**Project Name: Manhattan’s Real Estate Market**

**Yiming Ge**

**Introduction:**

Real estate market is one of the popular topics when it comes to investment. Manhattan, as the city that has a high population density, contains various types of real estate with different features, making the real estates in Manhattan, New York more attractive to people who want to invest in the real estate market.

Not only I but also many groups of people who may be interested in this topic and benefit from the result of this project. For real estates related companies, such as real estate developers and real estate brokers, knowing the results from this project can either help them better build or invest in properties in a certain area or with certain characteristics or help them better serve their customers by giving the most suitable property choices and recommendations. For people who want to invest in real estate or people who would like to know more about the real estate market, even people who are looking for a place to rent, the results of the project can provide useful insight. I will target the investors who are interested in investing in the properties in Manhattan and give recommendations for different people where, when and which to buy the house based on different situations.

**Analysis: (R and Tableau)**

1. Data collecting and combinations: module 3 sets & module 5 counting

I used the dataset from <https://www1.nyc.gov/site/finance/taxes/property-annualized-sales-update.page>

This dataset is a record of every building or building unit (apartment, etc.) sold in Manhattan in New York City property market over a 4-year period. (2016-2019)

1. Data Cleaning: module 1 logic & module 3 sets & module 4 functions & module 5 counting

Removement

Remove “apartment number” column because it has ~55% missing values

Remove "EASE.MENT" column because it only has NA Values

Remove "BLOCK" and "LOT" because they are meaningless in my model (NEIGHBORHOOD is enough)

Remove "TAX.CLASS.AT.PRESENT." and "BUILDING.CLASS.AT.PRESENT."

since there two columns are the same with the two at time of sale.

Remove rows of years before 1900 in year build (outlier)

Remove rows of NAs in sales date

Replacement

Replace missing values with median

Replace 0 with means

Combination

combine BUILDING CLASS CATEGORY

combine BUILDING.CLASS.AT.TIME.OF. SALE.

combine NEIGHBORHOOD

Create New Features

Price per square feet is calculated as sales price divided by gross square feet

Seasonality is calculated as the sales date are separated into different seasons (winter, spring, summer, fall) cut in four zone between month 3, 6, 9 and 12

Create Dummy Variables for Categorical Columns

0 means absence (False); 1 means presence (True)

Dummified BUILDING CLASS CATEGORY, BUILDING CLASS AT TIME OF SALE, TAX CLASS AT TIME OF SALE, SEASON

1. Modeling: Module 4 function, Module 10 relation, Module 12 Graphs and Trees

Use CatBoost to find the feature importance

CatBoost is an open-source software library developed by Yandex. It is based on gradient boosting. Since it uses oblivious trees or symmetric trees, it has high execution speed. Most important is that it is good at handling categorical features.

For the model parameters, I used RMSE (root-mean square deviation) as its loss function and used 100 iterations.

I deleted some overlap numerical features from the cleaned data (sale price, sale date, land square feet, gross square feet), divided the data into different neighborhood. Let y be the net price and x be all the categorical features. Finally select top 5 features that influence price. (Year Built, Units, Building Category, Season and Tax Class)

Use top 5 features to make a linear regression model

Let net price be y and the five features be x. Make a linear regression and get the coefficient for each feature in different neighborhood.

Based on coefficient, I found in the upper east the older the house built, the higher the net price will be (negative relation). However, in other six neighborhoods new houses are the most sought-after (positive relation). The commercial unit is in demand for people who want to buy real estate in East Village since 191.6 much larger than 105.3. The house will be bigger and more luxurious when the building category goes up. Only big and luxury house in financial district has positive influence for price. High tax class means the residents should pay more tax for it. Financial district real estate is the only one here has a negative relation as well.

1. Visualization: Module 4 function, Module 10 relation

Overall:

I draw the average price and total units’ density on the map. The average price from midtown to downtown is much higher than the upper town. West has more units than East.

I draw the relation diagram between avg gross square feet price and avg gross square feet for different neighborhoods. Midtown houses are large but moderately priced. East Village has highest price and lowest gross square feet.

Next, I draw the relation between seasonality and average price of these seven neighborhoods. In fall, Harlem, Financial District, Midtown have highest price and East Village. Greenwich Village, Upper Ease and West have lowest price for one year.

I used built-in time series model to make a price projection for the future price. The linear line shows that the real estate price is in a steady uptrend. There is a spike in 2018 because of the low interest rate (high tolerance rate for leverage) and the crash in stock market. People wants to invest in some safe place. Real estate’s market became most people’s target. The average price doubled in that year.

Different neighborhoods:

Financial District:

Map heat diagram shows the highest price and lowest price.

Relation diagram between year built and average price per square feet.

The newer the property, the lower the price normally would be

Relation diagram between seasonality and average price per square feet.

The lowest price is in summer. Property appreciation ability is low.

Greenwich & East Village:

Map heat diagram shows the highest price and lowest price.

Relation diagram between seasonality and average price per square feet.

The lowest price is in Fall. Property appreciation ability is high in East Village and moderate in Greenwich.

Unit Square price by square feet diagram.

Unit price decreases when the properties become larger.

Midtown:

Map heat diagram shows the highest price and lowest price.

Relation diagram between year built and average price per square feet.

The newer the property, the higher the price normally would be

Relation diagram between seasonality and average price per square feet.

The lowest price is in winter. Property appreciation ability is high.

Upper East:

Map heat diagram shows the highest price and lowest price.

Relation diagram between year built and average price per square feet.

The very new and very old properties are more expensive.

Relation diagram between seasonality and average price per square feet.

The lowest price is in Fall. Property appreciation ability is low.

Upper West:

Map heat diagram shows the highest price and lowest price.

Relation diagram between year built and average price per square feet.

New properties are expensive and old properties are cheap.

Relation diagram between seasonality and average price per square feet.

The lowest price is in Fall. Property appreciation ability is low.

Unit square price by building class diagram. K is retail building.

A is one family building. For more information check Appendix.

Harlem:

Map heat diagram shows the highest price and lowest price.

Relation diagram between seasonality and average price per square feet.

The lowest price is in winter. Property appreciation ability is high or tricky.

If there is a bubble in the market, the price of Harlem will jump high very quickly but when the market cooled down, it will directly go back to the normal trend.

1. Personas and recommendations:

I made five different personas and gave them recommendations where to buy based on previous analysis.

Ben Ford who is a travel photographer and plans to make investments with sufficient budget. I recommend him go to buy east village properties.

Angela Chan who is a incoming Columbia U biology professor and wants to live near school with medium-high budget. I recommend her to find properties in upper west near central park.

Anderson Williams is a employee of a big tech company with medium-high budget, I recommend him to buy properties in Midtown.

Clark Kent is a financial consultant and hates commuting with generous budget, I recommend him to buy properties in Financial District.

Peter Parker is a high school graduate and Manhattan local with tight budget, I recommend him to find properties in Harlem.

**Conclusion**

In this project, I digged into the dataset of property sales in Manhattan for the last 4 years (exclude pandemic period), which is from 2016 to 2019. I explored the dataset by having exploratory data analysis about the overall property sales information. Other than those, I built models to predict what the future property market would look like in different neighborhood in NYC, which features have most importance, and give out some recommendations regarding the property investments. And finally, I answered questions where to buy, when to buy and which to buy based on five different personas. Personally, I would like to make investment in NYC real estate market, East Village or Midtown will be my choice. I would buy the East Village property in Fall or buy the Midtown property in winter.

The weakness of this project is that the data I used is not very large and it only includes 2016 to 2019 sold properties (pre-pandemic). In the data cleaning part, I used the mean and median to replace the missing values which may cause inevitable result deviations. In the future study, I would recommend using larger dataset and includes some big financial events like financial crisis, trade war and covid-19 which will help the model to do the prediction better. I also recommend using a better way to deal with the missing data to reduce the error. If the data are large enough, the property with missing values could be directly removed from the dataset.

*Appendix:*

Data summary:

This dataset is a record of every building or building unit (apartment, etc.) sold in the New York City property market over a 4-year period.

Data is from the NYC Department of Finance.

<https://www1.nyc.gov/site/finance/taxes/property-rolling-sales-data.page>

**Dataset Descriptions:**

**Neighborhood:**

Department of Finance assessors determine the neighborhood name in the course of valuing properties. The common name of the neighborhood is generally the same as the name Finance designates. However, there may be slight differences in neighborhood boundary lines and some sub-neighborhoods may not be included.

**Building Class Category:**

**https://www1.nyc.gov/assets/finance/jump/hlpbldgcode.html#S**

This is a field that we are including so that users of the Rolling Sales Files can easily identify similar properties by broad usage (e.g. One Family Homes) without looking up individual Building Classes. Files are sorted by Borough, Neighborhood, Building Class Category, Block, and Lot.

**Tax Class at Present:**

**https://www1.nyc.gov/site/finance/taxes/property-tax-rates.page**

Every property in the city is assigned to one of four tax classes (Classes 1, 2, 3, and 4), based on the use of the property.

• Class 1: Includes most residential property of up to three units (such as one-, two-, and three-family homes and small stores or offices with one or two attached apartments), vacant land that is zoned for residential use, and most condominiums that are not more than three stories.

• Class 2: Includes all other property that is primarily residential, such as cooperatives and condominiums.

• Class 3: Includes property with equipment owned by a gas, telephone or electric company.

• Class 4: Includes all other properties not included in class 1,2, and 3, such as offices, factories, warehouses, garage buildings, etc. Glossary of Terms for Property Sales Files

**Block:**

A Tax Block is a subdivision of the borough on which real properties are located. The Department of Finance uses a Borough-Block-Lot classification to label all real property in the City. “Whereas” addresses describe the street location of a property, the block and lot distinguish one unit of real property from another, such as the different condominiums in a single building. Also, blocks and lots are not subject to name changes based on which side of the parcel the building puts its entrance on.

**Lot:**

A Tax Lot is a subdivision of a Tax Block and represents the property’s unique location.

**Easement:**

An easement is a right, such as a right of way, which allows an entity to make limited use of another’s real property. For example, MTA railroad tracks that run across a portion of another property.

**Building Class at Present:**

The Building Classification is used to describe a property’s constructive use. The first position of the Building Class is a letter that is used to describe a general class of properties (for example “A” signifies one-family homes, “O” signifies office buildings. “R” signifies condominiums). The second position, a number, adds more specific information about the property’s use or construction style (using our previous examples “A0” is a Cape Cod style one family home, “O4” is a tower type office building and “R5” is a commercial condominium unit). The term Building Class used by the Department of Finance is interchangeable with the term Building Code used by the Department of Buildings. See NYC Building Classifications.

**https://www.propertyshark.com/mason/text/nyc\_building\_class.html**

**Address:** The street address of the property as listed on the Sales File. Coop sales include the apartment number in the address field.

**Zip Code:** The property’s postal code

**Residential Units:**

The number of residential units at the listed property.

**Commercial Units:**

The number of commercial units at the listed property.

**Total Units:**

The total number of units at the listed property.

**Land Square Feet:**

The land area of the property listed in square feet.

**Gross Square Feet:**

The total area of all the floors of a building as measured from the exterior surfaces of the outside walls of the building, including the land area and space within any building or structure on the property.

**Year Built:**

Year the structure on the property was built.

**Building Class at Time of Sale:**

The Building Classification is used to describe a property’s constructive use. The first position of the Building Class is a letter that is used to describe a general class of properties (for example “A” signifies one-family homes, “O” signifies office buildings. “R” signifies condominiums). The second position, a number, adds more specific information about the property’s use or construction style (using our previous examples “A0” is a Cape Cod style one family home, “O4” is a tower type office building and “R5” is a commercial condominium unit). The term Building Class as used by the Department of Finance is interchangeable with the term Building Code as used by the Department of Buildings.

**Sales Price:**

The price paid for the property.

**Sale Date:**

The date the property sold.

**$0 Sales Price:**

A $0 sale indicates that there was a transfer of ownership without a cash consideration. There can be a number of reasons for a $0 sale including transfers of ownership from parents to children.

#####R CODE####

# Final Project

# Yiming Ge

library(readxl)

library(plyr)

#combine property sales from 2016 - 2019

#filenames <- list.files(path = "~/Desktop/Final Project", pattern = "\*", full.names=TRUE)

#rolling\_sales <- ldply(filenames, read\_xls)

#write.csv(rolling\_sales,"~/Desktop/Final Project/rolling\_sales.csv", row.names = FALSE)

rolling\_sales <- read.csv("~/Desktop/Final Project/rolling\_sales.csv", header=TRUE, stringsAsFactors=TRUE)

summary(rolling\_sales)

str(rolling\_sales)

head(rolling\_sales)

#data quality check

#setwd("~/Downloads")

#library(dataQualityR)

#checkDataQuality(data = rolling\_sales,

#out.file.num ="sa\_num.csv",

#out.file.cat= "sa\_cat.csv")

#sa\_num<-read.csv("~/Desktop/Final Project/sa\_num.csv")

#sa\_cat<-read.csv("~/Desktop/Final Project/sa\_cat.csv")

#View(sa\_num)

#View(sa\_cat)

########################### DATA CLEANING #############################

#replace space in column names with underline

#names(rolling\_sales) <- gsub(" ", "\_", names(rolling\_sales))

#remove rows with price less than $100,000 in the sales price column because they considered outliers

sales <- rolling\_sales[!rolling\_sales$SALE.PRICE < 10000,,drop=F]

sales <- sales[!sales$SALE.PRICE > 800000000,,drop=F]

#remove “apartment number” column because it has ~55% missing values

#remove "EASE.MENT" column because it only has NA Values

#remove "BLOCK" and "LOT"

#remove "TAX.CLASS.AT.PRESENT." and "BUILDING.CLASS.AT.PRESENT." as I found out that

#there two columns are the same with the two at time of sale.

col <- c("APARTMENT.NUMBER.", "EASE.MENT.", "BLOCK.", "LOT.",

"TAX.CLASS.AT.PRESENT.", "BUILDING.CLASS.AT.PRESENT.")

sales <- sales[,!names(sales) %in% col,drop=F]

#remove rows of years before 1900 in year build

table(rolling\_sales$YEAR.BUILT.)

sales <- sales[!sales$YEAR.BUILT < 1900,,drop=F]

summary(sales$YEAR.BUILT)

#remove rows of NAs in sales date

sales <- sales[!is.na(sales$SALE.PRICE) == 'TRUE',,drop=F]

#second time quality check

library(dataQualityR)

checkDataQuality(data = sales,

out.file.num ="sa\_num.csv",

out.file.cat= "sa\_cat.csv")

sa\_num<-read.csv("~/Desktop/Final Project/sa\_num.csv")

sa\_cat<-read.csv("~/Desktop/Final Project/sa\_cat.csv")

View(sa\_num)

View(sa\_cat)

#replace missing values with median

sales$RESIDENTIAL.UNITS.[is.na(sales$RESIDENTIAL.UNITS.)] <- median(na.omit(sales$RESIDENTIAL.UNITS.))

sales$COMMERCIAL.UNITS.[is.na(sales$COMMERCIAL.UNITS.)] <- median(na.omit(sales$COMMERCIAL.UNITS.))

sales$TOTAL.UNITS.[is.na(sales$TOTAL.UNITS.)] <- median(na.omit(sales$TOTAL.UNITS.))

sales$LAND.SQUARE.FEET.[is.na(sales$LAND.SQUARE.FEET.)] <- mean(na.omit(sales$LAND.SQUARE.FEET.))

sales$GROSS.SQUARE.FEET.[is.na(sales$GROSS.SQUARE.FEET.)] <- mean(na.omit(sales$GROSS.SQUARE.FEET.))

sales$YEAR.BUILT.[is.na(sales$YEAR.BUILT.)] <- median(na.omit(sales$YEAR.BUILT.))

#replace 0 with means

sales$LAND.SQUARE.FEET.[sales$LAND.SQUARE.FEET. == '0'] <- mean(na.omit(sales$LAND.SQUARE.FEET.))

sales$GROSS.SQUARE.FEET.[sales$GROSS.SQUARE.FEET. == '0'] <- mean(na.omit(sales$GROSS.SQUARE.FEET.))

#assign seasonality to sales date

library(lubridate)

getSeason <- function(input.date){

numeric.date <- 100\*month(input.date)+day(input.date)

## input Seasons upper limits in the form MMDD in the "break =" option:

cuts <- base::cut(numeric.date, breaks = c(0,301,0601,0901,1231))

# rename the resulting groups (could've been done within cut(...levels=) if "Winter" wasn't double

levels(cuts) <- c("Winter","Spring","Summer","Fall")

return(cuts)

}

sales$SEASON <- getSeason(sales$SALE.DATE.)

#combine BUILDING CLASS CATEGORY.

sales$BUILDING.CLASS.CATEGORY. <- substr(sales$BUILDING.CLASS.CATEGORY., 0,2)

#combine BUILDING.CLASS.AT.TIME.OF.SALE.

sales$BUILDING.CLASS.AT.TIME.OF.SALE. <- substr(sales$BUILDING.CLASS.AT.TIME.OF.SALE., 0, 1)

#convert tax class at time of sale to string

sales$TAX.CLASS.AT.TIME.OF.SALE. <- as.factor(sales$TAX.CLASS.AT.TIME.OF.SALE.)

#combine neighborhood

sales$NEIGHBORHOOD. <- as.character(sales$NEIGHBORHOOD.)

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "GREENWICH VILLAGE-CENTRAL"] <- "GREENWICH VILLAGE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "GREENWICH VILLAGE-WEST"] <- "GREENWICH VILLAGE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "HARLEM-CENTRAL"] <- "HARLEM"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "HARLEM-EAST"] <- "HARLEM"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "HARLEM-UPPER"] <- "HARLEM"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "HARLEM-WEST"] <- "HARLEM"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "MIDTOWN CBD"] <- "MIDTOWN"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "MIDTOWN EAST"] <- "MIDTOWN"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "MIDTOWN WEST"] <- "MIDTOWN"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "UPPER EAST SIDE (59-79)"] <- "UPPER EAST SIDE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "UPPER EAST SIDE (79-96)"] <- "UPPER EAST SIDE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "UPPER EAST SIDE (96-110)"] <- "UPPER EAST SIDE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "UPPER WEST SIDE (59-79)"] <- "UPPER WEST SIDE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "UPPER WEST SIDE (79-96)"] <- "UPPER WEST SIDE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "UPPER WEST SIDE (96-116)"] <- "UPPER WEST SIDE"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "WASHINGTON HEIGHTS LOWER"] <- "WASHINGTON HEIGHTS"

sales$NEIGHBORHOOD.[sales$NEIGHBORHOOD. == "WASHINGTON HEIGHTS UPPER"] <- "WASHINGTON HEIGHTS"

sales$NEIGHBORHOOD. <- as.factor(sales$NEIGHBORHOOD.)

table(sales$NEIGHBORHOOD.)

#dummified categorical data

library(caret)

library(dplyr)

sale.cat <- select(sales,

BUILDING.CLASS.CATEGORY.,

BUILDING.CLASS.AT.TIME.OF.SALE.,

TAX.CLASS.AT.TIME.OF.SALE.,

SEASON

)

str(sale.cat)

dataDummy <- dummyVars("~.",data=sale.cat, fullRank=F)

data.dummified <- as.data.frame(predict(dataDummy,sale.cat))

sales$names <- rownames(sales)

data.dummified$names <- rownames(data.dummified)

clean<-join(sales,data.dummified,type='left')

clean$

#write.csv(clean,"~/Desktop/Final Project/sales\_cleaned.csv", row.names = FALSE)

########################### Model #############################

rolling\_sales <- read.csv("~/Desktop/Final Project/sales\_cleaned.csv", header=TRUE, stringsAsFactors=TRUE)

rolling\_sales$netPrice<-rolling\_sales$SALE.PRICE./ rolling\_sales$GROSS.SQUARE.FEET.

library(dplyr)

rolling\_sales<-rolling\_sales %>%

select(-SALE.PRICE.,-SALE.DATE.,-LAND.SQUARE.FEET.,-GROSS.SQUARE.FEET.)

neighborhoodlist <- split(rolling\_sales, rolling\_sales$NEIGHBORHOOD.)

alphabetCity<-neighborhoodlist$`ALPHABET CITY`

chelsea<-neighborhoodlist$CHELSEA

chinatown<-neighborhoodlist$CHINATOWN

civicCenter<-neighborhoodlist$`CIVIC CENTER`

clinton<-neighborhoodlist$CLINTON

eastVillage<-neighborhoodlist$`EAST VILLAGE`

fashion<-neighborhoodlist$FASHION

financial<-neighborhoodlist$FINANCIAL

flatiron<-neighborhoodlist$FLATIRON

gramercy<-neighborhoodlist$GRAMERCY

greenwichVillage<-neighborhoodlist$`GREENWICH VILLAGE`

harlem<-neighborhoodlist$HARLEM

inwood<-neighborhoodlist$INWOOD

javitsCenter<-neighborhoodlist$`JAVITS CENTER`

kipsBay<-neighborhoodlist$`KIPS BAY`

littleItaly<-neighborhoodlist$`LITTLE ITALY`

lowerEastSide<-neighborhoodlist$`LOWER EAST SIDE`

manhattanValley<-neighborhoodlist$`MANHATTAN VALLEY`

midtown<-neighborhoodlist$MIDTOWN

morningsideHeights<-neighborhoodlist$`MORNINGSIDE HEIGHTS`

murrayHill<-neighborhoodlist$`MURRAY HILL`

rooseveltIsland<-neighborhoodlist$`ROOSEVELT ISLAND`

soho<-neighborhoodlist$SOHO

southbridge<-neighborhoodlist$SOUTHBRIDGE

tribeca<-neighborhoodlist$TRIBECA

upperEastSide<-neighborhoodlist$`UPPER EAST SIDE`

upperWestSide<-neighborhoodlist$`UPPER WEST SIDE`

washingtonHeights<-neighborhoodlist$`WASHINGTON HEIGHTS`

#catboost feature importance

library(catboost)

for (neighborhood in neighborhoodlist){

y\_train <- unlist(neighborhood[c('netPrice')])

X\_train <- neighborhood %>% select(-netPrice)

train\_pool <- catboost.load\_pool(data = X\_train, label = y\_train)

catboost<- catboost.train(learn\_pool = train\_pool,params = list(loss\_function = 'RMSE',iterations = 100))

FI<-catboost.get\_feature\_importance(catboost, pool = train\_pool, type = 'FeatureImportance',thread\_count = -1)

FI <- as.data.frame(FI, stringsAsFactors=FALSE)

FI$name<-rownames(FI)

print(neighborhood$NEIGHBORHOOD.[1])

print(FI[order(-FI[1]),])

}

#linear regression

print('linear regression')

for (neighborhood in neighborhoodlist){

linearMod <- lm(netPrice ~ BUILDING.CLASS.CATEGORY.+RESIDENTIAL.UNITS.+COMMERCIAL.UNITS.+TOTAL.UNITS.+YEAR.BUILT.+TAX.CLASS.AT.TIME.OF.SALE.+BUILDING.CLASS.AT.TIME.OF.SALE.+SEASON, data=neighborhood)

print(neighborhood$NEIGHBORHOOD.[1])

print(summary(linearMod))

}