NYC Real Estate Sales Analysis

Overview: In this analysis, we'll explore the NYC real estate dataset to uncover trends in sale prices across different neighborhoods and property types.We'll focus on understanding the relationship between property features (e.g., neighborhood, year built) and sale price.

Install apropriate libraries & packages

```
install.packages("tidyverse")
```

```
##
## The downloaded binary packages are in
##
/var/folders/k7/3hkxc3916d94sh54b7xgkrfh0000gn/T//RtmpFEhQ1y/downloaded_packa
ges
```

```
install.packages("knitr")
```

```
##
## The downloaded binary packages are in
##
/var/folders/k7/3hkxc3916d94sh54b7xgkrfh0000gn/T//RtmpFEhQ1y/downloaded_packa
ges
```

```
install.packages("scales")
```

```
##
## The downloaded binary packages are in
##
/var/folders/k7/3hkxc3916d94sh54b7xgkrfh0000gn/T//RtmpFEhQ1y/downloaded_packa
ges
```

```
install.packages("ggplot2")
```

```
##
## The downloaded binary packages are in
##
/var/folders/k7/3hkxc3916d94sh54b7xgkrfh0000gn/T//RtmpFEhQ1y/downloaded_packa
ges
```

```
library(ggplot2)
library(tidyverse)
library(knitr)
library(readr)
library(scales)
```

```
##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
##     discard

## The following object is masked from 'package:readr':
##
##     col_factor

library(dplyr)
library(lubridate)
```

Import document

```
nyc_dataset <- read_csv("~/Desktop/Data sets/nyc-rolling-sales.csv")

## New names:
## Rows: 84548 Columns: 22
## ── Column specification
## ────────────────────────────────────────────────────────────────────
Delimiter: "," chr
## (10): NEIGHBORHOOD, BUILDING CLASS CATEGORY, TAX CLASS AT PRESENT, BUIL...
dbl
## (10): ...1, BOROUGH, BLOCK, LOT, ZIP CODE, RESIDENTIAL UNITS, COMMERCIA...
lgl
## (1): EASE-MENT dttm (1): SALE DATE
## ℹ Use `spec()` to retrieve the full column specification for this data. ℹ
## Specify the column types or set `show_col_types = FALSE` to quiet this
message.
## • `` -> `...1`
```

Clean and prepare data

```
nyc_dataset <- nyc_dataset %>%
  select(-`EASE-MENT`, -`APARTMENT NUMBER`)

colnames(nyc_dataset) <- c("id", "borough", "neighborhood",
"building_class_category",
                           "tax_class_present", "block", "lot",
"building_class_present",
                           "address", "zip_code", "residential_units",
"commercial_units",
                           "total_units", "land_square_feet",
"gross_square_feet",
                           "year_built", "tax_class_sale_time",
"building_class_sale_time",
                           "sale_price", "sale_date")
```

```
# Convert sale_price to numeric and clean data
nyc_dataset <- nyc_dataset %>%
  mutate(sale_price = as.numeric(gsub("[^0-9.]", "", sale_price))) %>%
  filter(!is.na(sale_price))
```

Quick overview

```
colnames(nyc_dataset) #List of column names
```

```
##  [1] "id"                       "borough"
##  [3] "neighborhood"             "building_class_category"
##  [5] "tax_class_present"        "block"
##  [7] "lot"                      "building_class_present"
##  [9] "address"                  "zip_code"
## [11] "residential_units"        "commercial_units"
## [13] "total_units"              "land_square_feet"
## [15] "gross_square_feet"        "year_built"
## [17] "tax_class_sale_time"      "building_class_sale_time"
## [19] "sale_price"               "sale_date"
```

```
ncol(nyc_dataset) #How many columns are in data frame?
```

```
## [1] 20
```

```
nrow(nyc_dataset) #How many rows are in data frame?
```

```
## [1] 69987
```

```
dim(nyc_dataset)  #Dimensions of the data frame?
```

```
## [1] 69987    20
```

```
head(nyc_dataset)  #See the first 6 rows of data frame.
```

```
## # A tibble: 6 × 20
##       id borough neighborhood  building_class_category    tax_class_present
block
##    <dbl>   <dbl> <chr>         <chr>                      <chr>
<dbl>
## 1     4       1 ALPHABET CITY 07 RENTALS - WALKUP APART… 2A
392
## 2     7       1 ALPHABET CITY 07 RENTALS - WALKUP APART… 2B
402
## 3     8       1 ALPHABET CITY 07 RENTALS - WALKUP APART… 2A
404
## 4    10       1 ALPHABET CITY 07 RENTALS - WALKUP APART… 2B
406
## 5    13       1 ALPHABET CITY 08 RENTALS - ELEVATOR APA… 2
387
## 6    15       1 ALPHABET CITY 08 RENTALS - ELEVATOR APA… 2B
400
```

```
## # ℹ 14 more variables: lot <dbl>, building_class_present <chr>, address
<chr>,
## #   zip_code <dbl>, residential_units <dbl>, commercial_units <dbl>,
## #   total_units <dbl>, land_square_feet <chr>, gross_square_feet <chr>,
## #   year_built <dbl>, tax_class_sale_time <dbl>,
## #   building_class_sale_time <chr>, sale_price <dbl>, sale_date <dttm>
```

```
str(nyc_dataset)  #See list of columns and data types (numeric, character,
etc)
```

```
## tibble [69,987 × 20] (S3: tbl_df/tbl/data.frame)
##  $ id                      : num [1:69987] 4 7 8 10 13 15 16 17 18 19 ...
##  $ borough                 : num [1:69987] 1 1 1 1 1 1 1 1 1 1 ...
##  $ neighborhood            : chr [1:69987] "ALPHABET CITY" "ALPHABET CITY"
"ALPHABET CITY" "ALPHABET CITY" ...
##  $ building_class_category : chr [1:69987] "07 RENTALS - WALKUP
APARTMENTS" "07 RENTALS - WALKUP APARTMENTS" "07 RENTALS - WALKUP APARTMENTS"
"07 RENTALS - WALKUP APARTMENTS" ...
##  $ tax_class_present       : chr [1:69987] "2A" "2B" "2A" "2B" ...
##  $ block                   : num [1:69987] 392 402 404 406 387 400 373 373
373 373 ...
##  $ lot                     : num [1:69987] 6 21 55 32 153 21 40 40 40 40
...
##  $ building_class_present  : chr [1:69987] "C2" "C4" "C2" "C4" ...
##  $ address                 : chr [1:69987] "153 AVENUE B" "154 EAST 7TH
STREET" "301 EAST 10TH   STREET" "210 AVENUE B" ...
##  $ zip_code                : num [1:69987] 10009 10009 10009 10009 10009
...
##  $ residential_units       : num [1:69987] 5 10 6 8 24 10 0 0 0 0 ...
##  $ commercial_units        : num [1:69987] 0 0 0 0 0 0 0 0 0 0 ...
##  $ total_units             : num [1:69987] 5 10 6 8 24 10 0 0 0 0 ...
##  $ land_square_feet        : chr [1:69987] "1633" "2272" "2369" "1750" ...
##  $ gross_square_feet       : chr [1:69987] "6440" "6794" "4615" "4226" ...
##  $ year_built              : num [1:69987] 1900 1913 1900 1920 1920 ...
##  $ tax_class_sale_time     : num [1:69987] 2 2 2 2 2 2 2 2 2 2 ...
##  $ building_class_sale_time: chr [1:69987] "C2" "C4" "C2" "C4" ...
##  $ sale_price              : num [1:69987] 6625000 3936272 8000000 3192840
16232000 ...
##  $ sale_date               : POSIXct[1:69987], format: "2017-07-19"
"2016-09-23" ...
```

```
summary(nyc_dataset)  #Statistical summary of data. Mainly for numerics
```

```
##        id           borough      neighborhood      
building_class_category
##  Min.   :    4   Min.   :1.000   Length:69987       Length:69987      
##  1st Qu.: 4182   1st Qu.:2.000   Class :character   Class :character  
##  Median : 8989   Median :3.000   Mode  :character   Mode  :character  
##  Mean   :10288   Mean   :2.922                                        
##  3rd Qu.:15874   3rd Qu.:4.000                                        
##  Max.   :26738   Max.   :5.000                                        
##  tax_class_present      block            lot
building_class_present
##  Length:69987       Min.   :    1   Min.   :   1.0   Length:69987
##  Class :character   1st Qu.: 1348   1st Qu.:  22.0   Class :character
##  Mode  :character   Median : 3378   Median :  50.0   Mode  :character
##                     Mean   : 4196   Mean   : 373.8
##                     3rd Qu.: 6186   3rd Qu.: 709.0
##                     Max.   :16319   Max.   :9106.0
##    address             zip_code     residential_units commercial_units
##  Length:69987       Min.   :    0   Min.   :   0.0    Min.   :   0.0000
##  Class :character   1st Qu.:10306   1st Qu.:   0.0    1st Qu.:   0.0000
##  Mode  :character   Median :11209   Median :   1.0    Median :   0.0000
##                     Mean   :10741   Mean   :   1.9    Mean   :   0.1725
##                     3rd Qu.:11249   3rd Qu.:   2.0    3rd Qu.:   0.0000
##                     Max.   :11694   Max.   :1844.0    Max.   :2261.0000
##   total_units       land_square_feet   gross_square_feet    year_built
##  Min.   :   0.000   Length:69987       Length:69987       Min.   :   0
##  1st Qu.:   0.000   Class :character   Class :character   1st Qu.:1920
##  Median :   1.000   Mode  :character   Mode  :character   Median :1937
##  Mean   :   2.092                                         Mean   :1799
##  3rd Qu.:   2.000                                         3rd Qu.:1965
##  Max.   :2261.000                                         Max.   :2017
##  tax_class_sale_time building_class_sale_time   sale_price
##  Min.   :1.000       Length:69987             Min.   :0.000e+00
##  1st Qu.:1.000       Class :character         1st Qu.:2.250e+05
##  Median :2.000       Mode  :character         Median :5.300e+05
##  Mean   :1.642                                Mean   :1.276e+06
##  3rd Qu.:2.000                                3rd Qu.:9.500e+05
##  Max.   :4.000                                Max.   :2.210e+09
##    sale_date
##  Min.   :2016-09-01 00:00:00.00
##  1st Qu.:2016-11-30 00:00:00.00
##  Median :2017-02-28 00:00:00.00
##  Mean   :2017-02-27 21:22:54.48
##  3rd Qu.:2017-05-31 00:00:00.00
##  Max.   :2017-08-31 00:00:00.00
```

```
names(nyc_dataset)
```

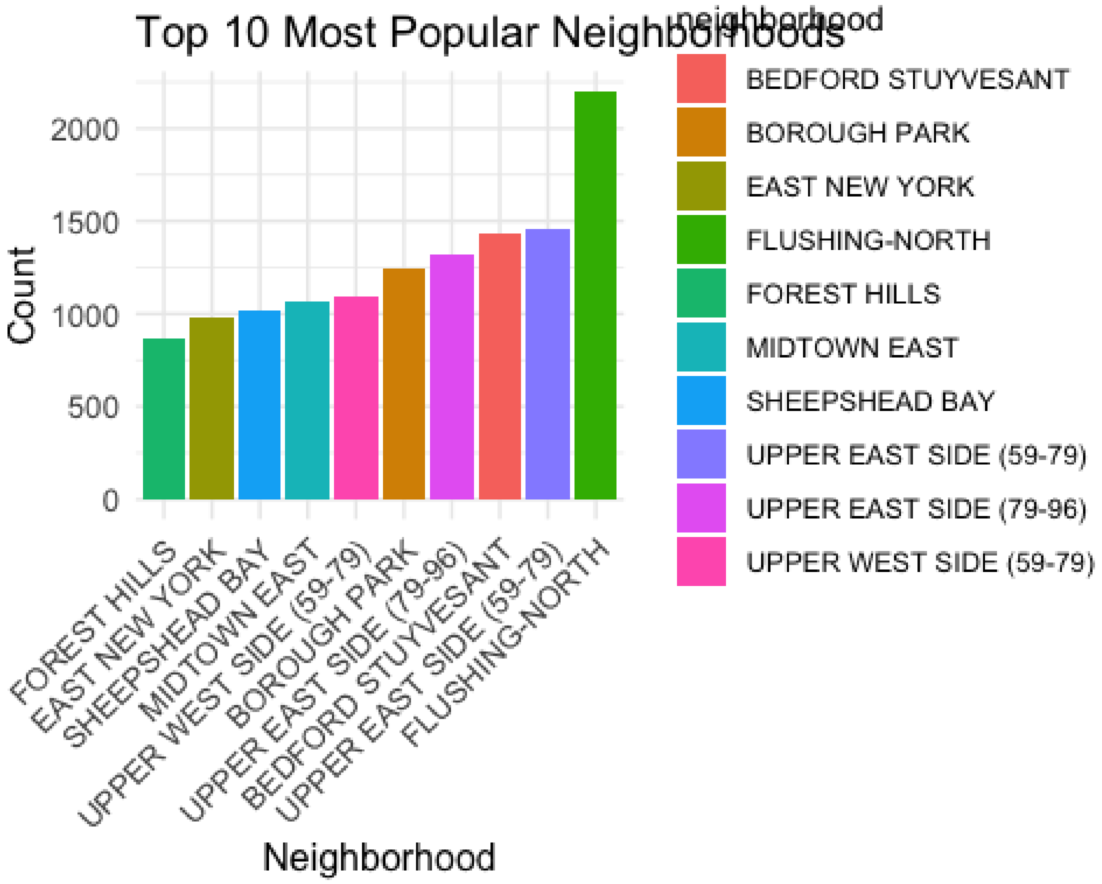
```
##  [1] "id"                      "borough"
##  [3] "neighborhood"            "building_class_category"
##  [5] "tax_class_present"       "block"
##  [7] "lot"                     "building_class_present"
##  [9] "address"                 "zip_code"
## [11] "residential_units"       "commercial_units"
## [13] "total_units"             "land_square_feet"
## [15] "gross_square_feet"       "year_built"
## [17] "tax_class_sale_time"     "building_class_sale_time"
## [19] "sale_price"              "sale_date"
```

Quick glance summary

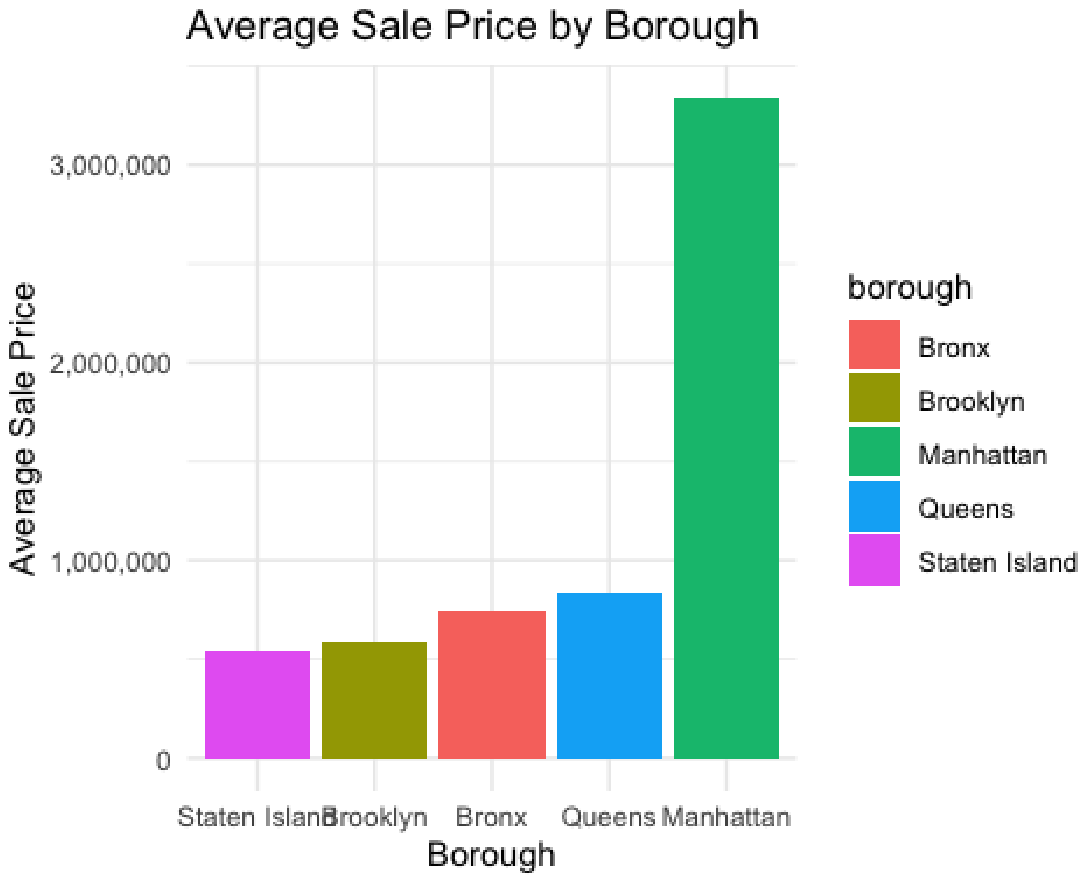
```
# Increase scipen to avoid scientific notation
options(scipen = 999)

# Summary statistics for sale price
summary(nyc_dataset$sale_price)

##       Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
##          0     225000     530000    1276456     950000 2210000000
```

Visualizations

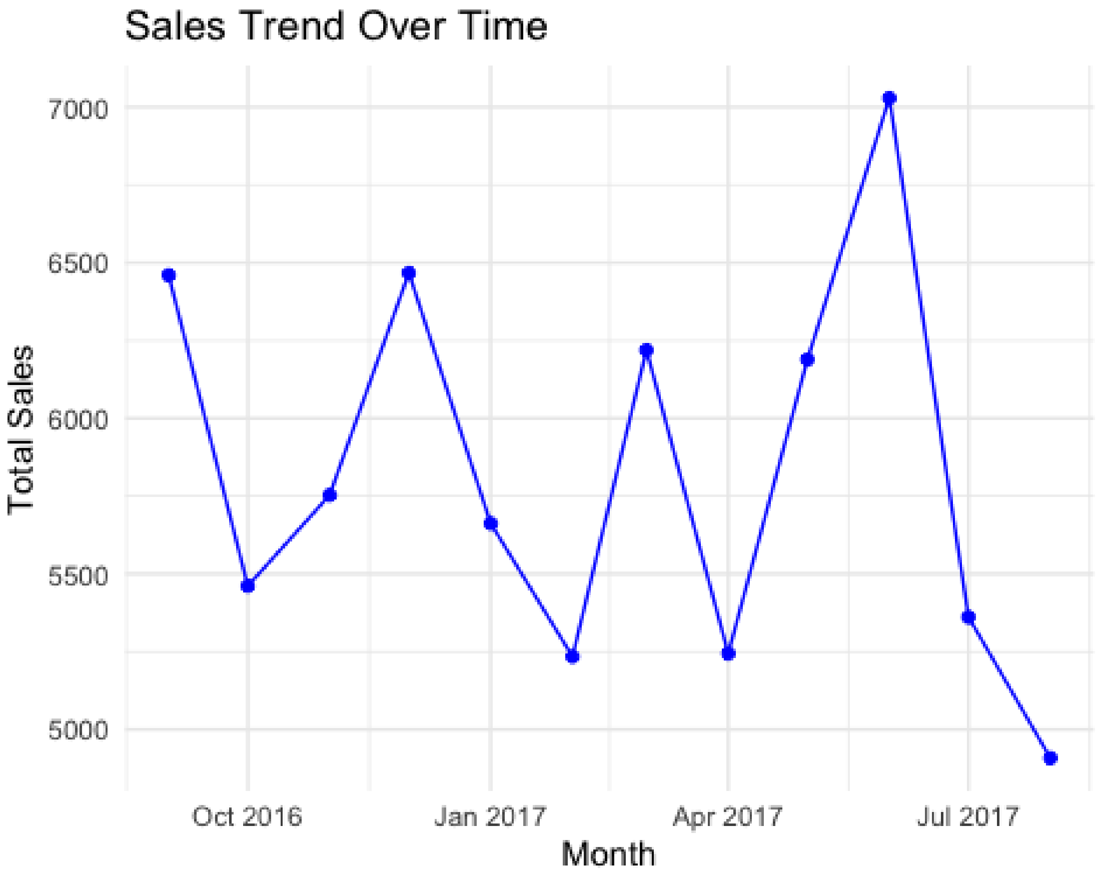
```
# Top 10 most popular neighborhoods
neighborhood_counts <- nyc_dataset %>%
  count(neighborhood)

top_neighborhoods <- neighborhood_counts %>%
  arrange(desc(n)) %>%
  top_n(10)

## Selecting by n

ggplot(top_neighborhoods, aes(x = reorder(neighborhood, n), y = n, fill =
neighborhood)) +
  geom_bar(stat = "identity") +
  labs(title = "Top 10 Most Popular Neighborhoods",
       x = "Neighborhood",
       y = "Count") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1, size = 10),
    axis.title = element_text(size = 12),
    plot.title = element_text(size = 14),
    axis.text.y = element_text(size = 10),
    plot.margin = margin(5, 5, 5, 0, "pt")
  )
```



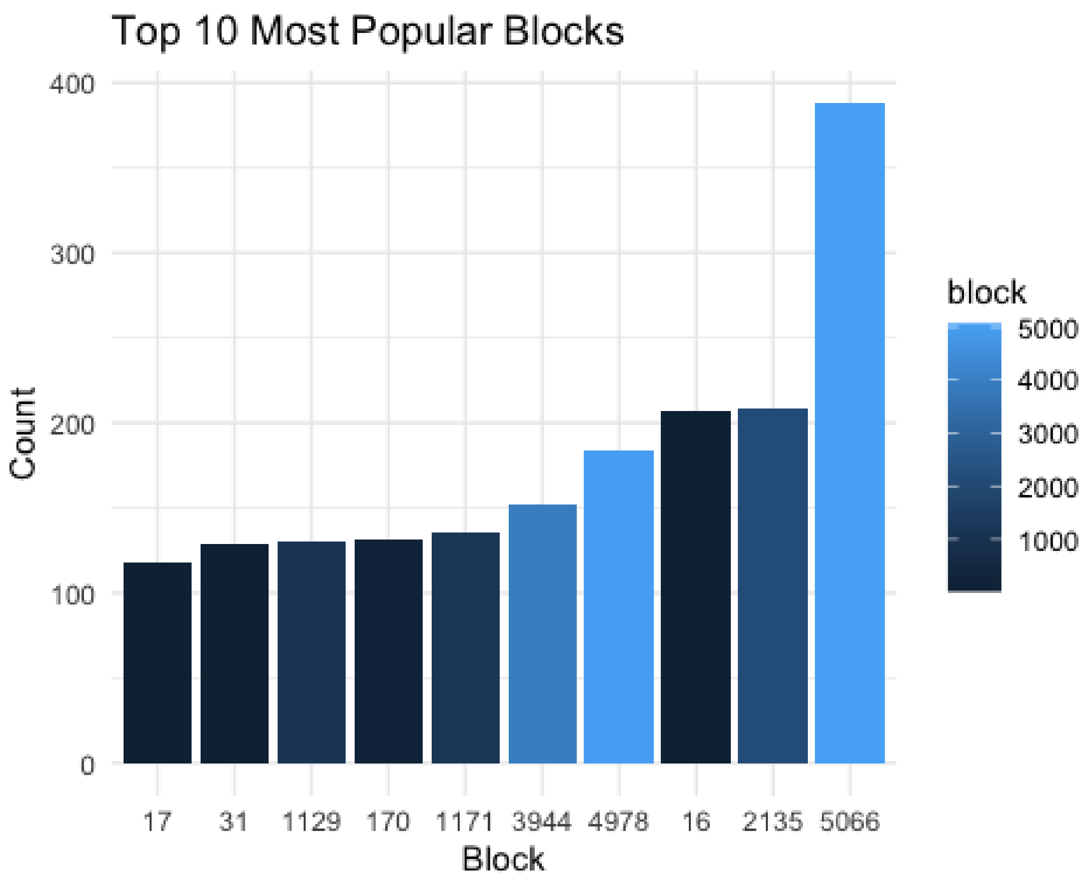
```r
# Average Sale Price by Borough
nyc_dataset <- nyc_dataset %>%
  mutate(borough = recode(borough,
                           `1` = "Manhattan",
                           `2` = "Brooklyn",
                           `3` = "Queens",
                           `4` = "Bronx",
                           `5` = "Staten Island"))

average_sale_price <- nyc_dataset %>%
  group_by(borough) %>%
  summarize(avg_sale_price = mean(sale_price, na.rm = TRUE))

ggplot(average_sale_price, aes(x = reorder(borough, avg_sale_price), y =
avg_sale_price, fill = borough)) +
  geom_bar(stat = "identity") +
  labs(title = "Average Sale Price by Borough",
       x = "Borough",
       y = "Average Sale Price") +
  scale_y_continuous(labels = scales::comma) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10),
        axis.title = element_text(size = 12),
        plot.title = element_text(size = 14),
        plot.margin = margin(5, 5, 5, 0, "pt")) +
  theme_minimal()
```



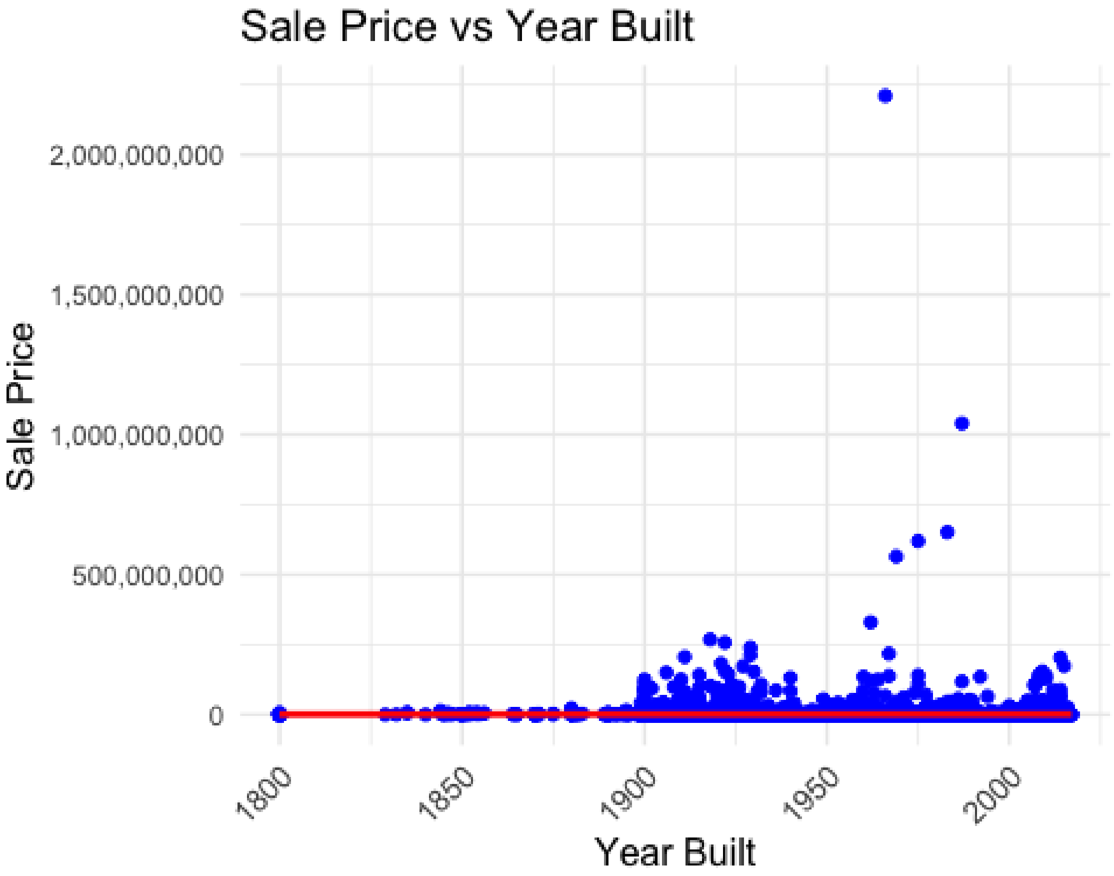
```
# Sales trend over time
monthly_sales <- nyc_dataset %>%
  mutate(month = floor_date(sale_date, "month")) %>%
  group_by(month) %>%
  summarize(total_sales = n())

ggplot(monthly_sales, aes(x = month, y = total_sales)) +
  geom_line(color = "blue") +
  geom_point(color = "blue") +
  labs(title = "Sales Trend Over Time",
       x = "Month",
       y = "Total Sales") +
  theme_minimal()
```



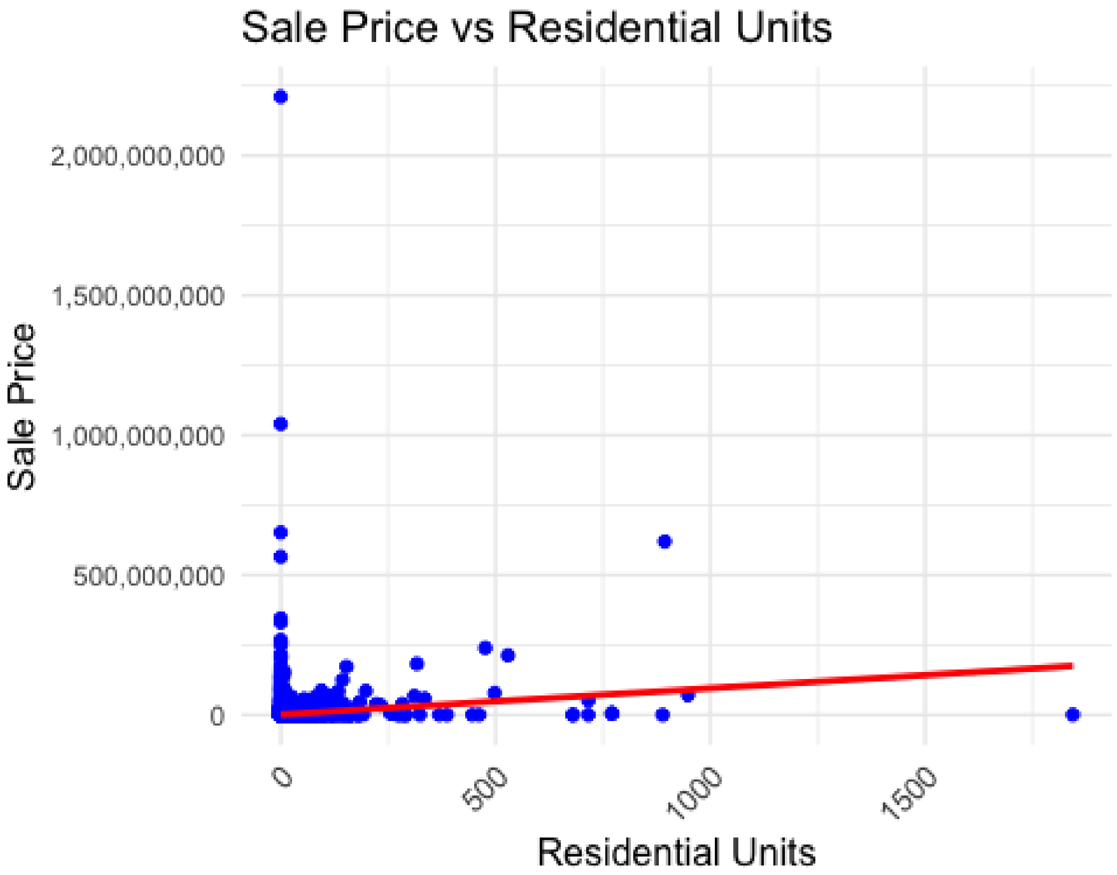
```r
# Top 10 most popular blocks
block_counts <- nyc_dataset %>%
  count(block)

top_blocks <- block_counts %>%
  arrange(desc(n)) %>%
  top_n(10)

## Selecting by n

ggplot(top_blocks, aes(x = reorder(block, n), y = n, fill = block)) +
  geom_bar(stat = "identity") +
  labs(title = "Top 10 Most Popular Blocks",
       x = "Block",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10),
        axis.title = element_text(size = 12),
        plot.title = element_text(size = 14),
        plot.margin = margin(5, 5, 5, 0, "pt")) +
  theme_minimal()
```



```
# Sale Price by Year Built
filtered_nyc_dataset <- nyc_dataset %>%
  filter(year_built != 0)

ggplot(filtered_nyc_dataset, aes(x = year_built, y = sale_price)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", color = "red") +
  labs(title = "Sale Price vs Year Built",
       x = "Year Built",
       y = "Sale Price") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10),
        axis.title = element_text(size = 12),
        plot.title = element_text(size = 14),
        plot.margin = margin(5, 5, 5, 0, "pt")) +
  scale_x_continuous(limits = c(1800, NA)) +
  scale_y_continuous(labels = scales::comma)

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

## Warning: Removed 1 row containing non-finite outside the scale range
## (`stat_smooth()`).
## Warning: Removed 1 row containing missing values or values outside the
scale range
## (`geom_point()`).
```



```r
# Sale Price vs Residential Units
ggplot(nyc_dataset, aes(x = residential_units, y = sale_price)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", color = "red") +
  labs(title = "Sale Price vs Residential Units",
       x = "Residential Units",
       y = "Sale Price") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10),
        axis.title = element_text(size = 12),
        plot.title = element_text(size = 14),
        plot.margin = margin(5, 5, 5, 0, "pt")) +
  scale_y_continuous(labels = scales::comma)

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



Conclusion: Looking at the data, Manhattan is, on average, the most expensive and most popular borough in New York City, and the 5066 block is the most popular block within NYC. Sales trends seem to drop in the months of October, February, and April, with a significant drop in August of 2017. The year built seems to have a very slight impact on sale price compared to the number of residential units, which has a much greater impact.