

Data 607 Final Project

Code Name: We-Showed-Up




We Showed Up

Team Members

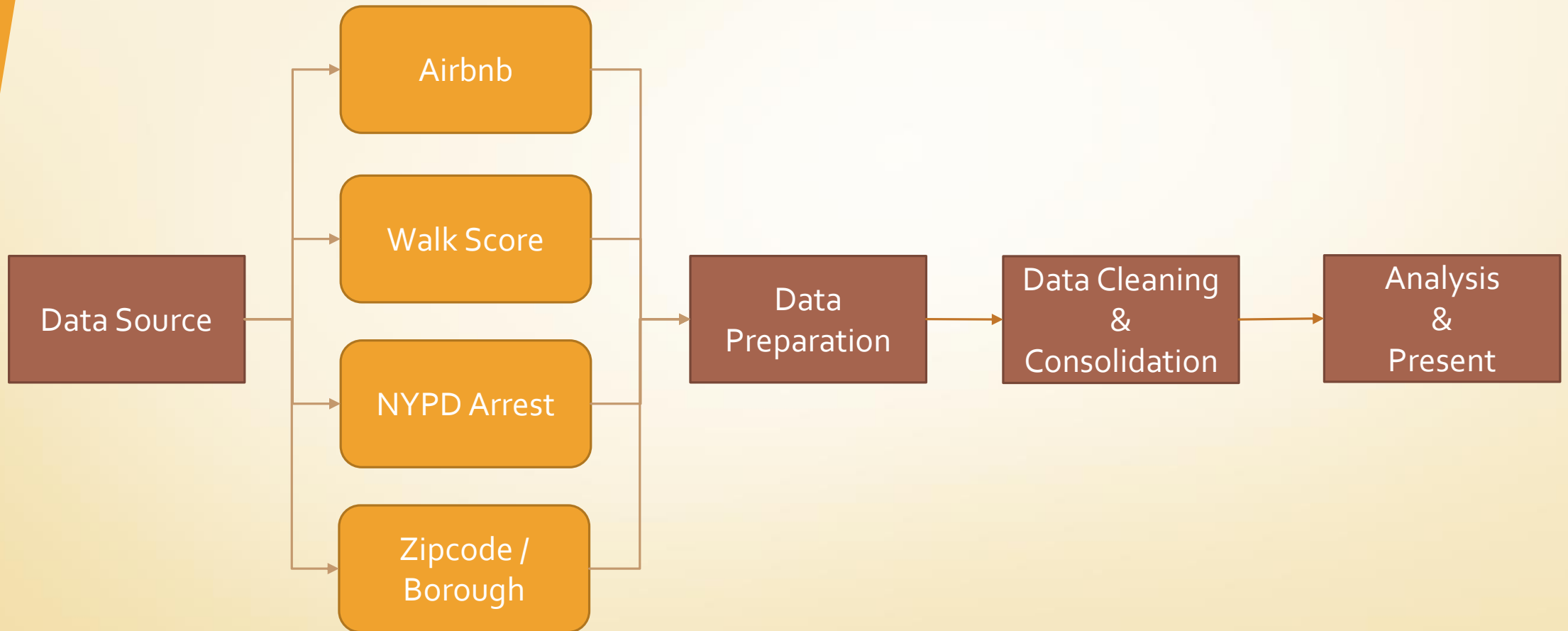
- Zhi Ying Chen (Class Group #01)
- Meng Qin Cai (Class Group #03)
- Fan Xu (Class Group #04)
- Sin Ying Wong (Class Group #04)





Can we find if there is any
relationship between Airbnb, Walk
Score, and Criminal Records in
New York City?

Work Flow



Data Preparation

- Where to collect data?
 - We want to know the relationship between Airbnb, Walk Score, and NYC Arrest Data
 - Collect data from Inside Airbnb, Walk Score, and NYPD Arrest websites
- How to collect data?
 - By excel file (.csv), web scraping, and API
- What to do next?
 - Read data into RStudio to tidy them
 - Join the tables by using different common information
 - Analyze and visualize them

Libraries

Packages

The following libraries were used in this project:

```
library(tidyverse)
library(methods)
library(knitr)
library(rvest)
library(RCurl)
library(RSocrata)
library(geosphere)
library(gridExtra)
```

Data 1 – Inside Airbnb

- ✓ Download directly



```
Airbnb_raw <- read_csv("https://raw.githubusercontent.com/oggyluky11/Data/master/listings_1.csv")
```

```
head(Airbnb_raw)
```

	id	last_scraped
	<dbl>	<chr>
	3647	9/13/2019
	3831	9/13/2019

Data 2 – Walk Scores Web Scraping

- ✓ Use distinct Borough and Zip Code data gathered from Inside Airbnb



Web Scrapping – 1. Combine borough and zip code

```
distinct_location <- Airbnb %>%  
  filter(!is.na(review_scores_rating)) %>%  
  select(neighbourhood_cleansed, zipcode) %>%  
  distinct() %>%  
  mutate(location = str_c(zipcode, ', ', neighbourhood_cleansed, ', NY')) %>%  
  mutate(url_tail = tolower(str_c(zipcode, '-', str_replace_all(neighbourhood_cleansed, '[:space:][:punct:]]', '\\-'), '-NY'))))  
  
distinct_location
```

zipcode <chr>	location <chr>	url_tail <chr>
11238	11238, Clinton Hill, NY	11238-clinton-hill-ny
10029	10029, East Harlem, NY	10029-east-harlem-ny
10016	10016, Murray Hill, NY	10016-murray-hill-ny
11216	11216, Bedford-Stuyvesant, NY	11216-bedford-stuyvesant-ny
10019	10019, Hell's Kitchen, NY	10019-hell-s-kitchen-ny
10025	10025, Upper West Side, NY	10025-upper-west-side-ny
10009	10009, East Village, NY	10009-east-village-ny
10002	10002, Chinatown, NY	10002-chinatown-ny
10036	10036, Hell's Kitchen, NY	10036-hell-s-kitchen-ny
11215	11215, South Slope, NY	11215-south-slope-ny

1-10 of 718 rows | 2-4 of 4 columns

Previous 1 2 3 4 5 6 ... 72 Next

Web Scraping – 2. Use their link format to extract scores

```
url_base <- 'https://www.walkscore.com/score/'
url_tail <- distinct_location$url_tail

for (location in url_tail[1:5]){

  url <- str_c(url_base, location)

  html_raw <- url %>%
    getURL() %>%
    read_html()

  print(url)

  walk_score_temp <- html_raw %>%
    html_node(xpath = "//img[contains(@src,'//pp.walk.sc/badge/walk/score')]") %>%
    html_attr('src') %>%
    str_extract('[0-9]+') %>%
    as.numeric()
```

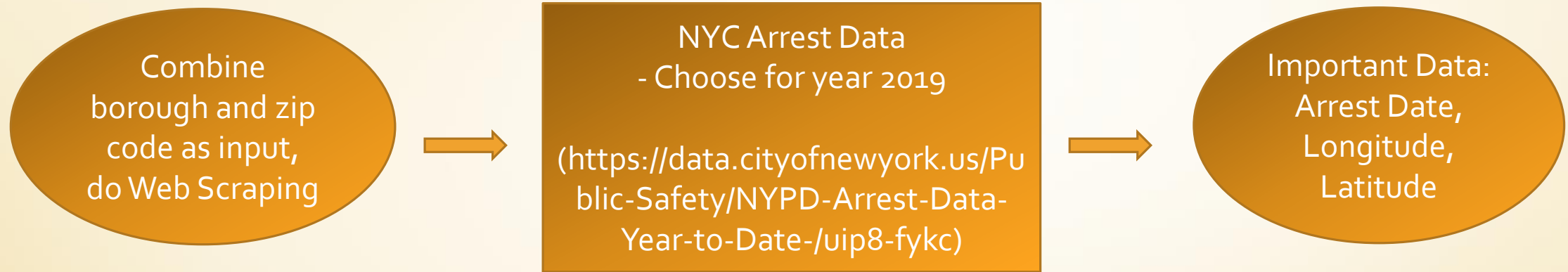
```
## [1] "https://www.walkscore.com/score/11238-clinton-hill-ny"
## [1] "https://www.walkscore.com/score/10029-east-harlem-ny"
## [1] "https://www.walkscore.com/score/10016-murray-hill-ny"
## [1] "https://www.walkscore.com/score/11216-bedford-stuyvesant-ny"
## [1] "https://www.walkscore.com/score/10019-hell-s-kitchen-ny"
```

scores

location <chr>	walk_score <dbl>	transit_score <dbl>	bike_score <dbl>
11238-clinton-hill-ny	96	100	90
10029-east-harlem-ny	99	100	88
10016-murray-hill-ny	99	100	85
11216-bedford-stuyvesant-ny	99	100	86
10019-hell-s-kitchen-ny	100	100	92

Data 3 – NYPD Arrest Open Data

- ✓ Use API to access the database



```
#Reand NYPD data into R
url <- "https://data.cityofnewyork.us/resource/uip8-fykc.json"
app_token <- "TfSVpyHY3KyAyimAxFDgFufJM"

nypd_raw <- read.socrata(url, app_token = app_token)
nypd_raw
```

perp_race <chr>	x_coord_cd <chr>	y_coord_cd <chr>	latitude <chr>	longitude <chr>
BLACK HISPANIC	990563	203120	40.72420015400007	-73.97722564299994
BLACK	1040611	190715	40.68997415500007	-73.79676854399997
WHITE	062080	160112	40.60612042000005	74.07657042000002

Data 4 – NYC Borough's Zip Codes

✓ Zip Codes <-> Borough



```
zip_borough_raw <- read_csv('https://raw.githubusercontent.com/oggyluky11/Data/master/zip_borough.csv')
```

zip <chr>	borough <chr>
10001	Manhattan
10002	Manhattan
10003	Manhattan

Data Consolidation – General

Prepare Data

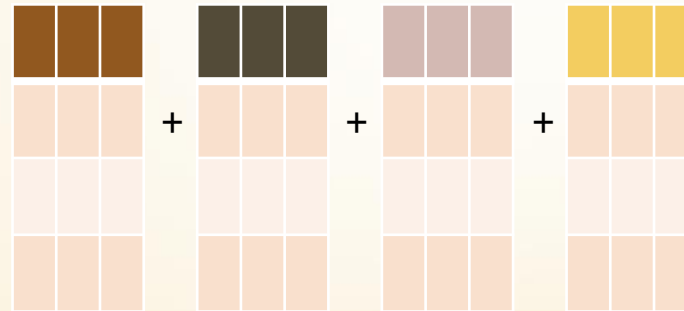
Data 1

Data 2

Data 3

Data 4

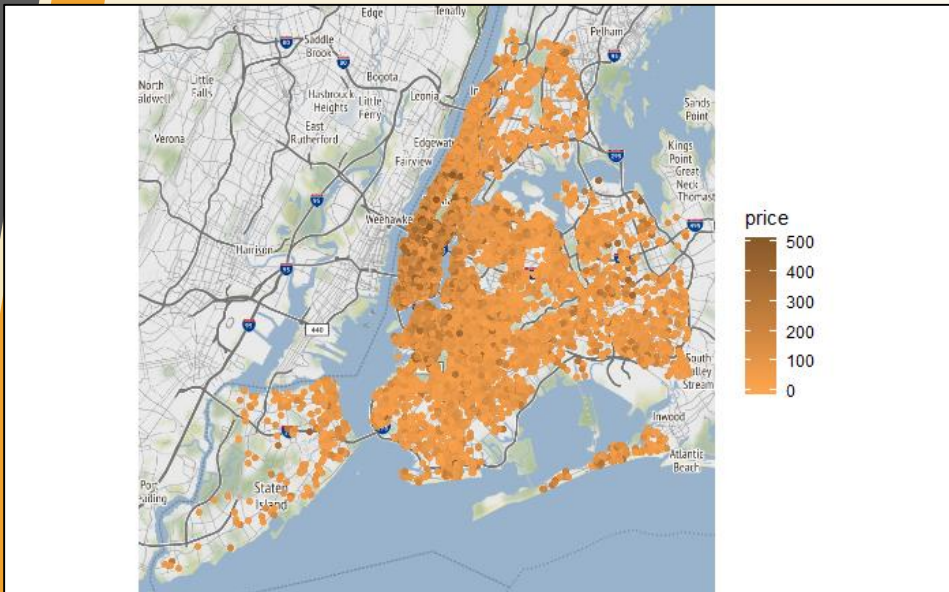
Join Data Table



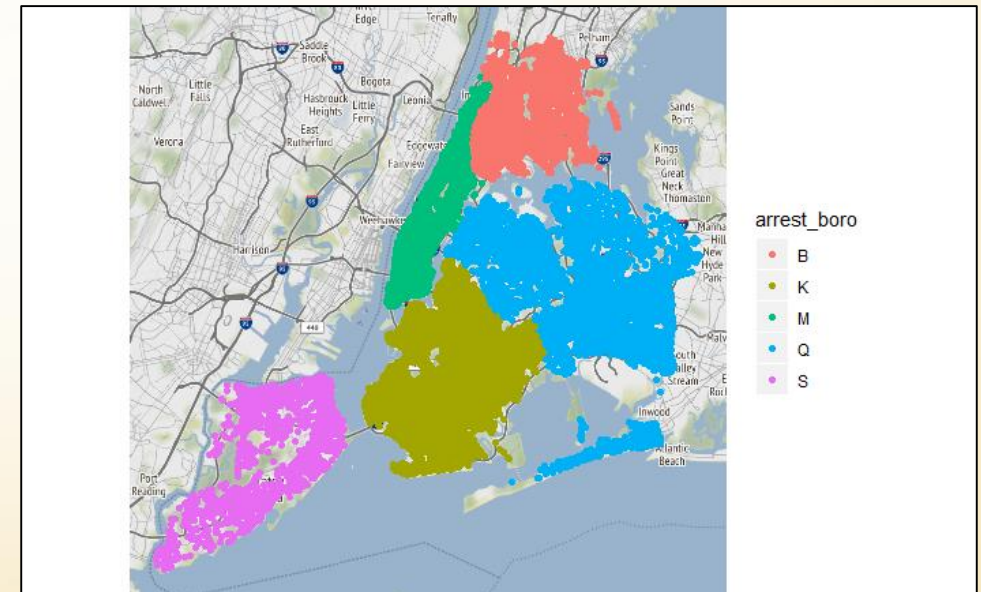
Remove NAs &
Select Columns

Data Consolidation – Challenge

Airbnb Data Table



NYPD Arrest Data Table

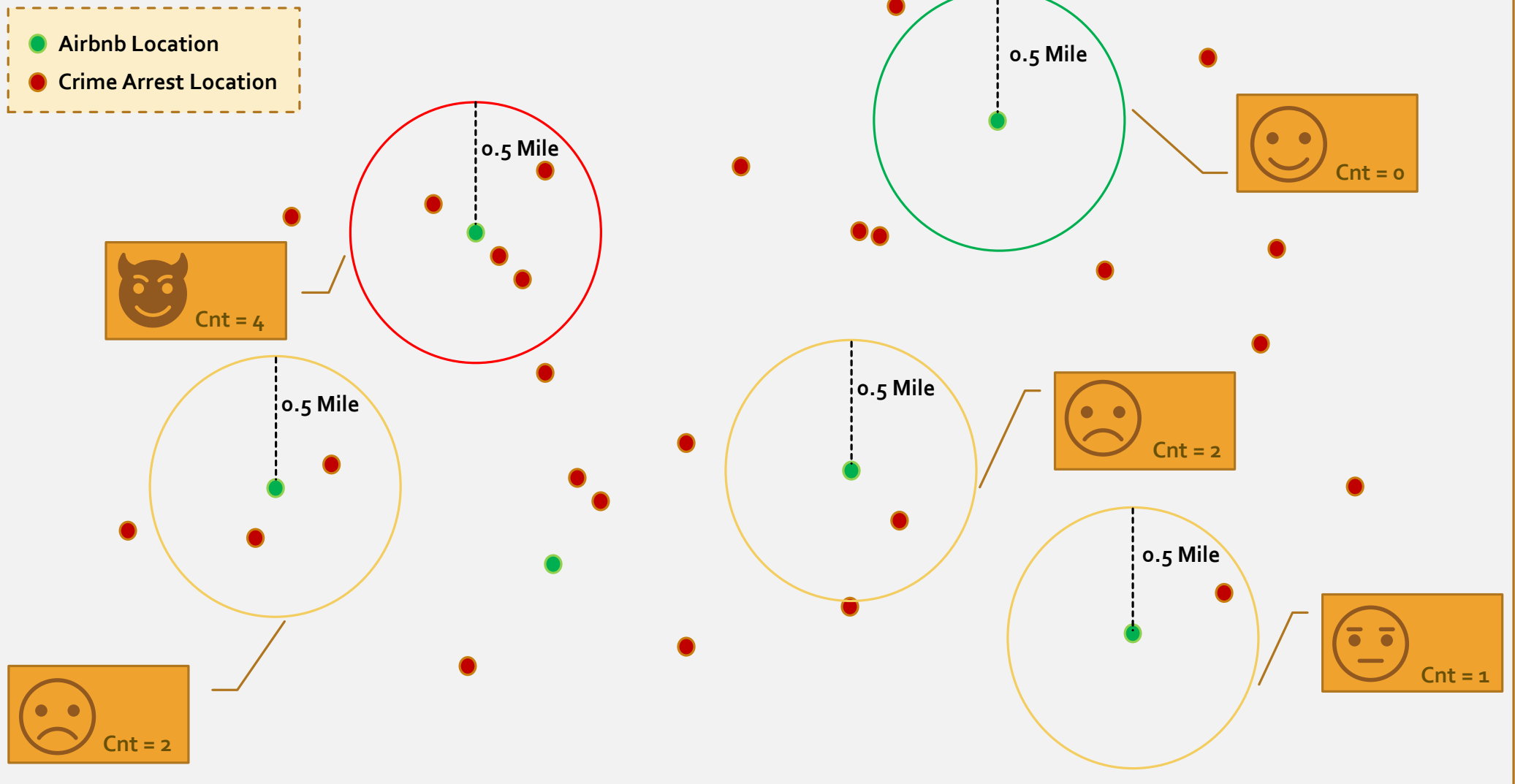


No general
key ?

Need to establish connection between these two data tables
Hint: Does crime activities often occurred in specific Airbnb locations?

Data Consolidation – Solution

Compute the number of crime arrests within 0.5 mile of each Airbnb location, the more, the worse 



Data Consolidation – Process

Compute the number of crime arrests within 0.5 mile of each Airbnb location, the more, the worse 

Use `dism()` function from [geosphere] package to compute the distance matrix of all Airbnb coordinates and crime arrest coordinates.

a. We have 40k+ Airbnb coordinates and 16K+ crime arrest coordinates. The matrix calculation is too large for R to handle.

b. Break the 40k x 16k distance matrix into 4 small matrices and compute the result separately.

c. Use `rowSums()` function to count the number of distance values that are less than 0.5.

d. Getting the count of crime arrests of each Airbnb location by sum up the results of the 4 small matrices

```
p_airbnb <- cbind(Airbnb$longitude, Airbnb$latitude)
p_crime_1 <- cbind(CrimeRecord$longitude[1:4000], CrimeRecord$latitude[1:4000])
p_crime_2 <- cbind(CrimeRecord$longitude[4001:8000], CrimeRecord$latitude[4001:8000])
p_crime_3 <- cbind(CrimeRecord$longitude[8001:12000], CrimeRecord$latitude[8001:12000])
p_crime_4 <- cbind(CrimeRecord$longitude[12001:16656], CrimeRecord$latitude[12001:16656])

#length(p2[,1])
#length(p1)
dm1 <- distm(p_airbnb, p_crime_1)/1609.344
dm2 <- distm(p_airbnb, p_crime_2)/1609.344
dm3 <- distm(p_airbnb, p_crime_3)/1609.344
dm4 <- distm(p_airbnb, p_crime_4)/1609.344

arrest_count_1 <- rowSums(dm1 <= 0.5)
arrest_count_2 <- rowSums(dm2 <= 0.5)
arrest_count_3 <- rowSums(dm3 <= 0.5)
arrest_count_4 <- rowSums(dm4 <= 0.5)

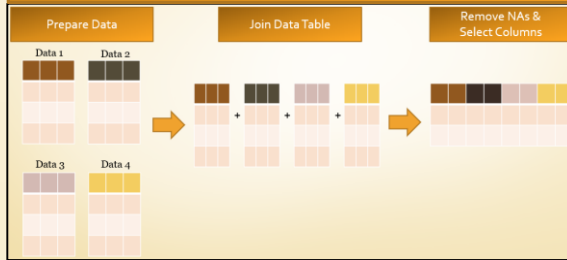
arrest_count_total <- arrest_count_1+arrest_count_2+arrest_count_3+arrest_count_4

head(arrest_count_1,20)
head(arrest_count_2,20)
head(arrest_count_3,20)
head(arrest_count_4,20)
head(arrest_count_total,20)

#arrest_count_total %>%
#data.frame() %>%
#rename(arrest_count_total = '.') %>%
#write.csv('D://DATA SCIENCE//DATA 607 FALL 2019//Homework//Final Project//arrest_count_total.csv', append =
FALSE, row.names = FALSE)
```


Data Consolidation – Combine Everything

Process Flow



a. Join all four dataset

b. Remove NAs

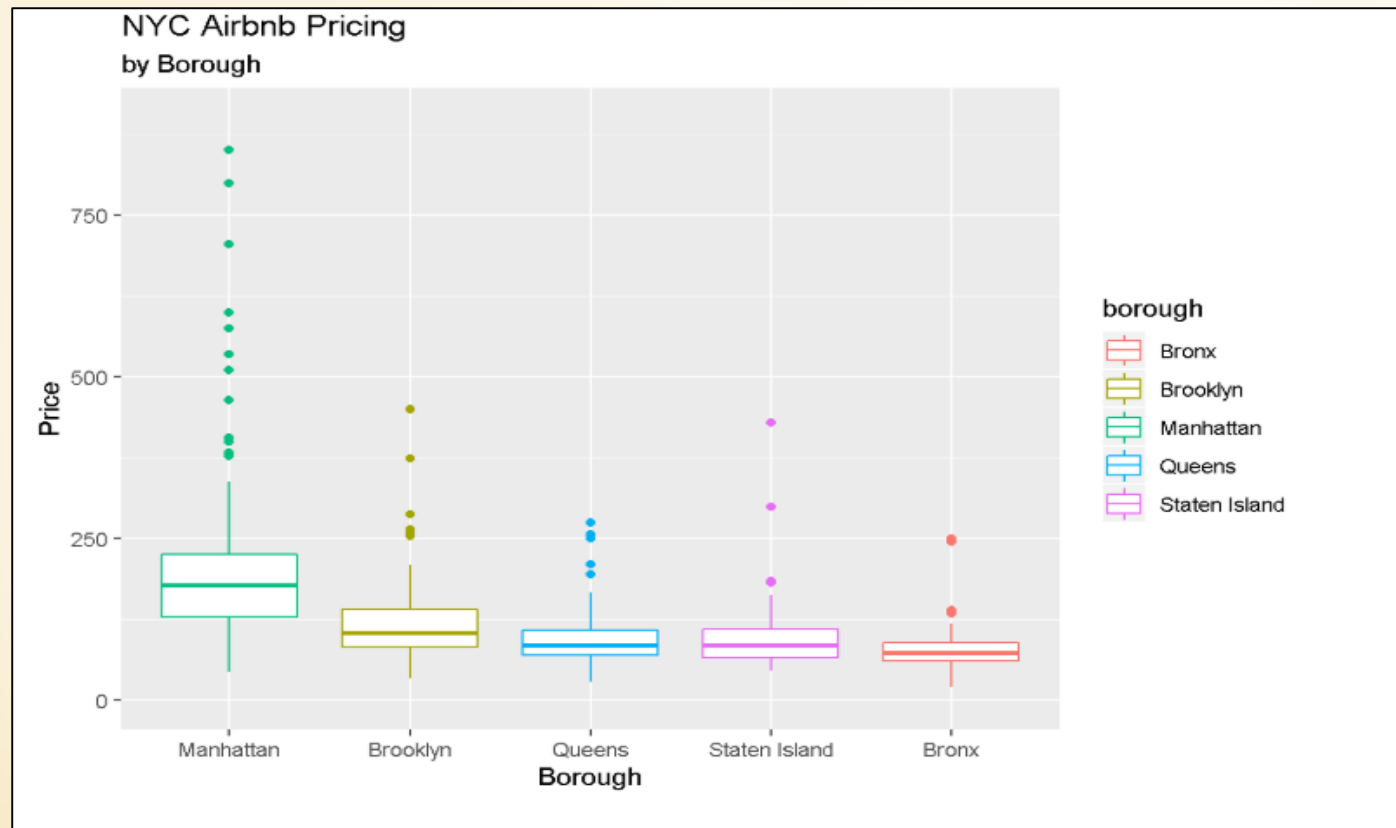
c. Select Columns that are needed

```
Airbnb_Scores <- Airbnb %>%
  mutate(arrest_count = arrest_count_total$arrest_count_total) %>%
  left_join(distinct_location, by = c('neighbourhood_cleansed', 'zipcode')) %>%
  left_join(scores, by = c('url_tail')) %>%
  left_join(zip_borough, by = c('zipcode' = 'zip')) %>%
  mutate_all(~na_if(str_trim(.), '')) %>%
  mutate(price = parse_number(price),
         review_scores_rating = parse_number(review_scores_rating),
         walk_score = parse_number(walk_score),
         transit_score = parse_number(transit_score),
         bike_score = parse_number(bike_score),
         latitude = parse_number(latitude),
         longitude = parse_number(longitude),
         arrest_count = parse_number(arrest_count)) %>%
  drop_na() %>%
  select(id,
         zipcode,
         neighbourhood_cleansed,
         borough,
         price,
         review_scores_rating,
         walk_score,
         transit_score,
         bike_score,
         arrest_count)
```

Airbnb_Scores

borough <chr>	price <dbl>	review_scores_rating <dbl>	walk_score <dbl>	transit_score <dbl>	bike_score <dbl>	arrest_count <dbl>
Brooklyn	115	93	99	100	93	88
Brooklyn	110	92	99	100	93	79
Brooklyn	120	95	99	100	86	139
Brooklyn	60	96	85	95	75	9
Manhattan	150	90	100	100	92	148

Analysis 1 – Pricing by Borough



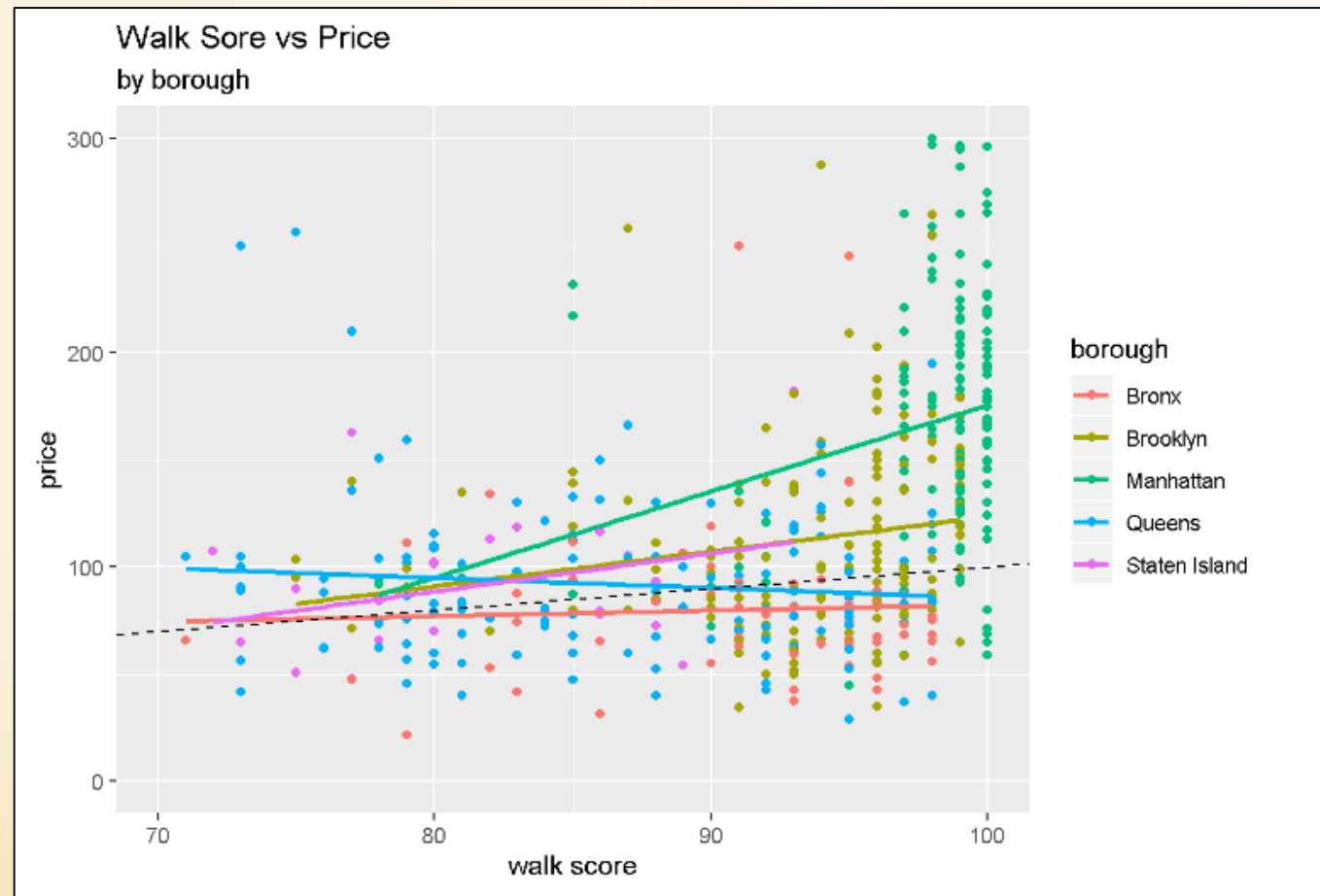
```
zipReviews %>%  
  mutate(borough = as.factor(borough)) %>%  
  ggplot(aes(x = fct_reorder(borough, desc(price)), y = price, color = borough)) +  
  geom_boxplot() +  
  ylim(0, 900) +  
  labs(title = 'NYC Airbnb Pricing',  
        subtitle = 'by Borough')+  
  xlab('Borough')+  
  ylab('Price')
```

Analysis 2 – Review Scores vs Price

```
p1<- ggplot(zipReviews, aes( x= review_scores_rating, y = price, color = borough)) +  
  geom_point() +  
  xlim(70, 100) +  
  ylim(0, 900) +  
  labs(title = 'Review Scores vs Price',  
        subtitle = 'by Borough')+  
  xlab('Review Score')+  
  ylab('Price')+  
  geom_smooth(method = 'lm',se = FALSE)+  
  geom_abline(color="black",linetype = 2)+  
  theme(legend.position = "none")  
  
p2 <- zipReviews %>%  
  ggplot(aes(review_scores_rating, fill = borough))+  
  geom_histogram(binwidth = 1) +  
  xlim(70, 100) +  
  facet_grid(rows = vars(fct_reorder(borough, review_scores_rating)))  
  
grid.arrange(p1, p2, nrow = 1)
```



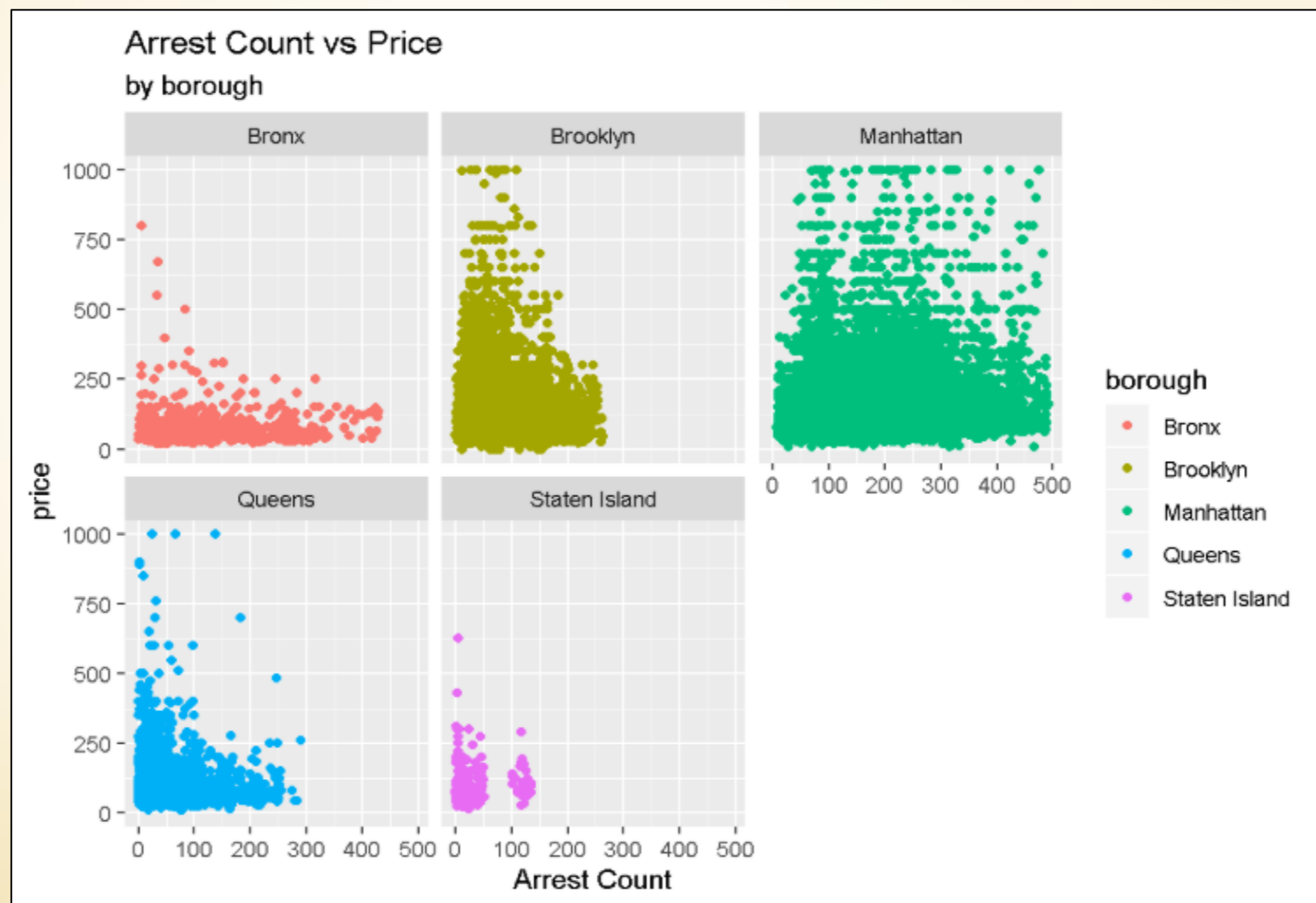
Analysis 3 – Walk Score vs Price



```
ggplot(zipReviews, aes(x = walk_score, y = price, color = borough))+  
  geom_point()+xlim(70,100)+ylim(0,300)+  
  labs(title="Walk Score vs Price",subtitle="by borough")+  
  xlab("walk score")+ylab("price")+  
  geom_smooth(method = 'lm',se = FALSE)+  
  geom_abline(color="black",linetype = 2)
```

Analysis 4 – Criminal arrest count by Borough

```
ggplot(Airbnb_Scores, aes(x = arrest_count, y = price, color = borough))+  
  geom_point()+  
  #xlim(70, 100)+  
  ylim(0, 1000)+  
  labs(title="Arrest Count vs Price", subtitle="by borough")+  
  xlab("Arrest Count")+ylab("price")+  
  facet_wrap(~borough)
```



Conclusion

We focused on investigating Airbnb's pricing with other information on hand. We used review scores, walk scores, and arrest counts from the Airbnb dataset, walk scores, and criminal records.

Manhattan has the most expensive pricing on Airbnb among all five NYC boroughs. The higher the review scores of the Airbnb listings, the higher the price of them. On the other hand, the more than arrest counts within 0.5 mile from an Airbnb listing, the lower the set price of it.

However, there is no obvious relationship between walk score and Airbnb's pricing. It may be because the transit is convenient in NYC. Besides walking, we also have subway, buses, free ferry and bikes. This study may have a different result if the location is set in a less condense and less commute-friendly city.



Thank You!