STATS 15 Final Project – NYC Apartment Rental Prices

By Megha Ravi, Aryan Gupta, Haniya Malik, and Saloni Panth

SECTION 1 - BACKGROUND

1.1 INSPIRATION:

New York is the most populous city in the United States. Since property prices are so steep, about 70% of residents rent properties, contrasting how about 65% of residents in the US own their properties. Additionally, as a major metropolis, New York has one of the highest rent prices. Our dataset focuses on the area of Manhattan, where the rent prices are even higher. The reason behind this is because there is less housing per capita than other major cities in the US. Therefore we decided we will investigate numerous factors that could affect the rent prices of the apartments in New York. We are interested in answering the question "what affects the rent price of apartments in Manhattan?".

RENT PRICE:

The rent price for an apartment is the monthly payment for the property and in New York it typically includes utilities such as water and gas. However, electricity and internet bills are usually the responsibility of the tenant.

1.2 DATASET BACKGROUND:

Where did we get our data? StreetEasy is the website from which the dataset was scraped, in June of 2016. StreetEasy primarily gets its data from two sources. The first is the New York City department of finance sales data. This data includes sales, prices, dates, and basic information about the building. The second source is through listings posted by brokers. This is where the majority of the data on StreetEasy comes from.

HOW LISTINGS CREATED ON STREETEASY:

Listings on StreetEasy are posted through your agent account by selecting your rentals and clicking create new rental. You then are able to enter the address and unit number, create the listing, and then add as much detail to it as possible.

1.3 NEIGHBORHOODS:

Upper East Side: This is considered one of the more posh neighborhoods of New York and it occupies 1.776 square miles.

Greenwich Village: This is known for its artistic residents and alternative culture, it occupies 0.29 square miles.

Midtown: Midtown is home to times square and is the city's primary central business district. It occupies 2.25 square miles.

Soho: Soho is a top shopping destination known for its fashionable crowds, it occupies 1 square mile.

Central Harlem: It is the heart of a historic African American neighborhood. It occupies 1.4 square miles.

Central Park South: This is home to central park, it occupies 1.31 square miles.

East Harlem: It is a showcase to Puerto Rican culture, it occupies 4.4 square miles.

Midtown East: It has easy access to anywhere in the city and is home to waterfront views. It occupies 1.15 square miles.

Battery Park City: It is set along the Hudson River. It occupies 0.21 square miles.

Flatiron: This is home to office high rises and tall apartment buildings. It occupies 0.39 square miles.

Lower East Side: In this neighborhood, older buildings and alleyways are mixed in with upscale apartments. It occupies 0.84 square miles.

East Village: This is home to many bars and performance spaces. It occupies 0.77 square miles.

Gramercy Park: This is a well to do residential area with historic bars and fashionable restaurants. It occupies 1.13 square miles.

Financial District: This is home to wall street and skyscrapers. It occupies 0.45 square miles.

Chelsea: It has mainly low rise apartment buildings and is home to attractions such as the New York highline. It occupies 0.77 square miles.

Inwood: Inwood is mainly a mix of commercial streets. It occupies 0.91 square miles.

Tribeca: It is known for old industrial buildings and historical commercial buildings. It occupies 0.33 square miles.

Washington Heights: It is New York's little Dominican Republic and is known for being safe. It occupies 1.66 square miles.

Chinatown: It is one of the oldest Chinese enclaves. It occupies 0.77 square miles.

Roosevelt Island: Relative to New York this area has a high crime rate. It occupies 1.78 square miles.

West Village: Fashionable neighborhood with designer boutiques and restaurants, it occupies 0.41 square miles.

Midtown West: This is also known as Hell's kitchen. It occupies 0.84 square miles.

Midtown South: It is home to the garment district. It occupies 0.37 square miles.

Hamilton Heights: It contains the historic district of Sugar Hill. It occupies 0.42 square miles.

Stuyvesant Town/PCV: It is a large post World War ll residential area. It occupies 0.23 square miles.

Morningside Heights: It is home to several religious institutions. It occupies 0.47 square miles.

Little Italy: It is known for its large Italian population. It occupies roughly 2 square miles.

Nolita: It is known for its designer jewelry shops and home design stores. It occupies 4 blocks.

Long Island City: It is known for its high rises with nice views to Manhattan. It occupies 3.33 square miles.

Upper West Side: It is home to performing arts institutions. It occupies 1.93 square miles.

Lower West Side: It has its own village vibe with lots of colorful art. It occupies 2.8 square miles.

1.4 VARIABLES OF THE DATASET:

EXPLANATORY VARIABLES:

Bedrooms: The amount of rooms in an apartment.

Bathrooms: The amount of bathrooms in an apartment.

Size_sqft: The size per square feet of an apartment.

Min_to_subway: The distance in minutes to the nearest subway station.

Floor: The floor that the apartment is on.

Building_age_yrs: How old the building the apartment is in is in years.

No_fee: Whether or not there is a broker's fee for the apartment.

Has_roofdeck: Whether or not there is a patio outside the building.

Has_washer_dryer: Whether or not the apartment includes a washer and dryer.

Has_doorman: Whether or not the building has a doorman.

Has_elevator: Whether or not there is an elevator in the building.

Has_dishwasher: Whether or not there is a dishwasher in the apartment.

Has_patio: Whether or not there is a patio in the apartment.

Has_gym: Whether or not the apartment building has a gym.

neighborhood: which neighborhood the apartment is located in out of the 31 unique neighborhoods in our dataset.

RESPONSE VARIABLE:

Rent: The rent per month for the particular apartment.

METRIC:

Subway stations play a large role in a metropolitan city like New York. Due to congestion and traffic many people do not own cars and rely on the subway as a primary mode of transportation. There are currently 151 subway stations in Manhattan, and our dataset records the distance in minutes to the nearest subway from the particular property. We use this variable as a metric to create scores for apartments and they are classified as "close" or "far".

SECTION 2 - DATA LOADING AND CLEANUP

2.1 - Loading, joining and cleaning our data

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
new_york <- read.csv("manhattan.csv")

rental_dataset1 <- new_york %>%
  select(-rental_id, -borough)

head(rental_dataset1)
```

```
##    rent bedrooms bathrooms size_sqft min_to_subway floor building_age_yrs
## 1  2550        0         1       480             9     2               17
## 2 11500        2         2      2000             4     1               96
## 3  4500        1         1       916             2    51               29
## 4  4795        1         1       975             3     8               31
## 5 17500        2         2      4800             3     4              136
## 6  3800        3         2      1100             3     5              101
##   no_fee has_roofdeck has_washer_dryer has_doorman has_elevator has_dishwasher
```

```
## 1      1            1                0           0            1              1
## 2      0            0                0           0            0              0
## 3      0            1                0           1            1              1
## 4      0            0                0           1            1              1
## 5      0            0                0           1            1              1
## 6      0            0                0           0            0              0
##   has_patio has_gym      neighborhood
## 1         0       1   Upper East Side
## 2         0       0 Greenwich Village
## 3         0       0           Midtown
## 4         0       1 Greenwich Village
## 5         0       1              Soho
## 6         0       0    Central Harlem
```

The first thing to do is check if there are an NA values in our data set.

```
rental_dataset1 %>%
  summarise_all(funs(sum(is.na(.))))
```

```
## Warning: `funs()` was deprecated in dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
```

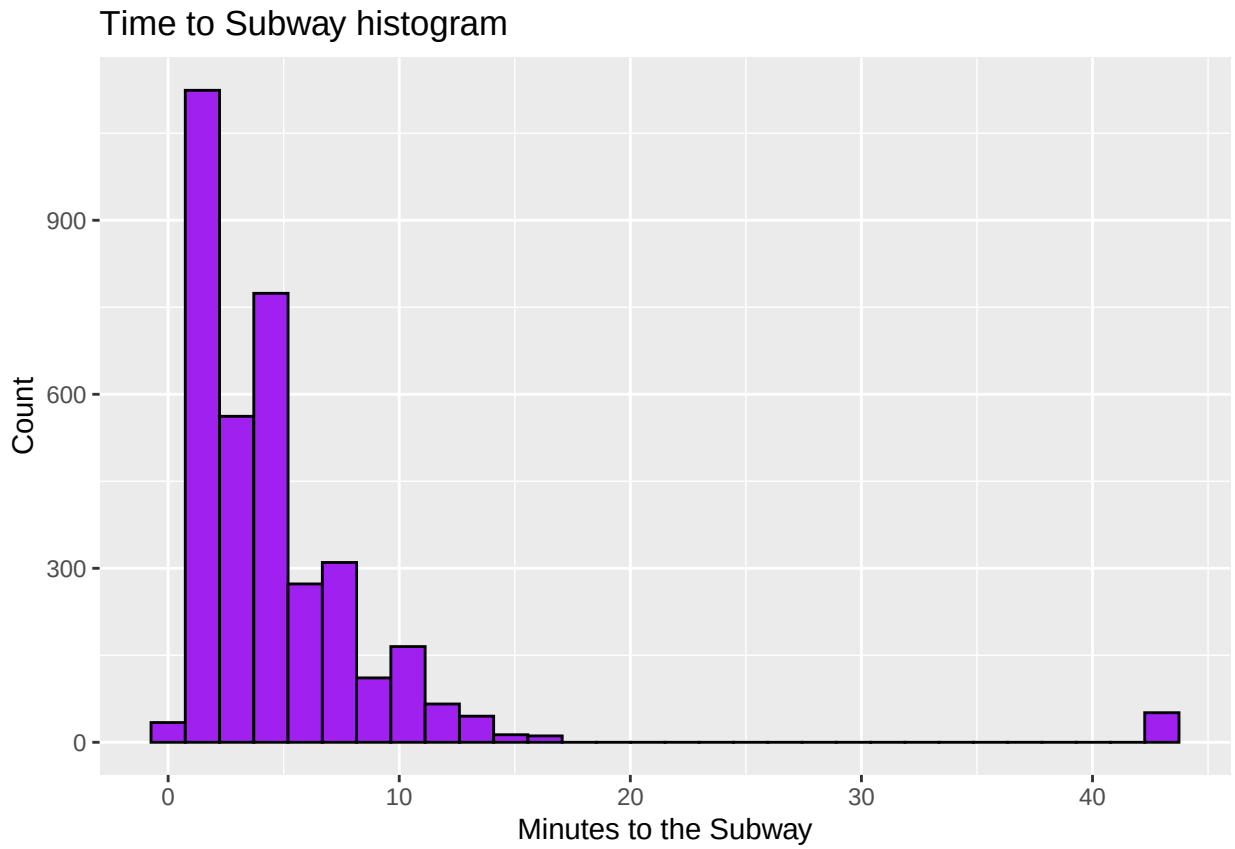
```
##   rent bedrooms bathrooms size_sqft min_to_subway floor building_age_yrs no_fee
## 1    0        0         0         0             0     0                0      0
##   has_roofdeck has_washer_dryer has_doorman has_elevator has_dishwasher
## 1            0                0           0            0              0
##   has_patio has_gym neighborhood
## 1         0       0            0
```

After summarizing the dataset, we can see that there are no NA values which means we can proceed.

2.2 - FEATURE ENGINEERING

METRIC - Minutes to the Subway

```
rental_dataset1 %>%
  ggplot(aes(x = min_to_subway)) +
  geom_histogram(fill='purple', color='black', bins = 30) +
xlab("Minutes to the Subway") + ylab("Count") + ggtitle("Time to Subway histogram")
```



The distribution is slightly skewed to the right, with most times being between 0 and 10 minutes to the nearest subway. Interestingly enough, let's look at the times that are 40 minutes or greater to the subway which could constitute as outliers.

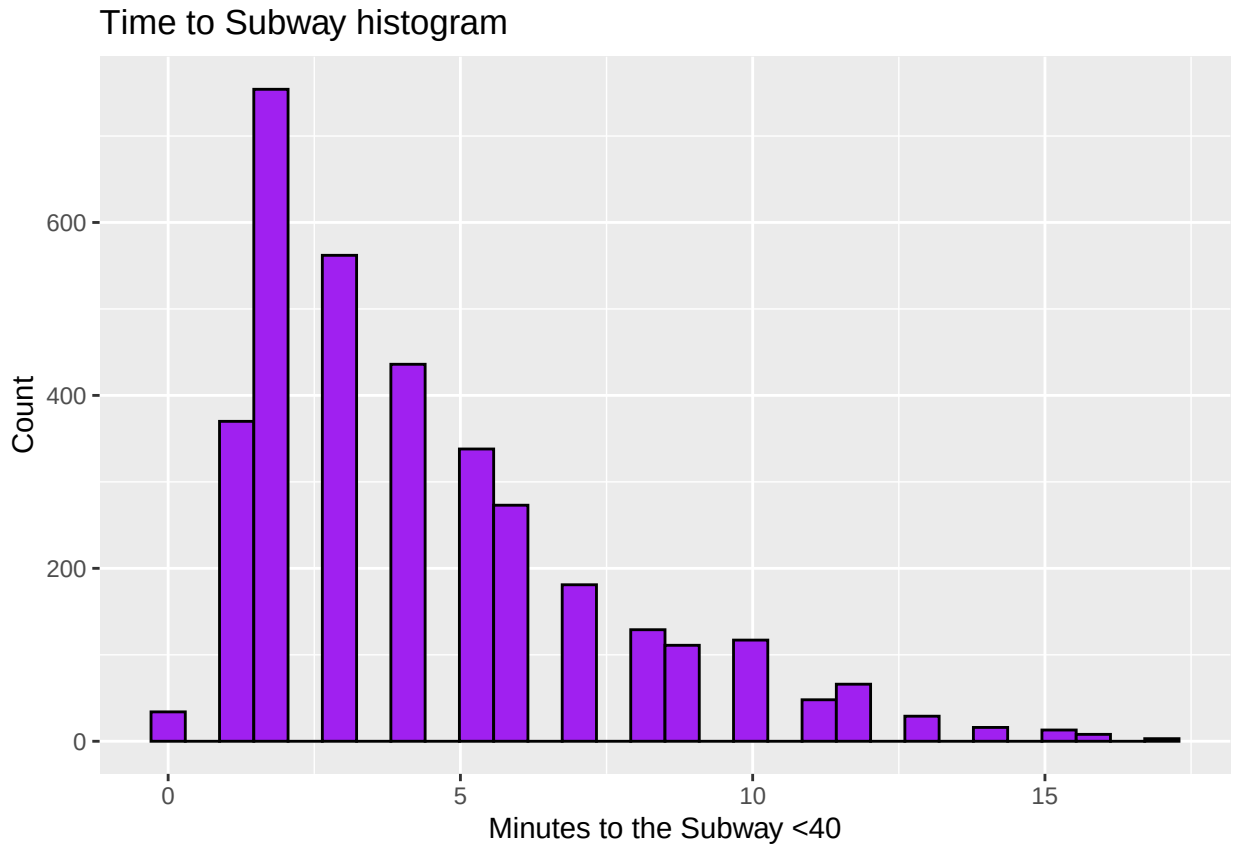
```
rental_dataset1 %>%
  filter(min_to_subway>40) %>%
  head()
```

```
##    rent bedrooms bathrooms size_sqft min_to_subway floor building_age_yrs
## 1  2950        1         1       550            43    17               14
## 2  3625        1         1       650            43    19               14
## 3  4890        1         1       815            43     7                8
## 4  4995        2         2      1153            43    16                9
## 5 13750        3         3      1920            43     8                8
## 6 13500        3         3      1868            43    32                8
##   no_fee has_roofdeck has_washer_dryer has_doorman has_elevator has_dishwasher
## 1      1            1                0           1            1              0
## 2      1            0                0           0            0              0
## 3      1            0                0           0            1              0
## 4      0            0                0           0            0              0
## 5      1            0                0           0            0              0
## 6      0            0                0           0            0              0
##   has_patio has_gym     neighborhood
## 1         0       0  Upper East Side
## 2         0       0  Upper East Side
## 3         0       0  Upper West Side
## 4         0       0 Roosevelt Island
```

```
## 5          0        0  Upper West Side
## 6          1        0  Upper West Side
```

Due to prior knowledge and after checking StreetEasy, we decided to drop these values as it does not make sense for someone to walk 40 or more minutes to the subway from an apartment in Manhattan. In these 5 neighborhoods, there are generally nearby subway stations that takes less than 40 minutes to walk to from an apartment. Even after dropping these rows, we will have enough values in our dataset. Let's now look at the graph without the 40 or more minute values to the subway present.

```
rental_dataset1 <- rental_dataset1 %>%
  filter(min_to_subway<40)

rental_dataset1 %>%
    ggplot(aes(x = min_to_subway)) +
  geom_histogram(fill='purple', color='black', bins = 30) +
xlab("Minutes to the Subway <40") + ylab("Count") + ggtitle("Time to Subway histogram")
```



Furthermore, in order for the time to the subway from an apartment to be significant in our analysis, we decided to group the time into two categories: "close" and "far". To group the time into close/far, times below the mean time can be described as "close" and times above the mean time can be described as "far".

```
mean_min_to_subway = mean(rental_dataset1$min_to_subway)
mean_min_to_subway
```

```
## [1] 4.414851
```

```
rental_dataset1 <- rental_dataset1 %>%
  add_column(time_status= NA)

for(i in 1:nrow(rental_dataset1)){
  if(rental_dataset1[i, 5] < mean_min_to_subway){
   rental_dataset1[i, 17] = "close"
  } else{
       rental_dataset1[i, 17] = "far"
  }

}

rental_dataset1 %>%
  select(neighborhood, min_to_subway, time_status) %>%
  head()
```

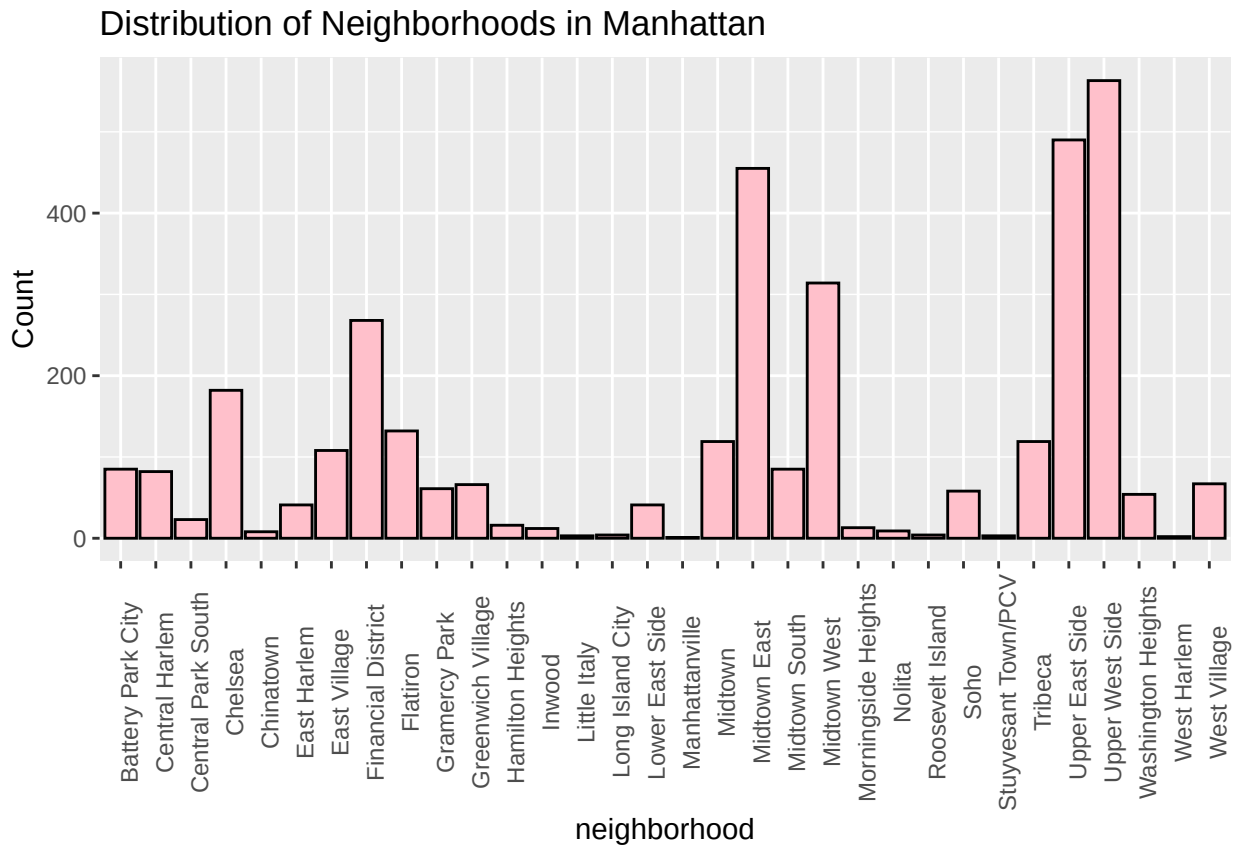
```
##        neighborhood min_to_subway time_status
## 1   Upper East Side             9         far
## 2 Greenwich Village             4       close
## 3           Midtown             2       close
## 4 Greenwich Village             3       close
## 5              Soho             3       close
## 6    Central Harlem             3       close
```

This method makes sense as with prior knowledge of NYC, times greater than around 5 minutes to the nearest subway station can be seen as far, while times less than around 5 minutes can be seen as close.

Neighborhoods

```
library(tidyverse)
rental_dataset1 %>%
  ggplot(aes(x = neighborhood)) +
  geom_bar(fill = 'pink', color = 'black', bins = 30) +
  xlab("neighborhood") + ylab("Count") +
  ggtitle("Distribution of Neighborhoods in Manhattan") +
  theme(axis.text.x = element_text(angle = 90))
```

```
## Warning: Ignoring unknown parameters: bins
```



As we can see, there are many neighborhoods in Manhattan; in our dataset, there are rent prices collected from 32 neighborhoods. However, because there are so many neighborhoods, it may be difficult to clearly analyze the impact of neighborhood on rent price, especially since some neighborhoods do not have much data in comparison to others. Thus, we will pick and focus on a few neighborhoods with relatively adequate amount of data and that are geographically in various places in Manhattan: Central Harlem, Upper West Side, Chelsea, Financial District, Midtown West, and the Upper East Side.

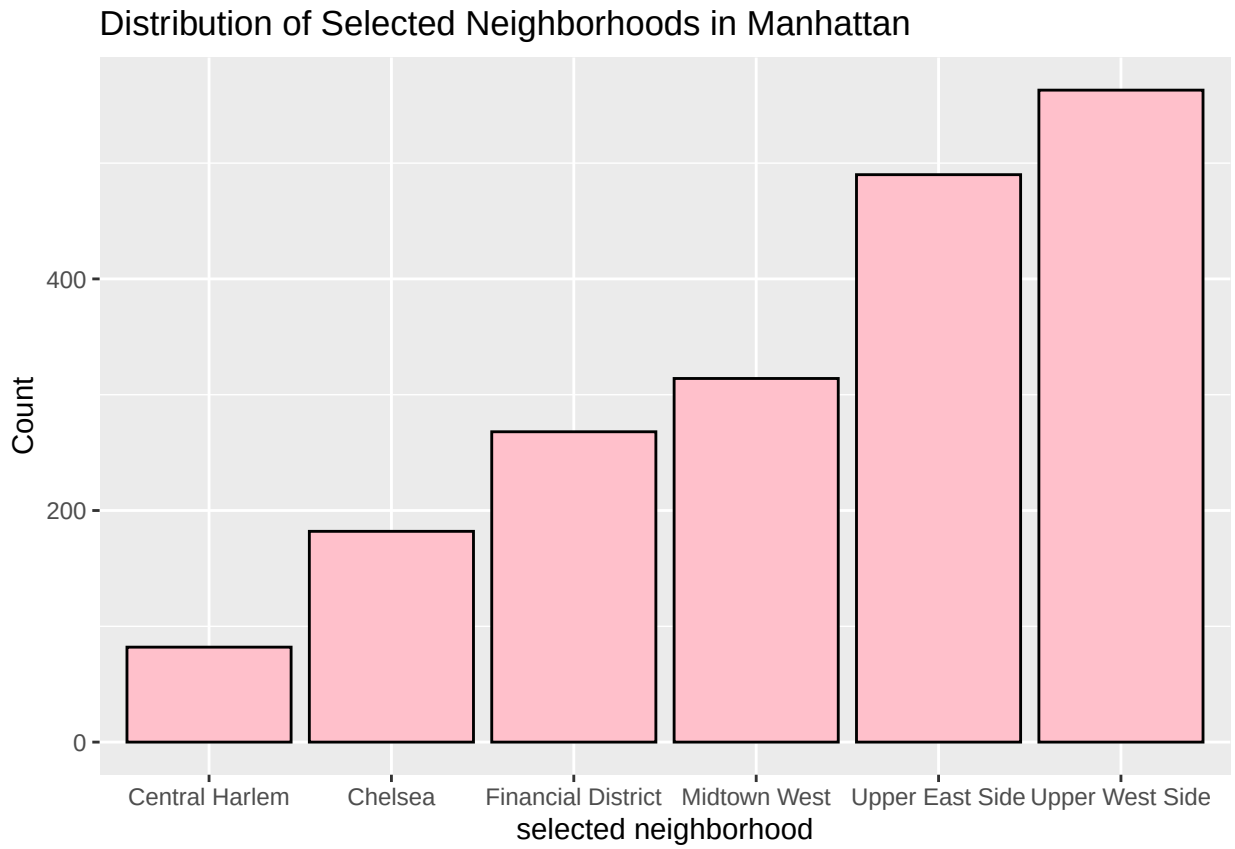
To make our future analysis simpler, we will add a column to our dataset that specifies the six selected neighborhoods and adds NA values for the other neighborhoods since we do not want to include them.

```
rental_dataset1$neigh_selected[rental_dataset1$neighborhood == c("Central Harlem")]= "Central Harlem"
rental_dataset1$neigh_selected[rental_dataset1$neighborhood == c("Upper West Side")] = "Upper West Side
rental_dataset1$neigh_selected[rental_dataset1$neighborhood == c("Chelsea")] = "Chelsea"
rental_dataset1$neigh_selected[rental_dataset1$neighborhood == c("Financial District")] = "Financial Di
rental_dataset1$neigh_selected[rental_dataset1$neighborhood == c("Midtown West")] = "Midtown West"
rental_dataset1$neigh_selected[rental_dataset1$neighborhood == c("Upper East Side")] = "Upper East Side

rental_dataset1 %>%
  select(neighborhood, neigh_selected) %>%
  head()
```

```
##        neighborhood  neigh_selected
## 1   Upper East Side Upper East Side
## 2 Greenwich Village            <NA>
## 3           Midtown            <NA>
## 4 Greenwich Village            <NA>
## 5              Soho            <NA>
```

```
## 6    Central Harlem  Central Harlem
```

```
rental_dataset1 %>%
 filter(!is.na(neigh_selected)) %>%
  ggplot(aes(x = neigh_selected)) +
  geom_bar(fill = 'pink', color = 'black') +
  xlab("selected neighborhood") + ylab("Count") +
  ggtitle("Distribution of Selected Neighborhoods in Manhattan")
```



In addition to neighborhood, we also think that it would be interesting to see if the geographical groupings of Upper, Midtown, and Downtown Manhattan have an effect on the rent. We will group the different neighborhoods based on their location in Manhattan.

```
  unique(rental_dataset1$neighborhood)
```

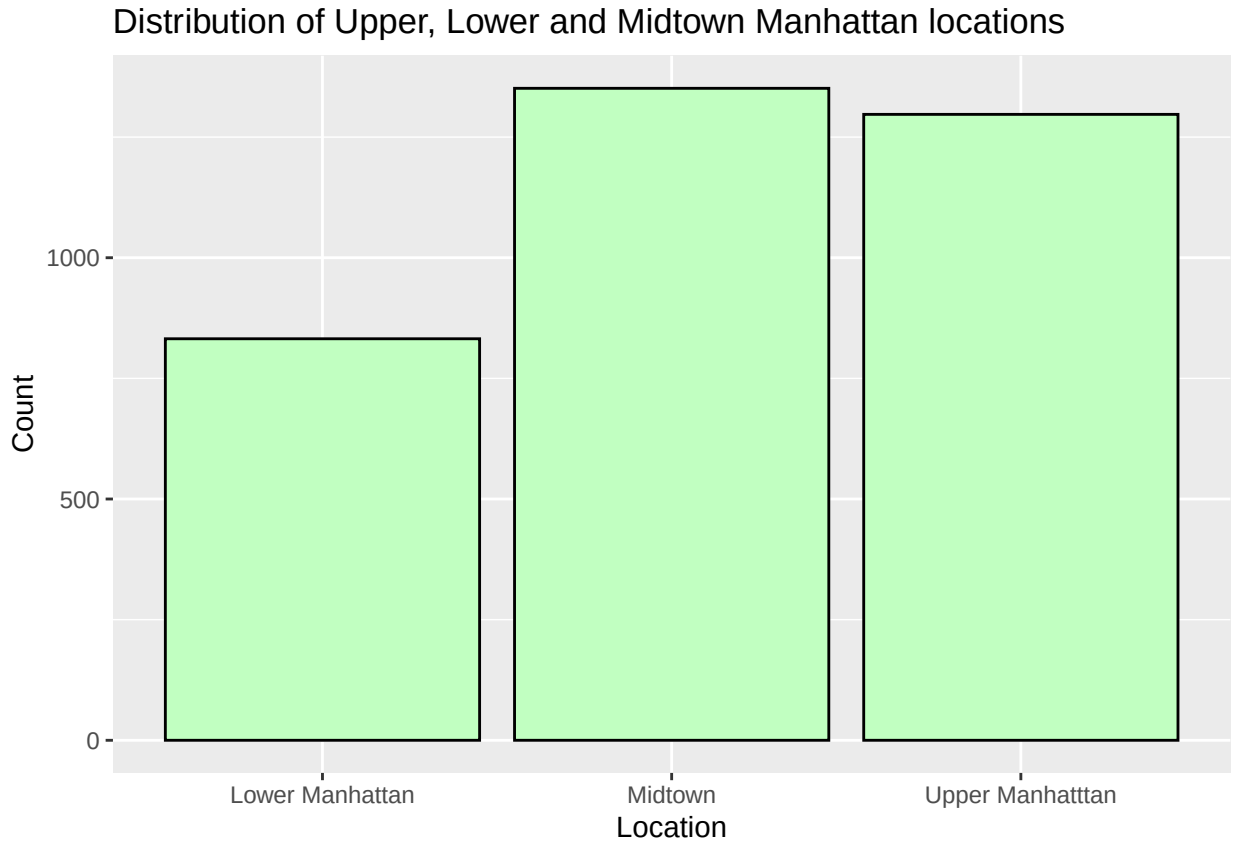
```
##  [1] "Upper East Side"     "Greenwich Village"   "Midtown"
##  [4] "Soho"                "Central Harlem"      "Midtown East"
##  [7] "Battery Park City"   "Flatiron"            "East Village"
## [10] "Midtown West"        "Upper West Side"     "Lower East Side"
## [13] "Tribeca"             "Gramercy Park"       "East Harlem"
## [16] "West Village"        "Central Park South"  "Chelsea"
## [19] "Financial District"  "Inwood"              "Midtown South"
## [22] "Washington Heights"  "Chinatown"           "Hamilton Heights"
## [25] "Stuyvesant Town/PCV" "Morningside Heights" "Little Italy"
## [28] "Roosevelt Island"    "Nolita"              "West Harlem"
## [31] "Long Island City"    "Manhattanville"
```

We will be grouping the different neighborhoods in three sections depending on where they are located in Manhattan. We chose to include all the neighborhoods except Roosevelt Island as it is considered part of both Lower and Upper Manhattan.

```
#Neighborhoods in Upper Manhattan
rental_dataset1$location[rental_dataset1$neighborhood == c("Upper East Side")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Manhattanville")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Upper West Side")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Hamilton Heights")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("West Harlem")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Central Park South")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Morningside Heights")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Central Harlem")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Inwood")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Upper East Side")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Washington Heights")] = "Upper Manhatttan"
rental_dataset1$location[rental_dataset1$neighborhood == c("East Harlem")] = "Upper Manhatttan"


#Neighborhoods in Midtown
rental_dataset1$location[rental_dataset1$neighborhood == c("Midtown")] = "Midtown"
rental_dataset1$location[rental_dataset1$neighborhood == c("Midtown South")] = "Midtown"
rental_dataset1$location[rental_dataset1$neighborhood == c("Midtown East")] = "Midtown"
rental_dataset1$location[rental_dataset1$neighborhood == c("Flatiron")] = "Midtown"
rental_dataset1$location[rental_dataset1$neighborhood == c("Stuyvesant Town/PCV")] = "Midtown"
rental_dataset1$location[rental_dataset1$neighborhood == c("Midtown West")] = "Midtown"
rental_dataset1$location[rental_dataset1$neighborhood == c("Gramercy Park")] = "Midtown"
rental_dataset1$location[rental_dataset1$neighborhood == c("Chelsea")] = "Midtown"

#Neighborhoods in Lower Manhattan
rental_dataset1$location[rental_dataset1$neighborhood == c("Financial District")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Chinatown")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Little Italy")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Nolita")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Soho")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Lower East Side")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Battery Park City")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Tribeca")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("Greenwich Village")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("East Village")] = "Lower Manhattan"
rental_dataset1$location[rental_dataset1$neighborhood == c("West Village")] = "Lower Manhattan"
```

Let's plot the distribution of the location of Upper, Lower, and Midtown Manhattan.

```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
ggplot(aes(x = location)) +
  geom_bar(fill = 'darkseagreen1', color = 'black') +
  xlab("Location") + ylab("Count") +
  ggtitle("Distribution of Upper, Lower and Midtown Manhattan locations")
```

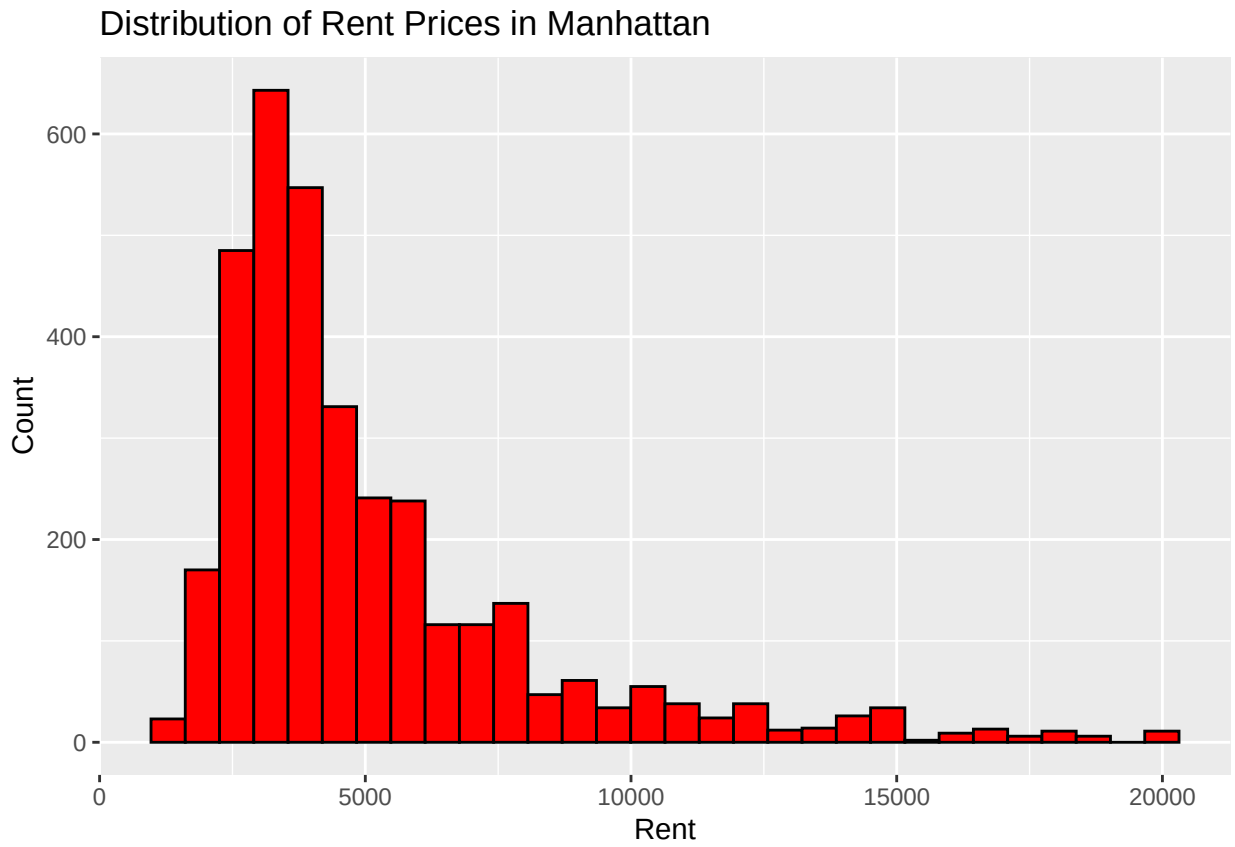


There are about the same number of apartments in Midtown and Upper Manhattan, while about 3/4 the apartments in Lower Manhattan compared to Midtown and Upper Manhattan.

SECTION 3 - EXPLORATORY DATA ANALYSIS

3.1: UNIVARIATE PLOTS RENT Let's plot the distribution of the rent price to check for outliers.

```
rental_dataset1 %>%
  ggplot(aes(x = rent)) +
  geom_histogram(fill = 'red', color = 'black', bins = 30) +
  xlab("Rent") + ylab("Count") + ggtitle("Distribution of Rent Prices in Manhattan")
```



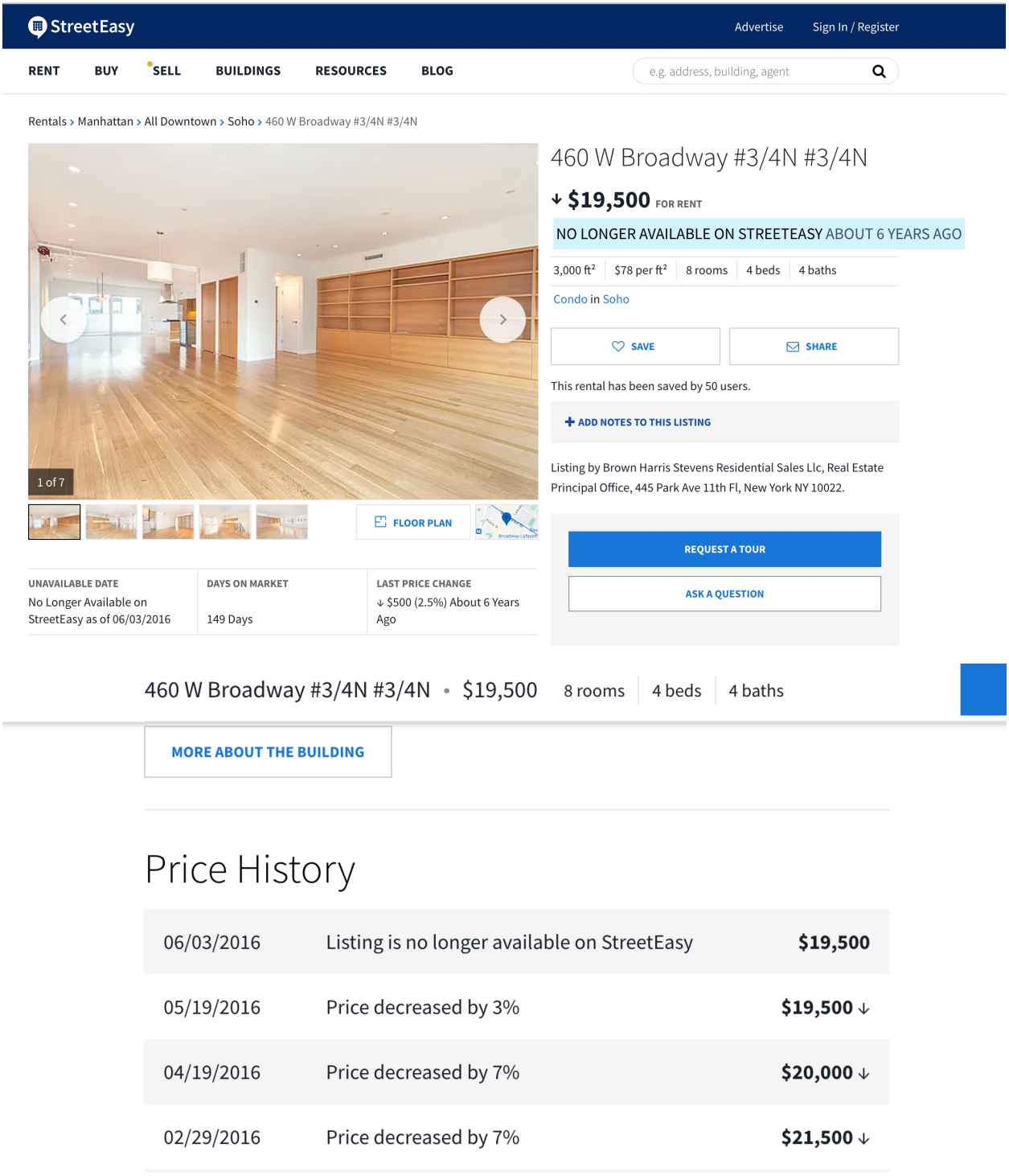
As we can see, the distribution of the rent prices is skewed to the right. It looks like there are possible outliers around 20,000 dollars. To be more precise, let's determine and examine the possible outliers by calculating the standard deviation, z-scores and mean of the rent prices. After filtering by $|z| > 2$, we can investigate these high values with similar apartment listings from the same neighborhood in 2016 on StreetEasy.

```
rent_mean <- mean(rental_dataset1$rent)
rent_sd <- sd(rental_dataset1$rent)

rental_dataset1 <- rental_dataset1 %>%
  mutate(rent_z_score = (rent-rent_mean)/rent_sd)

rental_dataset1 %>%
  filter(abs(rent_z_score) > 2) %>%
  select(rent, neighborhood, bedrooms, bathrooms, size_sqft) %>%
  arrange(desc(rent)) %>%
  head()
```

```
##    rent    neighborhood bedrooms bathrooms size_sqft
## 1 20000            Soho        3         3      2600
## 2 20000         Tribeca        2         2      2200
## 3 20000 Upper East Side        4         4      2160
## 4 20000        Flatiron        3         3      2783
## 5 20000        Flatiron        3         2      2376
## 6 20000    Midtown East        3         3      2500
```

When we see these possible outliers that are skewing the data,

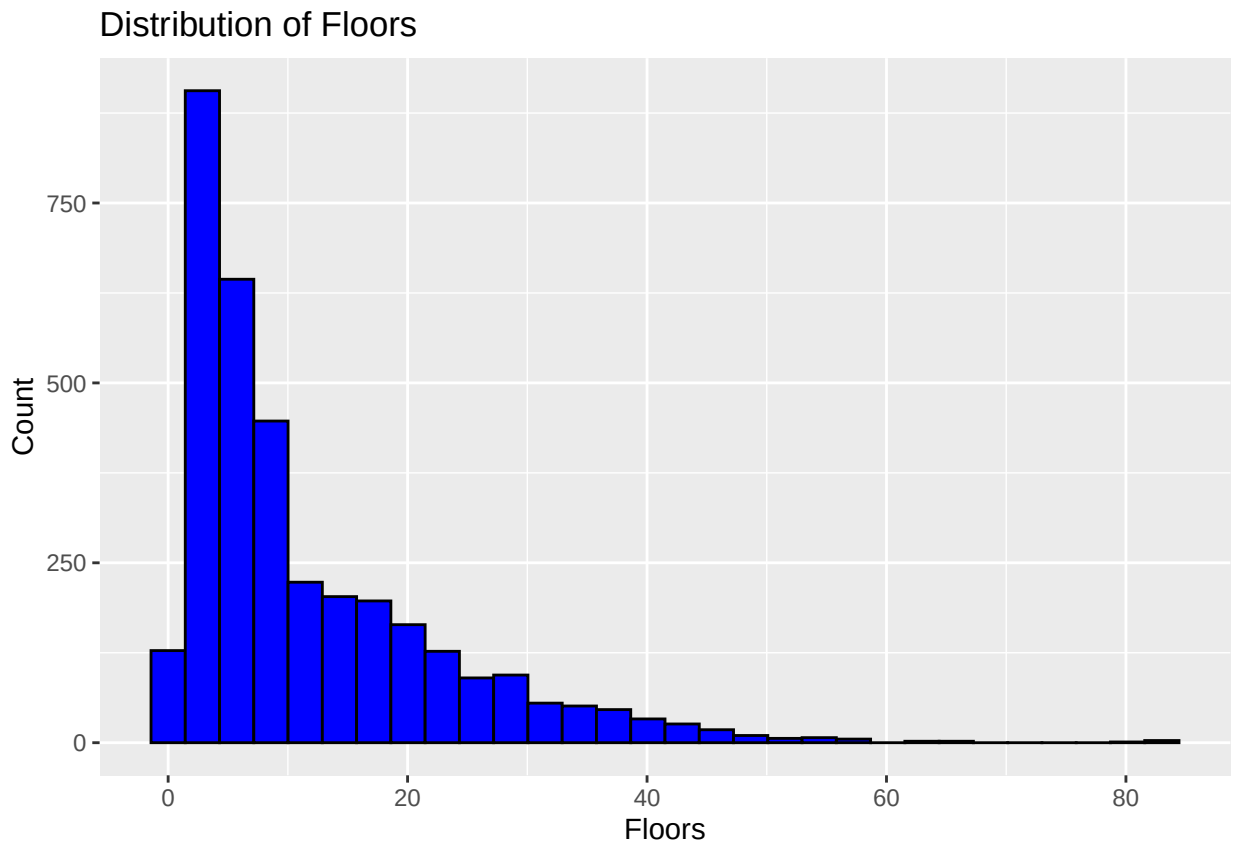
that the highest rent price is 20,000 per month, in neighborhoods like Soho and Tribeca. Let us look at similar apartment listings from StreetEasy in 2016 to determine whether these prices were legitimate at the time.



In the Price History, we can see that the apartment was once 20,000 dollars in 2016. After investigating this rent listing in the Soho neighborhood, we determine that the high rent price of 20,000 dollars is legitimate and that we can keep high rent prices such as these in our dataset.

Floors

```
ggplot(rental_dataset1, aes(x = floor)) +
  geom_histogram(color = "black", fill = "blue", bins = 30) +
  xlab("Floors") +
  ylab("Count") + ggtitle("Distribution of Floors")
```

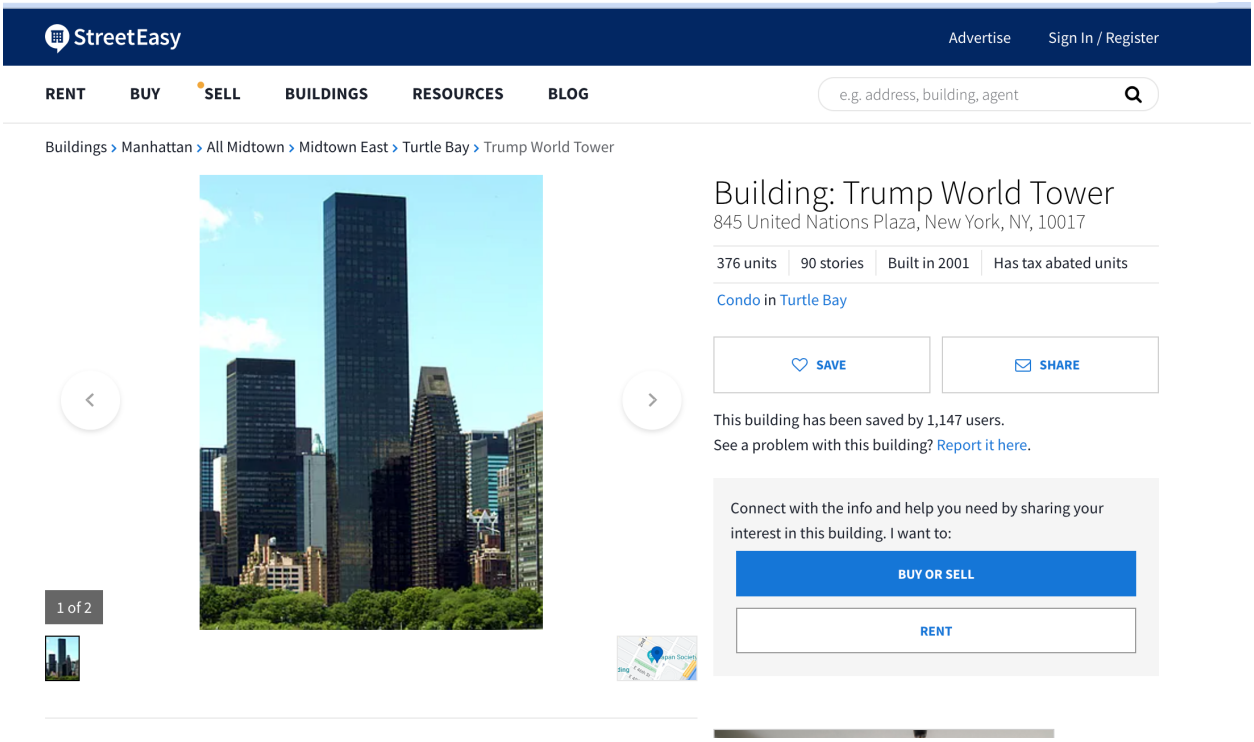


In the distribution of floors, which is skewed to the right, we can see that there are possible outliers around 75 floors. Let's investigate this.

```
rental_dataset1 %>%
  select(neighborhood,floor) %>%
  filter(floor > 75) %>%
  arrange(desc(floor))
```

```
##      neighborhood floor
## 1    Midtown East    83
## 2    Midtown East    83
## 3    Midtown East    83
## 4 Upper West Side    81
```

To determine if these high values of floors are legitimate, let's look on StreetEasy to verify.



As we can see, apartments can realistically have high floors such as floor 83. Thus, we can keep the high floor values in our dataset.

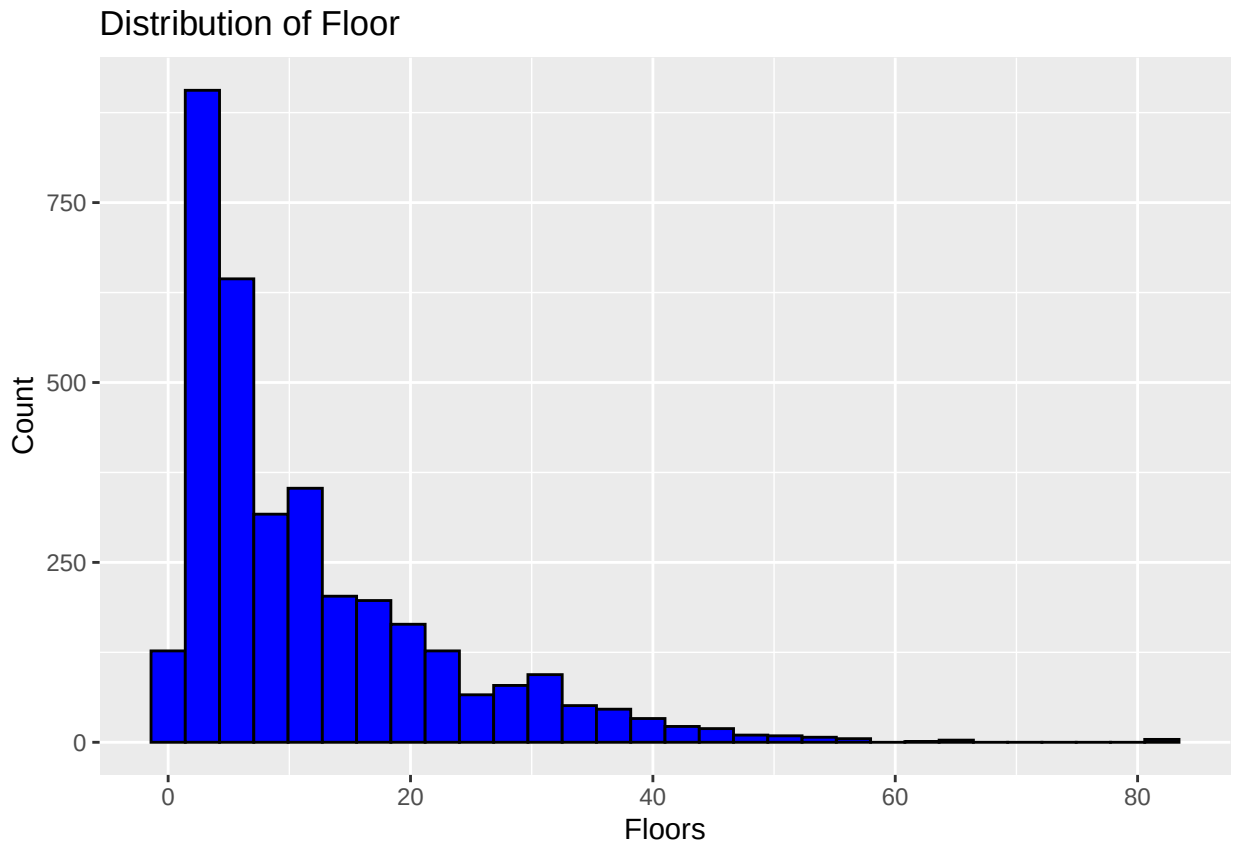
We also see a possible outlier around a floor of 0.

```
rental_dataset1 %>%
  select(neighborhood,floor) %>%
  filter(floor < 1)
```

```
##     neighborhood floor
## 1 Central Harlem     0
```

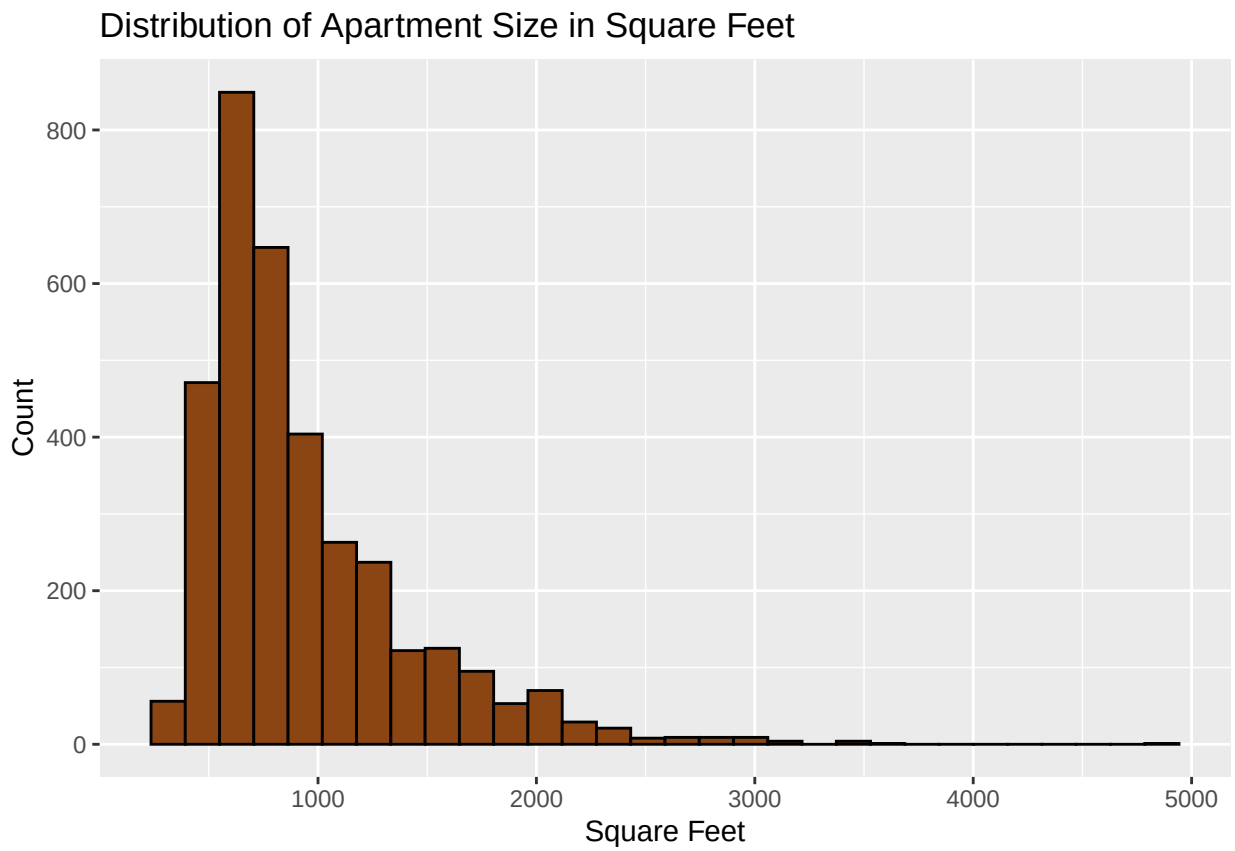
We decide to take out this value with floor 0 because the label of floor "0" is uncommon and we do not have enough data from StreetEasy to verify that this apartment could be in the basement of a building or below ground.

```
rental_dataset1 <- rental_dataset1 %>%
  filter(floor > 0)

ggplot(rental_dataset1, aes(x = floor)) +
  geom_histogram(color = "black", fill = "blue", bins = 30)+ xlab("Floors") +
  ylab("Count") + ggtitle("Distribution of Floor")
```



** Sqft**

```
library(ggplot2)

ggplot(rental_dataset1, aes(x = size_sqft)) +
  geom_histogram(color = "black", fill = "chocolate4", bins = 30) +
  xlab("Square Feet") +
  ylab("Count") + ggtitle("Distribution of Apartment Size in Square Feet")
```

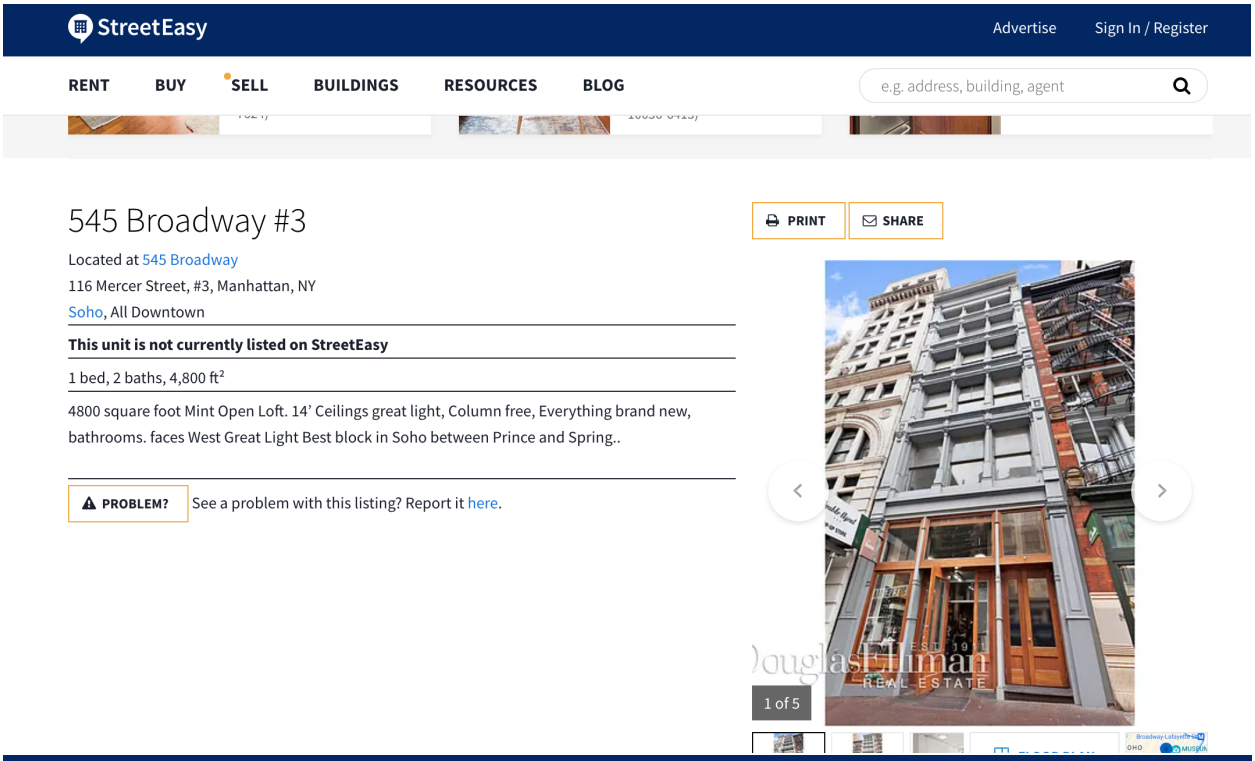


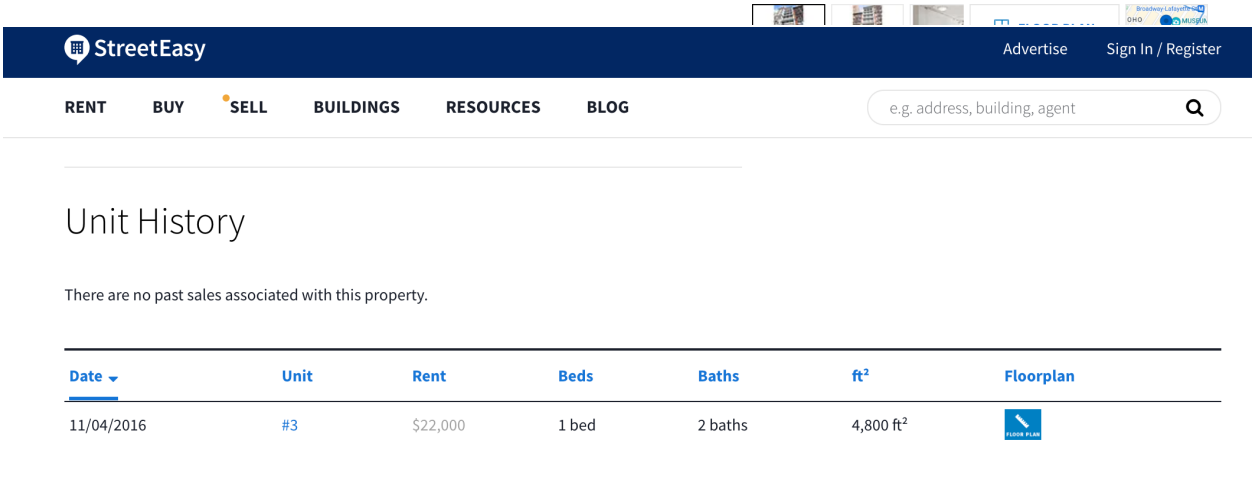
The distribution of the size in square feet is skewed right. We can see that there is a possible outlier around 3500 square feet; let's investigate this.

```
rental_dataset1 %>%
  filter(size_sqft > 3500) %>%
  select(neighborhood, size_sqft, rent, bedrooms)
```

```
##   neighborhood size_sqft  rent bedrooms
## 1         Soho      4800 17500        2
## 2 Midtown West      3680 17900        3
```

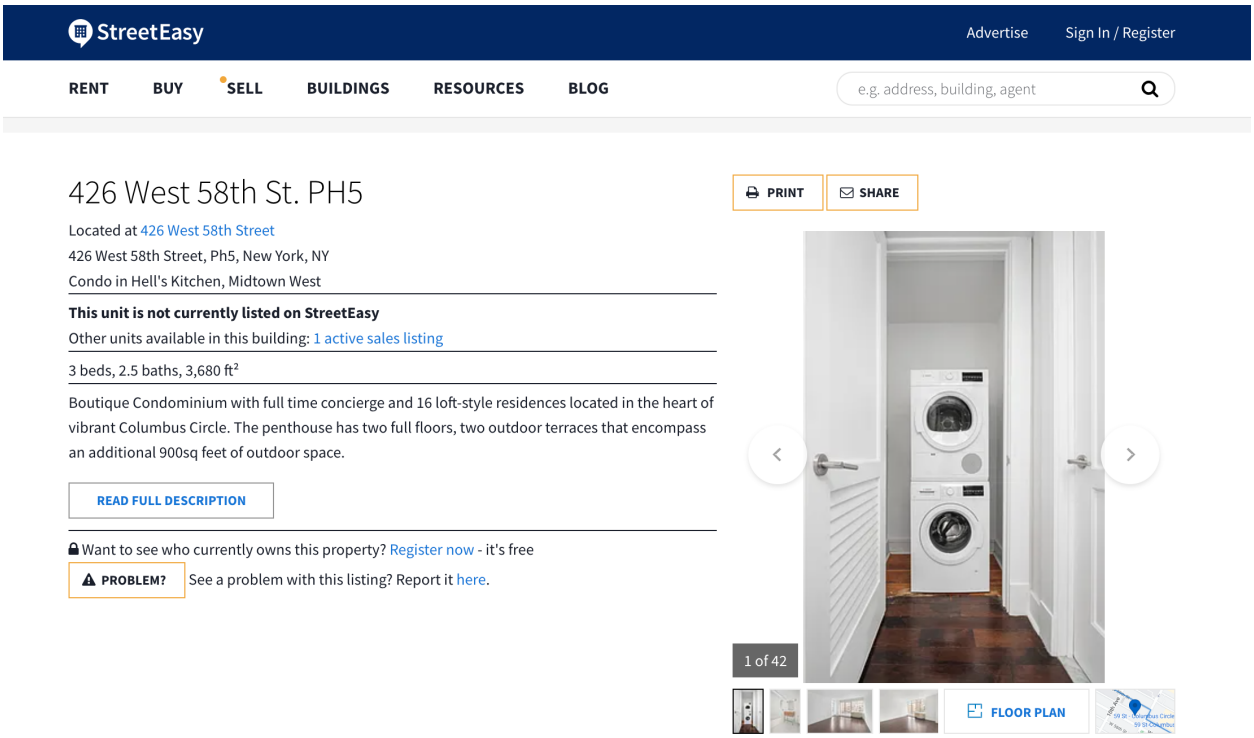
As we can see, a possible outlier is 4800 square feet in the Soho neighborhood. Let us find an apartment in the Soho neighborhood with a relatively similar size in square feet to determine if this is legitimate.





The rent price the apartment was rented at was 22,000 dollars, while the rent price in our dataset at the time the listing was scraped is set at 17,500 dollars; the 4,500 dollars difference is substantial. Because StreetEasy does not have data on the previous rent price of this listing at times earlier in the year closer to June, we decide to drop this value as the rent price difference is quite substantial and cannot be thoroughly verified.

Let's investigate the other possible outlier of the Midtown West Apartment.



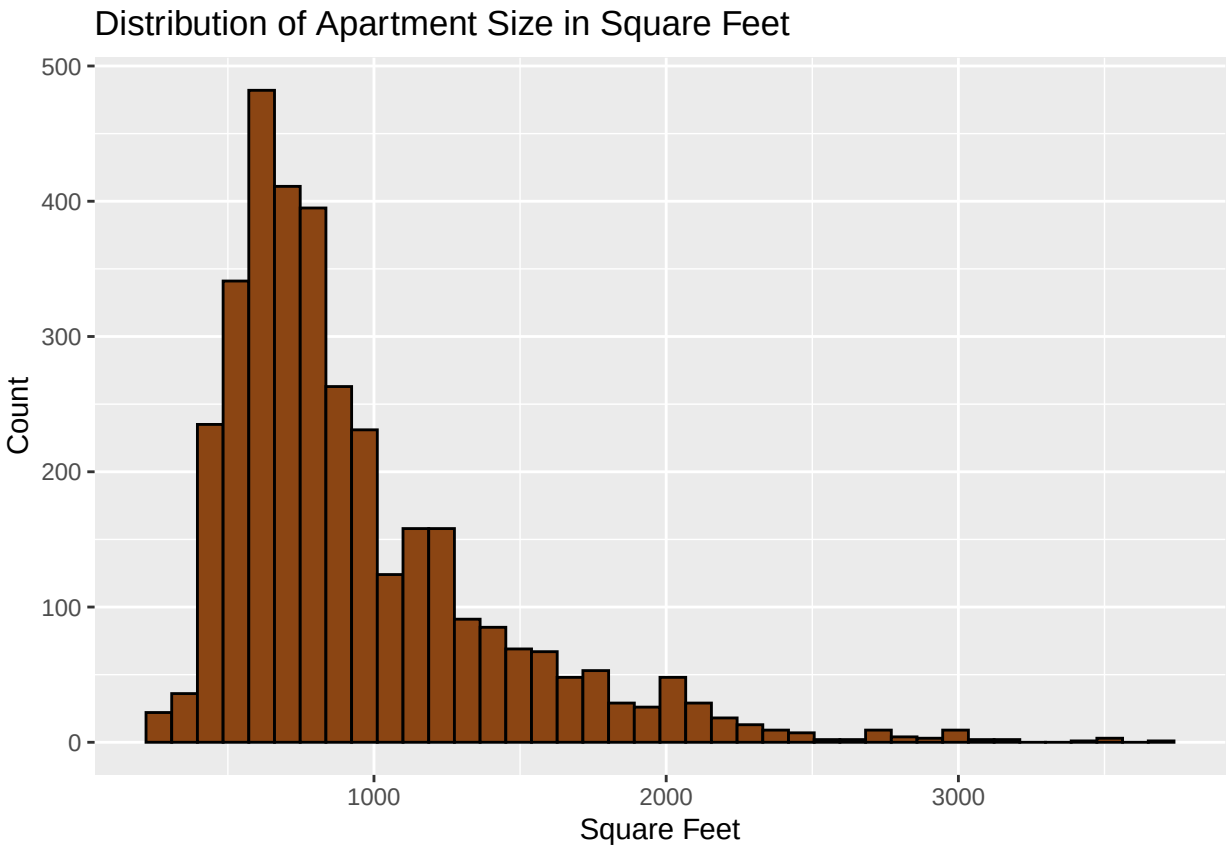
Unit History

Date	Unit	Price	Listing status	Beds	Baths	ft²	Floorplan
06/06/2022	#PH5	$6,995,000	No Longer Available on StreetEasy	3 beds	2.5 baths	3,680 ft²	

Date	Unit	Rent	Beds	Baths	ft²	Floorplan
06/29/2016	#PH5	$17,900	3 beds	2.5 baths	3,680 ft²	
04/12/2011	#PH5	$17,500	3 beds	2.5 baths	3,680 ft²	
03/27/2010	#PH5	$19,000	3 beds	2.5 baths	3,680 ft²	

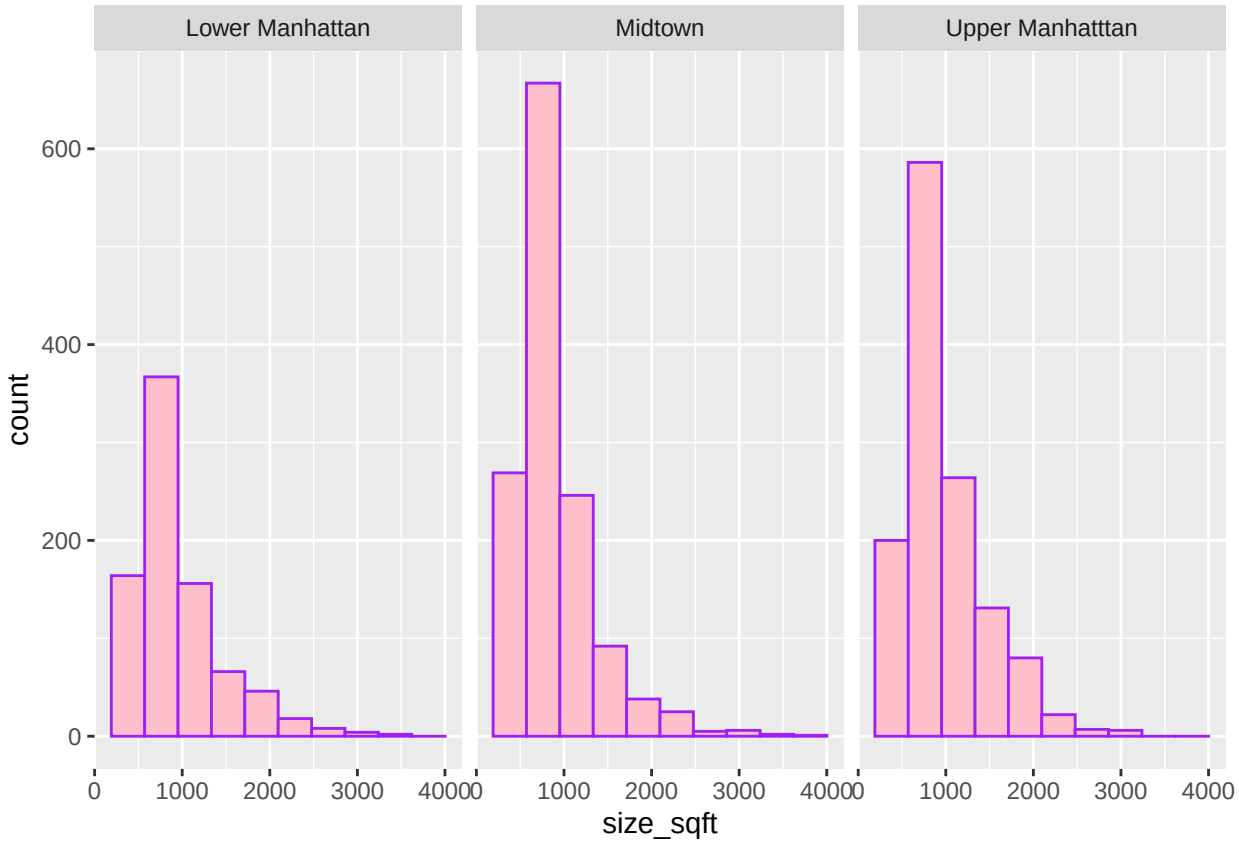
The rent price and square feet size are the same, so we determine that this high value of 3680 square feet is legitimate and that we can keep it in our dataset. As we replot the distribution of size in square feet, the distribution looks much more reasonable.

```
rental_dataset1 <- rental_dataset1 %>%
  filter(size_sqft < 4800)

ggplot(rental_dataset1, aes(x = size_sqft)) +
  geom_histogram(color = "black", fill = "chocolate4", bins = 40) + xlab("Square Feet") +
  ylab("Count") + ggtitle("Distribution of Apartment Size in Square Feet")
```



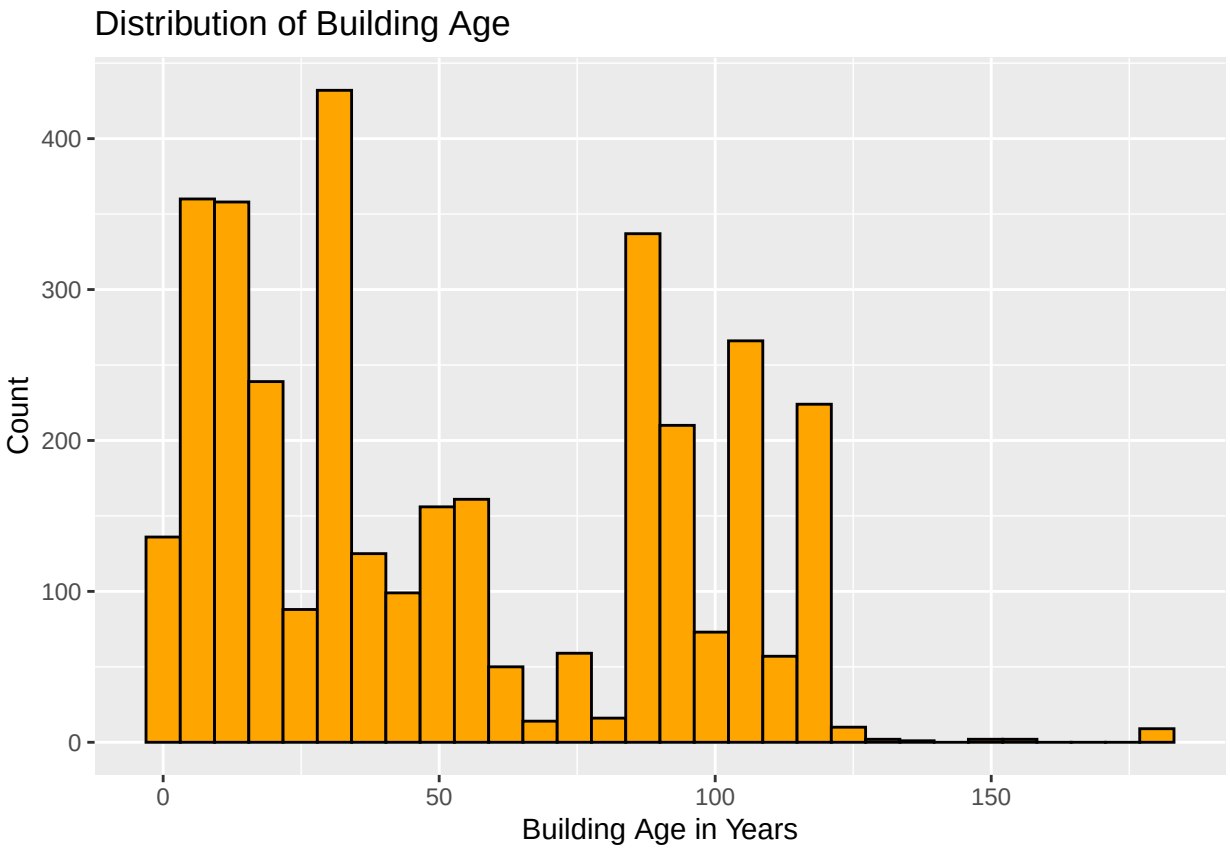
Next, we look at the square feet of the apartments depending on which neighborhood type they are in.

```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
  ggplot(aes(x=size_sqft)) +
  geom_histogram(bins = 10, color = "purple", fill = "pink") +
  facet_wrap(~location)
```



*BUILDING AGE**

```
rental_dataset1 %>%
  ggplot(aes(x = building_age_yrs)) +
  geom_histogram(fill='orange', color='black', bins = 30) +
xlab("Building Age in Years") + ylab("Count") + ggtitle("Distribution of Building Age")
```

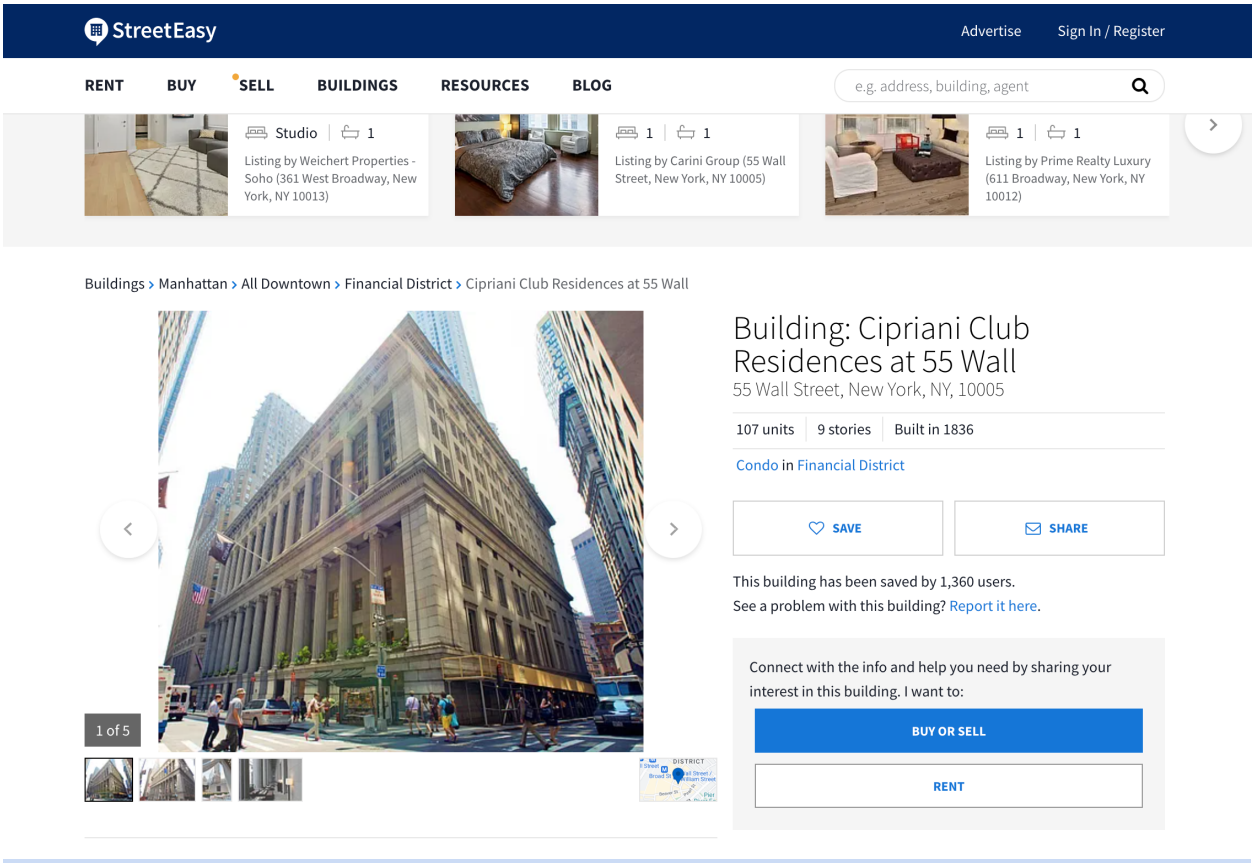


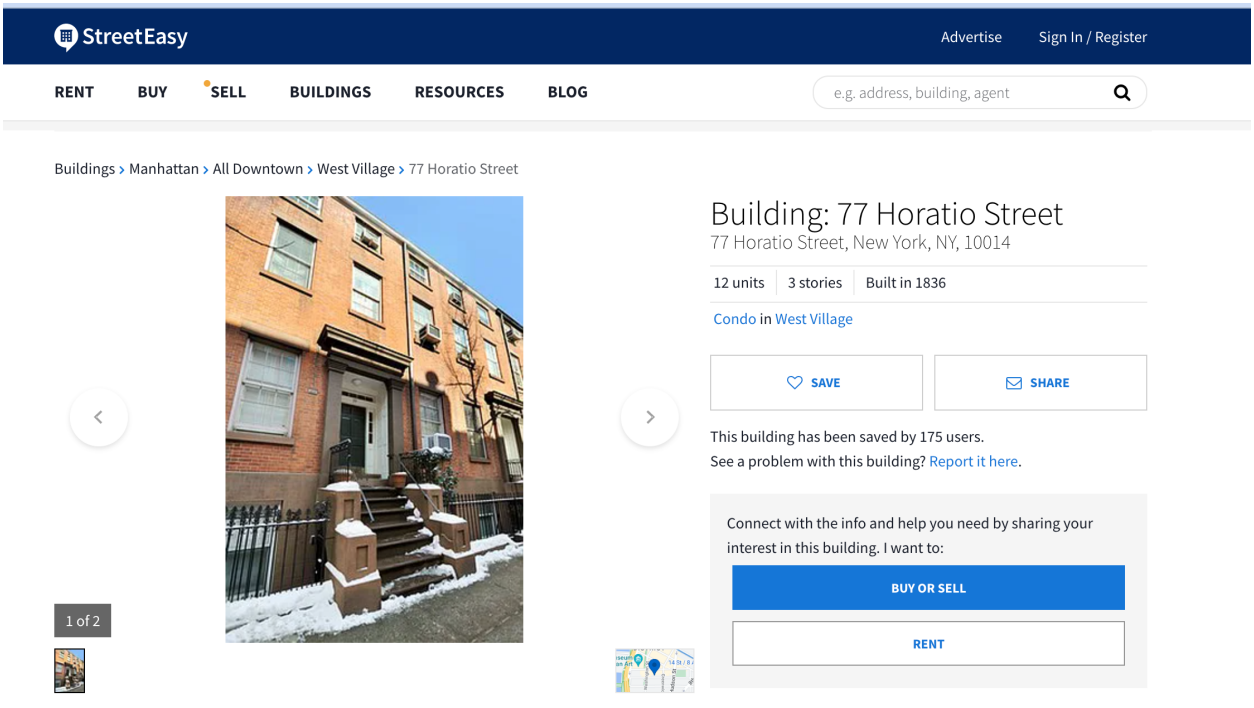
As we can see, the distribution is relatively even with a possible outlier at around 175 years. Let's look at these high building age values.

```
rental_dataset1 %>%
  filter(building_age_yrs > 175) %>%
  select(neighborhood, building_age_yrs) %>%
  arrange(desc(building_age_yrs)) %>%
  head()
```

```
##         neighborhood building_age_yrs
## 1 Financial District              180
## 2 Financial District              180
## 3 Financial District              180
## 4       West Village             180
## 5 Financial District              180
## 6 Financial District              180
```

In order to verify that buildings can realistically have an age of 180 years, let's look at on StreetEasy for buildings built in 1836, since in 2016, a building age of 180 years would be constructed in 1836.



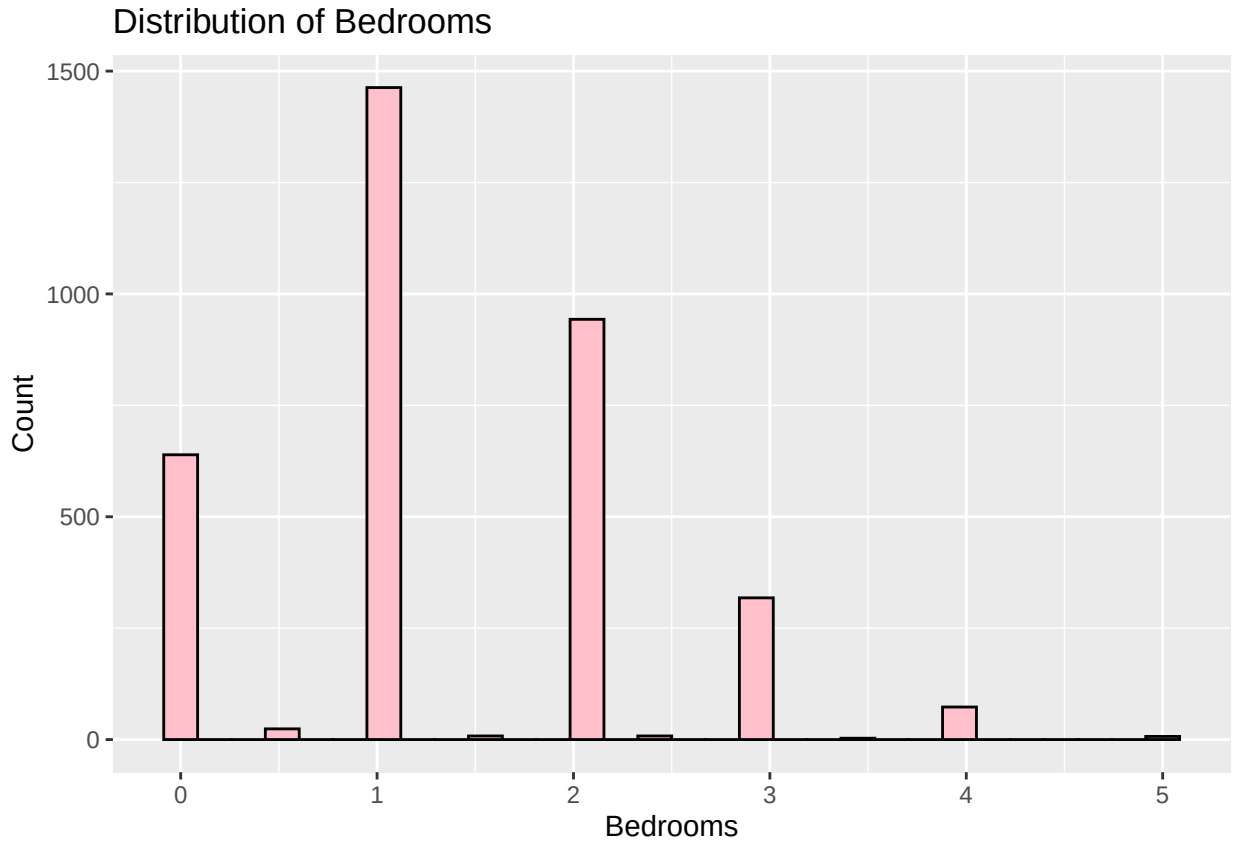


As we can see in the sample buildings, a building that is 180 years old is legitimate, such as in the Financial District and West Village, where most of these high values are occurring. Thus, we can keep these values in our dataset.

BEDROOMS

```
rental_dataset1 %>%
  ggplot(aes(x = bedrooms)) + geom_histogram(fill='pink', color='black') +
xlab("Bedrooms") + ylab("Count") + ggtitle("Distribution of Bedrooms")
```

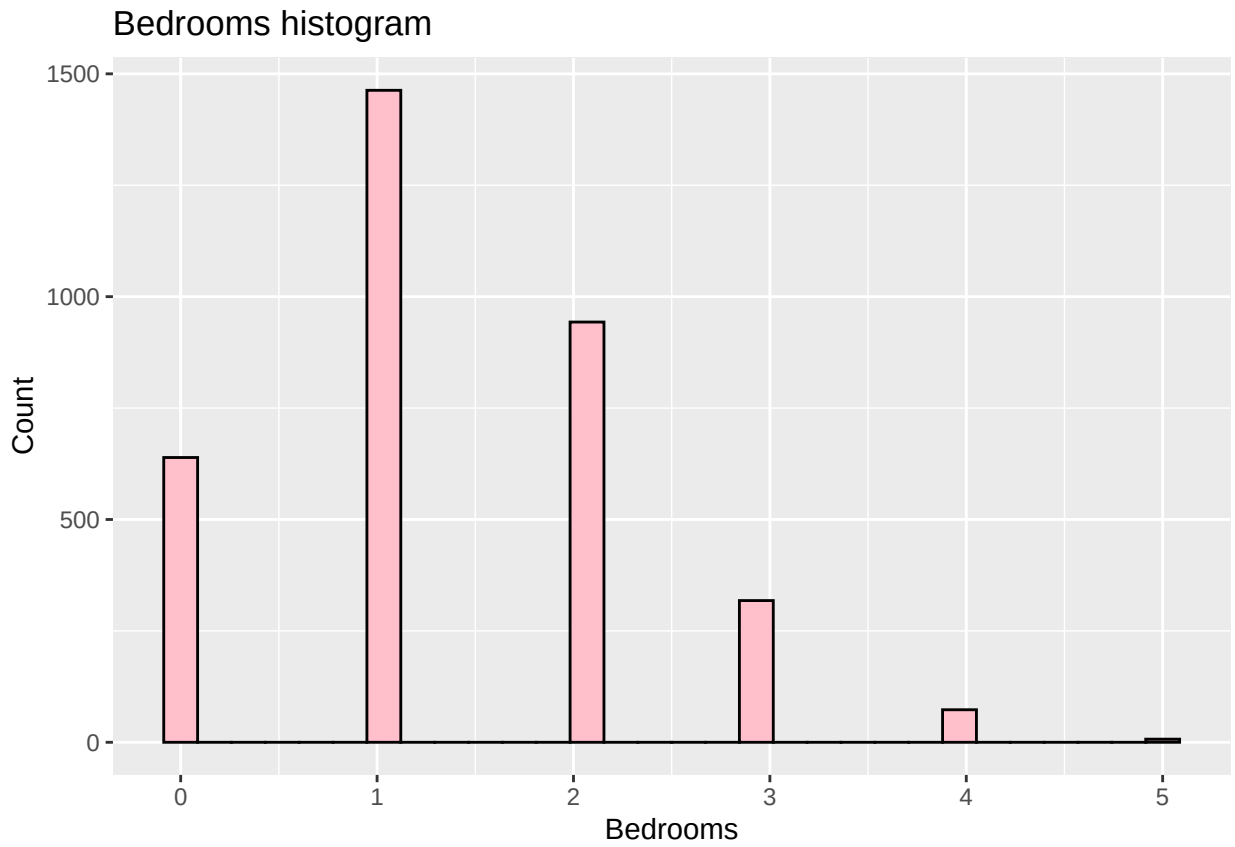
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



The distribution is fairly even with a slight skew to the right. There is an abundance of whole number bedrooms with a slight number of half-bedrooms. We do not need to investigate any outliers because it is reasonable to have 5 bedrooms in an apartment. Furthermore, 0 bedrooms mostly indicates that the apartment is a studio rental.

```
rental_dataset1 <- rental_dataset1 %>%
    filter(bedrooms %in% c("0","1","2","3","4","5"))


rental_dataset1 %>%
  ggplot(aes(x = bedrooms)) + geom_histogram(fill='pink', color='black') +
xlab("Bedrooms") + ylab("Count") + ggtitle("Bedrooms histogram")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

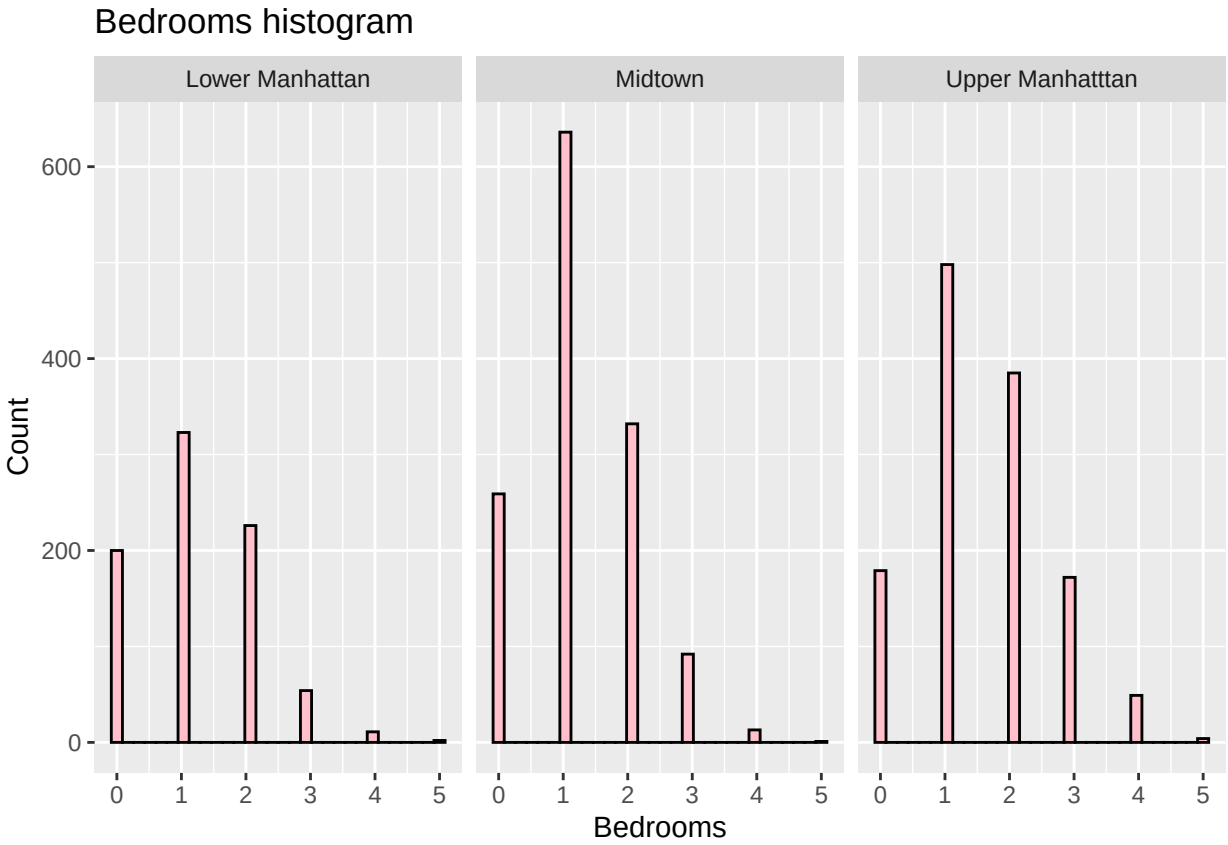


After removing the half bedrooms we can most clearly see this unimodal graph with the peak being at 1 bedrooms. The graph still skews to the right.

Now we will examine if the bedrooms change depending on type of neighborhood.

```
rental_dataset1 %>%
    filter(!is.na(location)) %>%
  ggplot(aes(x = bedrooms)) + geom_histogram(fill='pink', color='black') +
xlab("Bedrooms") + ylab("Count") +
  ggtitle("Bedrooms histogram") + facet_wrap(~location)
```

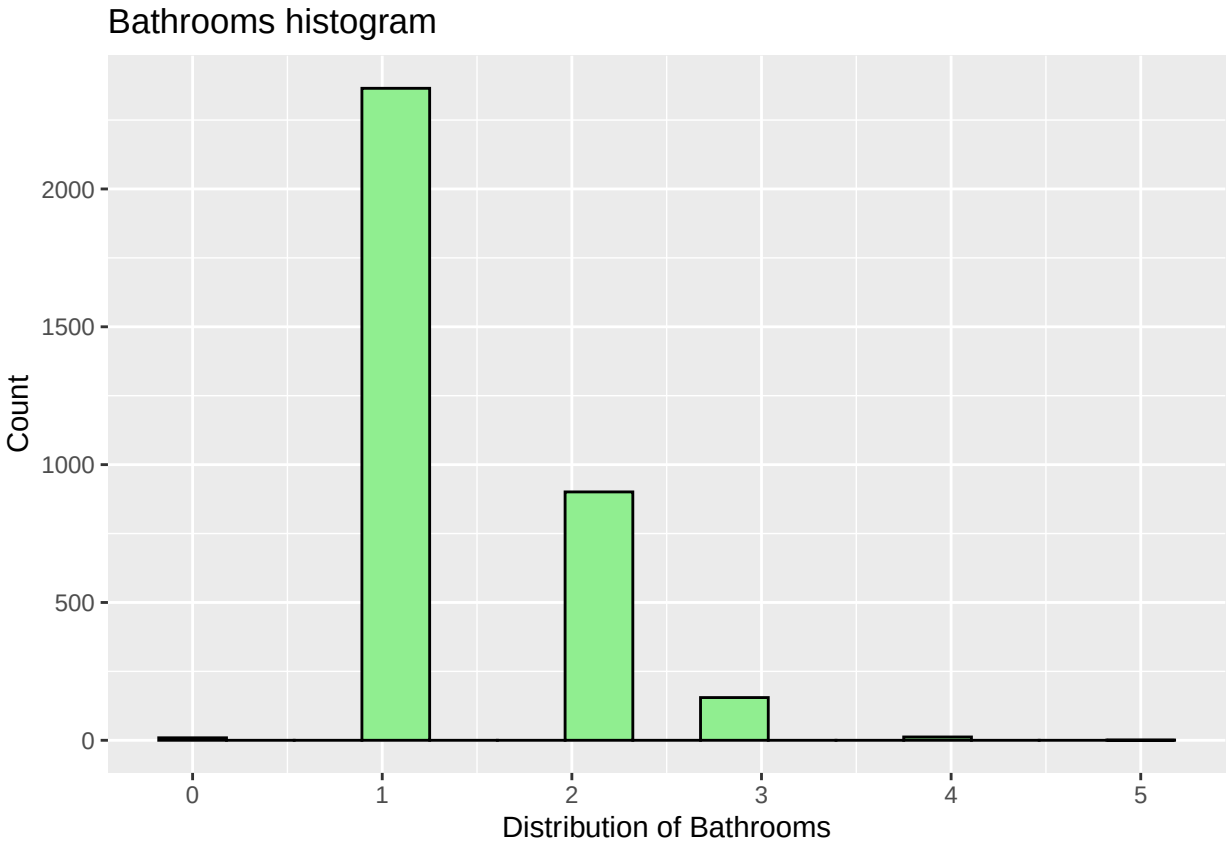
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Not surprisingly, they dont. The peak bedrooms is still one and the general shape of the graph is same for all 3 locations.

Bathrooms

```
rental_dataset1 %>%
  ggplot(aes(x = bathrooms)) +
  geom_histogram(fill='lightgreen', color='black', bins = 15)+
xlab("Distribution of Bathrooms") + ylab("Count") + ggtitle("Bathrooms histogram")
```



The distribution is fairly skewed right with most apartments having 1-2 bathrooms, however there are no major outliers. In Manhattan, it is reasonable to have 0 bathrooms or 5 bathrooms depending on the apartment rental.

Amenities

Let's see how common the amenities are in Manhattan.

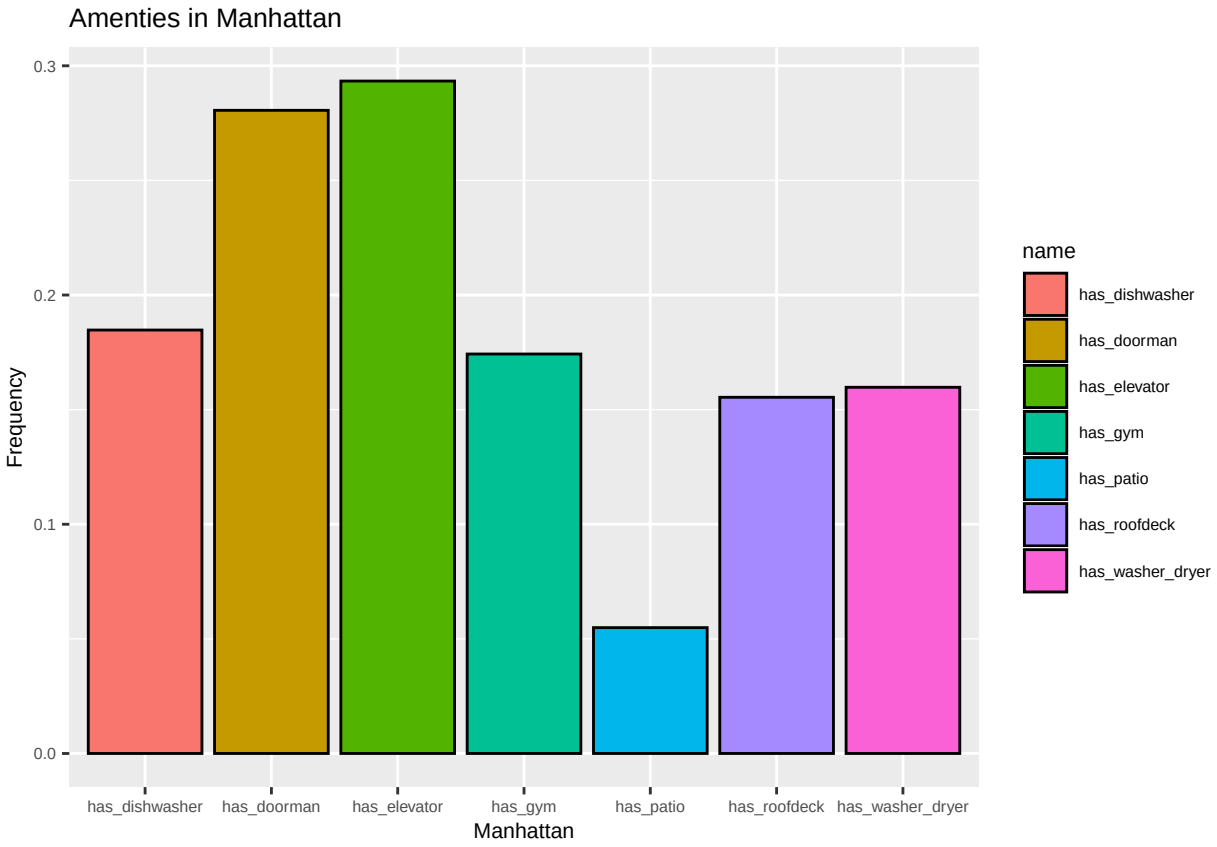
```
amenities <- rental_dataset1 %>%
  select(has_roofdeck,has_washer_dryer,has_doorman,has_elevator,has_dishwasher,has_patio,has_gym)
  name_of_amenities = colnames(amenities)
amenities_frequencies <- tibble(colMeans(amenities))
add_column(amenities_frequencies,name_of_amenities)
```

```
## # A tibble: 7 x 2
##   `colMeans(amenities)` name_of_amenities
##                   <dbl> <chr>
## 1                 0.155  has_roofdeck
## 2                 0.160  has_washer_dryer
## 3                 0.281  has_doorman
## 4                 0.293  has_elevator
## 5                 0.185  has_dishwasher
## 6                 0.0549 has_patio
## 7                 0.174  has_gym
```

Not surprisingly, most apartments in Manhattan don't have a lot of amenities. The most frequent amenity in Manhattan is an elevator while the most uncommon amenity is a patio.

Let's see the distribution of amenities across all of the Manhattan apartment listings.

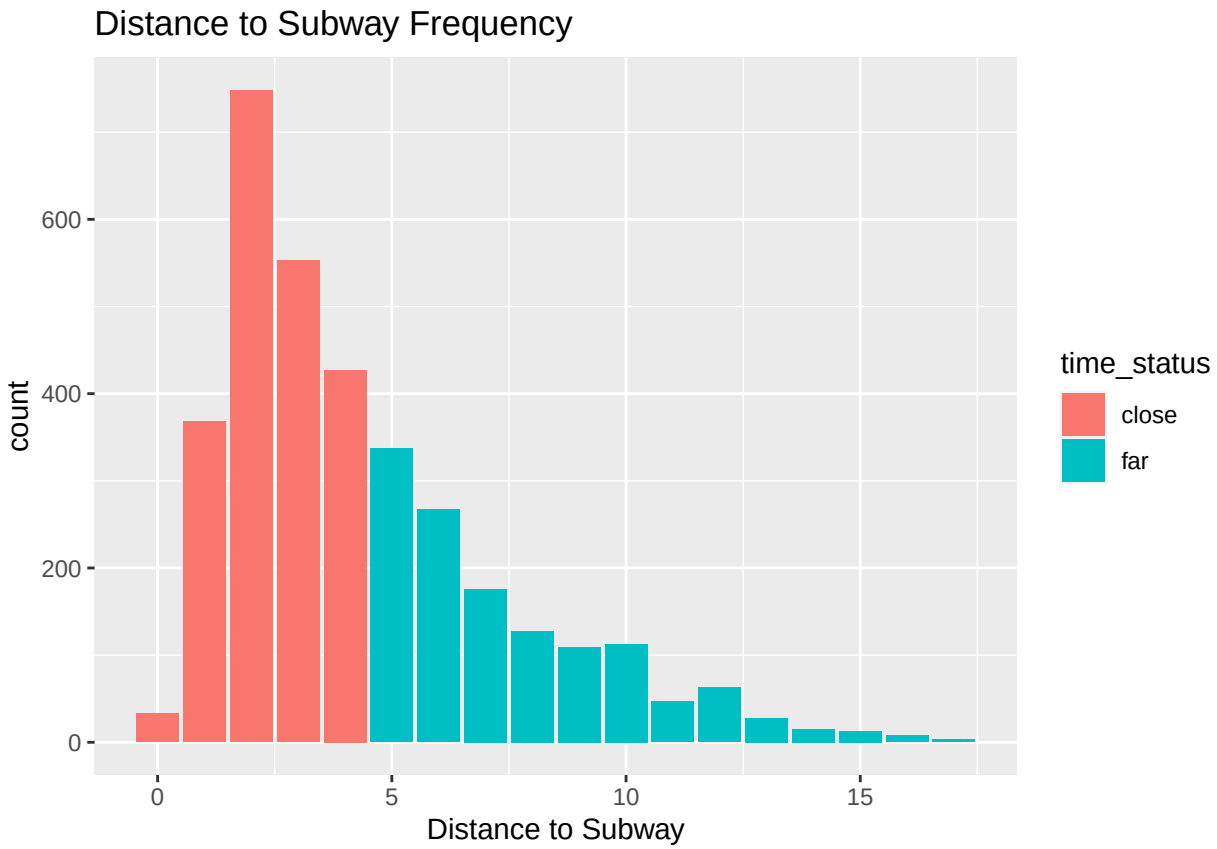
```
rental_dataset1 %>% pivot_longer(cols = starts_with("has")) %>%
  select(name, value) %>%
  group_by(name) %>%
  summarize(prop = mean(value)) %>%
  ggplot(aes(x = name, y = prop, fill = name))+
  theme(text = element_text(size=8)) +
  geom_col(color = "black" ) + xlab("Manhattan") + ylab("Frequency") + ggtitle("Amenties in Manhattan")
```



Distance to Subway

Let's look at the subway variable again but the time looking at the mean time it takes for a person to get to the subway. Let's first look again at the breakdown of count vs distance to subway, this time factoring in the time_status variable.

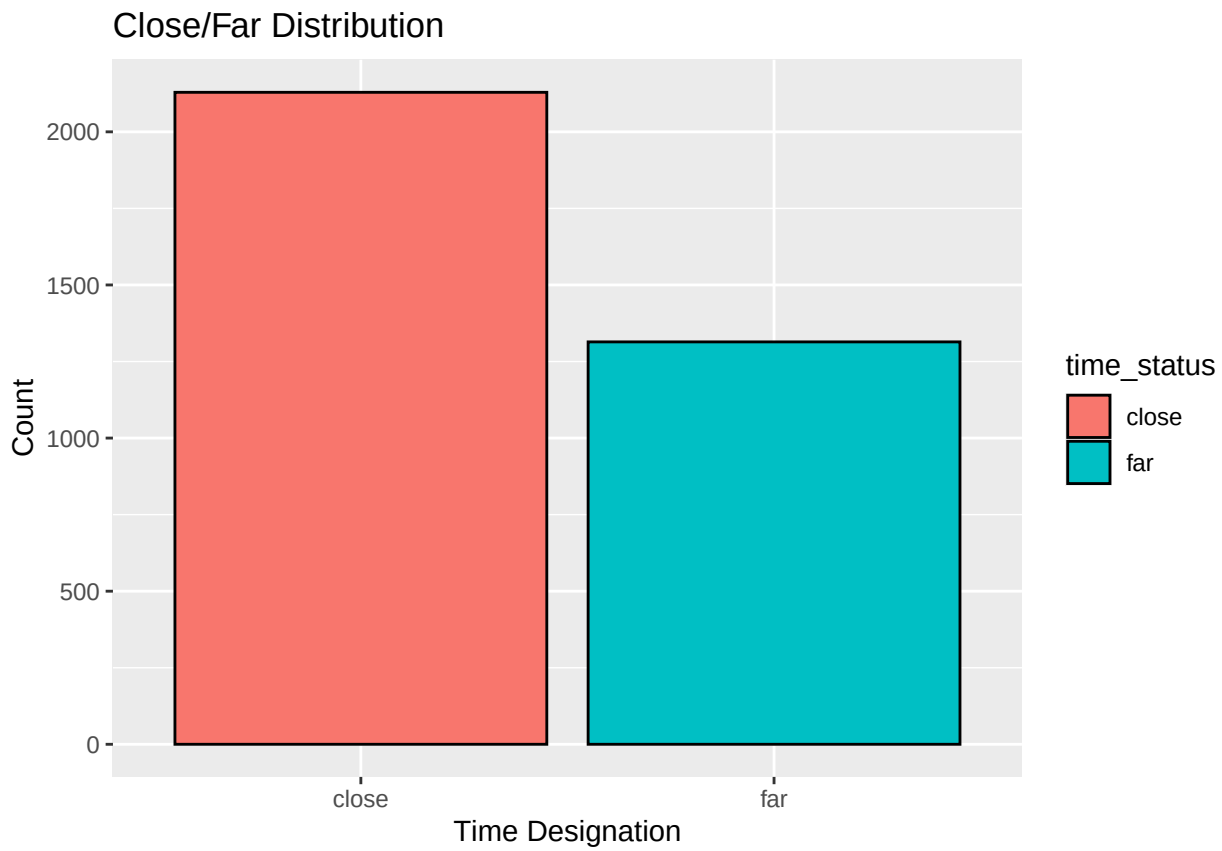
```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
  ggplot(aes(x=min_to_subway, fill = time_status))+ geom_bar(position = "dodge") +
  xlab("Distance to Subway") + ggtitle("Distance to Subway Frequency")
```



The distribution is skewed to the right. The distribution of the minutes to the subway has two main peaks; for times that are "close", the peak is around 2 minutes, while for times that are "far", the peak is around 5-6 minutes. Because we assigned our close/far groupings according to the mean, it is important to note that 5 minutes is included as "far".

Let's look at the split between the close and far times.

```
rental_dataset1 %>%
  ggplot(aes(x = time_status, fill = time_status)) + geom_bar(color='black') +
xlab("Time Designation") + ylab("Count") + ggtitle("Close/Far Distribution")
```



When comparing the Close/Far Distribution graph to the previous Time to Subway Distribution graph, It makes sense that there are more apartments considered close to the subway station as close times under 5 minutes had much greater frequencies than far times greater than 5. This makes sense as Manhattan has a good metro system and has many subway lines.

SECTION 3.2: BIVARIATE PLOTS

Bedrooms

```
rental_dataset1 %>%
ggplot(aes(x=bedrooms, y= size_sqft)) +
geom_point(color = "purple") +
geom_smooth(method="lm", color="purple") +
xlab("Bedrooms") + ylab("Square Feet")+
  ggtitle("Bedrooms vs. Square Feet")
```

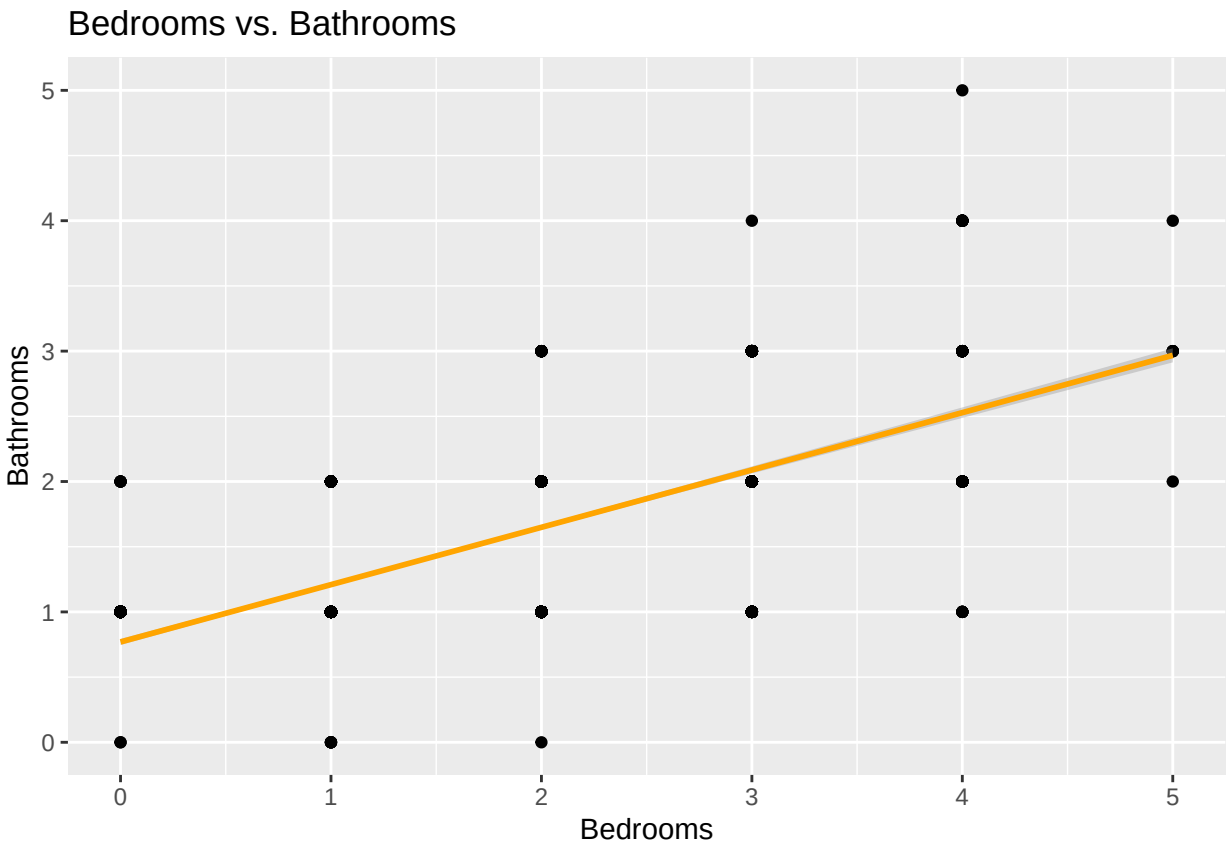
```
## `geom_smooth()` using formula 'y ~ x'
```

As expected, We can see there is a positive correlation; as the square feet increases, so does the number of bedrooms. This makes sense as biggest apartments can fit more bedrooms.

Next we will compare the amount of bedrooms with the amount of bathrooms.

```
rental_dataset1 %>%
ggplot(aes(x=bedrooms, y= bathrooms)) +
geom_point(color = "black") +
geom_smooth(method="lm", color="orange") +
xlab("Bedrooms") + ylab("Bathrooms") +
  ggtitle("Bedrooms vs. Bathrooms")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

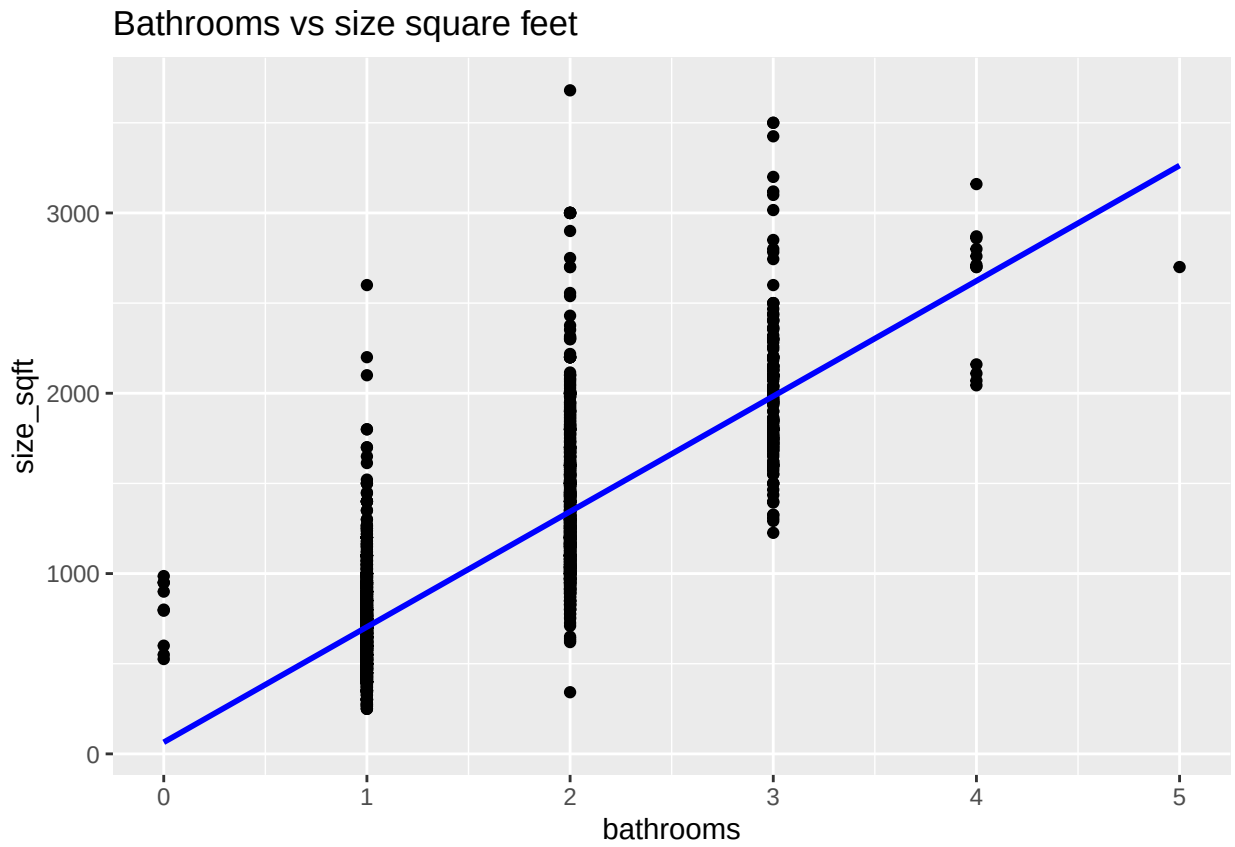


Similarly to the previous graph we can see that bedrooms in fact does have an affect on bathrooms. The more bedrooms there are, the more bathrooms there will be this. This also makes a lot of sense since a big apartment can fit more bathrooms.

Bathrooms

```
rental_dataset1 %>%
ggplot(aes(x=bathrooms, y= size_sqft)) +
geom_point(color = "black") +
geom_smooth(method="lm", se = FALSE, color = 'blue') +
 ggtitle("Bathrooms vs size square feet")
```

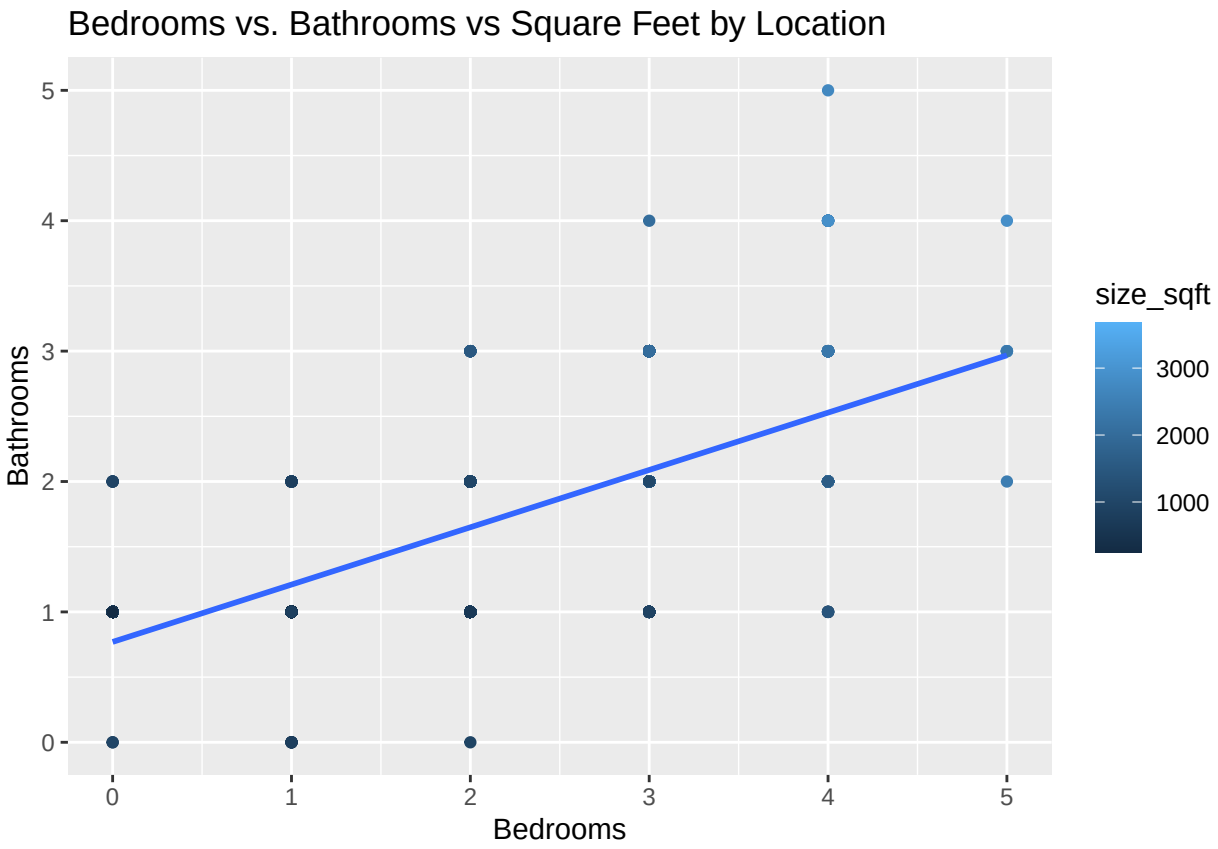
```
## `geom_smooth()` using formula 'y ~ x'
```



As expected we see that bathrooms do have an impact on the square feet of the apartment. This is expected as more bathrooms usually have more bedrooms which in turn leads to a bigger apartment.

```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
ggplot(aes(x=bedrooms, y= bathrooms, color = size_sqft)) +
geom_point() +
geom_smooth(method="lm", se = FALSE) +
xlab("Bedrooms") + ylab("Bathrooms")+
  ggtitle("Bedrooms vs. Bathrooms vs Square Feet by Location")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Building Age

We're curious to see if comparing the age of the building to the floor you live on has any correlation.

```
rental_dataset1 %>%
ggplot(aes (x= building_age_yrs, y = floor)) +
geom_point(color = "black") +
geom_smooth(method="lm", se = FALSE, color = 'blue') +
 ggtitle("Floor vs Building Age ")+
  xlab("Age of Building") +ylab("Floor")
```

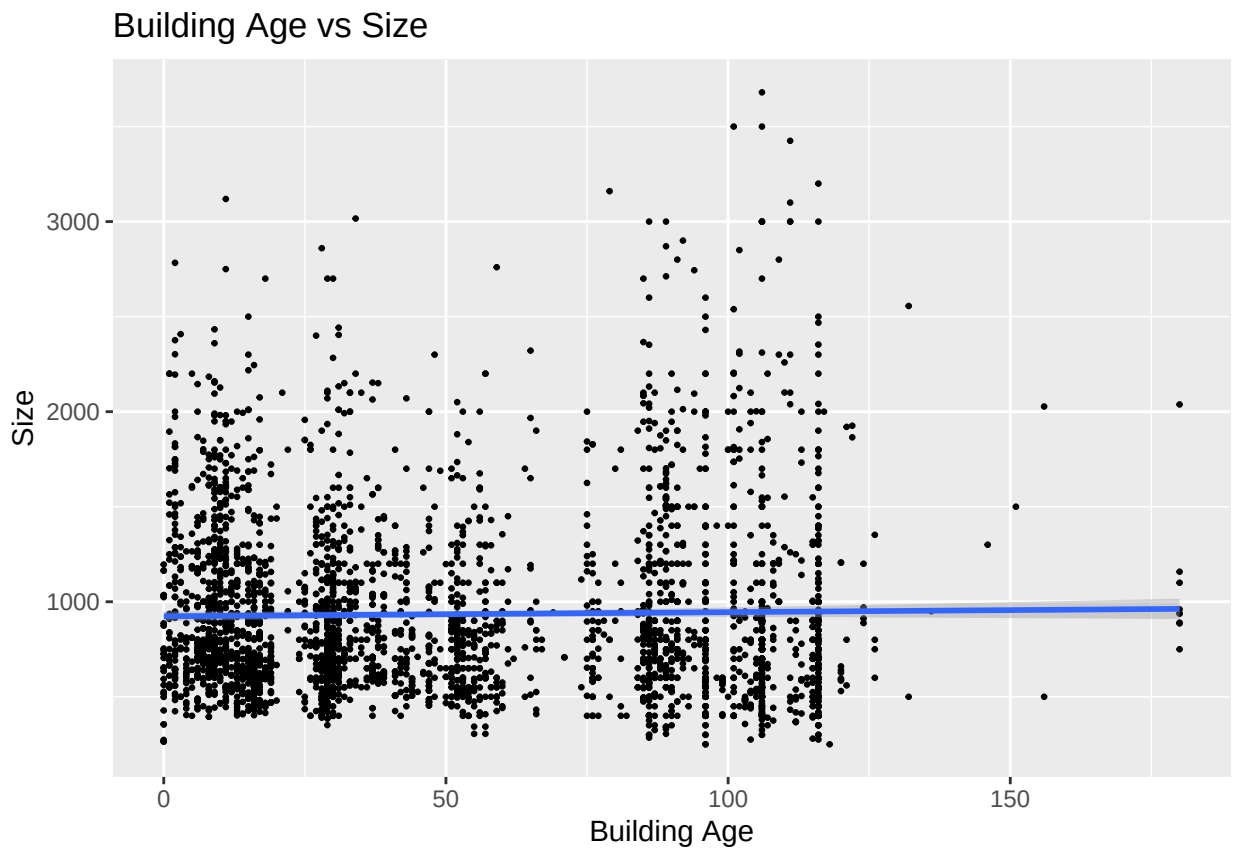
```
## `geom_smooth()` using formula 'y ~ x'
```

We can see that there is little to no correlation between the age of building and the floor.

Building Age Years vs Square Feet

```
ggplot(rental_dataset1, aes(x = building_age_yrs, y = size_sqft)) +
geom_point(size = 0.5, color = "black") +
  geom_smooth(method = "lm") +
  xlab("Building Age") + ylab("Size") + ggtitle("Building Age vs Size")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

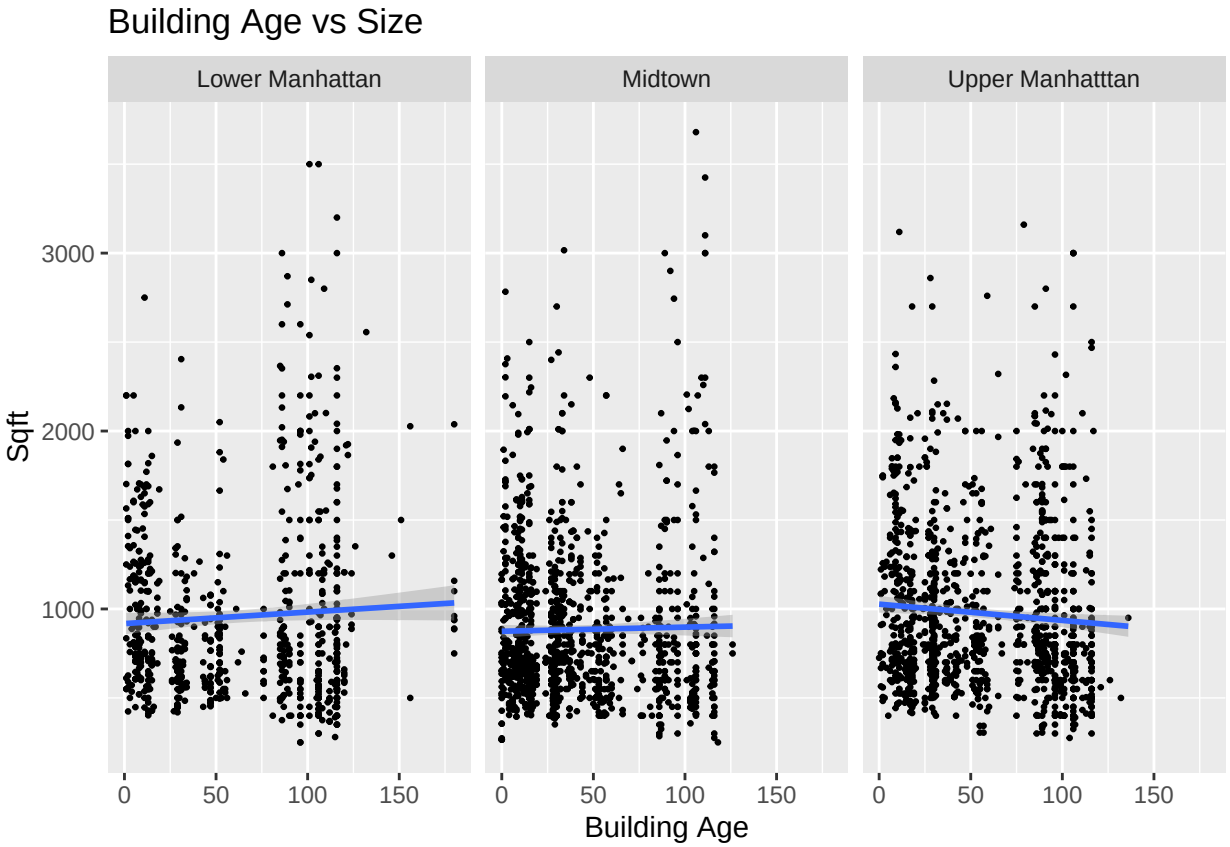


The following scatterplot explores the relationship between how old a building is, and the size of the building to explore the question: Do newer buildings have apartments larger or smaller in size? We draw the conclusion from the graph that there is little to no correlation between the two variables, and newer buildings can have apartments of any size.

Lets see if there is any correlation when we factor in location.

```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
ggplot(aes(x = building_age_yrs, y = size_sqft)) +
geom_point(size = 0.5, color = "black") +
  geom_smooth(method = "lm") +
  xlab("Building Age") + ylab("Sqft")+
  ggtitle("Building Age vs Size") + facet_wrap(~location)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

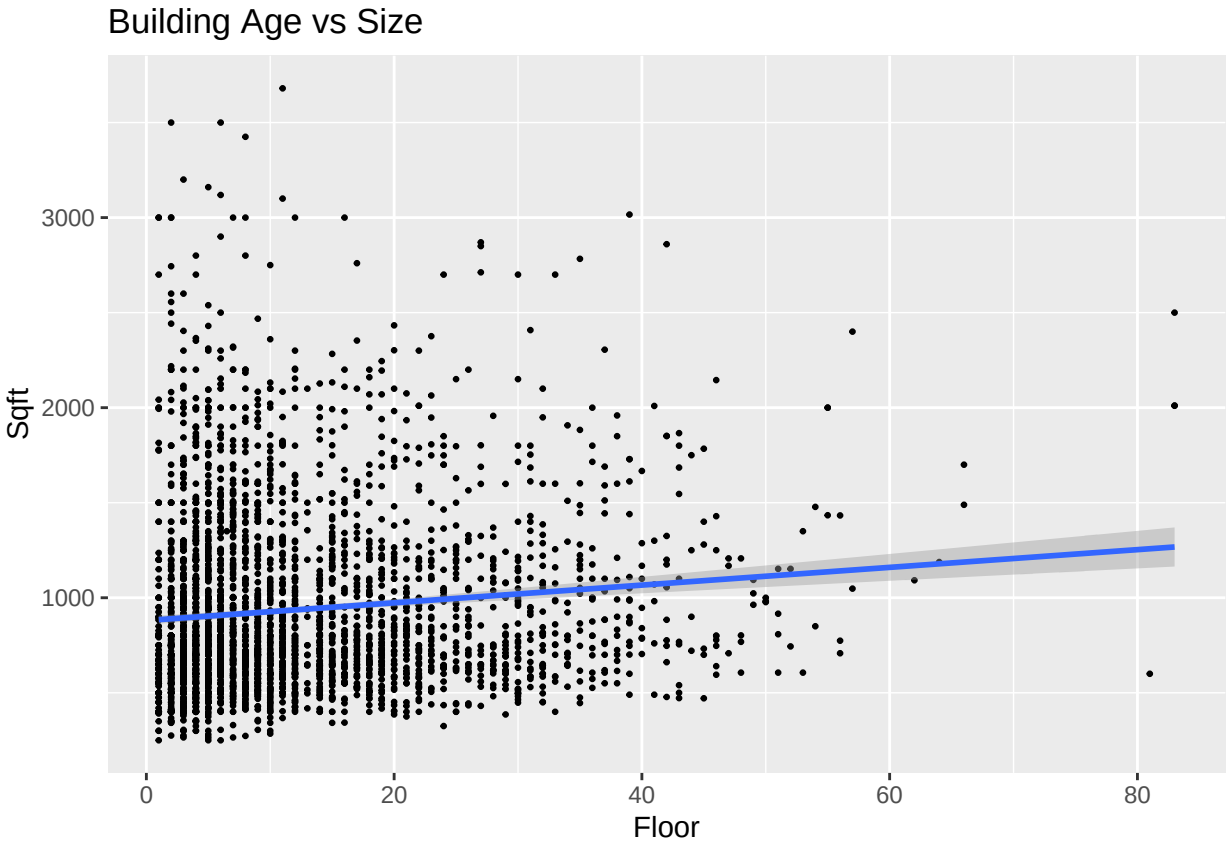


We can see that even if we factor based on location there is still little to no correlation between building age and sqft.

Floor

```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
ggplot(aes(x = floor, y = size_sqft)) +
geom_point(size = 0.5, color = "black") +
  geom_smooth(method = "lm") +
  xlab("Floor") + ylab("Sqft")+ ggtitle("Building Age vs Size")
```

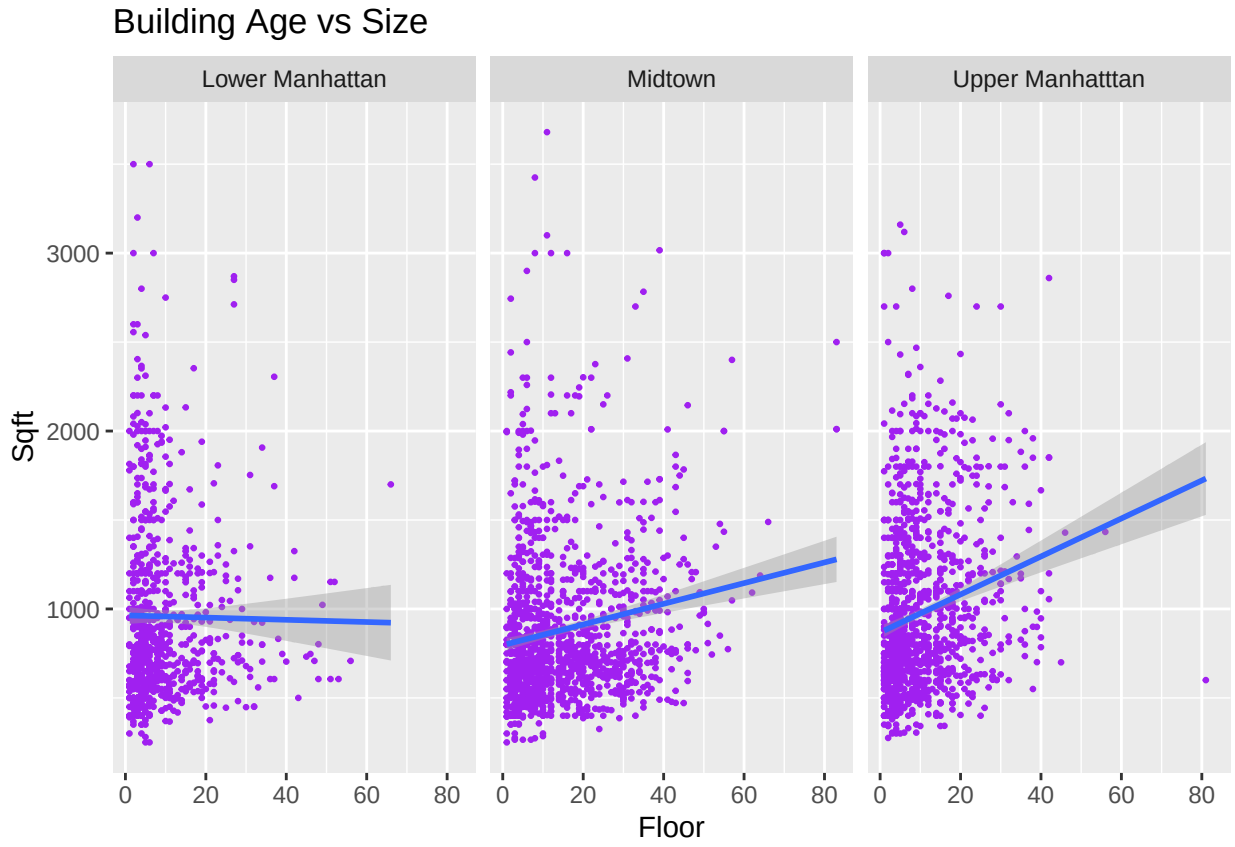
```
## `geom_smooth()` using formula 'y ~ x'
```



We can see that there is little to none correlation between floor and sqft. The floor you are on does not determine the square feet of the apartment.

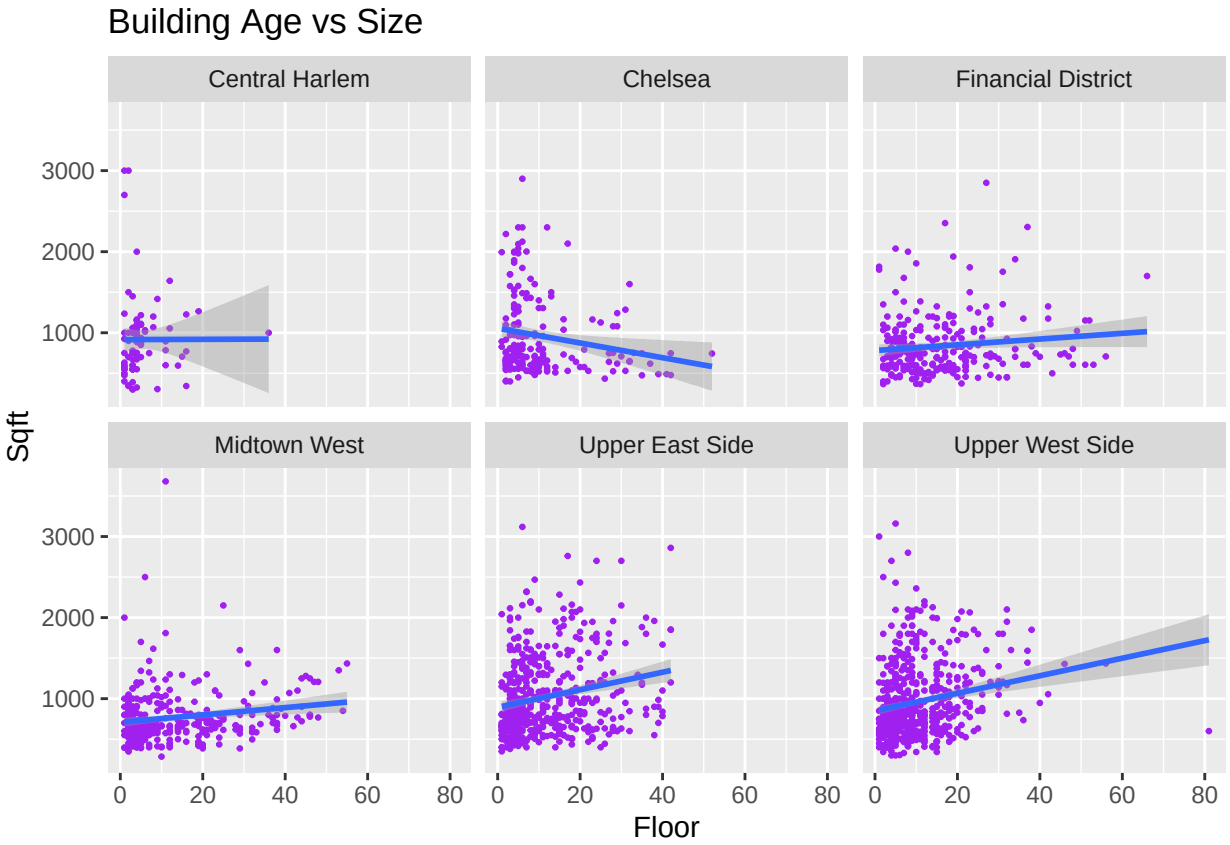
```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
ggplot(aes(x = floor, y = size_sqft)) +
geom_point(size = 0.5, color = "purple") +
  geom_smooth(method = "lm") +   xlab("Floor") +
  ylab("Sqft") + ggtitle("Building Age vs Size") + facet_wrap(~location)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
rental_dataset1 %>%
  filter(!is.na(neigh_selected)) %>%
ggplot(aes(x = floor, y = size_sqft)) +
geom_point(size = 0.5, color = "purple") + geom_smooth(method = "lm") +
  xlab("Floor") + ylab("Sqft") + ggtitle("Building Age vs Size")+
  facet_wrap(~neigh_selected)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

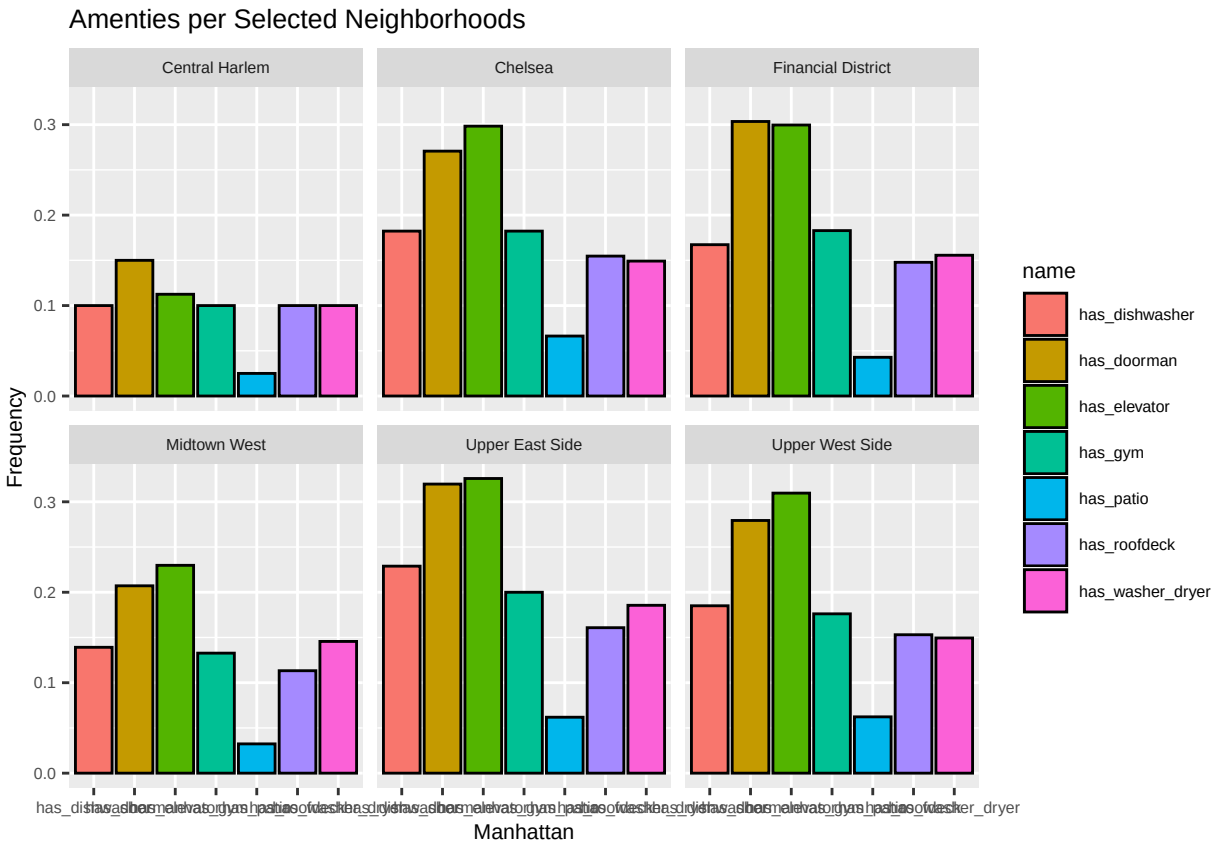


Even after factoring in differnt neighborhoods and different locations we can see that there is close to no correlation between floor and sqft.

Amenities Let's look at the distribution of amenities by neighborhood.

```
rental_dataset1 %>% pivot_longer(cols = starts_with("has")) %>%
  select(neigh_selected, name, value) %>%
  group_by(neigh_selected, name) %>%
  summarize(prop = mean(value)) %>%
  filter(!is.na(neigh_selected))%>%
  ggplot(aes(x = name, fill = name, y = prop))+
  theme(text = element_text(size=8)) +
  geom_col(color = "black", position = "dodge") +
  xlab("Manhattan") + ylab("Frequency") +
  ggtitle("Amenties per Selected Neighborhoods") + facet_wrap(~neigh_selected)
```

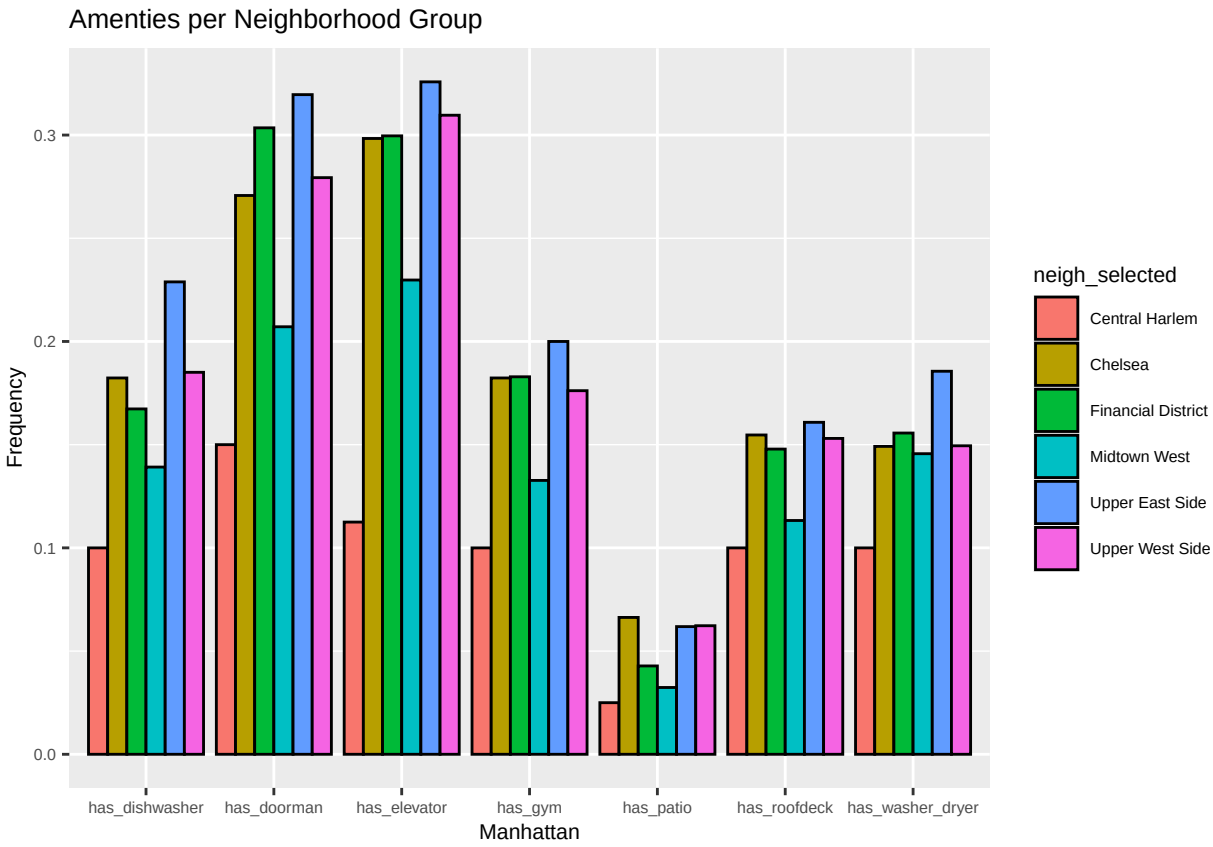
```
## `summarise()` has grouped output by 'neigh_selected'. You can override using
## the `.groups` argument.
```



Let's look at the distribution of amenities by neighborhood group.

```
rental_dataset1 %>% pivot_longer(cols = starts_with("has")) %>%
  select(neigh_selected, name, value) %>%
  group_by(neigh_selected, name) %>%
  summarize(prop = mean(value)) %>%
  filter(!is.na(neigh_selected))%>%
  ggplot(aes(x = name, fill = neigh_selected, y = prop))+
  theme(text = element_text(size=8)) +
  geom_col(color = "black", position = "dodge") +
  xlab("Manhattan") + ylab("Frequency") + ggtitle("Amenties per Neighborhood Group")
```

```
## `summarise()` has grouped output by 'neigh_selected'. You can override using
## the `.groups` argument.
```

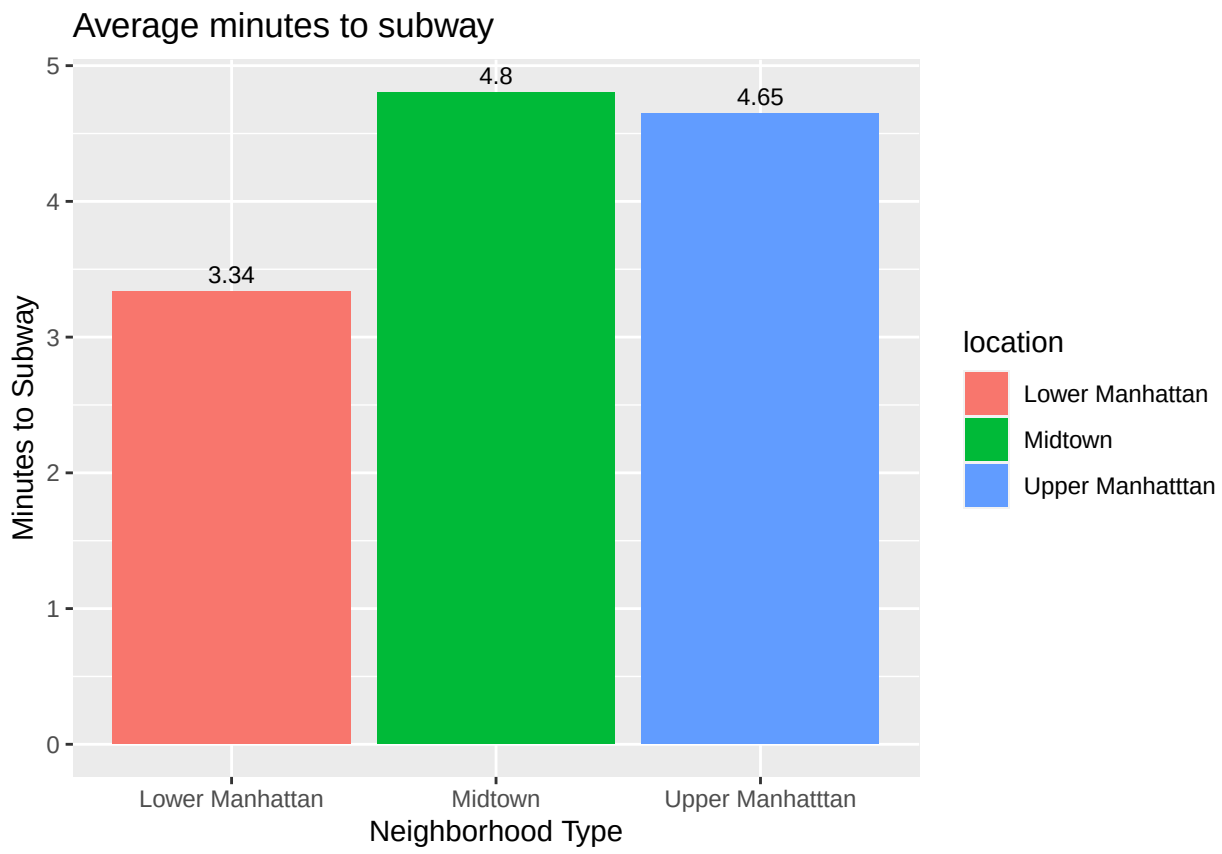


From the graph we can see that there is a lot of variation in which amenities in apartment depending on which neighborhood you live in. Even after factoring different neighborhoods we can see most apartments do not have a lot of amenities.

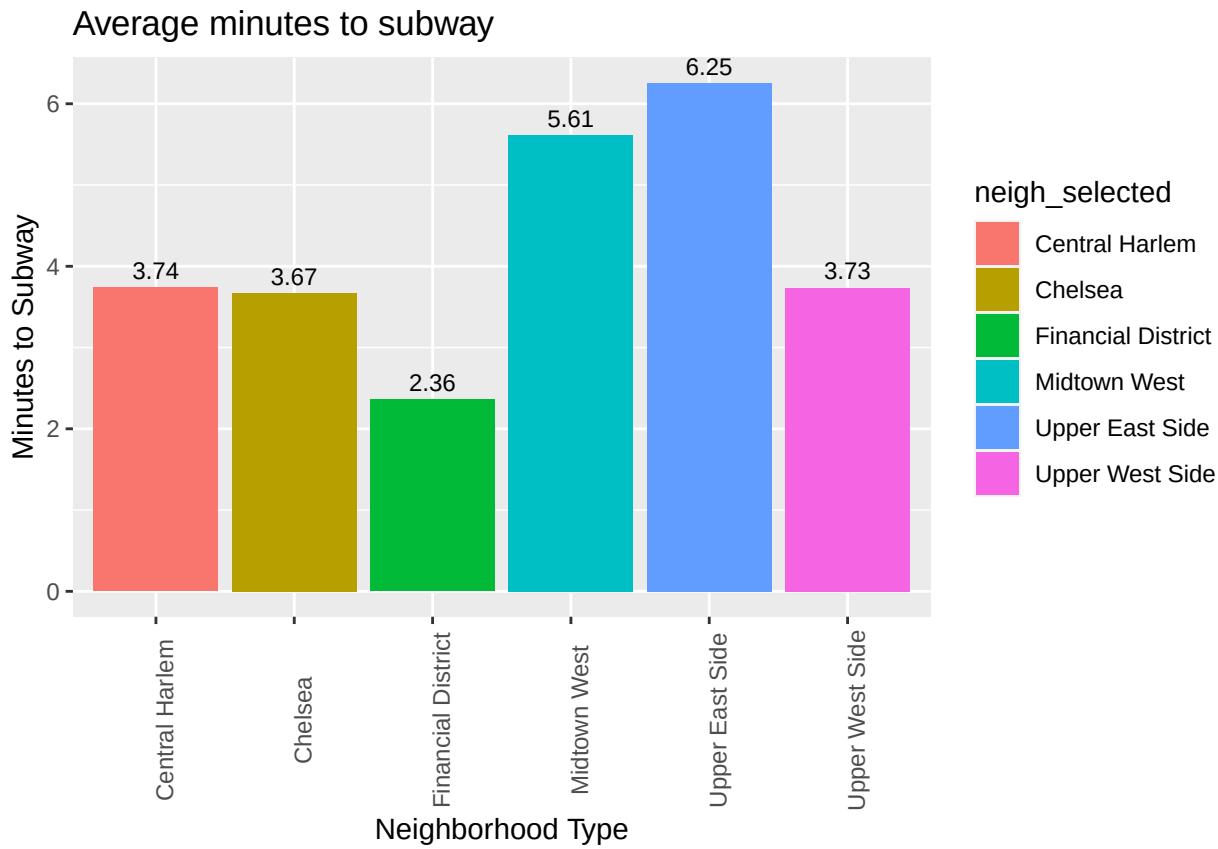
Subway Time vs Neighborhood and Location

We are now trying to figure out the average minutes it takes to the subway depending on neighborhood and the location you live.

```
rental_dataset1 %>%
  group_by(location) %>%
  filter(!is.na(location)) %>%
  summarize(value = mean(min_to_subway)) %>%
  ggplot(aes(x=location, y = value, fill = location)) + geom_col() + xlab("Neighborhood Type") + ylab("
```




```
rental_dataset1 %>%
  group_by(neigh_selected) %>%
  filter(!is.na(neigh_selected)) %>%
  summarize(value = mean(min_to_subway)) %>%
  ggplot(aes(x=neigh_selected, y = value, fill = neigh_selected)) + geom_col() +
  xlab("Neighborhood Type") + ylab("Minutes to Subway")+ ggtitle("Average minutes to subway ")+
  geom_text(aes(label = round(value,2)), vjust = -0.5, size=3)+
  theme(axis.text.x = element_text(angle = 90))
```

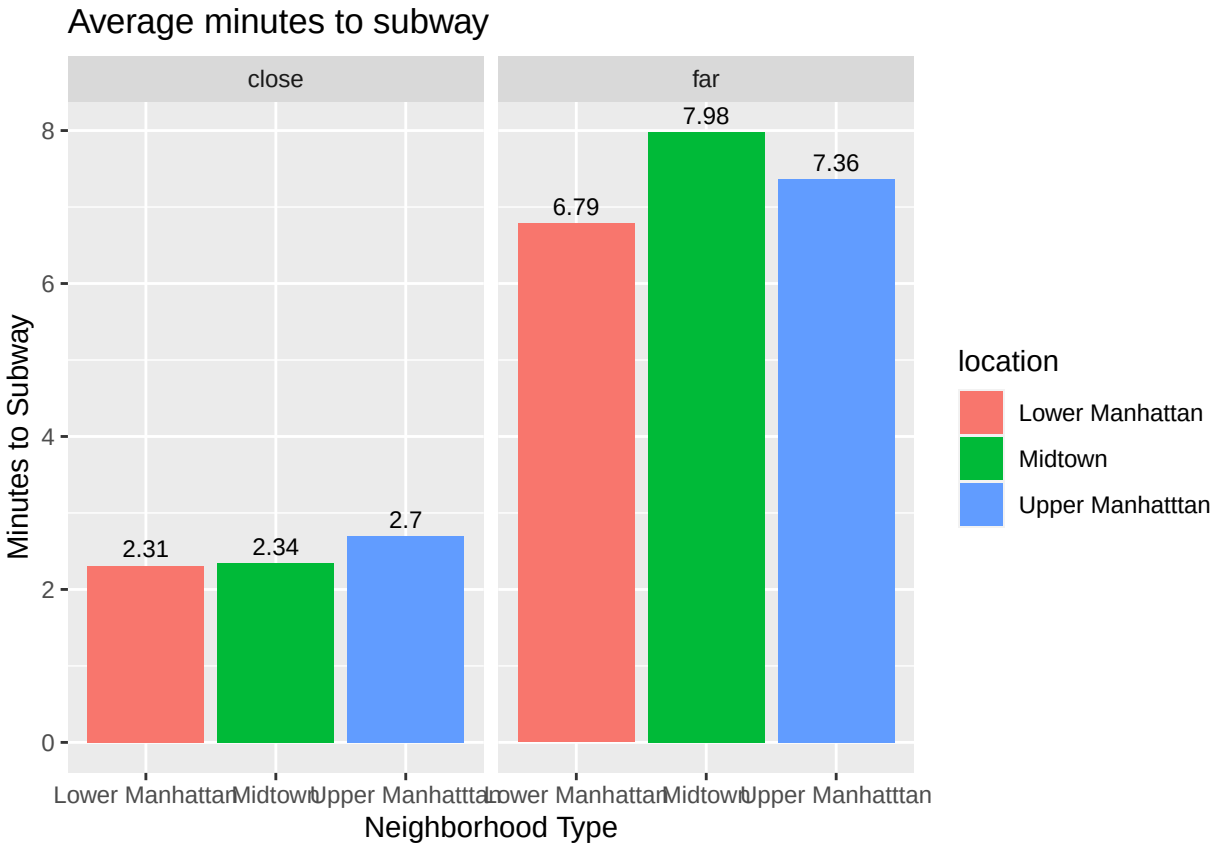


This is interesting we can see that there is a lot of variation on the time it takes to get to the subway depending on which specific neighborhood you live in. The Financial District is the closest to the subway while the Upper East Side takes the most amount of the time. The difference when factoring location decreases but still exists. We can see that apartments in Lower Manhattan are at least a minute closer to the subway than apartments in Midtown and Upper Manhattan. Apartments are about the same distance to the subway in Midtown and Upper Manhattan.

Let's look at this graph again but this time comparing on close and far neighborhoods.

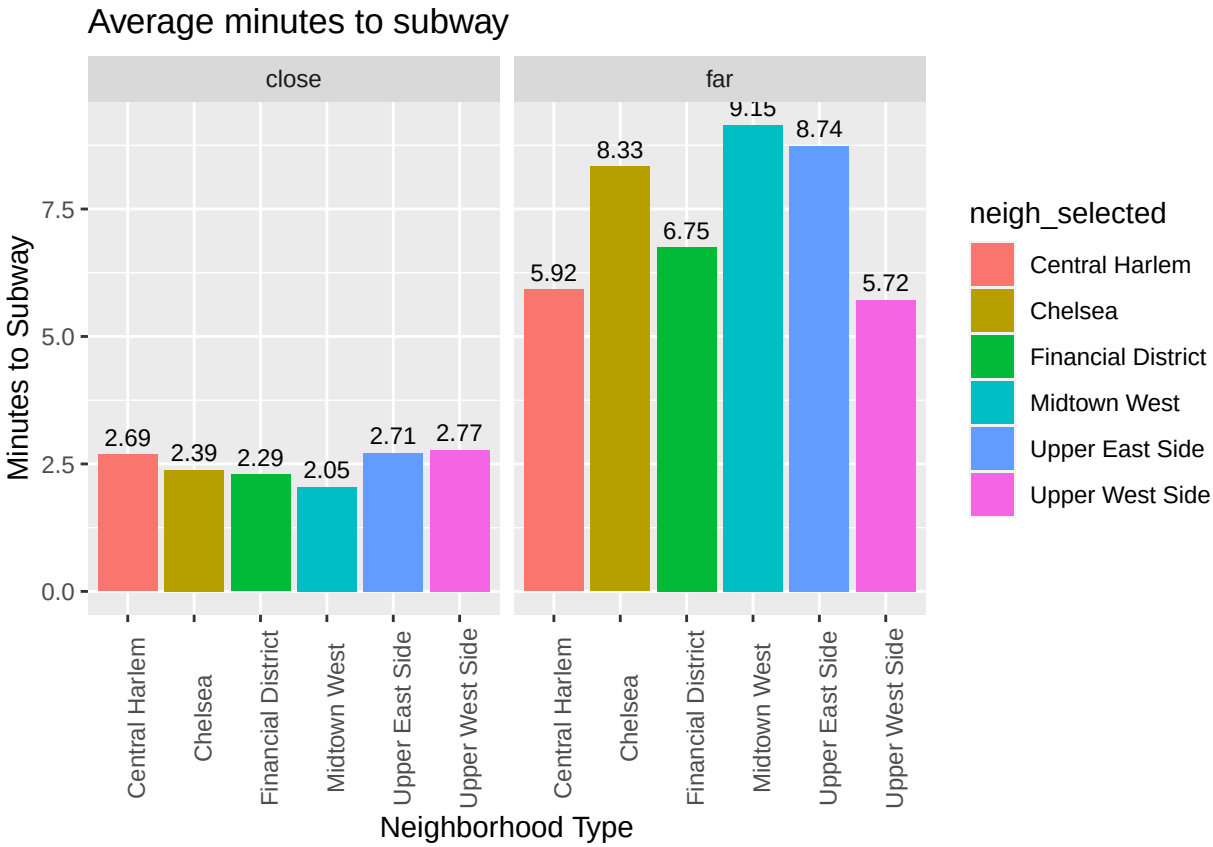
```
rental_dataset1 %>%
  group_by(location,time_status) %>%
  filter(!is.na(location)) %>%
  summarize(value = mean(min_to_subway)) %>%
  ggplot(aes(x=location, y = value, fill = location))+ geom_col() + xlab("Neighborhood Type") + ylab("M
```

```
## `summarise()` has grouped output by 'location'. You can override using the
## `.groups` argument.
```




```
rental_dataset1 %>%
  group_by(neigh_selected,time_status) %>%
  filter(!is.na(neigh_selected)) %>%
  summarize(value = mean(min_to_subway)) %>%
  ggplot(aes(x=neigh_selected, y = value, fill = neigh_selected)) +
  geom_col() + xlab("Neighborhood Type") + ylab("Minutes to Subway") + ggtitle("Average minutes to subwa
```

```
## `summarise()` has grouped output by 'neigh_selected'. You can override using
## the `.groups` argument.
```



Interestingly we can see that the average minutes to the subway doesn't change that much for neighborhoods that are already close to the subway. Neighborhoods and Locations that are far from the subway have more variation.

3.3: PRICE VS OTHER VARIABLES

Sqft vs Rent

The most important variable we expect in determining price of an apartment rental is square feet. We believe that the greater the square feet, the higher the rent can be expected per month.

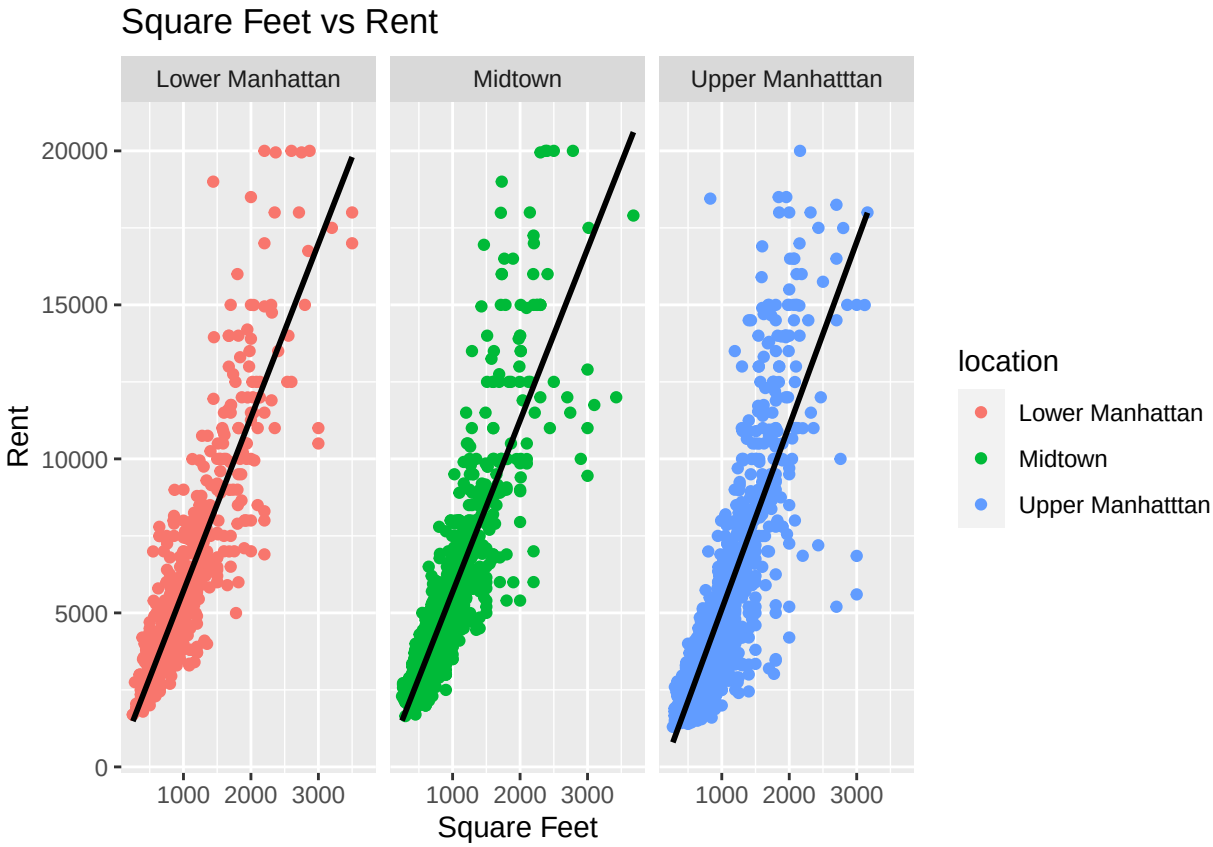
```
ggplot(rental_dataset1, aes(x = size_sqft, y = rent)) +
  geom_point() + xlab("Square Feet") + ylab("Rent")+ ggtitle("Square Feet vs Rent vs Bedrooms") + scale_
```

```
## `geom_smooth()` using formula 'y ~ x'
```

In this scatter plot we have mapped out the square feet variable on the x axis, and the rent variable on the y axis. Altogether this scatter plot demonstrates a moderate positive correlation between all three of these variables. Therefore we can guess on average that the greater the square footage of a property, the more the rent goes up. Additionally, the amount of bedrooms increases according to square footage. Let's look at this graph again this time controlling for the different neighborhoods.

```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
  ggplot(aes(x=size_sqft, y=rent, color = location)) + geom_point() + facet_wrap(~location)+xlab("Squar
```

```
## `geom_smooth()` using formula 'y ~ x'
```



As guessed the correlation holds through even for the different location. For the most part, the larger the apartment rental the higher the rent. Let's check one last time this time looking at individual neighborhoods.

```
rental_dataset1 %>%
  group_by(neigh_selected) %>%
  summarize(mean_rent = mean(rent), mean_sqft = mean(size_sqft))  %>%
  arrange(desc(mean_rent))
```

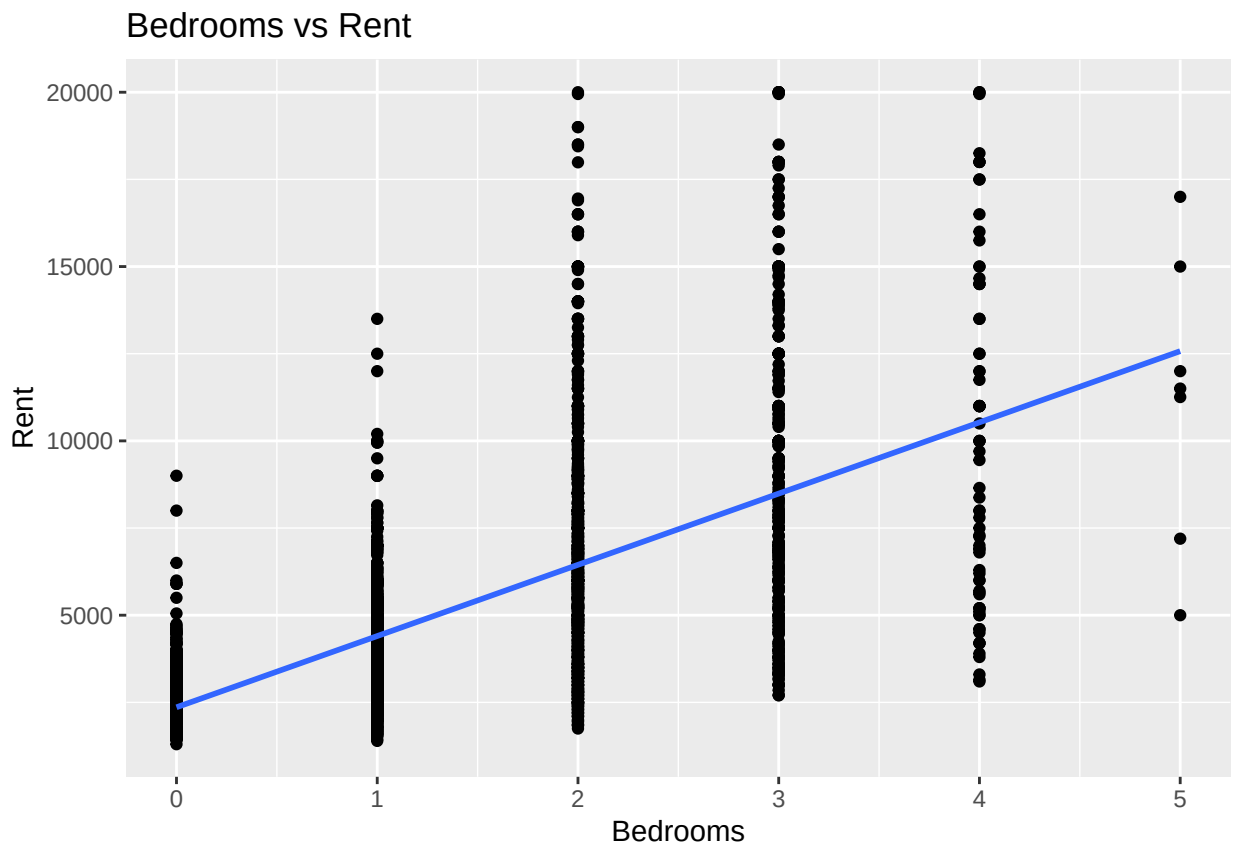
```
## # A tibble: 7 x 3
##   neigh_selected     mean_rent mean_sqft
##   <chr>                  <dbl>     <dbl>
## 1 Chelsea                6145.      961.
## 2 Upper East Side        5560.     1023.
## 3 <NA>                   5234.      944.
## 4 Upper West Side        5225.      959.
## 5 Financial District     4294.      838.
## 6 Midtown West           4148.      772.
## 7 Central Harlem         2974.      917.
```

Interesting it shows that when factoring for individual neighborhoods the correlation weakens. A reason by this might be that some neighborhoods are just that expensive because of many factors even if they have smaller apartments.

Beds vs Rent

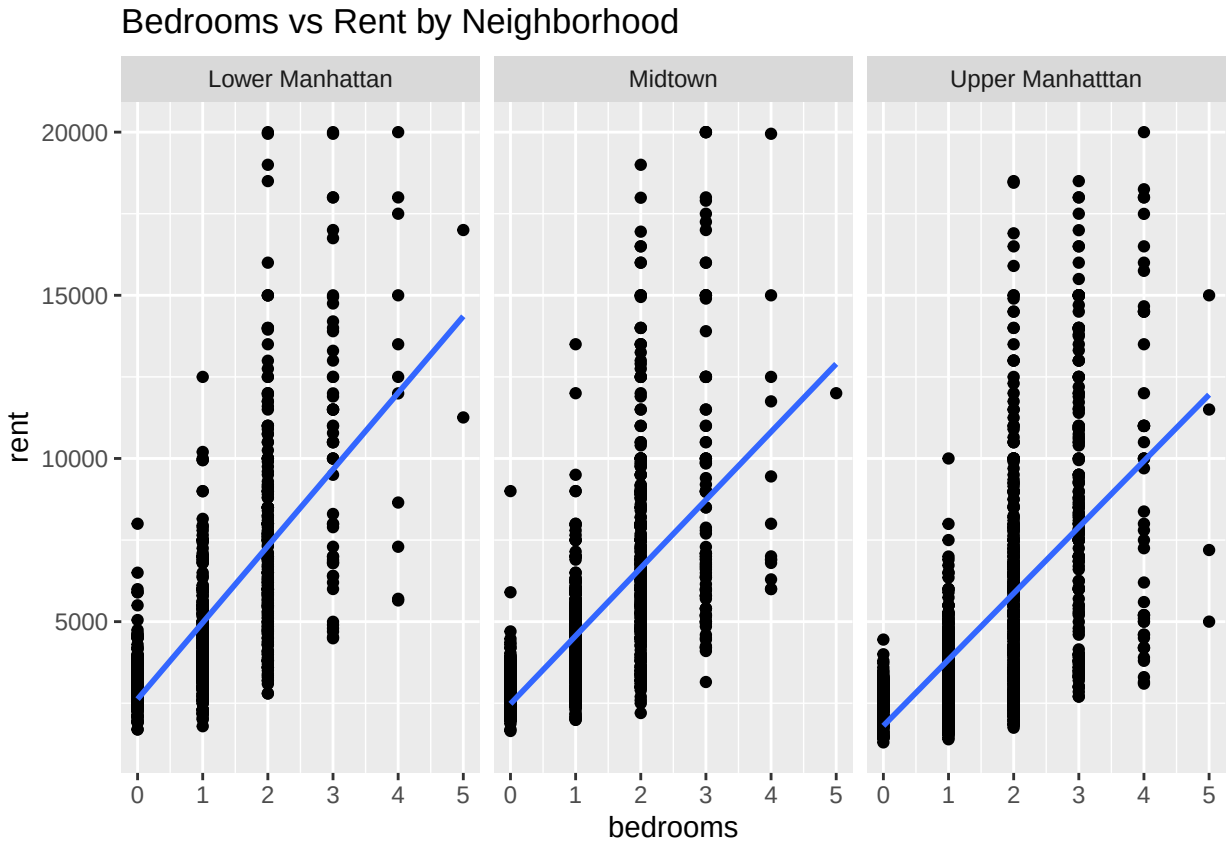
```
rental_dataset1 %>%
ggplot(aes(x=bedrooms, y= rent)) +
geom_point() +
geom_smooth(method="lm", se = FALSE) +
 ggtitle("Bedrooms vs Rent") + xlab("Bedrooms") + ylab("Rent")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

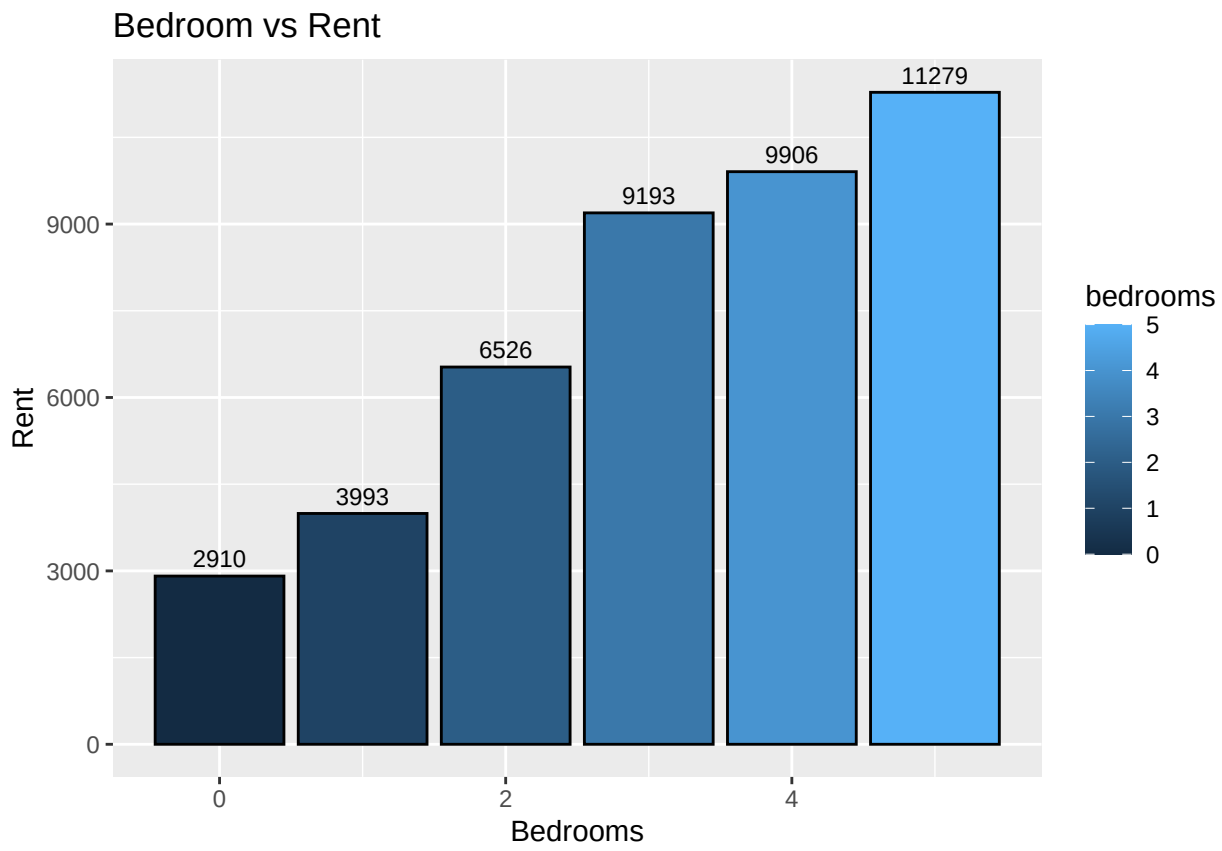



```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
           ggplot(aes(x=bedrooms, y= rent)) +
           geom_point() + geom_smooth(method="lm", se = FALSE)+
           ggtitle("Bedrooms vs Rent by Neighborhood") + facet_wrap(~location)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
rental_dataset1 %>%
  group_by(bedrooms) %>%
  summarize(value = mean(rent)) %>%
  ggplot(aes(x=bedrooms, y = value, fill = bedrooms ))+ geom_col(color = "black")+geom_text(aes(label =
```

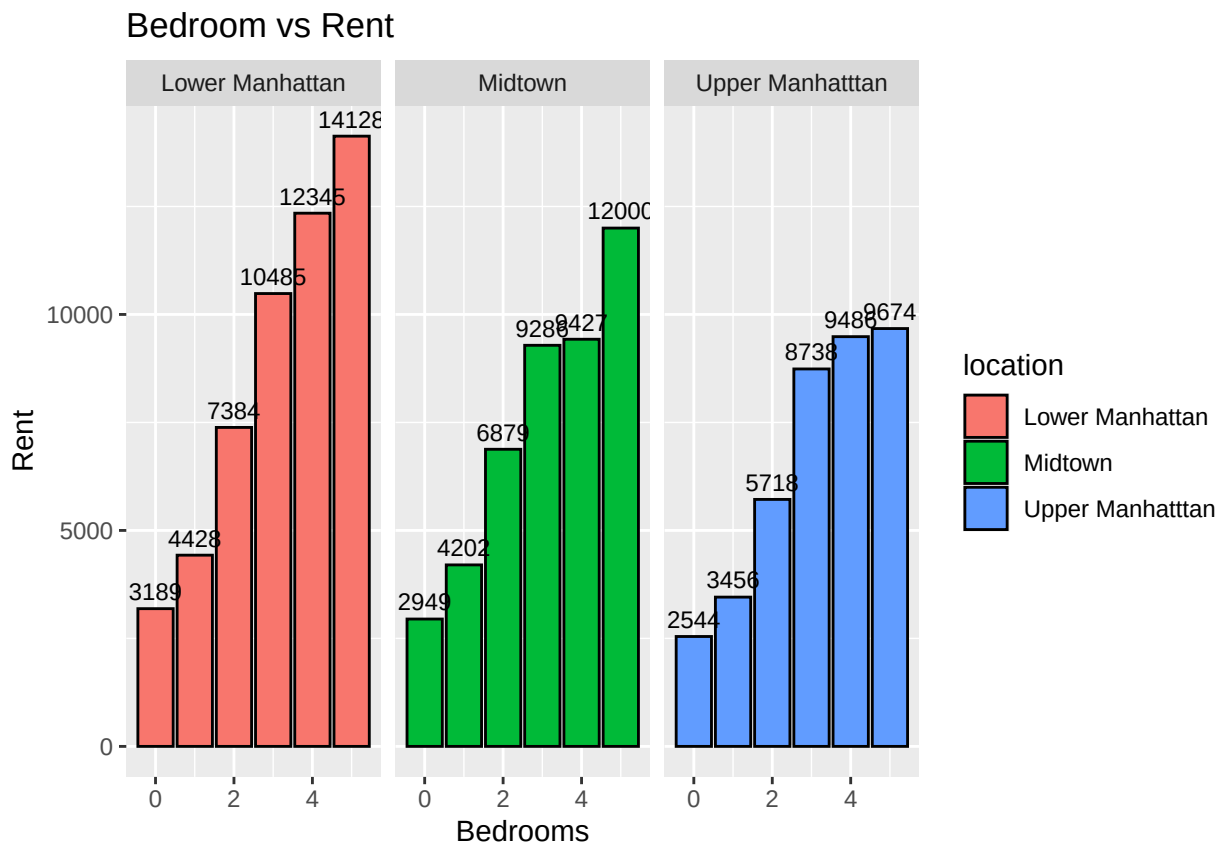


We can see that the amount of bedrooms in the apartment has a clear correlation with the average rent. The more bedrooms there are in the apartment the higher the average rent price. This makes a lot of sense as we previously saw that bedrooms correlate with square feet which also has a strong effect on rent price. There is a big spike in rent price from 1 to 2 bedrooms with rent spiking on average around 2550 dollars and there is another spike in rent price from 2 to 3 bedrooms with rent spiking around 2700 dollars.

Let's see if the spike between 1-2 bedrooms and 2-3 bedrooms exists when we factor in for the diffent neighbrhoods.

```
rental_dataset1 %>%
  group_by(bedrooms, location) %>%
  filter(!is.na(location)) %>%
  summarize(value = mean(rent)) %>%
  ggplot(aes(x=bedrooms, y = value, fill = location ))+ geom_col(color = "black") + geom_text(aes(label
```

```
## `summarise()` has grouped output by 'bedrooms'. You can override using the
## `.groups` argument.
```

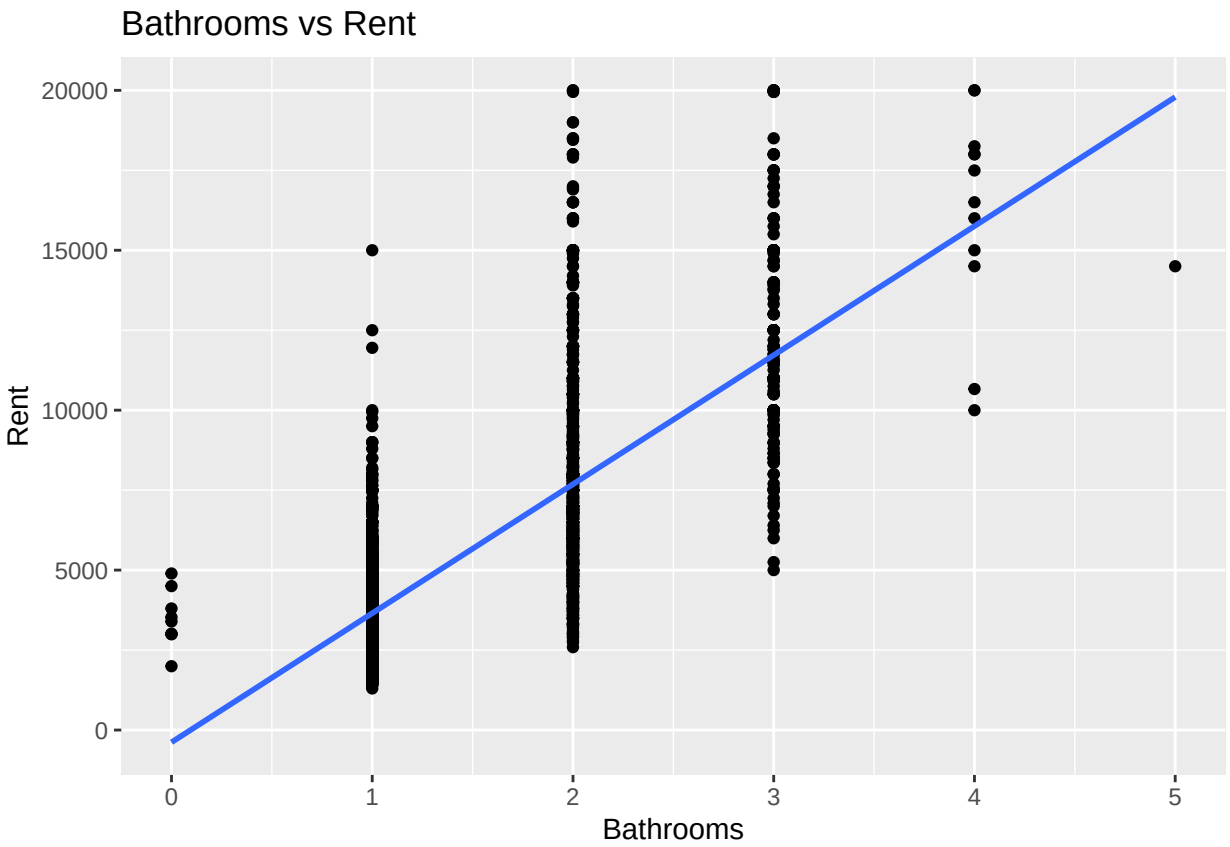


We can see that there is a similar spike in price between 1-2 bedrooms and 2-3 bedrooms in all three neighborhoods

Bathrooms vs Rent

```
rental_dataset1 %>%
           ggplot(aes(x=bathrooms, y= rent)) +
           geom_point() + geom_smooth(method="lm", se = FALSE)+
           ggtitle("Bathrooms vs Rent") + xlab("Bathrooms") + ylab("Rent")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



As expected there is an increase between bathrooms and rent. This makes a lot of sense as a bigger apartment can fit more bathrooms hence costing more money.

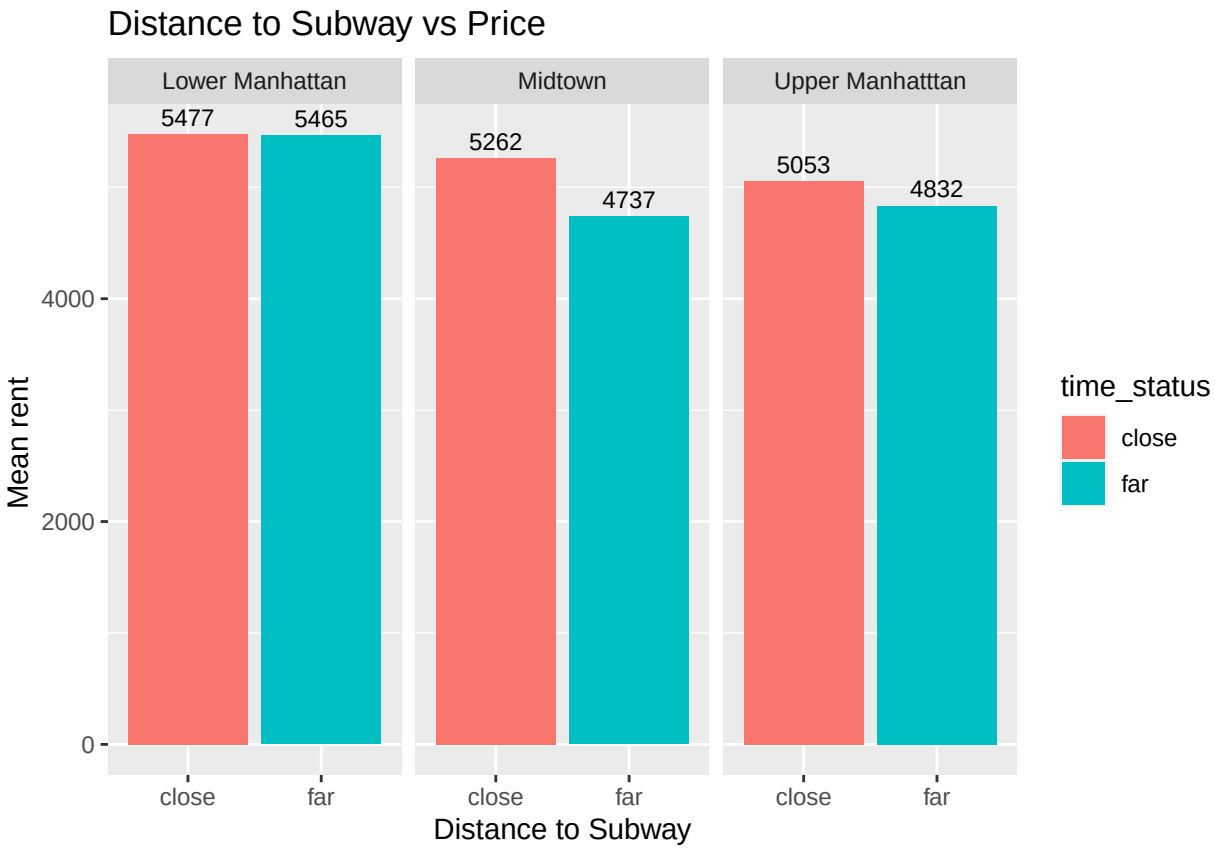
Distance to Subway vs Rent

```
rental_dataset1 %>%
  group_by(time_status) %>%
  summarize( value = mean(rent)) %>%
  ggplot(aes(x= time_status, y = value, fill = time_status))+ geom_col() + geom_text(aes(label = round(
```

We can see that there is a clear difference is average rent price between an apartment that is close to the subway and an apartment that is further away from a subway. The difference in price between a far and close apartment is about 400 dollars. Lets see now if that difference also holds between the different neighborhoods.

```
rental_dataset1 %>%
  group_by(time_status,location) %>%
  summarize( value = mean(rent)) %>%
  filter(!is.na(location)) %>%
  ggplot(aes(x= time_status, y = value, fill = time_status)) + geom_col() + geom_text(aes(label = round
```

```
## `summarise()` has grouped output by 'time_status'. You can override using the
## `.groups` argument.
```



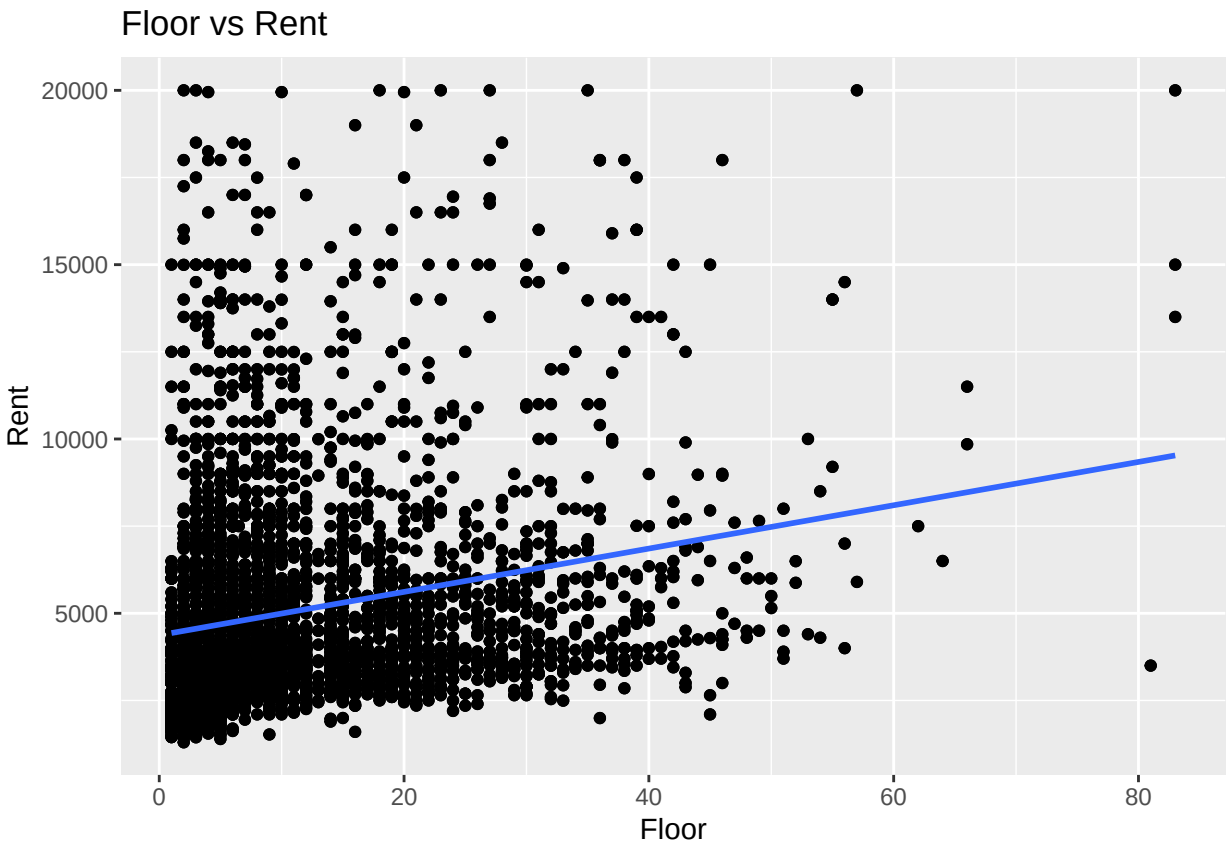
The correlation weakens but there is still a difference in rent price between close and far neighborhoods. In all three locations, an apartment that was close to the subway cost more than an apartment that was further away from the subway. A reason why the difference in price in Lower Manhattan might be so little is because of the fact that Lower Manhattan is just so expensive irregardless of the length of the subway.

Floor vs Rent

We think that floor can have an impact on rent. The higher the floor, the higher the rent the owner might charge because of the better views outside the apartment.

```
rental_dataset1 %>%
  ggplot(aes(x= floor, y = rent)) + geom_point() + xlab("Floor")+ ylab("Rent") +ggtitle("Floor vs Rent
  geom_smooth(method = 'lm', se = FALSE)
```

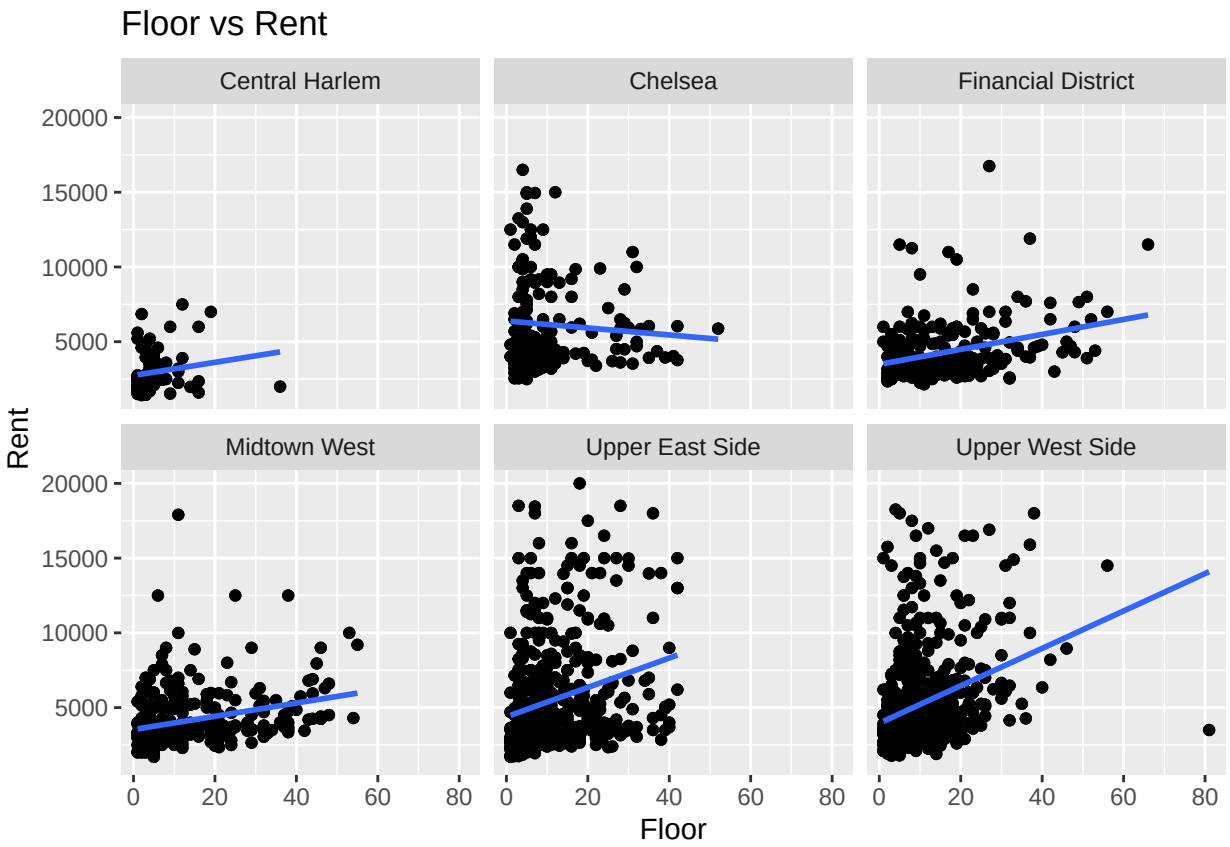
```
## `geom_smooth()` using formula 'y ~ x'
```



From the graph we can see the which floor you live does not have an impact on the rent price. The owner does not take in to account which floor the apartment rental is when determining what price to charge.

```
rental_dataset1 %>%
  filter(!is.na(neigh_selected)) %>%
  ggplot(aes(x= floor, y = rent)) + geom_point() + xlab("Floor")+ ylab("Rent") +ggtitle("Floor vs Rent
  geom_smooth(method = 'lm', se = FALSE) + facet_wrap(~neigh_selected)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

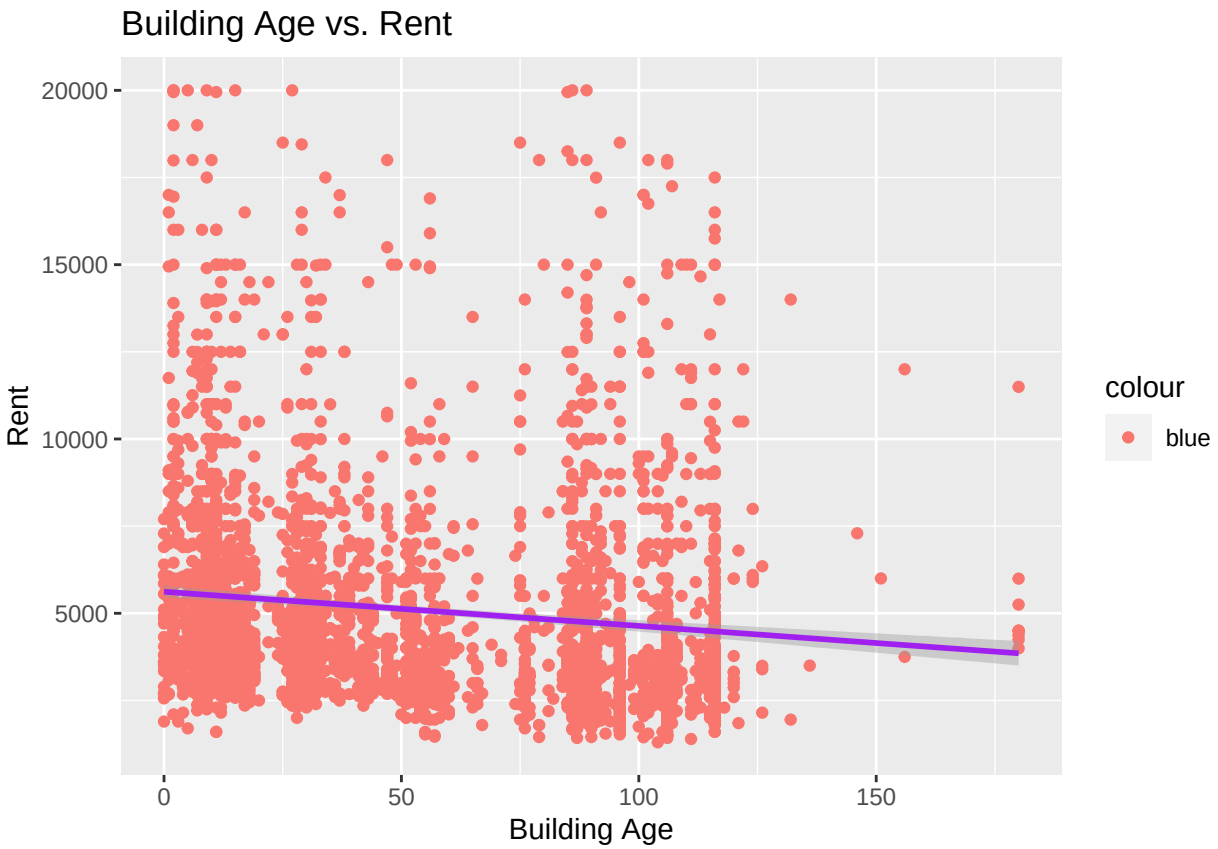


Overall, while the trend may seem weak for all the data in Manhattan, when we facet by the selected neighborhoods, we can see that there is actually a stronger, positive correlation between floor and rent in certain neighborhoods. This is evident in the Upper West Side and the Upper East Side.

Building Age vs Rent

```
rental_dataset1 %>%
ggplot(aes(x=building_age_yrs, y= rent, color = "blue")) +
geom_point() + geom_smooth(method="lm", color="purple") +
xlab("Building Age") + ylab("Rent")+
  ggtitle("Building Age vs. Rent")
```

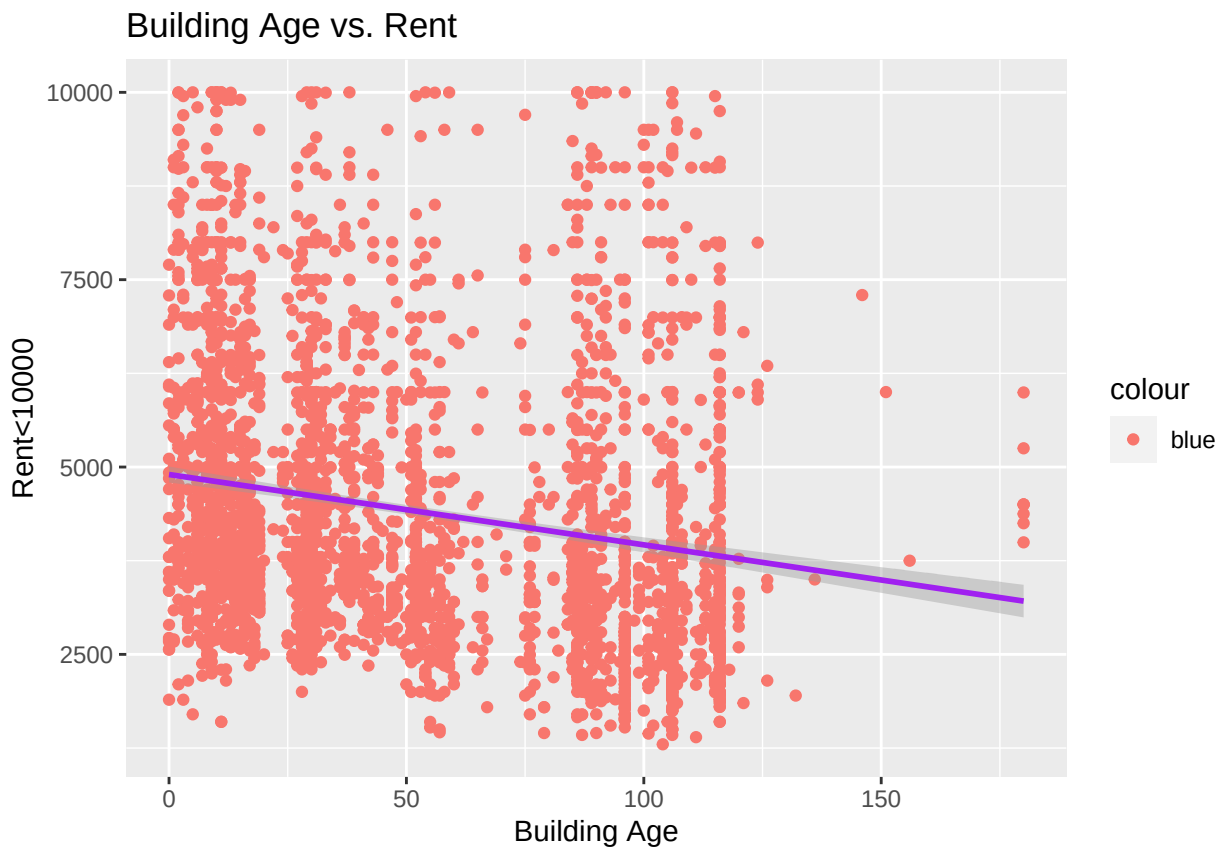
```
## `geom_smooth()` using formula 'y ~ x'
```



There seems to be a very weak negative correlation between building age and rent. For the most part the owner can charge whatever rental price he wants regardless of the age of the building. Lets see if the correlation increases if we break up the price range.

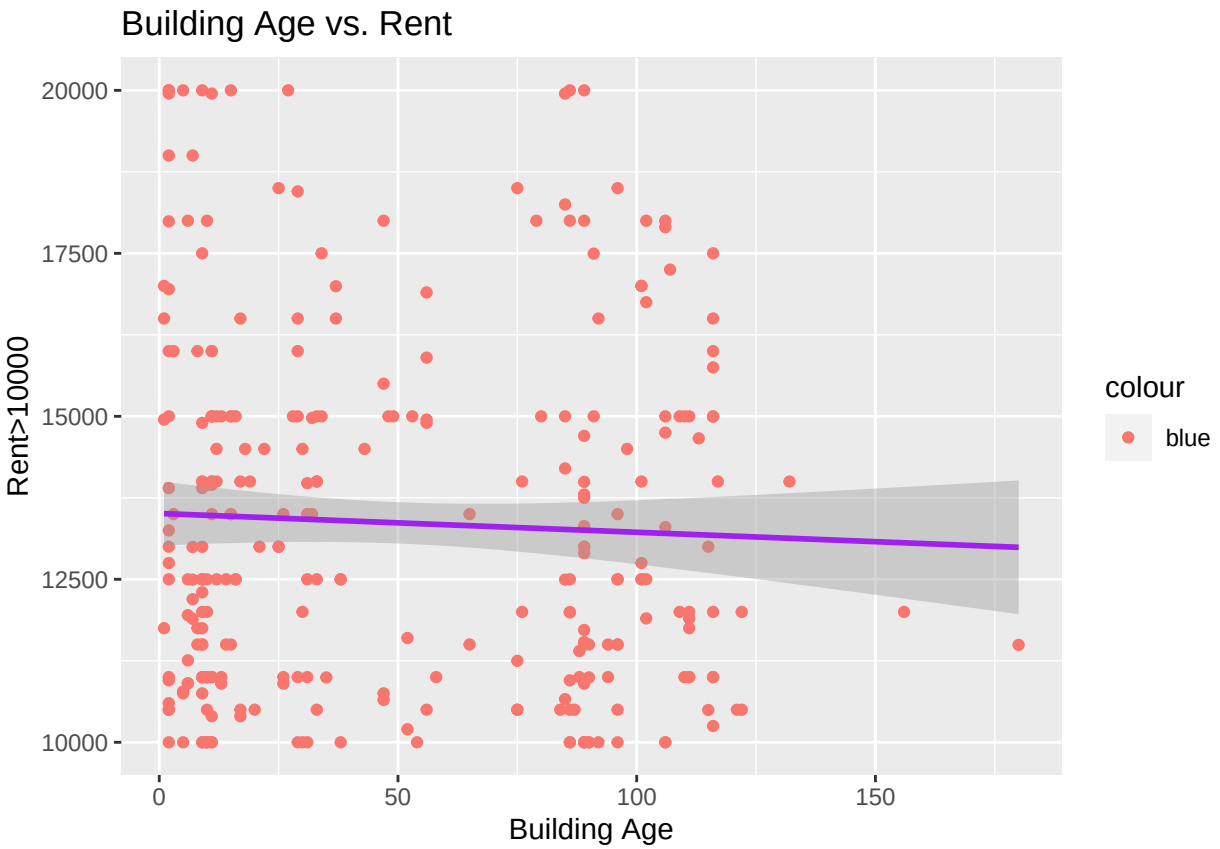
```
rental_dataset1 %>%
filter(rent<=10000) %>%
ggplot(aes(x=building_age_yrs, y= rent, color = "blue")) +
geom_point() + geom_smooth(method="lm", color="purple") +
xlab("Building Age") + ylab("Rent<10000") +
  ggtitle("Building Age vs. Rent")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
rental_dataset1 %>%
filter(rent>=10000) %>%
ggplot(aes(x=building_age_yrs, y= rent, color = "blue")) +
geom_point() + geom_smooth(method="lm", color="purple") +
xlab("Building Age") + ylab("Rent>10000") +
  ggtitle("Building Age vs. Rent")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Even after creating two graphs based on price we can see that the correlation really does not change. There is weak negative correlation under 5,000 between building age and rent, but when you look at rent prices above 5,000 dollars there is zero correlation. For the most part, the owner does not take into the account of the building age when thinking what rent to charge.

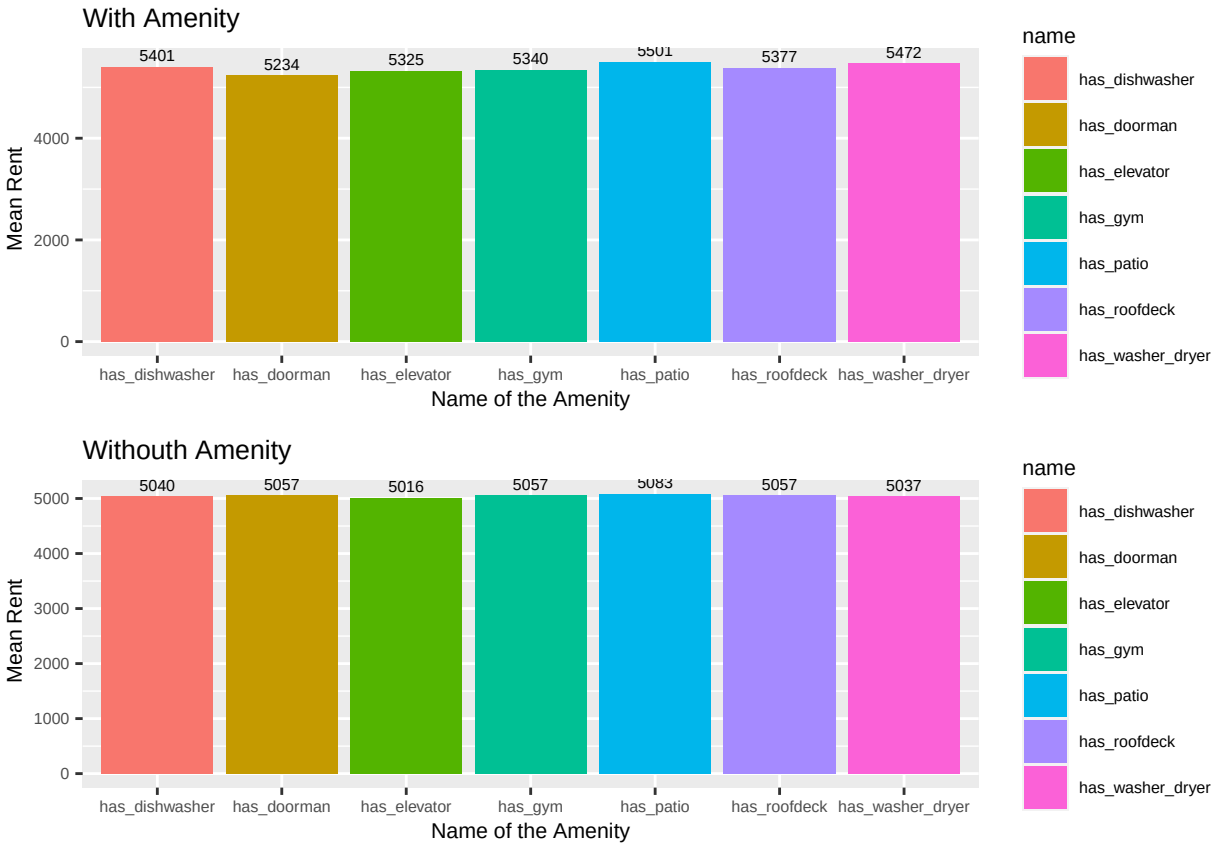
Amenities vs Rent

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(ggrepel)

g1 <- rental_dataset1 %>%
  pivot_longer(cols = starts_with("has")) %>%
  filter(value==1) %>%
  group_by(name) %>%
  summarize(mean_rent = mean(rent)) %>%
  ggplot(aes(x=name,y= mean_rent, fill = name)) +
  geom_col() +theme(text = element_text(size=8))+ geom_text(aes(label = round(mean_rent)), vjust = -0.5
```

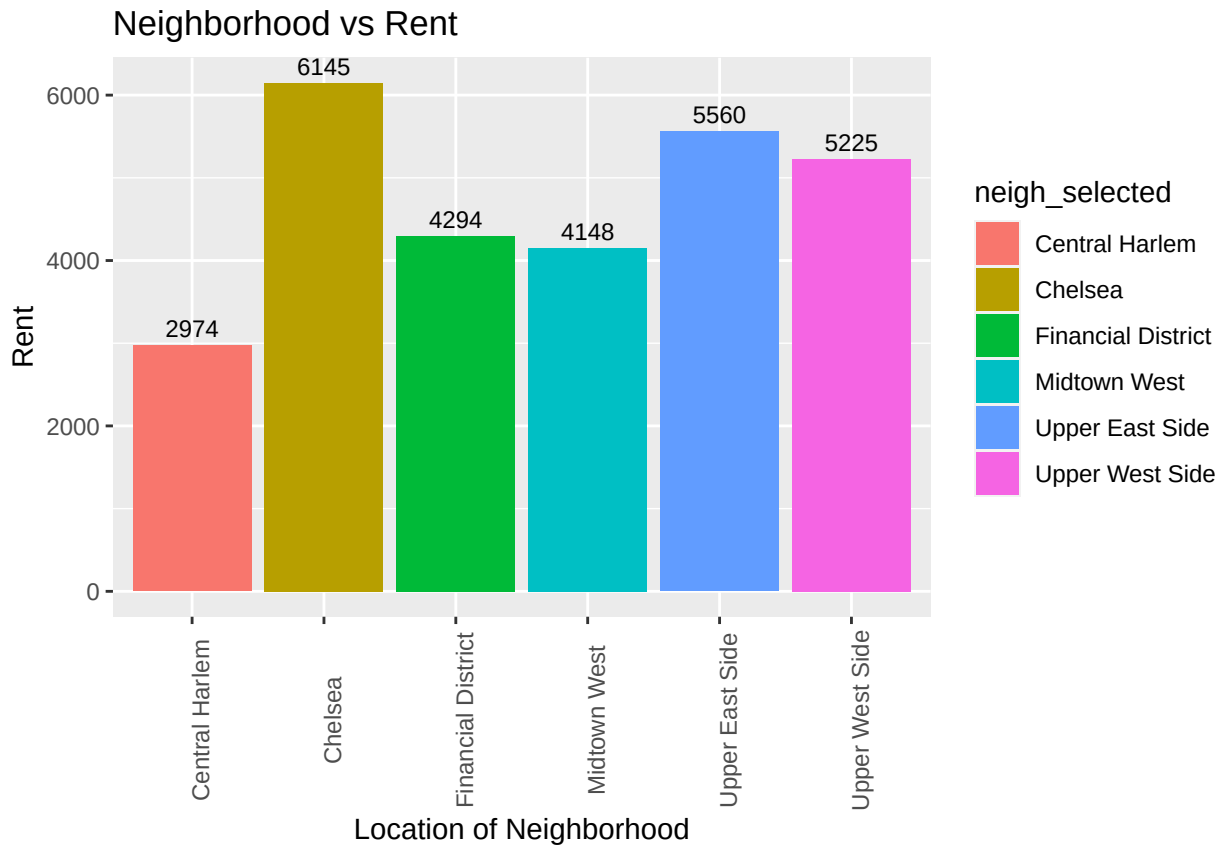
```
g2 <- rental_dataset1 %>%
  pivot_longer(cols = starts_with("has")) %>%
  filter(value==0) %>%
  group_by(name) %>%
  summarize(mean_rent = mean(rent)) %>%
  ggplot(aes(x=name,y= mean_rent, fill = name)) + geom_col()+theme(text = element_text(size=8)) +
  geom_text(aes(label = round(mean_rent)), vjust = -0.5, size=2)+ xlab("Name of the Amenity") +
  ylab(" Mean Rent") + ggtitle("Withouth Amenity")

grid.arrange(g1,g2)
```



We can see that amenities clearly has an effect on the price of a rental apartment with some amenities having a greater affect on rent than other amenities

Neighborhood vs Rent

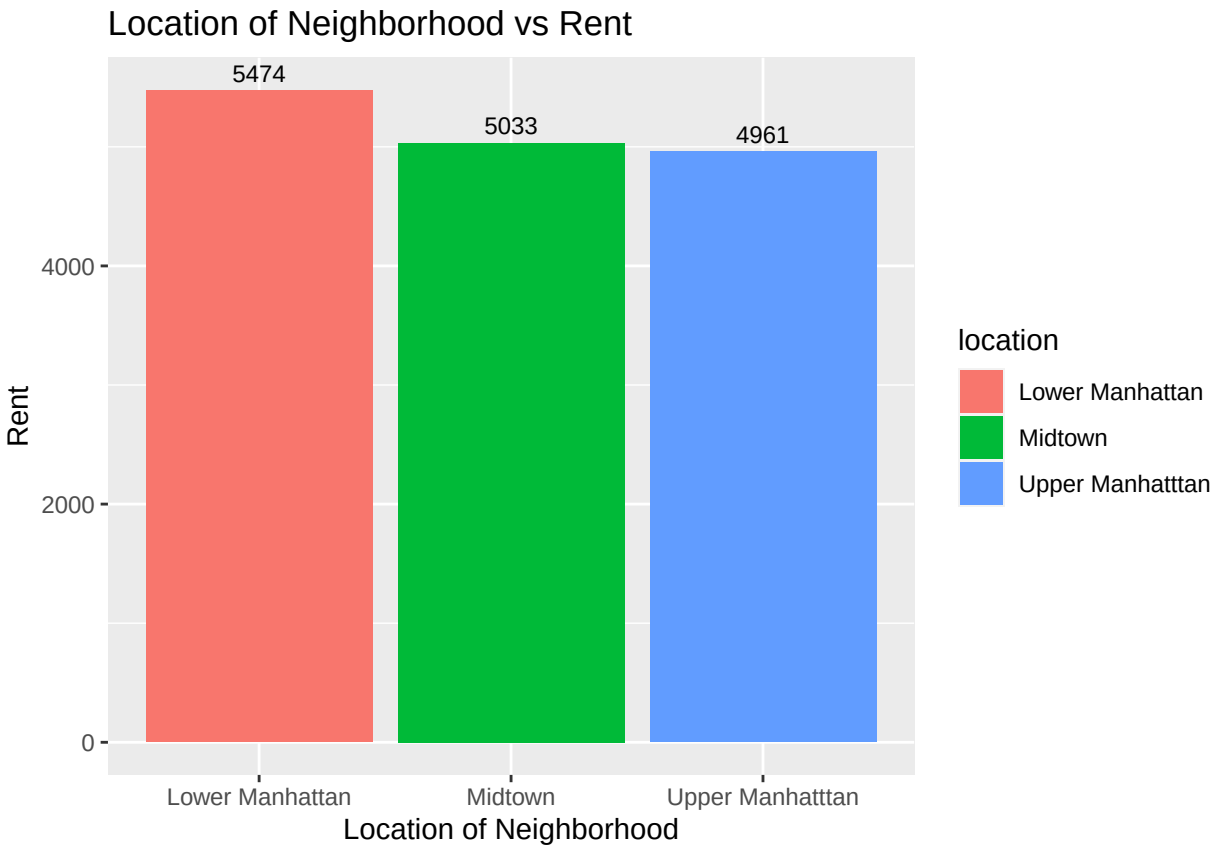
```
rental_dataset1 %>%
  filter(!is.na(neigh_selected)) %>%
  group_by(neigh_selected) %>%
  summarize(value = mean(rent)) %>%
  ggplot(aes(x=neigh_selected,y = value, fill = neigh_selected))+ geom_col() + xlab("Location of Neighb
  ggtitle("Neighborhood vs Rent")+ geom_text(aes(label = round(value)), vjust = -0.5, size=3) +
  theme(axis.text.x = element_text(angle = 90))
```



This graph shows that the neighborhood you live in has a strong influence on how much an apartment will cost. From the graph we can see the Central Harlem is the cheapest neighborhood to live in while Chelsea is the most expensive neighborhood to live in.

Location vs Rent

```
rental_dataset1 %>%
  filter(!is.na(location)) %>%
  group_by(location) %>%
  summarize(value = mean(rent)) %>%
  ggplot(aes(x=location,y = value, fill = location))+ geom_col() + xlab("Location of Neighborhood") + yl
```



As seen in the previous graph, Lower Manhattan is more expensive than both Midtown and Upper Manhattan by about 500 dollars. Midtown and Upper Manhattan are both similar in price.

3.4: CHOOSING THE VARIABLES FOR OUR MODEL

Based on our exploratory data analysis, if we were to pick just two variables at this stage, we would pick the size in square feet and neighborhood to be the most significant factors that affect rent price. However, we also want to include other variables in our model to determine which variables affect rent price through linear regression and to understand how much variation of the rent price can be explained by the different variables .

SECTION 4: MODELLING

We choose to use a linear regression model with multiple explanatory variables. Let's create and use a training data set to build our model.

CREATING TEST AND TRAIN DATASET

```
library(caTools)
set.seed(1)
dataset_split <- sample.split(rental_dataset1, SplitRatio = 0.8)

train <- subset(rental_dataset1, dataset_split == "TRUE")
test <- subset(rental_dataset1, dataset_split == "FALSE")
```

Because we have categorical variables such as neighborhood and location, we want to ensure that the linear regression model includes them appropriately.

```
rental_dataset1$neighborhood = as.factor(rental_dataset1$neighborhood)
```

UNIVARIATE ANALYSIS AND INDIVIDUAL R^2 VALUES

First, let's look at univariate models for each variable. We can take the R^2 values and take a first glimpse at how much the variation of rent price can be explained by the variables, separately. We can also see if our assumptions about the most significant variables are reflected through the individual linear model.

BEDROOMS

```
bedrooms_lm <- lm(rent~bedrooms, data = train)
df <- data.frame(bedrooms = seq(0,5,1))

predict <- df %>%
  mutate(rent_price = predict(bedrooms_lm, df))
predict
```

```
##   bedrooms rent_price
## 1        0   2348.156
## 2        1   4392.756
## 3        2   6437.355
## 4        3   8481.955
## 5        4  10526.554
## 6        5  12571.154
```

```
bedrooms_rs <- summary(bedrooms_lm)$r.squared
```

The model reflects our assumption that the positive relationship between rent price and bedrooms is relatively strong.

BATHROOMS

```
bathrooms_lm <- lm(rent~bathrooms, data = train)
df <- data.frame(bathrooms = seq(1,5,1))

predict <- df %>%
  mutate(rent_price = predict(bathrooms_lm, df))
predict
```

```
##   bathrooms rent_price
## 1         1   3642.594
## 2         2   7673.919
## 3         3  11705.244
## 4         4  15736.569
## 5         5  19767.894
```

```
bathrooms_rs <- summary(bathrooms_lm)$r.squared
```

The model reflects our assumption that the positive relationship between rent price and bathrooms is relatively strong.

SIZE_SQFT

```
sqft_lm <- lm(rent~size_sqft, data = train)
df <- data.frame(size_sqft = seq(500,3500,500))

predict <- df %>%
  mutate(rent_price = predict(sqft_lm, df))
predict
```

```
##   size_sqft rent_price
## 1       500   2628.917
## 2      1000   5465.290
## 3      1500   8301.663
## 4      2000  11138.035
## 5      2500  13974.408
## 6      3000  16810.781
## 7      3500  19647.153
```

```
sqft_rs <- summary(sqft_lm)$r.squared
```

The model illustrates that the positive relationship between rent price and size is relatively strong, as expected.

BUILDING AGE

```
ba_lm <- lm(rent~building_age_yrs, data = train)
df <- data.frame(building_age_yrs= seq(10,180, 20))

predict <- df %>%
  mutate(rent_price = predict(ba_lm, df))
predict
```

```
##   building_age_yrs rent_price
## 1               10   5481.699
## 2               30   5303.654
## 3               50   5125.609
## 4               70   4947.564
## 5               90   4769.518
## 6              110   4591.473
## 7              130   4413.428
## 8              150   4235.383
## 9              170   4057.338
```

```
ba_rs <- summary(ba_lm)$r.squared
```

As expected from our exploratory data analysis, the model illustrates the weak negative correlation between building age and rent price.

HAS_WASHER_DRYER

```
ws_lm <- lm(rent~has_washer_dryer, data = train)
df <- data.frame(has_washer_dryer= unique(train$has_washer_dryer))

predict <- df %>%
  mutate(rent_price = predict(ws_lm, df))
predict
```

```
##   has_washer_dryer rent_price
## 1                0   5046.387
## 2                1   5381.420
```

```
ws_rs <- summary(ws_lm)$r.squared
```

The linear model shows that there is a decent jump in price, about 500 dollars, to get an apartment that includes a laundry unit, which is similar to the difference in average rent price found in our exploratory data analysis.

```
dw_lm <- lm(rent~has_dishwasher, data = train)
dw_rs <- summary(dw_lm)$r.squared

p_lm <- lm(rent~has_patio, data = train)
p_rs <- summary(p_lm)$r.squared

rd_lm <- lm(rent~has_roofdeck, data = train)
rd_rs <- summary(rd_lm)$r.squared

dm_lm <- lm(rent~has_doorman, data = train)
dm_rs <- summary(dm_lm)$r.squared

he_lm <- lm(rent~has_elevator, data = train)
he_rs <- summary(he_lm)$r.squared

gym_lm <- lm(rent~has_gym, data = train)
gym_rs <- summary(gym_lm)$r.squared

neighborhood_lm <- lm(rent~neighborhood, data = train)
neighborhood_rs <- summary(neighborhood_lm)$r.squared

name <- c("bedrooms", "bathrooms", "size_sqft",
          "building age", "has_washer_dryer", "has_dishwasher",
          "has_patio", "has_roofdeck", "has_doorman", "has_elevator",
          "has_gym","neighborhood")
r_squared <- c(bedrooms_rs, bathrooms_rs, sqft_rs, ba_rs, ws_rs, dw_rs, p_rs, rd_rs, dm_rs, he_rs, gym_
df_r <- data.frame(name, r_squared)
df_r %>%
  arrange(desc(r_squared))
```

```
##                name    r_squared
## 1         size_sqft 0.7345755034
## 2         bathrooms 0.5842202969
## 3          bedrooms 0.3995497612
## 4      neighborhood 0.1300320918
## 5      building age 0.0123465789
## 6    has_dishwasher 0.0017083927
## 7  has_washer_dryer 0.0015252582
## 8      has_elevator 0.0012457541
## 9         has_patio 0.0011529886
## 10     has_roofdeck 0.0006867992
## 11          has_gym 0.0004786394
## 12      has_doorman 0.0004743641
```

Similar to our prediction, the size in square feet is the most significant factor here with the largest r-squared value. It also makes sense that size in square feet is followed by bathrooms and bedrooms. However, the r-squared value of neighborhood is lower than we thought it would be.

MULTIVARIATE ANALYSIS

```
rent_lm <- lm(rent~ size_sqft + bathrooms + floor + building_age_yrs +
                              has_roofdeck + has_washer_dryer + has_doorman +
                              has_elevator + has_dishwasher + has_patio + has_gym
                            + neighborhood, train)
summary(rent_lm)
```

```
##
## Call:
## lm(formula = rent ~ size_sqft + bathrooms + floor + building_age_yrs +
##     has_roofdeck + has_washer_dryer + has_doorman + has_elevator +
##     has_dishwasher + has_patio + has_gym + neighborhood, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6954.6  -599.2   -83.7   437.0 13273.0
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    -4.632e+02  1.873e+02  -2.473 0.013448 *
## size_sqft                       4.510e+00  9.712e-02  46.435  < 2e-16 ***
## bathrooms                       9.452e+02  7.632e+01  12.384  < 2e-16 ***
## floor                           2.893e+01  2.882e+00  10.039  < 2e-16 ***
## building_age_yrs               -5.710e+00  7.728e-01  -7.389 1.96e-13 ***
## has_roofdeck                   -5.014e+01  9.112e+01  -0.550 0.582195
## has_washer_dryer                1.278e+02  8.253e+01   1.548 0.121640
## has_doorman                    -6.720e+01  8.937e+01  -0.752 0.452116
## has_elevator                    7.419e+01  9.167e+01   0.809 0.418406
## has_dishwasher                 -1.688e+01  7.947e+01  -0.212 0.831770
## has_patio                      -9.555e+01  1.170e+02  -0.816 0.414356
## has_gym                         3.217e+01  9.987e+01   0.322 0.747403
## neighborhoodCentral Harlem     -1.826e+03  2.416e+02  -7.557 5.59e-14 ***
## neighborhoodCentral Park South  7.680e+02  3.598e+02   2.135 0.032858 *
## neighborhoodChelsea             8.801e+02  2.072e+02   4.247 2.24e-05 ***
## neighborhoodChinatown          -3.126e+02  6.352e+02  -0.492 0.622624
## neighborhoodEast Harlem        -1.372e+03  3.038e+02  -4.514 6.64e-06 ***
## neighborhoodEast Village        4.797e+02  2.269e+02   2.114 0.034582 *
## neighborhoodFinancial District -3.079e+02  2.005e+02  -1.535 0.124850
## neighborhoodFlatiron            8.227e+02  2.169e+02   3.793 0.000152 ***
## neighborhoodGramercy Park       5.999e+02  2.548e+02   2.355 0.018602 *
## neighborhoodGreenwich Village   9.173e+02  2.531e+02   3.624 0.000296 ***
## neighborhoodHamilton Heights   -1.549e+03  4.184e+02  -3.702 0.000218 ***
## neighborhoodInwood             -1.959e+03  4.659e+02  -4.205 2.69e-05 ***
## neighborhoodLittle Italy        2.660e+02  8.091e+02   0.329 0.742339
## neighborhoodLong Island City   -1.134e+03  8.076e+02  -1.404 0.160512
## neighborhoodLower East Side    -2.993e+02  2.981e+02  -1.004 0.315332
## neighborhoodManhattanville     -8.528e+02  1.376e+03  -0.620 0.535545
## neighborhoodMidtown             6.847e+01  2.279e+02   0.300 0.763852
## neighborhoodMidtown East       -3.559e+02  1.864e+02  -1.910 0.056257 .
```

```
## neighborhoodMidtown South       -8.653e+01  2.368e+02  -0.365 0.714843
## neighborhoodMidtown West        -2.074e+02  1.928e+02  -1.075 0.282301
## neighborhoodMorningside Heights -1.091e+03  4.892e+02  -2.230 0.025827 *
## neighborhoodNolita               1.204e+03  5.441e+02   2.213 0.026950 *
## neighborhoodRoosevelt Island    -1.161e+03  8.087e+02  -1.436 0.151144
## neighborhoodSoho                 1.776e+03  2.702e+02   6.574 5.86e-11 ***
## neighborhoodStuyvesant Town/PCV -7.885e+01  8.070e+02  -0.098 0.922176
## neighborhoodTribeca              9.780e+02  2.223e+02   4.400 1.13e-05 ***
## neighborhoodUpper East Side     -1.303e+02  1.856e+02  -0.702 0.482563
## neighborhoodUpper West Side      2.145e+01  1.842e+02   0.116 0.907301
## neighborhoodWashington Heights  -1.915e+03  2.773e+02  -6.906 6.18e-12 ***
## neighborhoodWest Harlem         -1.288e+03  1.376e+03  -0.936 0.349443
## neighborhoodWest Village         1.748e+03  2.598e+02   6.729 2.08e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1364 on 2712 degrees of freedom
## Multiple R-squared:  0.815,  Adjusted R-squared:  0.8121
## F-statistic: 284.4 on 42 and 2712 DF,  p-value: < 2.2e-16
```

We took out bedrooms because it can be seen as a proxy for size_sqft, also because the high correlation between bedrooms and size_sqft may increase error due to multicollinearity. Our model is relatively strong, with an R^2 value of about 0.83, meaning that about 81% of the variation of the rent price can be attributed to our variables, collectively.

PREDICTING PRICES WITH TEST DATASET

```
library(Metrics)

df <- data.frame(rent = test$rent, neighborhood = test$neighborhood)


test_rent <- df %>%
  mutate(predicted_rent = predict(rent_lm, test), diff = abs(predicted_rent - rent))

test_rent %>%
  head(n= 10) %>%
  select(-neighborhood)
```

```
##     rent predicted_rent      diff
## 1   4795       5873.156 1078.1564
## 2   1995       1259.142  735.8585
## 3   2995       3512.385  517.3855
## 4   4950       4732.459  217.5414
## 5  10000       9753.533  246.4675
## 6  10904       9006.292 1897.7079
## 7   2100       1605.238  494.7621
## 8   5485       5344.885  140.1150
## 9   1875       2177.763  302.7629
## 10  2400       2030.335  369.6651
```

```
average_diff<- mean(test_rent$diff)
```

```
root_mse <- rmse(test_rent$rent, test_rent$predicted_rent)

average_diff
```

```
## [1] 851.8417
```

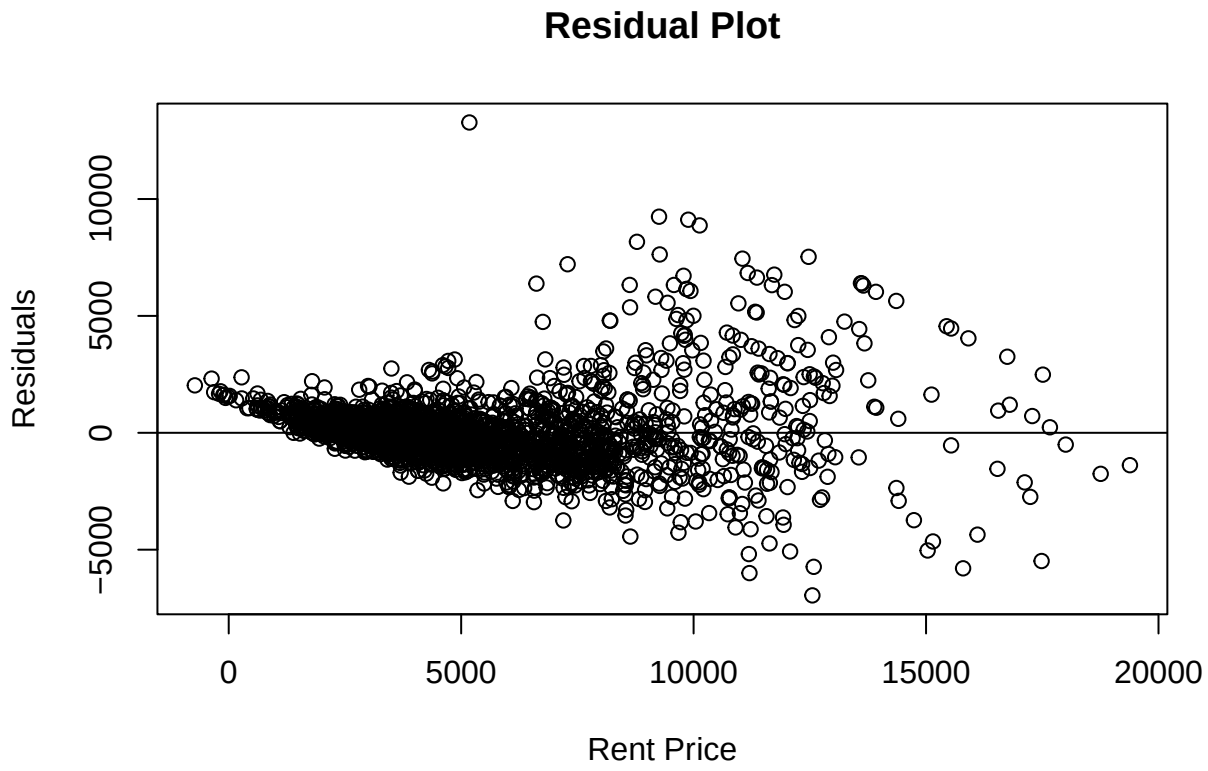
```
root_mse
```

```
## [1] 1295.129
```

Our prediction performance is not very accurate, with a average difference of around 756 dollars and a root squared mean of around 1138 dollars. However, it is not terribly off either. One could say that our prediction performance is moderately weak.

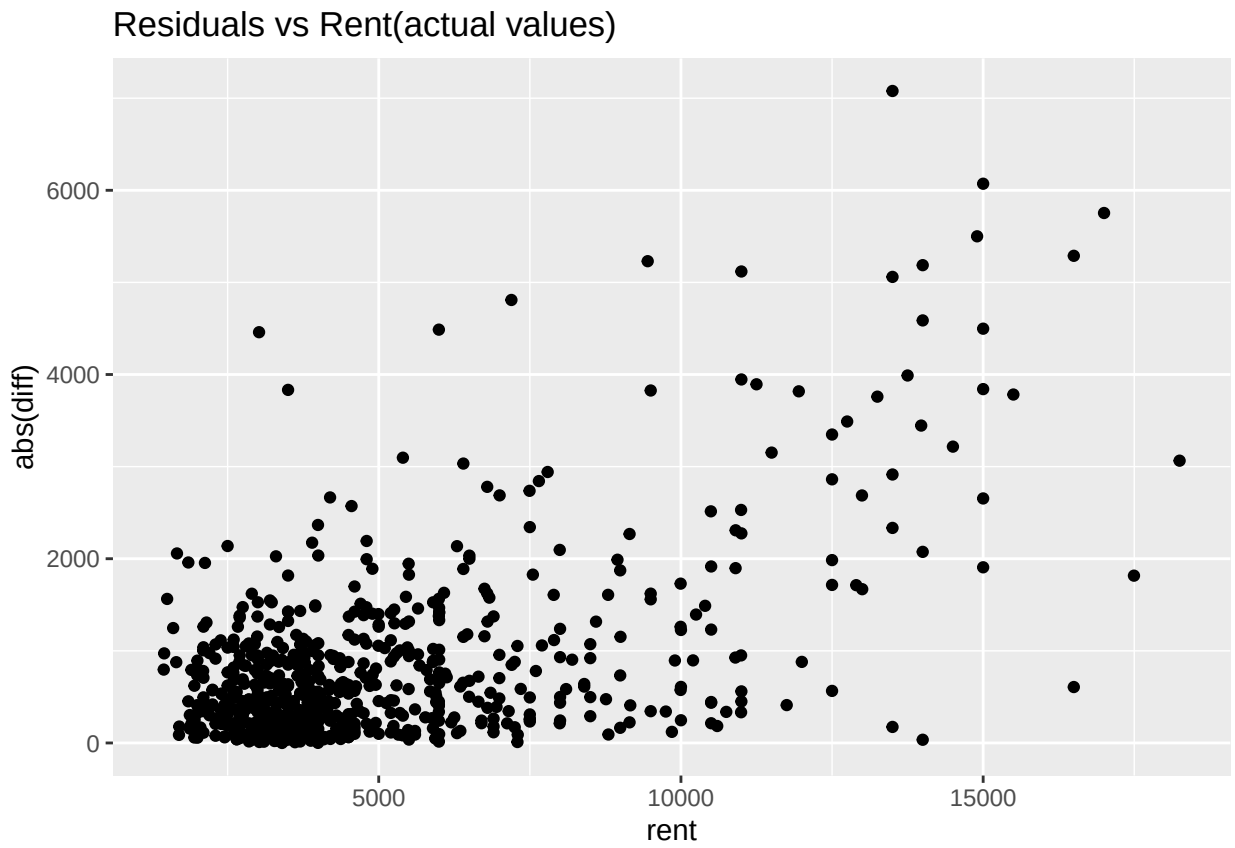
VISUALIZING OUR MODEL WITH RESIDUALS

```
res<- resid(rent_lm)
plot(fitted(rent_lm), res, ylab = "Residuals", xlab = "Rent Price", main = "Residual Plot")
abline(0,0)
```



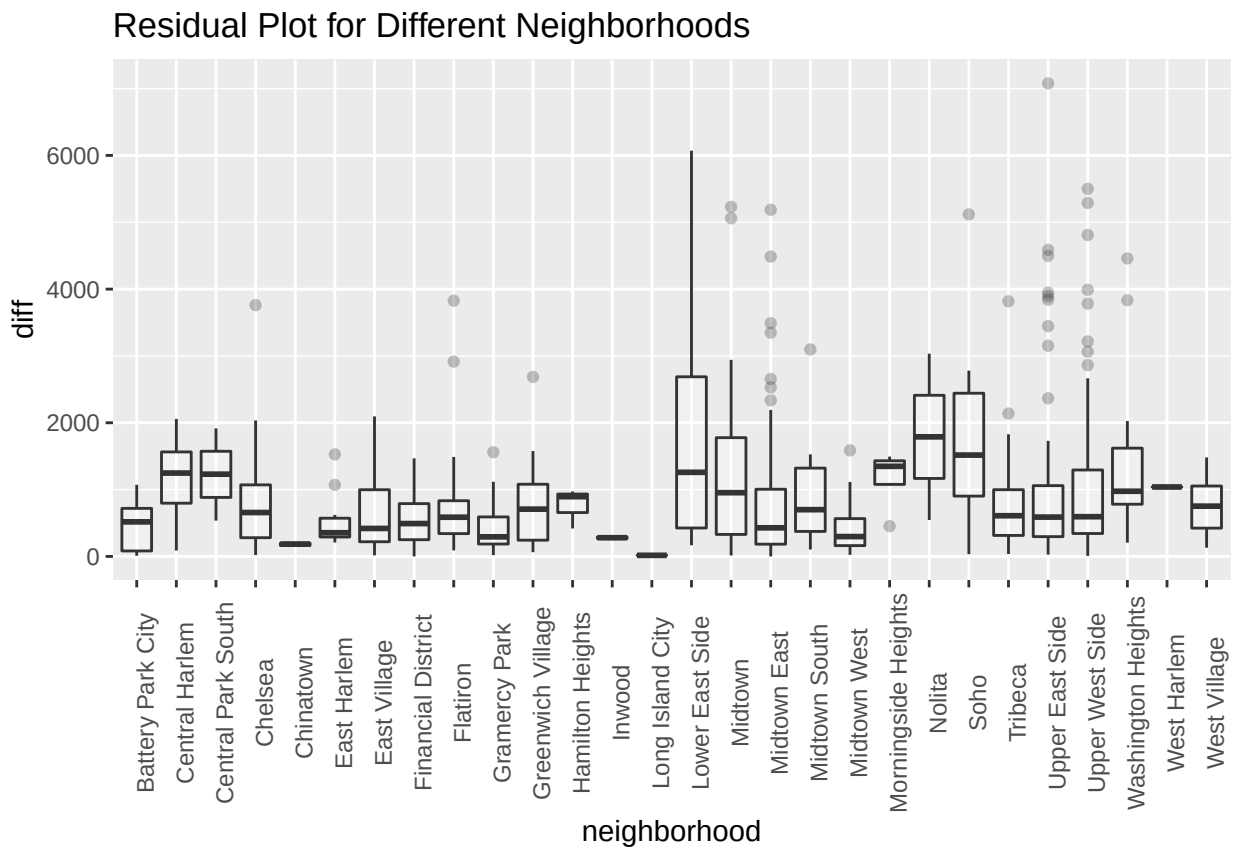
With our residual plot, we can see that the spread of residuals of our model tend to be higher for rent prices starting from about 8,000 dollars. It is clear that our model has both underfit and overfit when predicting rent prices.

```
ggplot(test_rent, aes(y = abs(diff), x = rent)) + geom_point()+
  ggtitle("Residuals vs Rent(actual values)")
```



We can see that for rents until about 5000 dollars, the model performs best as the difference between the actual and predicted price mainly ranges from 0 to 1250 dollars. However, the model performs worse and the differences are spread out past 5000 dollars, especially in the higher price ranges.

```
test_rent %>%
  ggplot(aes(x = neighborhood, y = diff)) + geom_boxplot(alpha = 0.3) +
  theme(axis.text.x = element_text(angle = 90))+ ggtitle("Residual Plot for Different Neighborhoods")
```



It is interesting that for some neighborhoods, our model was not good at predicting the rent prices within certain neighborhoods. For Central Park South, Soho, Nolita, and Washington Heights, the differences between the predicted rent prices and actual rent prices was the greatest.

IMPROVING OUR PREDICTED RENT PRICES WITH OUR MODEL

Even though our R^2 values were high for our linear model, our predicted prices were way off. Perhaps this is because there are too many neighborhoods. Initially, we wanted to compare all the present neighborhoods because we wanted to explore the groupings of "neighborhood' and"upper/lower/mid Manhattan". However, we realize that because there are so many neighborhoods, this may lead to increased sources of error.

Let's remove data that include the neighborhoods we identified earlier that our model did not predict rent prices well for: Central Park South, Soho, Nolita, and Washington Heights.

```
train2 <- train %>%
  filter(!neighborhood == c("Central Park South", "Soho", "Nolita", "Washington Heights"))
```

```
## Warning in neighborhood == c("Central Park South", "Soho", "Nolita", "Washington
## Heights"): longer object length is not a multiple of shorter object length
```

```
rent_lm1 <- lm(rent~ size_sqft  + floor + bathrooms + building_age_yrs +has_roofdeck + has_washer_dryer
has_elevator + has_dishwasher + has_patio + has_gym
                              + neighborhood, data = train2)

df <- data.frame(rent = train2$rent)

train2_rent <- df %>%
```

```
  mutate(predicted_rent = predict(rent_lm1, train2), diff = abs(predicted_rent - rent))

train2_rent %>%
  head(n= 10)
```

```
##     rent predicted_rent       diff
## 1   2550       2496.463   53.53688
## 2  11500      10876.091  623.90898
## 3   4500       5916.033 1416.03289
## 4   3800       4122.507  322.50658
## 5  15000      10860.662 4139.33785
## 6   4650       4025.425  624.57510
## 7   6920       7612.105  692.10518
## 8   4875       5270.776  395.77620
## 9   4850       4927.418   77.41816
## 10  3700       4410.571  710.57080
```

```
mean(train2_rent$diff)
```

```
## [1] 836.8752
```

Unfortunately, removing neighborhoods that had greater residuals did not greatly improve our rent prediction, as the mean difference is about 874 dollars compared to our initial test, which was 766 dollars - our predictions worsened.

Let's see if we can improve our prediction prices further only by selecting a few neighborhoods that are near each other in Manhattan; with less neighborhoods, perhaps our model can fit the data better. Let's also filter the rent prices to be 5000 dollars or less.

```
train3 <- train %>%
  filter(neighborhood == c("Upper East Side", "Manhattanville", "Upper West Side", "Hamilton Heights",
  filter(rent < 5000)
```

```
## Warning in neighborhood == c("Upper East Side", "Manhattanville", "Upper West
## Side", : longer object length is not a multiple of shorter object length
```

```
rent_lm2 <- lm(rent~ size_sqft + floor + bathrooms + building_age_yrs +has_roofdeck + has_washer_dryer +

df <- data.frame(rent = train3$rent)

train3_rent <- df %>%
  mutate(predicted_rent = predict(rent_lm2, train3), diff = abs(predicted_rent - rent))

train3_rent %>%
  head(n= 10)
```

```
##    rent predicted_rent      diff
## 1  2550       2583.121  33.12098
## 2  3375       3404.756  29.75598
## 3  2950       3465.344 515.34418
## 4  4995       4491.882 503.11841
```

```
## 5  4095       3787.580 307.42047
## 6  2799       3095.874 296.87387
## 7  3195       2983.421 211.57863
## 8  3100       3477.455 377.45523
## 9  1750       1402.115 347.88497
## 10 2695       2971.249 276.24915
```

```
mean(train3_rent$diff)
```

```
## [1] 314.5931
```

Filtering the rent to be less than 5000 dollars and limiting our data to a small number of neighborhoods significantly improved our rent prediction. We went from a mean difference of 766 and 874 dollars to about 276 dollars, which is a significant improvement.

COEFFICIENT ANALYSIS

```
rent_lm <- lm(rent~ size_sqft + bathrooms + floor + building_age_yrs +
                              has_roofdeck + has_washer_dryer + has_doorman +
                              has_elevator + has_dishwasher + has_patio + has_gym
                            + neighborhood, train)
summary(rent_lm)
```

```
##
## Call:
## lm(formula = rent ~ size_sqft + bathrooms + floor + building_age_yrs +
##     has_roofdeck + has_washer_dryer + has_doorman + has_elevator +
##     has_dishwasher + has_patio + has_gym + neighborhood, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6954.6  -599.2   -83.7   437.0 13273.0
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -4.632e+02  1.873e+02  -2.473 0.013448 *
## size_sqft                      4.510e+00  9.712e-02  46.435  < 2e-16 ***
## bathrooms                      9.452e+02  7.632e+01  12.384  < 2e-16 ***
## floor                          2.893e+01  2.882e+00  10.039  < 2e-16 ***
## building_age_yrs              -5.710e+00  7.728e-01  -7.389 1.96e-13 ***
## has_roofdeck                  -5.014e+01  9.112e+01  -0.550 0.582195
## has_washer_dryer               1.278e+02  8.253e+01   1.548 0.121640
## has_doorman                   -6.720e+01  8.937e+01  -0.752 0.452116
## has_elevator                   7.419e+01  9.167e+01   0.809 0.418406
## has_dishwasher                -1.688e+01  7.947e+01  -0.212 0.831770
## has_patio                     -9.555e+01  1.170e+02  -0.816 0.414356
## has_gym                        3.217e+01  9.987e+01   0.322 0.747403
## neighborhoodCentral Harlem    -1.826e+03  2.416e+02  -7.557 5.59e-14 ***
## neighborhoodCentral Park South 7.680e+02  3.598e+02   2.135 0.032858 *
## neighborhoodChelsea            8.801e+02  2.072e+02   4.247 2.24e-05 ***
## neighborhoodChinatown         -3.126e+02  6.352e+02  -0.492 0.622624
## neighborhoodEast Harlem       -1.372e+03  3.038e+02  -4.514 6.64e-06 ***
## neighborhoodEast Village       4.797e+02  2.269e+02   2.114 0.034582 *
```

```
## neighborhoodFinancial District  -3.079e+02  2.005e+02  -1.535 0.124850
## neighborhoodFlatiron             8.227e+02  2.169e+02   3.793 0.000152 ***
## neighborhoodGramercy Park        5.999e+02  2.548e+02   2.355 0.018602 *
## neighborhoodGreenwich Village    9.173e+02  2.531e+02   3.624 0.000296 ***
## neighborhoodHamilton Heights    -1.549e+03  4.184e+02  -3.702 0.000218 ***
## neighborhoodInwood              -1.959e+03  4.659e+02  -4.205 2.69e-05 ***
## neighborhoodLittle Italy         2.660e+02  8.091e+02   0.329 0.742339
## neighborhoodLong Island City    -1.134e+03  8.076e+02  -1.404 0.160512
## neighborhoodLower East Side     -2.993e+02  2.981e+02  -1.004 0.315332
## neighborhoodManhattanville      -8.528e+02  1.376e+03  -0.620 0.535545
## neighborhoodMidtown              6.847e+01  2.279e+02   0.300 0.763852
## neighborhoodMidtown East        -3.559e+02  1.864e+02  -1.910 0.056257 .
## neighborhoodMidtown South       -8.653e+01  2.368e+02  -0.365 0.714843
## neighborhoodMidtown West        -2.074e+02  1.928e+02  -1.075 0.282301
## neighborhoodMorningside Heights -1.091e+03  4.892e+02  -2.230 0.025827 *
## neighborhoodNolita               1.204e+03  5.441e+02   2.213 0.026950 *
## neighborhoodRoosevelt Island    -1.161e+03  8.087e+02  -1.436 0.151144
## neighborhoodSoho                 1.776e+03  2.702e+02   6.574 5.86e-11 ***
## neighborhoodStuyvesant Town/PCV -7.885e+01  8.070e+02  -0.098 0.922176
## neighborhoodTribeca              9.780e+02  2.223e+02   4.400 1.13e-05 ***
## neighborhoodUpper East Side     -1.303e+02  1.856e+02  -0.702 0.482563
## neighborhoodUpper West Side      2.145e+01  1.842e+02   0.116 0.907301
## neighborhoodWashington Heights  -1.915e+03  2.773e+02  -6.906 6.18e-12 ***
## neighborhoodWest Harlem         -1.288e+03  1.376e+03  -0.936 0.349443
## neighborhoodWest Village         1.748e+03  2.598e+02   6.729 2.08e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1364 on 2712 degrees of freedom
## Multiple R-squared:  0.815,  Adjusted R-squared:  0.8121
## F-statistic: 284.4 on 42 and 2712 DF,  p-value: < 2.2e-16
```

We will ignore variables with p-values less than 0.05, since they are not statistically significant and cannot explain the variation of the rent price.

The variables that are statistically significant include size_sqft, bathrooms, floor, building_age_yrs, and some neighborhoods such as Central Park South, Central Harlem, Financial District, etc.

Out of these, certain neighborhoods have the largest absolute value coefficients compared to the reference, which is Battery Park City. The top 3 neighborhoods with the largest absolute value coefficients include Washington Heights at about -1992 dollars, West Village at 1855 dollars, and Central Harlem at -1707 dollars. In comparison to Battery Park City rent prices, Washington Heights is about 1992 dollars less, Central Harlem is 1707 dollars less, and the West Village is 1855 dollars more. As we can see, the rent prices vary greatly by neighborhood, thus reflecting that neighborhood is a significant factor in apartment rent prices.

Out of the other statistically significant variables, bathrooms seem to effect rent price the most as an increase of one bathroom yields an average increase of about 992 dollars. Interestingly, having a washer/dryer adds an average of 210 dollars, while the size in square feet, adds around 4 dollars per square foot.

In terms of floor, an increase in one floor adds about 30 dollars to the rent price, which is expected as generally in Manhattan a higher floor means a better view. The building age seems to not matter much with a coefficient of -6.

INSIGHTS

With our linear model, it is evident that our selected variables affect the rent price greatly, the most important variables being neighborhood and bathrooms. This makes sense, as Manhattan is such a diverse and populous

city with various historical factors influencing the creation of specific neighborhoods. Specifically in New York, it makes sense that rent prices vary greatly among different neighborhoods – location of Uptown, Midtown and Downtown don't matter too much in comparison.

While the R^2 value of our linear model was relatively high around 0.82, the predictive power of our linear model was certainly limited. While neighborhoods certainly affect the rent price, it causes a lot of variation among prices which the linear model might not have been able to notice.

Why did our linear model not work as well as we had thought? Well, in Manhattan and specifically for our dataset, there are probably many other factors that are not included in this dataset that affect the rent price. For instance, having a view of the skyline, whether your apartment faces a garage, or it overlooks the Hudson River – there are so many other factors in NYC that could affect the rent price which are not in our dataset. This thought is even more emphasized by the fact that Manhattan is so diverse and such as distinct city from others. The predictive power of our model most likely was not able to pick up on these other variables.

With our predictions, we were able to improve them by sub-setting and resizing our dataset. Because rent prices vary so greatly across Manhattan, it makes sense that our model could predict the apartment prices of a few neighborhoods with 5000 dollars or less– this price range could be seen as more of a "typical" experience of someone looking to rent an apartment. It was more difficult for our model to predict higher prices; which, again, could be explained by the existence of other factors beyond the data set.

IMPROVEMENTS

In terms of improvements, before modeling, we would restrict the data set to a few significant neighborhoods with relatively equal amounts of data in order to make our linear regression more precise. Predicting prices through the entire geographical area of Manhattan probably introduced more error. Furthermore, we could also improve our "close/far" metric; we used the mean of the times to distinguish between close and far, but perhaps 5 minutes was too low to be considered "far". The "close/far" metric may have been more significant in both our analysis and modelling if we had chosen a higher number such as 7 or 8 minutes.

Also, to address the outside factors that could have influenced rent price in Manhattan, we would probably also add another dataset with information on factors such as the skyline view of the apartment or if it faces a garage – this could make our predictive power much better.

CONCLUSION

While some relationships between the rent price and variables were relatively weak, we found that there were relatively strong correlations between rent price and bathrooms, bedrooms, size in square feet, and neighborhood. As for the minutes to the subway, we thought that it would have a much greater impact on the rent price, but our exploratory data analysis reflected that the minutes to the subway did not affect the rent price that much. It was suprising that most of the amenities did not really affect the rent price, as we thought that having a doorman would be significant in Manhattan, when in contrast, it seems that having a washer/dryer unit is more significant. Furthermore, in terms of the floor, we had thought that the higher the floor, the more expensive the listing – the correlation was not as strong as we had thought. Perhaps, even if an apartment is on a higher floor, if its view is not considered to be aesthetically pleasing or if the view is blocked somehow, the rent price could be affected by this. In general, after our exploratory data analysis and modeling, we see that neighborhood is arguably the most important factor along with the bathrooms an apartment has and the size in square feet. With our predictions, when restricting the amount of neighborhoods, the impact on the accuracy was significant. Having a washer-dryer unit is also important, but less in comparison to the former variables. Overall, floor and building age tended to have a weaker relationship to rent price and thus its predictive power was weaker as well. By looking at our exploratory data analysis, linear model and its coefficients, and the predicted prices of our model, we were able to determine the significant factors affecting rent price in Manhattan.