# HW2_cq2203

*Chang Qu (cq2203)*

*2018/9/26*

## Contents



## Part 1

i. Load the data into a dataframe called housing.

```
housing <- read.csv(file = "/Users/courtneyqu/Desktop/GR5206/NYChousing.csv", header = TRUE)
```

ii. How many rows and columns does the dataframe have?

```
nrow(housing)
```

```
## [1] 2506
```

```
ncol(housing)
```

```
## [1] 22
```

There are 2506 rows and 22 columns in this dataframe.

iii. Run the command: apply(is.na(housing), 2, sum), and explain, in words, what this does.

```
apply(is.na(housing), 2, sum)
```

```
##                             UID                    PropertyName
##                               0                               0
##                             Lon                             Lat
##                              15                              15
##                        AgencyID                            Name
##                               0                               0
##                           Value                         Address
##                              52                               0
##                  Violations2010                       REACNumber
##                               0                            1873
##                         Borough                              CD
##                               0                               0
##             CityCouncilDistrict                     CensusTract
##                              10                               0
##                    BuildingCount                       UnitCount
##                               0                               0
##                        YearBuilt                           Owner
##                               0                               0
##                     Rental.Coop                OwnerProfitStatus
##                               0                               0
##      AffordabilityRestrictions StartAffordabilityRestrictions
##                               0                               5
```

This command first use is.na function to the housing data, which will indicate which element is missing. If the element is miss, it will return the TRUE, which is equivalent as 1. Then we use apply function to sum over columns of the dataframe. If it gives 0 for the given variable, it means there is no missing value in this variable. And the number shows under each variable is the number of missing values in this given variable.

iv. Remove the rows of the dataset for which the variable Value is NA.

```
housing <- housing[-which(is.na(housing$Value)==1),]
```

v. How many rows did you remove with the previous call? Does this agree with your result from (iii)?

The row I removed is $2506 - 2454 = 52$, which agrees with the result in part iii.

vi. Create a new variable in the dataset called logValue that is equal to the logarithm of the property's Value. What are the minimum, median, mean, and maximum values of logValue?

```
logValue <- log(housing$Value)
housing <- cbind(housing, logValue)
summary(housing$logValue)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    8.41   12.49   13.75   13.68   14.80   20.47
```

Therefore, minmimum of logValue is $ 8.41$, median is 13.75, mean is 13.68, maximum is 20.47.

vii. Create a new variable in the dataset called logUnits that is equal to the logarithm of the number of units in the property. The number of units in each piece of property is stored in the variable UnitCount.

```
logUnits <- log(housing$UnitCount)
housing <- cbind(housing, logUnits)
```

viii. Create a new variable in the dataset called after 1950 which equals TRUE if the property was built in or after 1950 and FALSE otherwise.

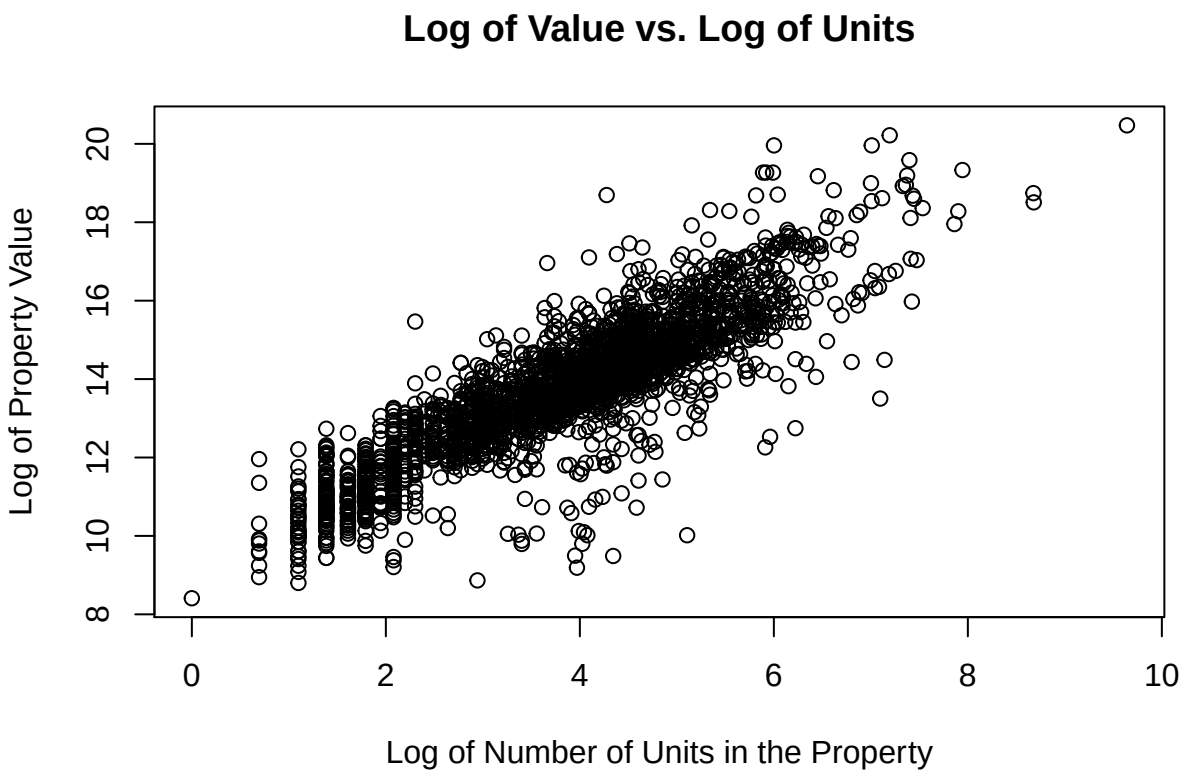
```
housing$after1950 <- ifelse(housing$YearBuilt>1950, TRUE, FALSE)
```

## Part 2

The column Borough contains the Borough of each property and is one of either Bronx, Manhattan, Staten Island, Brooklyn, or Queens.

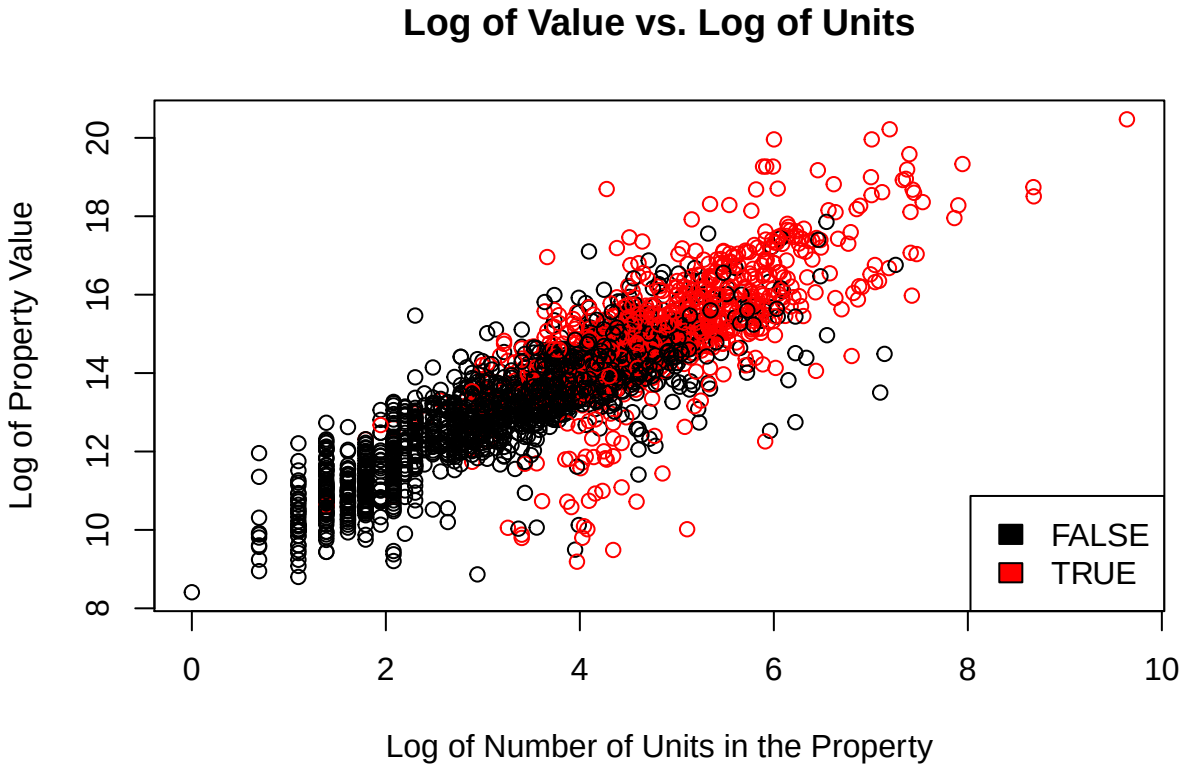
i. Plot propertylogValue against property logUnits.

```
plot(housing$logUnits, housing$logValue,
     ylab = "Log of Property Value",
     xlab = "Log of Number of Units in the Property",
     main = "Log of Value vs. Log of Units")
```




ii. Make the same plot as above, but now include the argument col = factor(housing$after1950). Describe this plot and the covariation between the two variables. What does the coloring in the plot tell us?

```
plot(housing$logUnits, housing$logValue,
     ylab = "Log of Property Value",
     xlab = "Log of Number of Units in the Property",
     col = factor(housing$after1950),
     main = "Log of Value vs. Log of Units")
legend("bottomright", legend = levels(factor(housing$after1950)),fill = unique(factor(housing$after1950
```




From this plot, we can tell that there are strong positive linear relationship between the log of property value and the log of number of units in the property, which means the higher the log of number of units in the property, the higher the log of property value. Because logrithm is a non-linear transformation. It means there is a strong covriantion between the property value and the number of units in the property.

This plot divides all the observation based on the year the property built. Red dots are properties built after 1950, and black dots are properties built before 1950. We can tell that there are more red dots on the upper right corner of the plot, which means properties built after 1950 have higher property values and higher number of units in the property.

iii. The cor() function calculates the correlation coefficient between two variables. Whatis the correlation between propertylogValue and propertylogUnits in (i) the whole data, (ii) just Manhattan (iii) just Brooklyn (iv) for properties built after 1950 (v) for properties built before 1950?

```
cor(housing$logValue, housing$logUnits)
```

```
## [1] 0.8727348
```

```
cor(housing[housing$Borough=="Manhattan","logValue"], housing[housing$Borough=="Manhattan","logUnits"])
```

```
## [1] 0.8830348
```

```
cor(housing[housing$Borough=="Brooklyn","logValue"], housing[housing$Borough=="Brooklyn","logUnits"])
```

```
## [1] 0.9102601
```

```
cor(housing[housing$after1950=="TRUE","logValue"], housing[housing$after1950=="TRUE","logUnits"])
```

```
## [1] 0.7285898
```

```
cor(housing[housing$after1950=="FALSE","logValue"], housing[housing$after1950=="FALSE","logUnits"])
```

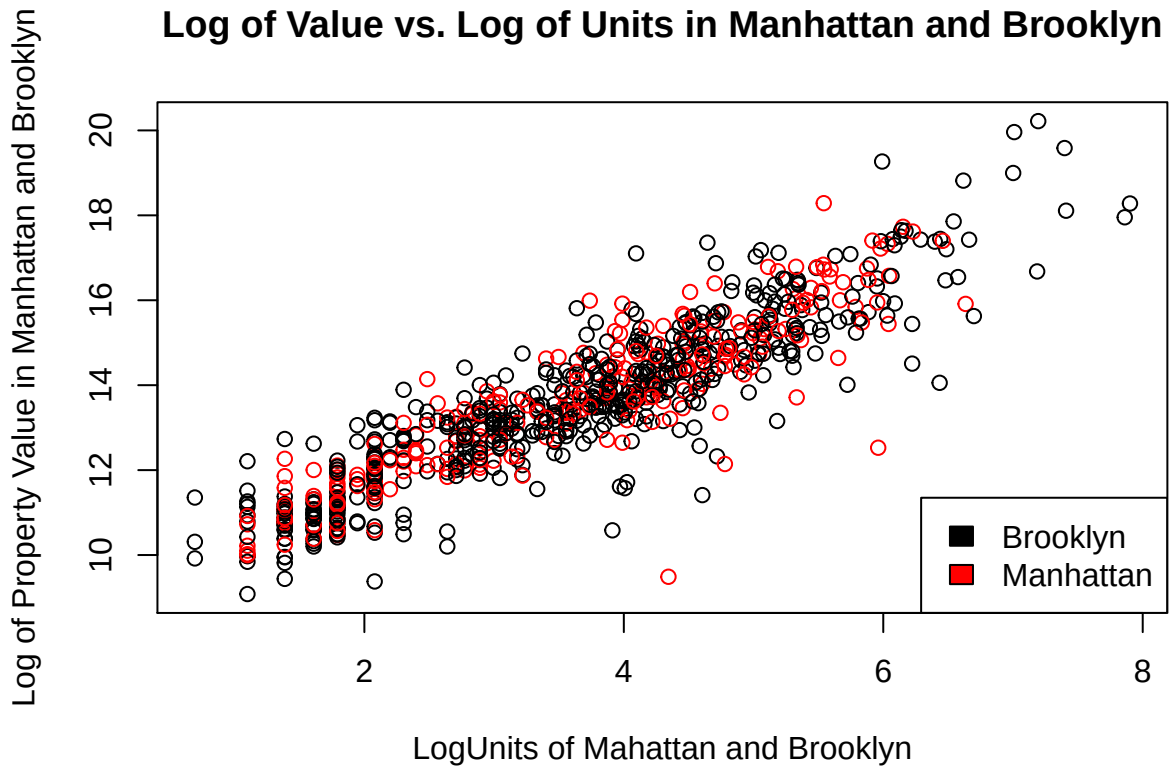
```
## [1] 0.8630975
```

The correlation between property logValue and property logUnits in the whole data is 0.8727348. The correlation between property logValue and property logUnits in just Manhattan is 0.8830348. The correlation between property logValue and property logUnits just Brooklyn is 0.9102601. The correlation between property logValue and property logUnits built after 1950 is 0.7285898. The correlation between property logValue and property logUnits built before 1950 is 0.8630975.

iv. Make a single plot showing propertylogValue against propertylogUnits for Manhattan and Brooklyn.

```
plot(housing[housing$Borough== c("Manhattan","Brooklyn"),"logUnits"],
     housing[housing$Borough==c("Manhattan","Brooklyn"),"logValue"],
     ylab = "Log of Property Value in Manhattan and Brooklyn",
     xlab = "LogUnits of Mahattan and Brooklyn",
     main = "Log of Value vs. Log of Units in Manhattan and Brooklyn",
     col= factor(housing$Borough==c("Manhattan","Brooklyn")))

legend("bottomright", legend = levels(factor(housing$Borough[which(housing$Borough==c("Manhattan","Brook
       fill = unique(factor(housing$Borough==c("Manhattan","Brooklyn"))))
```




v. Consider the following block of code. Give a single line of R code which gives the same final answer as the block of code. There are a few ways to do this.

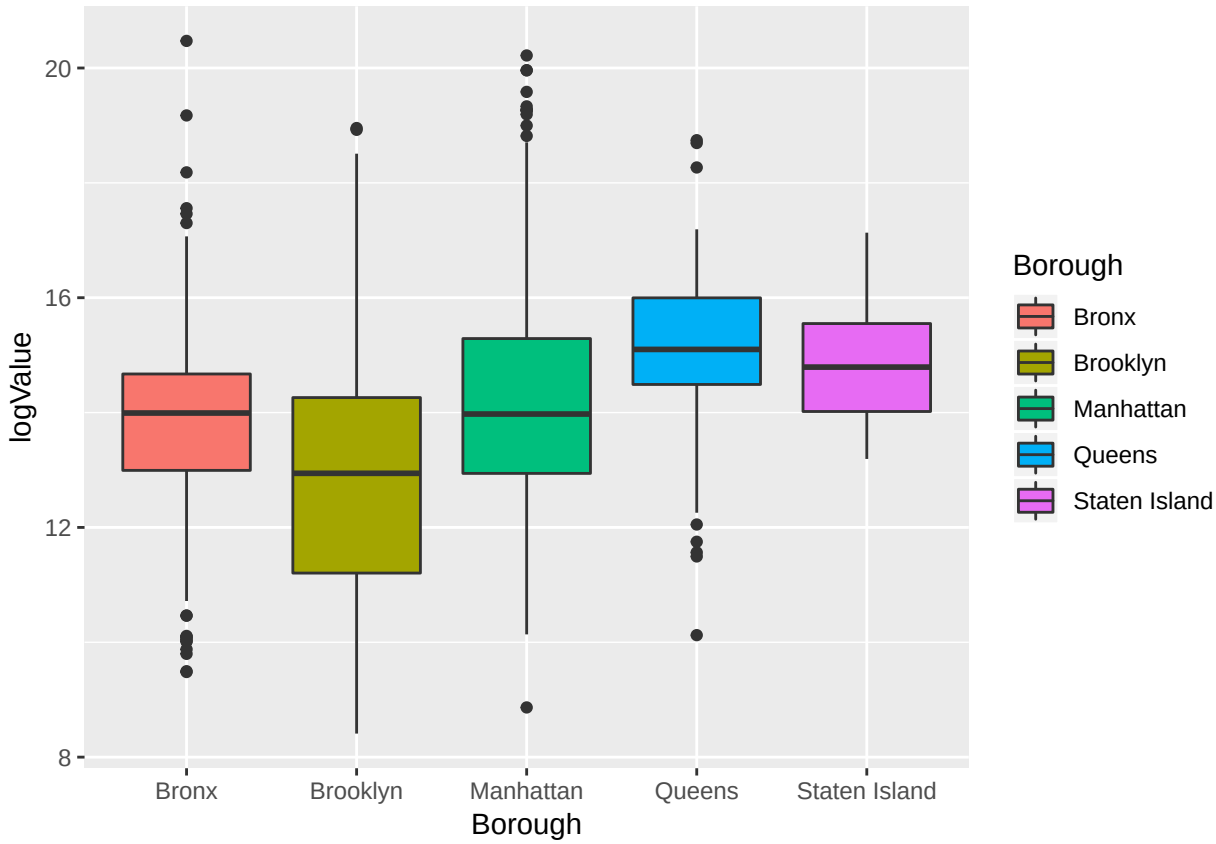
```
median(housing$Value[(which(housing$Borough=="Manhattan"))])
```

```
## [1] 1172362
```

This code gives the same result as the chunck of code, which gives us the median of property value in Manhattan, which is 1172362.

vi. Make side-by-side box plots comparing propertylogValueacross the five boroughs.

```
library(ggplot2)
ggplot(housing, aes(Borough, logValue, fill=Borough))+
  geom_boxplot()
```




vii. For five boroughs, what are the median property values?

```
tapply(housing$Value, housing$Borough, median)
```

```
##         Bronx      Brooklyn     Manhattan        Queens Staten Island
##       1192950        417610       1172362       3611700       2654100
```

From the result above, we can see that the median property value of Bronx is 1192950, for Brooklyn is 417610, for Manhattan is 1172362, for Queens is 3611700, for Staten Island is 2654100.