   Stainless_Appliances Renovated Office_Space Days_Till_Available
## 1                    1         1            0                  47
##   Day_of_the_week_recorded                Unique_ID Estiamted_Vacancy
## 1                Wednesday 0105071210MassApartments        0.02083333
```

After examining the columns, we noticed that there is a column for Unique_ID, which led us to believe that some apartments may be listed more than once. We want to know how many entries/rows there are in the dataset as a whole, as well as how many unique rows are present.

```
kable(
  equity_apts_data %>%
    summarise(total_records=n()) ,caption="Total Apartments Listed")
```

Table 1: Total Apartments Listed

| total_records |
|---:|
| 62810 |

```
kable(
  equity_apts_data %>%
  distinct(Unique_ID) %>% summarise(total_records=n())
  ,caption="Total Unique Apartments Listed")
```

Table 2: Total Unique Apartments Listed

| total_records |
|---:|
| 5305 |

After seeing that some apartments are listed multiple times, we want to further see how many unique records are found in each city.

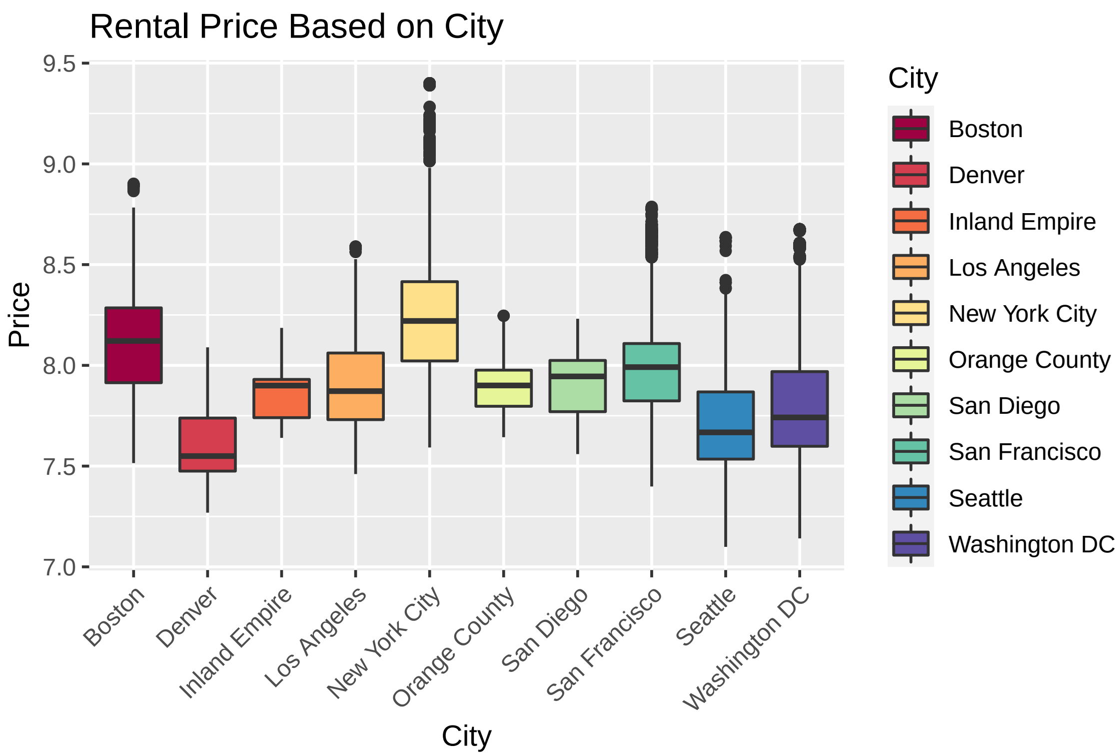
```
kable(
  equity_apts_data %>%
    group_by(City) %>% distinct(Unique_ID) %>% summarise(total_records=n())
  ,caption="Total Unique Apartments Listed by City")
```

Table 3: Total Unique Apartments Listed by City

| City | total_records |
|---|---:|
| Boston | 758 |
| Denver | 117 |
| Inland Empire | 71 |
| Los Angeles | 639 |
| New York City | 701 |
| Orange County | 134 |
| San Diego | 152 |
| San Francisco | 1017 |
| Seattle | 463 |
| Washington DC | 1253 |

Now that we know that there are 10 cities found in the dataset, we want to get the maximum rent, minimum rent, and median rent per city. Because there are so many cities to look at, being able to visualize this as a box plot will be beneficial to us.

```
equity_apts_data %>%
  ggplot(aes(x=City, y = log(Price), fill=City))+ geom_boxplot() +
  labs(title="Rental Price Based on City", x = "City", y = "Price") +
  theme(axis.text.x = element_text(hjust = 1, angle = 45))+
  scale_fill_brewer(palette="Spectral")
```

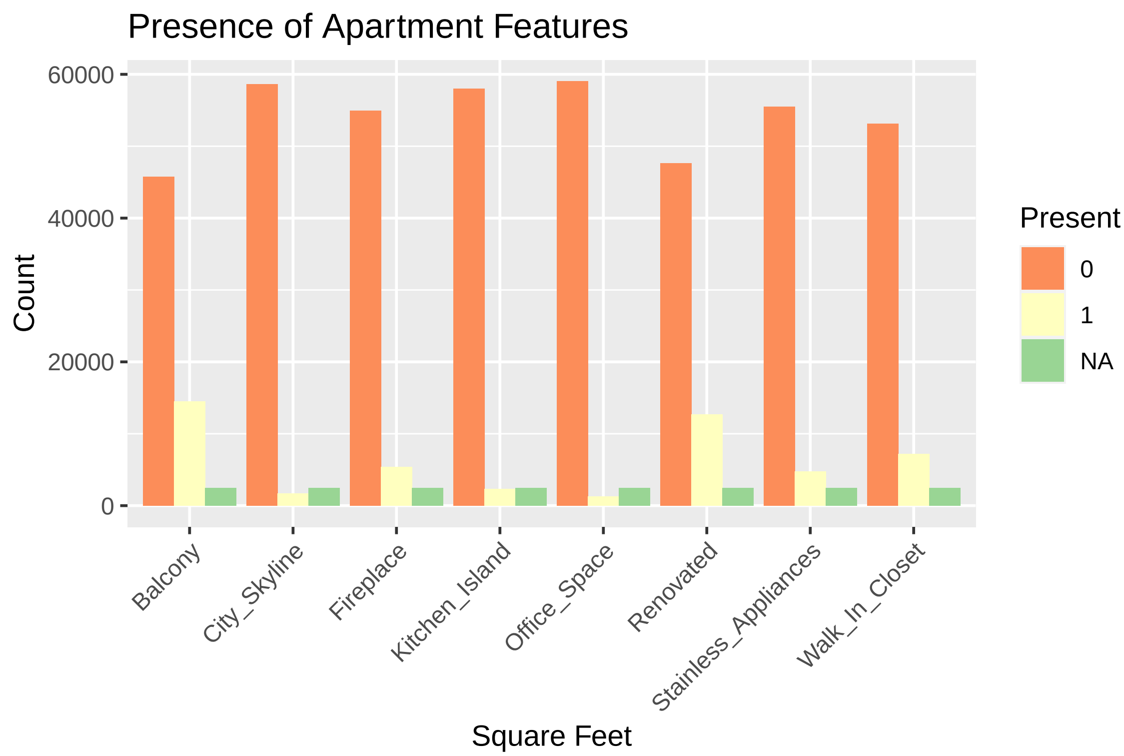



Many of the columns in the dataset were boolean values for whether a specific entry had an apartment feature. We want to determine which features are most commonly found in apartments.

```
# Select only apartment features
aptsFeatures <- equity_apts_data %>%
  select(Balcony, Walk_In_Closet, Fireplace, City_Skyline, Kitchen_Island,
         Stainless_Appliances, Renovated, Office_Space)

# pivot longer for the bar chart
pivoted <- aptsFeatures %>%
  pivot_longer(everything(), names_to = "Feature", values_to = "Present") %>%
  group_by(Feature, Present) %>%
  summarise(count=n())

pivoted %>%
  ggplot(aes(x=Feature, y = count, fill=Present))+
  geom_bar(stat = "identity",position = "dodge") +
  labs(title="Presence of Apartment Features", x = "Square Feet", y = "Count") +
  theme(axis.text.x = element_text(hjust = 1, angle = 45))+
  scale_fill_brewer(palette="Spectral")
```




## 2. Structuring

Now that we have done some discovery analysis on our dataset, we need to structure it to make the data more organized and consequently our analysis, easier. First, we checked the data types of each of the Feature columns and noticed that they are of type Character. Since we will be using this data in plots, it will be beneficial to convert them into numeric data types as below.

```
# Checking the data types of columns
checktypes <- equity_apts_data %>%
  select(Northern_Exposure, Southern_Exposure, Eastern_Exposure, Western_Exposure,
         Balcony, Walk_In_Closet, Fireplace, City_Skyline, Kitchen_Island,
         Stainless_Appliances, Renovated, Office_Space, Days_Till_Available)
sapply(checktypes, typeof)
```

```
##    Northern_Exposure    Southern_Exposure     Eastern_Exposure
##          "character"          "character"          "character"
##     Western_Exposure              Balcony       Walk_In_Closet
##          "character"          "character"          "character"
##            Fireplace         City_Skyline       Kitchen_Island
##          "character"          "character"          "character"
## Stainless_Appliances            Renovated         Office_Space
##          "character"          "character"          "character"
##  Days_Till_Available
##          "character"
```

```
# All of the above columns are characters, converting them to integer
equity_apts_data$Northern_Exposure <- as.integer(equity_apts_data$Northern_Exposure)
equity_apts_data$Southern_Exposure <- as.integer(equity_apts_data$Southern_Exposure)
equity_apts_data$Eastern_Exposure <- as.integer(equity_apts_data$Eastern_Exposure)
equity_apts_data$Western_Exposure <- as.integer(equity_apts_data$Western_Exposure)
equity_apts_data$Balcony <- as.integer(equity_apts_data$Balcony)
```

```
equity_apts_data$Walk_In_Closet <- as.integer(equity_apts_data$Walk_In_Closet)
equity_apts_data$Fireplace <- as.integer(equity_apts_data$Fireplace)
equity_apts_data$City_Skyline <- as.integer(equity_apts_data$City_Skyline)
equity_apts_data$Kitchen_Island <- as.integer(equity_apts_data$Kitchen_Island)
equity_apts_data$Stainless_Appliances <- as.integer(equity_apts_data$Stainless_Appliances)
equity_apts_data$Renovated <- as.integer(equity_apts_data$Renovated)
equity_apts_data$Office_Space <- as.integer(equity_apts_data$Office_Space)
equity_apts_data$Days_Till_Available <- as.integer(equity_apts_data$Days_Till_Available)

#Checking data type of Beds
typeof(equity_apts_data$Beds)
```

```
## [1] "integer"
```

```
#Changing type from integer to character
equity_apts_data$Beds <- as.character(equity_apts_data$Beds)
```

Next we noticed that there are two columns - Eastern and Western Exposure, which denote having a view of a Sunrise/Sunset. To make it easier to work with, we combined these two columns into one column - Sunrise/Sunset. The value of this column is 1 if the apartment has either Eastern/Western exposure, and 0 if it has neither.

```
# Define a new column - If either of the values are 1, make the new column value 1
# Else make the new column value 0
equity_apts_data <- equity_apts_data %>%
  mutate(Sunrise_Sunset = ifelse(equity_apts_data$Eastern_Exposure==1 |
                                   equity_apts_data$Western_Exposure==1, 1,0))

kable(
equity_apts_data %>%
    select(Eastern_Exposure, Western_Exposure, Sunrise_Sunset) %>% slice(1:5),
    caption="Addition of Sunrise/Sunset Column")
```

Table 4: Addition of Sunrise/Sunset Column

| Eastern_Exposure | Western_Exposure | Sunrise_Sunset |
|---:|---:|---:|
| 0 | 1 | 1 |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |

We also observed that the Address column contained the zipcode. In order to use the zipcode to get the latitude and longitude of a place, we extracted it from the Address and added the data into a new column - Zipcode. We dropped the addresses which did not have a Zipcode.

```
# Add new column 'Zipcode'. Use substring and pick last 5 characters
equity_apts_data$zipcode <- as.numeric(substr(equity_apts_data$Address,
          nchar(equity_apts_data$Address)-4, nchar(equity_apts_data$Address)))

# Remove rows with NA in Zipcode column
equity_apts_data <- equity_apts_data[-c(which(is.na(equity_apts_data$zipcode))),]

# Add leading 0 to Zipcodes with just 4 characters
for (i in 1:nrow(equity_apts_data)) {
    if(nchar(equity_apts_data$zipcode[i]) == 4) {
      equity_apts_data$zipcode[i] <- paste0("0", equity_apts_data$zipcode[i])}}
```

## 3. Cleaning

We now have neatly structured data but it still has invalid/incorrect values which need to be cleaned up. First we checked the column names in the data. The last column was spelled incorrectly, so we corrected it to make it easier to work with and also confirmed that the column name had been changed.

```
# Check column names for misspellings ("last one"Estiamated Vacancy is incorrect)
colnames(equity_apts_data)
```

```
##  [1] "X"                        "Price"
##  [3] "Beds"                     "Baths"
##  [5] "sq.ft"                    "Floor"
##  [7] "Move_in_date"             "building_id"
##  [9] "unit_id"                  "URL"
## [11] "Day_Recorded"             "Amenity"
## [13] "Apartment.Name"           "Address"
## [15] "City"                     "Units"
## [17] "Northern_Exposure"        "Southern_Exposure"
## [19] "Eastern_Exposure"         "Western_Exposure"
## [21] "Balcony"                  "Walk_In_Closet"
## [23] "Fireplace"                "City_Skyline"
## [25] "Kitchen_Island"           "Stainless_Appliances"
## [27] "Renovated"                "Office_Space"
## [29] "Days_Till_Available"      "Day_of_the_week_recorded"
## [31] "Unique_ID"                "Estiamted_Vacancy"
## [33] "Sunrise_Sunset"           "zipcode"
```

```
# Check number of columns to determine which one to change
ncol(equity_apts_data)
```

```
## [1] 34
```

```
# Correct last column name and check for correction
colnames(equity_apts_data)[32] <- "Estimated_Vacancy"
colnames(equity_apts_data)[32]
```

```
## [1] "Estimated_Vacancy"
```

## 4. Enriching

Now that we have clean data, the next task is to enrich the data by adding useful information to it. Using the Zipcodes that we extracted earlier, we used the zipcodeR package to obtain the latitude and longitude information for each of the zipcodes so we can later create a heat map. We also noticed that there are many apartments with zero pricing and removed those records because they may impact our plots. Because we may want to plot by longitude and latitude, it's important that we also remove the NA's from those two columns to avoid any errors later on.

```
# Adding latitude and longitude by merging zipcode db with our table
equity_apts_data <- merge(equity_apts_data, zip_code_db[c("zipcode", "lat", "lng")], by = "zipcode")
names(equity_apts_data)[names(equity_apts_data) == "lat"] <- "Latitude"
names(equity_apts_data)[names(equity_apts_data) == "lng"] <- "Longitude"

# Remove NA columns in latitude and longitude columns
equity_apts_data <- subset(equity_apts_data, !(is.na(equity_apts_data$Latitude)))
equity_apts_data <- subset(equity_apts_data, !(is.na(equity_apts_data$Longitude)))

# Check if any apartment has zero price
unique(equity_apts_data$Price == 0)
```

```
## [1] FALSE  TRUE
```

```r
# Remove all the rows with zero pricing
equity_apts_data <- subset(equity_apts_data, equity_apts_data$Price!=0)

# Confirm there are no apartments with zero price
unique(equity_apts_data$Price == 0)
```

```
## [1] FALSE
```

Often times the number of bedrooms and bathrooms can have an impact on the the price per month of an apartment. We're creating a new column that indicates the relationship between bedrooms and bathrooms which can be used for analysis later.

```r
equity_bed_bath <-
  equity_apts_data %>%
  mutate( bed_bath_bin=
            case_when(
              Beds==Baths ~ "Beds = Bath ",
              Beds>Baths ~ "Beds > Baths",
              Beds<Baths ~ "Beds < Baths"))
```

### 5. Validating

Now we are almost done with Data Wrangling. The last task is to do a final validation of our data. As part of validating, we want to confirm that the changes we made in the previous 4 steps were successful. We start off with checking for NAs in the address column.

```r
unique(is.na(equity_apts_data$Address))
```

```
## [1] FALSE
```

Next, we checked for NAs in the City column because without that information the data has no meaning. Additionally it will create a new category as NA in the City column.

```r
unique(is.na(equity_apts_data$City))
```

```
## [1] FALSE
```

Next, we made sure there are no negative values in the Days till available column. Logically, if the days till available column is negative, then a tenant has already moved in and the rent price would already be fixed at that point.

```r
unique(equity_apts_data$Days_Till_Available<0)
```

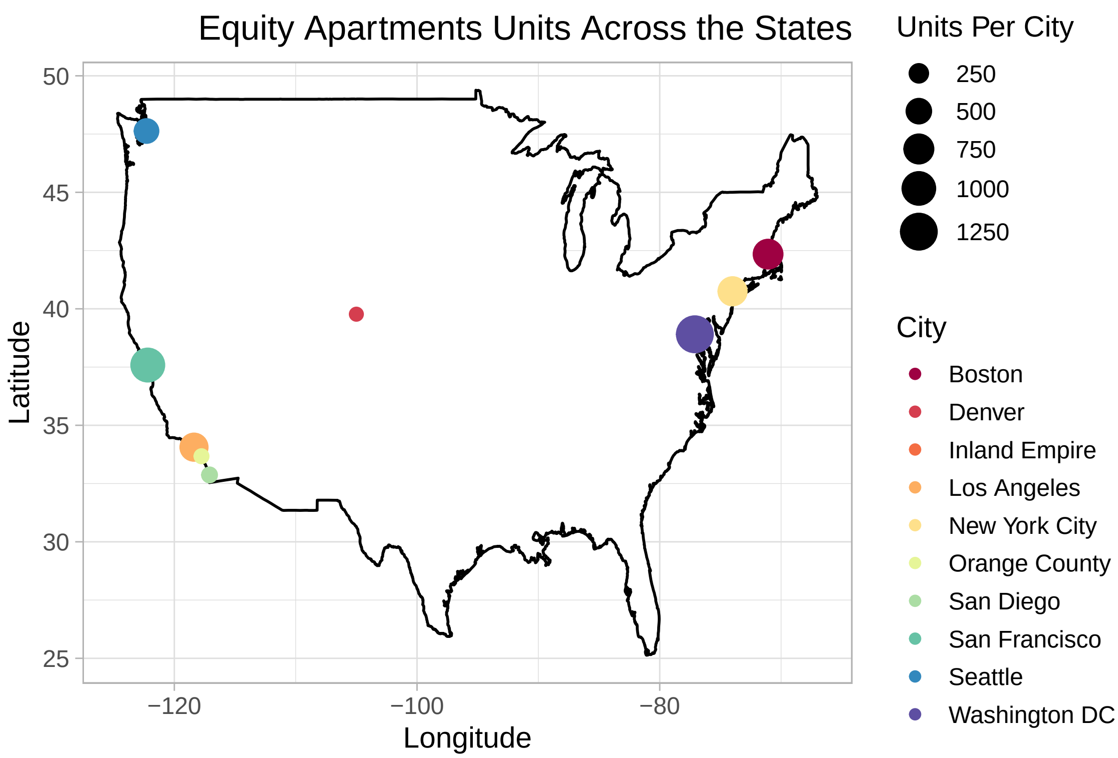
```
## [1]    NA FALSE
```

## Business Questions:

**Question 1: Equity Apartments has properties in multiple cities throughout the United States. Plot the number of apartments by city to determine where the most apartments are located. Plotting on a US map will help renters visualize where in the country their options are if they would like to rent from Equity Apartments.**

```r
usa <- map_data('usa')
summaryCoord <- equity_apts_data %>%
                distinct(Unique_ID, .keep_all = TRUE) %>%
                group_by(City) %>%
                summarise(UnitsPerCity = n(), meanLat = mean(Latitude),
                          meanLong = mean(Longitude))

ggplot() + geom_polygon(data = usa, aes(x=long, y = lat, group = group),
              fill = "white", color = "black") +
  theme_linedraw() + theme_light() +
  labs(x="Longitude", y="Latitude") +
  ggtitle("Equity Apartments Units Across the States") +
  theme(plot.title = element_text(hjust = 1), legend.key.size = unit(0.5, 'cm')) +
  geom_point(data = summaryCoord , aes(x = meanLong, y = meanLat,
                                 size = UnitsPerCity, color = City)) +
  labs(size = "Units Per City")+ scale_colour_brewer(palette="Spectral")
```



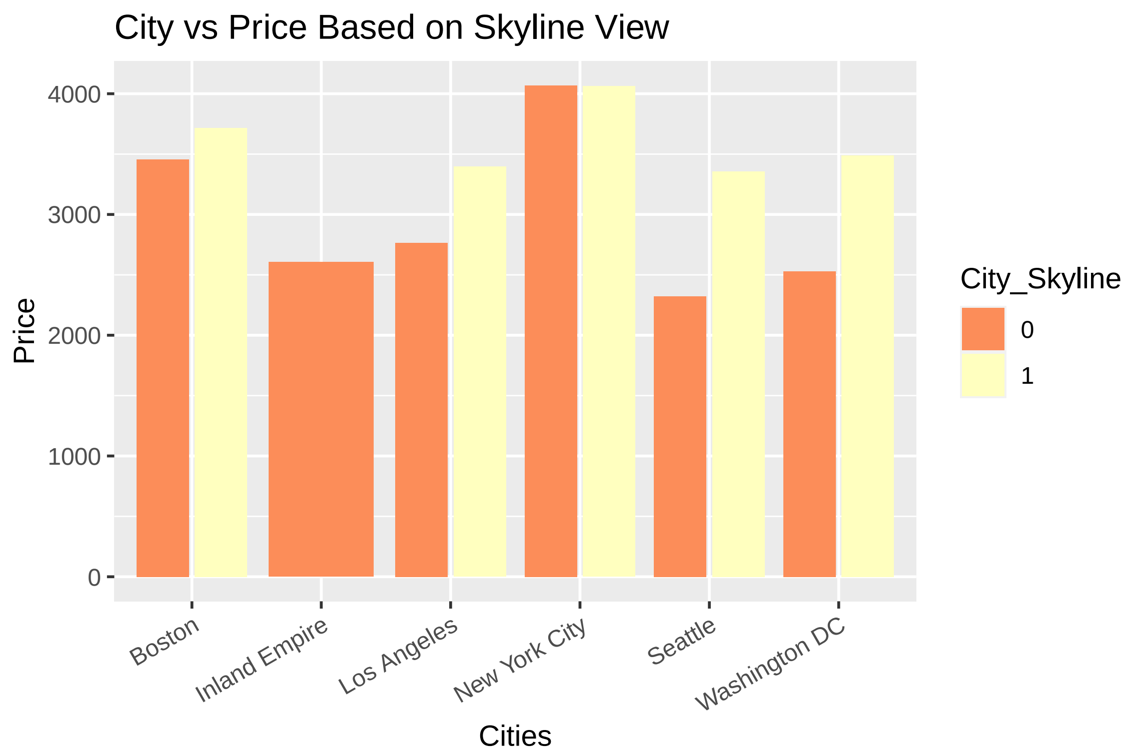

**Observations** We made the following observations from the map above:

- Washington DC has the highest number of Equity Apartment units.
- Inland Empire and Denver have the lowest number of Equity units across the country.
- There are a greater number of Equity Apartment properties on the West Coast compared to the East Coast.

**Question 2: Does having a city skyline view impact the price of the apartment? If it does, property owners will be able to charge a greater amount of rent for apartments with a skyline view.**

```r
# Unique values
equity_apts_data$City_Skyline <- as.character(equity_apts_data$City_Skyline)
```

```r
equity_apts_data %>%
 subset((City_Skyline == 0 | City_Skyline == 1)
        & (City == "Boston" | City == "New York City" | City == "Los Angeles" | City == "Seattle"
                    | City == "Washington DC" | City == "Inland Empire")) %>%
  group_by(City, City_Skyline) %>%
  summarise(avgprice = mean(Price)) %>%
  ggplot(aes(x = City, y = avgprice, fill = City_Skyline)) +
  geom_bar(stat = "Identity", position="dodge2") +
  labs(x="Cities", y="Price", title = "City vs Price Based on Skyline View") +
  theme(axis.text.x = element_text(hjust = 1, angle = 30)) +
  scale_fill_brewer(palette="Spectral")
```




**Observations** We made the following observations from the bar chart above:

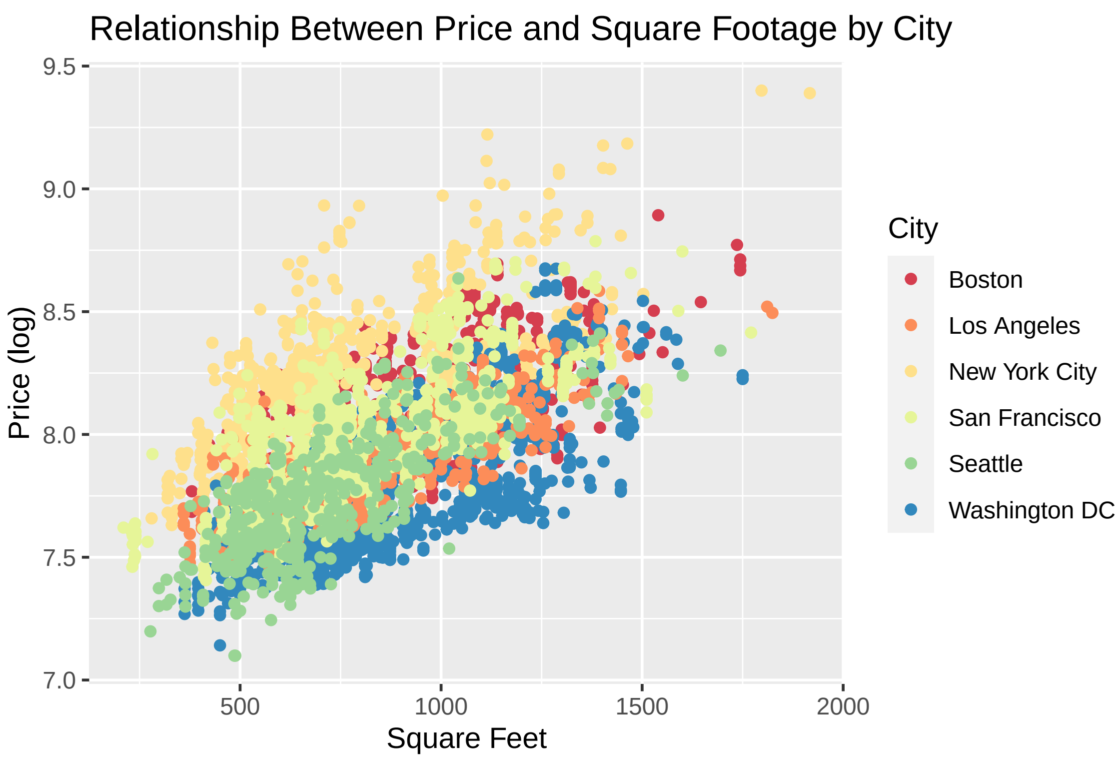
- Across the big cities such as Boston and Seattle, apartments with a skyline view tend to be more expensive than those with no view.
- In New York City, the skyline view availability seems to have no impact on the pricing.
- Inland Empire has no units with a city skyline in the dataset.

**Question 3: What is the relationship between the square footage of an apartment and the price of it? In different cities, the price per square foot can vary greatly. Landlords and property owners should be aware of the relationship between price of the apartment and the total square footage so they can charge rent accordingly.**

```r
equity_apts_data_u <- equity_apts_data %>% distinct(Unique_ID, .keep_all = TRUE)

apts_data_u_cities <- equity_apts_data_u %>%
  filter(City %in% c("Boston", "Los Angeles", "New York City", "San Francisco",
                     "Seattle", "Washington DC"))
```

```r
ggplot(apts_data_u_cities, aes(x=sq.ft, y=log(Price), color=City)) +
  geom_point() +
  labs(title="Relationship Between Price and Square Footage by City") +
  labs(x = "Square Feet") + labs(y = "Price (log)") +
  scale_colour_brewer(palette="Spectral")
```

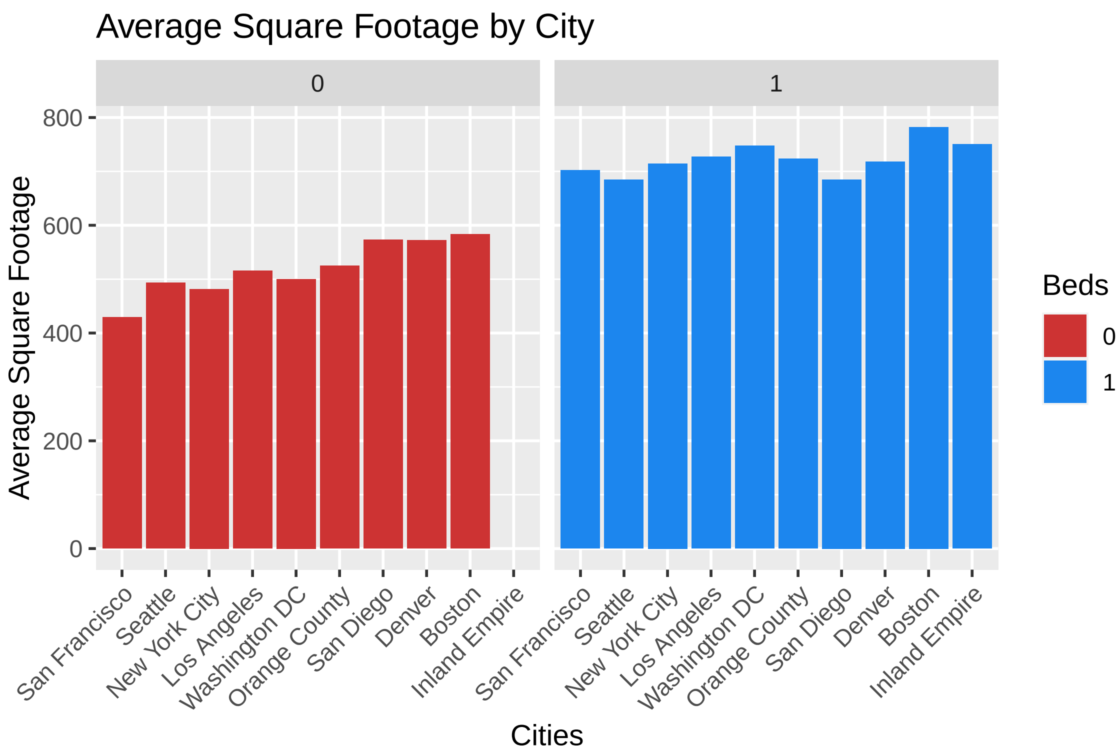



**Observations** We made the following observations from the scatter plot above:

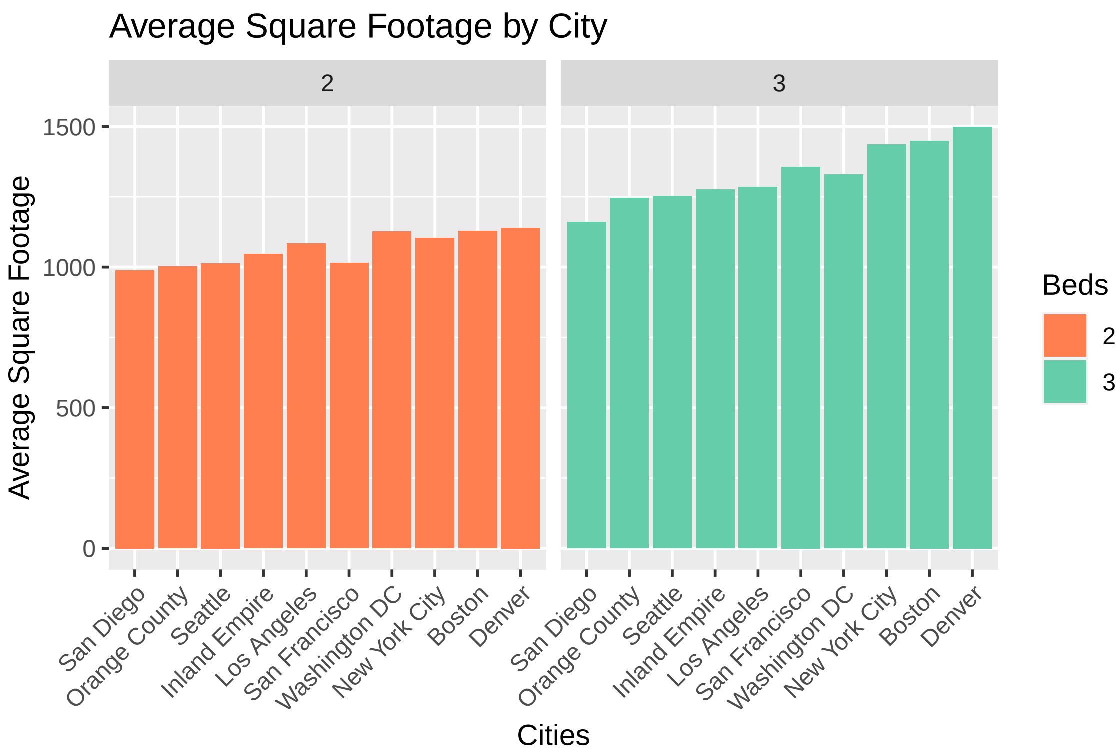
- New York City tends to have the highest rent for the same square footage when compared to other cities.
- Regardless of square footage, Washington DC tends to have the lowest rent.
- As the square footage of an apartment increases, the price begins to vary more, and we can see this because the points are much more spread out.

**Question 4: Which cities have apartments with higher square footage? Renters may want to live in cities with specific square footages so it is important for them to be aware of where they will find the right sized apartment.**

```r
# 0 and 1 Beds
equity_apts_data %>% distinct(Unique_ID, .keep_all = TRUE) %>%
                     subset(Beds==0 | Beds==1) %>%
                     group_by(City, Beds) %>%
                     summarise(AvgSqByBed = mean(sq.ft)) %>%
                     arrange(desc(AvgSqByBed)) %>%
  ggplot(aes(x = reorder(`City`, AvgSqByBed), y = AvgSqByBed, fill = Beds)) +
  geom_bar(stat = "Identity") +
  theme(axis.text.x=element_text(hjust=1, angle=45))+
  scale_fill_manual(values=c("brown3", "dodgerblue2"))+
  labs(x="Cities", y="Average Square Footage", title = "Average Square Footage by City")+
  facet_grid(.~Beds)
```




```r
# 2 and 3 Beds
equity_apts_data %>% distinct(Unique_ID, .keep_all = TRUE) %>%
                     subset(Beds == 2 | Beds == 3) %>%
                     group_by(City, Beds) %>%
                     summarise(AvgSqByBed = mean(sq.ft)) %>%
                     arrange(desc(AvgSqByBed)) %>%
  ggplot(aes(y = AvgSqByBed, x = reorder(`City`, AvgSqByBed), fill = Beds)) +
  geom_bar(stat = "Identity") +
  theme(axis.text.x = element_text(hjust = 1, angle = 45))+
  scale_fill_manual(values=c("coral", "aquamarine3")) +
  labs(x = "Cities", y = "Average Square Footage", title = "Average Square Footage by City") +
  facet_grid(.~Beds)
```

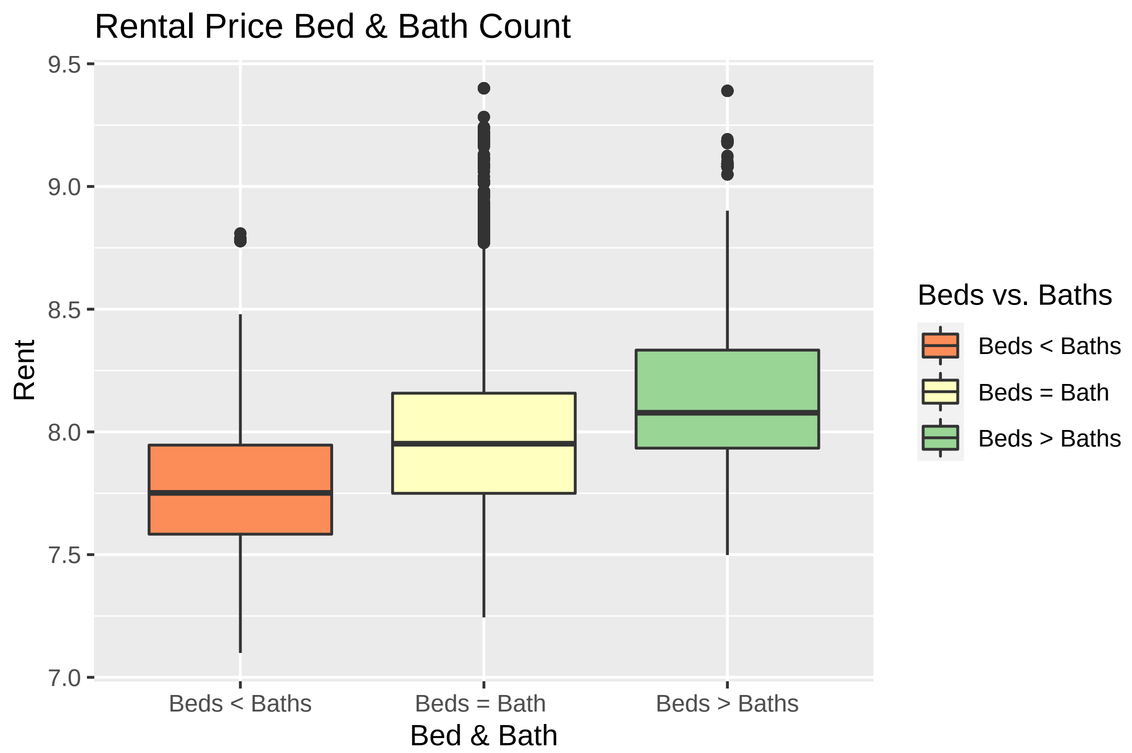



**Observations** We made the following observations from the bar chart above:

- In 2 and 3 bedroom apartments, Denver has the largest units so renters looking for more space should look there.
- In studios and 1 bed apartments, Boston has the largest sized units.
- Inland Empire has no Equity studio listings in our database.
- Overall, San Diego has the smallest apartments on average, so regardless of price renters will have smaller spaces there.

**Question 5: An apartment having a private bathroom means it has the same number of bedrooms and bathrooms. While many apartment have the same number of bathrooms as bedrooms, some have a greater number of bathrooms or bedrooms. So, does an apartment having a private bathroom make it more expensive than an apartment without one?**

```
equity_bed_bath %>%
  ggplot(aes(x=bed_bath_bin, y = log(Price), fill=bed_bath_bin))+
  geom_boxplot() +
  labs(title="Rental Price Bed & Bath Count") +
  labs(x = "Bed & Bath", y = "Rent", fill="Beds vs. Baths") +
  scale_fill_brewer(palette="Spectral")
```




**Observations** We made the following observations from the box plot above:

- The rent price of the middle 50% apartments with a great number of bedrooms than bathrooms is greater than the middle 50% of apartments with a greater number of bathrooms than bedrooms.
- The median rent price for apartments where the number of bedrooms equals the number of bathrooms is equal to the third quartile of when the number of bedrooms is less than the number of bathrooms.

**Question 6: For each of the top 5 cities with the most units in the dataset, what is the average rent price? For new renters trying to lease with Equity Apartments, they should know the cities in which they will have the most options within their price range.**

```
Cities_with_most_units <- equity_apts_data %>%
  distinct(Unique_ID, .keep_all = TRUE) %>%
  group_by(City) %>%
  summarise(Number_of_Units_per_City = n())

Average_Rent <- equity_apts_data %>%
  group_by(City) %>%
  summarise(Average_Rent = round(mean(Price), 2))

Average_Rent_of_Cities <- merge(Cities_with_most_units, Average_Rent, by="City") %>%
  arrange(desc(Number_of_Units_per_City)) %>% slice(1:5)

kable(
  Average_Rent_of_Cities
  ,caption="Average Rent Price of Top 5 Cities with Most Units")
```

Table 5: Average Rent Price of Top 5 Cities with Most Units

| City | Number_of_Units_per_City | Average_Rent |
|---|---|---|
| Washington DC | 1253 | 2521.27 |
| San Francisco | 1017 | 3018.01 |
| Boston | 739 | 3451.60 |
| New York City | 698 | 4027.23 |
| Los Angeles | 638 | 2779.53 |

**Observations** We made the following observations from the table:

- New York City has the highest average rent price followed by Boston.
- Washington DC has the lowest average rent price among the 5 major cities listed and also the most number of units per city. An individual hoping to rent with Equity Apartments more affordable cities would be best off considering Washington DC as their top option.

**Question 7: Often times, having a good view can impact the price a tenant will pay for an apartment leading landlords and property owners to make more per month off a listing. Depending on if an apartment has eastern (sunrise) or western (sunset) exposure, does Equity Apartments see a difference in average rent price?**

```
kable(
  equity_apts_data_u %>%
    drop_na(Sunrise_Sunset) %>%
    group_by(Sunrise_Sunset) %>%
    summarise(total_records=n(), average_rent = mean(Price))
  ,caption="Rent Based on Sunrise/Sunset Exposure")
```

Table 6: Rent Based on Sunrise/Sunset Exposure

| Sunrise_Sunset | total_records | average_rent |
|---|---|---|
| 0 | 2829 | 2913.843 |
| 1 | 2220 | 3051.574 |

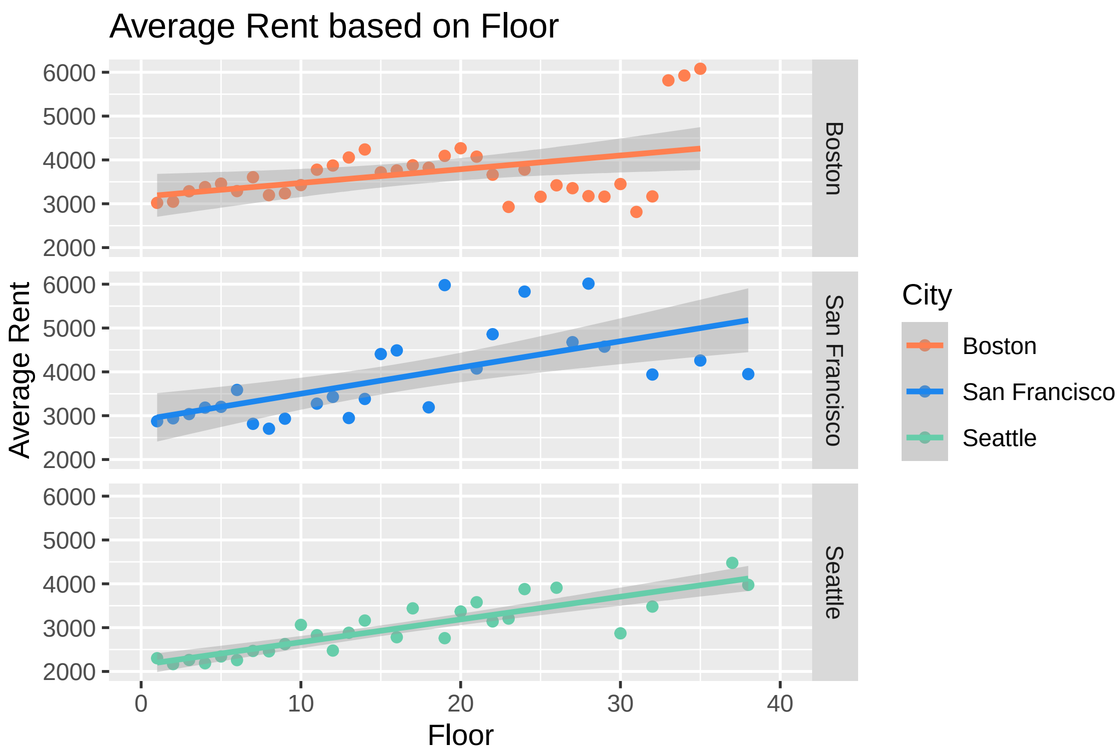
**Observations** We made the following observations from the table:

- Average rent for an apartment with either a sunrise or sunset view is about $140 greater than apartments without a sunrise or sunset view.
- There are a greater number of apartments that face the Sunrise/Sunset than those that don't.

**Question 8: Sometimes having an apartment on a higher floor can mean having a better view, or other times the units on the highest floors are even considered penthouse suites. What relationship does the floor an apartment is on have on the price of the apartment?**

```
  equity_apts_data %>%
  distinct(Unique_ID, .keep_all = TRUE) %>%
  subset(City=="Boston" | City=="Seattle" | City == "San Francisco") %>%
```

```
  group_by(City, Floor) %>%
  summarise(Average_Price=mean(Price)) %>%
  ggplot(aes(x=Floor, y=Average_Price, color=City)) +
  geom_point()+
  xlim(0, 40) +
  geom_smooth(method="lm")+
  labs(x="Floor", y="Average Rent", title = "Average Rent based on Floor")+
  facet_grid(City~.)+
  scale_colour_manual(values=c("coral", "dodgerblue2", "aquamarine3"))
```




**Observations** We made the following observations from the scatter plots:

- All 3 major cities plotted here show an increasing trend in terms of Floor vs. Price. The higher up the apartment is located, the higher the rent is.
- Boston shows a steep increase at topmost floors so renters looking for more affordable options may want to consider lower floors.

**Question 9: What specific apartment features does each city offer as a whole? Knowing which features are most common in specific cities can help renters get an idea of what features their potential apartments may have.**

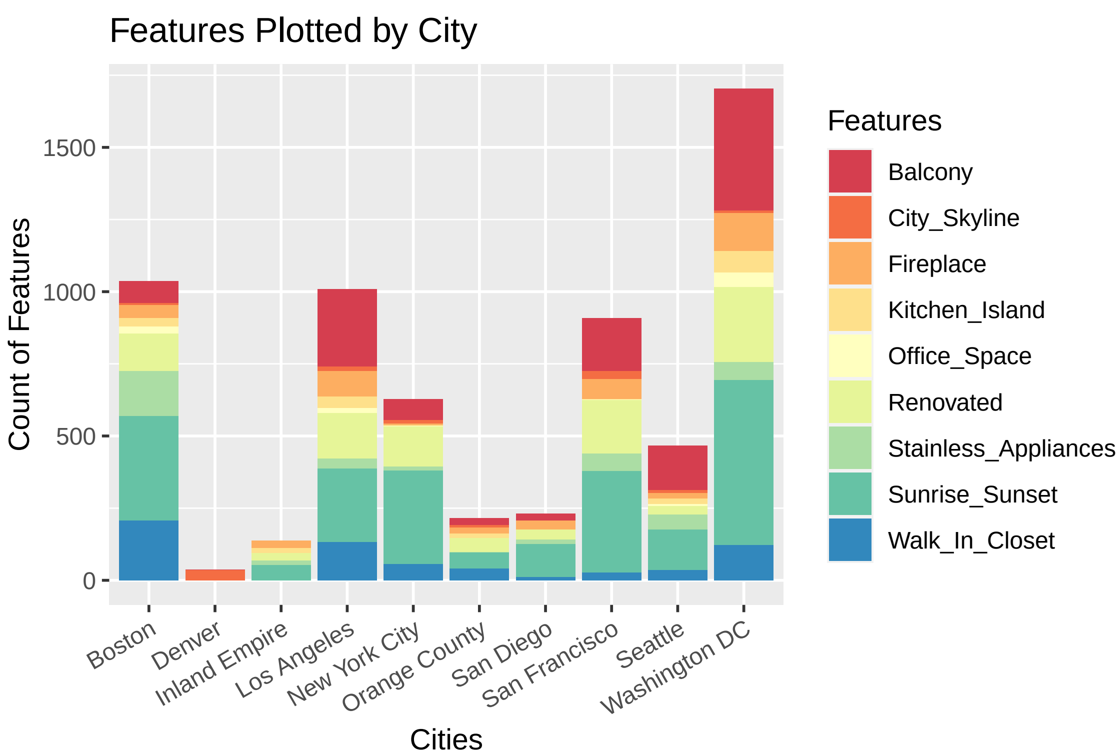
```
equity_apts_data$City_Skyline <- as.numeric(equity_apts_data$City_Skyline)

equity_apts_data_longform <- equity_apts_data %>%
  distinct(Unique_ID, .keep_all = TRUE) %>%
  pivot_longer(-c(zipcode, X, Price, Beds, Baths, sq.ft, Floor, Move_in_date, building_id,
                unit_id, URL, Day_Recorded, Amenity, Apartment.Name, Address, City, Units,
                Northern_Exposure, Southern_Exposure, Eastern_Exposure, Western_Exposure,
```

```
                   Days_Till_Available, Day_of_the_week_recorded, Unique_ID, Estimated_Vacancy,
                   Latitude, Longitude),
               names_to = "Features",
               values_to = "Yes_No",
               values_drop_na = TRUE)

equity_apts_data_longform <- subset(equity_apts_data_longform, Yes_No==1)

equity_apts_data_longform %>%
  ggplot(aes(x=City, fill=Features))+
  geom_bar()+
  labs(x="Cities", y="Count of Features", title = "Features Plotted by City")+
  theme(axis.text.x=element_text(hjust=1, angle=30))+
  scale_fill_brewer(palette="Spectral")
```




**Observations** We made the following observations from the graph:

- The only feature Denver has in the dataset is a city skyline view.
- Inland Empire apartments in the dataset have neither balconies nor walk-in-closets.
- Not many Boston apartments have balconies, but Boston has the highest count of walk-in closets out of all of the cities.
- The most common feature of apartments in all cities is a view of sunrise/sunset.
- Only a few New York City apartments have Balconies and walk-in closets, and no NYC apartments in the dataset have fireplaces.

## Conclusion

Going into this project, we knew that living in New York City was quite expensive but after our investigation, we learned that not only is it the most expensive out of all of Equity Residential's locations, but it also does not have nearly as many features as apartments in Washington DC and Boston. We also drew the conclusion that specific features do indeed raise the rent of the apartment and some of these included facing either the sunrise or sunset, the floor of an apartment, and also having a greater number of bedrooms than bathrooms. With this information, hopefully an individual looking to rent the cheapest apartment possible will know to avoid New York City and also pick an apartment with a fewer number of bedrooms.

Throughout this report and project, all three students (Reha, Niraj, and Sindhu) contributed equally to the efforts.