Initial-code_Project-code.R

HRandhaw

2020-11-16

```r
#CMTH 8120

#Load Dataset
housing <- read.csv('Housingdata.csv', header = TRUE)

#Data Summary
summary(housing)
```

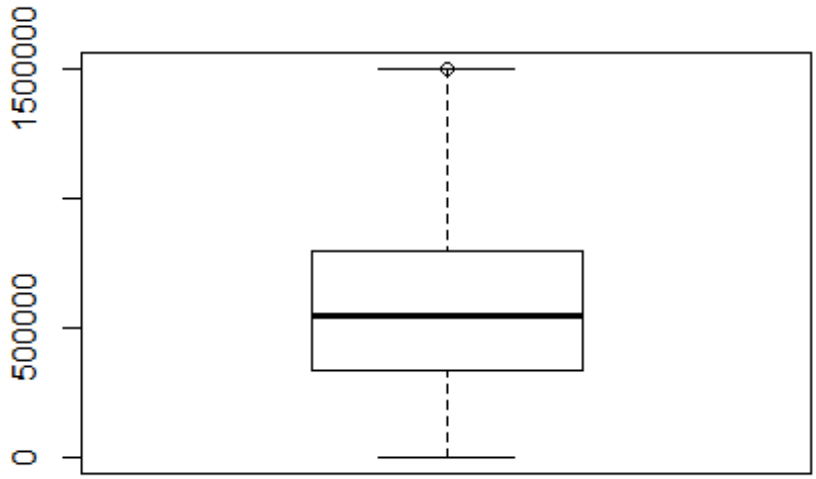
```
##        X             BOROUGH                      NEIGHBORHOOD
##  Min.   :     1   Min.   :1.000   FLUSHING-NORTH         : 10975
##  1st Qu.: 86266   1st Qu.:2.000   UPPER EAST SIDE (59-79):  7235
##  Median :172530   Median :3.000   UPPER EAST SIDE (79-96):  6310
##  Mean   :172530   Mean   :2.996   UPPER WEST SIDE (59-79):  5886
##  3rd Qu.:258795   3rd Qu.:4.000   BEDFORD STUYVESANT     :  5793
##  Max.   :345059   Max.   :5.000   MIDTOWN WEST           :  5648
##                                   (Other)                :303212
##                      BUILDING.CLASS.CATEGORY TAX.CLASS.AS.OF.FINAL.ROLL
##  01 ONE FAMILY DWELLINGS        : 57674     1      :160549
##  02 TWO FAMILY DWELLINGS        : 49164     2      :121762
##  10 COOPS - ELEVATOR APARTMENTS : 38905     4      : 26053
##  13 CONDOS - ELEVATOR APARTMENTS: 36966     2A     :  9935
##  01  ONE FAMILY DWELLINGS       : 18618     2C     :  7535
##  02  TWO FAMILY DWELLINGS       : 16508     1A     :  5950
##  (Other)                        :127224     (Other): 13275
##      BLOCK            LOT          EASE.MENT
##  Min.   :    1   Min.   :   1.0   Mode:logical
##  1st Qu.: 1330   1st Qu.:  22.0   NA's:345059
##  Median : 3361   Median :  50.0
##  Mean   : 4307   Mean   : 367.7
##  3rd Qu.: 6383   3rd Qu.: 438.0
##  Max.   :16350   Max.   :9139.0
##
##  BUILDING.CLASS.AS.OF.FINAL.ROLL                      ADDRESS
##  D4     : 51368                  1335 AVENUE OF THE AMERIC:   940
##  R4     : 47613                  102 WEST 57TH STREET     :   755
##  A1     : 28523                  1335 AVENUE OF THE AMER  :   239
##  A5     : 23580                  550 VANDERBILT AVENUE    :   238
##  B2     : 20348                  131-03 40TH   ROAD       :   236
##  B1     : 20043                  131-05 40TH   ROAD       :   231
##  (Other):153584                  (Other)                  :342420
##  APARTMENT.NUMBER    ZIP.CODE     RESIDENTIAL.UNITS COMMERCIAL.UNITS
##         :270269   Min.   :    0   1      :139086    0      :302884
```

```
##  4       :  1196   1st Qu.:10305   0       : 86766    1       : 19028
##  3A      :  1115   Median :11209   2       : 66529            : 15978
##  3B      :  1051   Mean   :10764   3       : 19375    2       :  3924
##  2B      :  1032   3rd Qu.:11357           : 15978    3       :  1069
##  2       :  1005   Max.   :11697   4       :  5419    4       :   582
##  (Other): 69391   NA's   :15      (Other): 11906    (Other):  1594
##   TOTAL.UNITS     LAND.SQUARE.FEET GROSS.SQUARE.FEET   YEAR.BUILT
##  1       :153857   - 0    : 75244   - 0    : 86377    1,920  : 25229
##  2       : 66385   0       : 47504   0       : 35894    0       : 23934
##  0       : 65530           : 15980           : 15977    1,930  : 20634
##  3       : 22708   2,000  : 15710   2,400  :  1634    1,925  : 17990
##          : 15978   2,500  : 14083   3,000  :  1395    1,910  : 14221
##  4       :  6115   4,000  : 12224   1,600  :  1388    1,950  : 13087
##  (Other): 14486   (Other):164314   (Other):202394    (Other):229964
##  TAX.CLASS.AT.TIME.OF.SALE BUILDING.CLASS.AT.TIME.OF.SALE   SALE.PRICE
##  Min.   :1.000             D4     : 51372                 0      :105065
##  1st Qu.:1.000             R4     : 49647                 10     :  3241
##  Median :1.000             A1     : 28595                 650,000:  1724
##  Mean   :1.654             A5     : 23655                 550,000:  1655
##  3rd Qu.:2.000             B2     : 20357                 600,000:  1636
##  Max.   :4.000             B1     : 20049                 450,000:  1633
##                            (Other):151384                 (Other):230105
##       SALE.DATE         Latitude       Longitude      Community.Board
##  08/15/2019:   907   Min.   :40.50   Min.   :-74.25   Min.   :101
##  06/30/2016:   745   1st Qu.:40.65   1st Qu.:-73.98   1st Qu.:209
##  06/27/2019:   643   Median :40.71   Median :-73.94   Median :313
##  06/28/2019:   615   Mean   :40.71   Mean   :-73.93   Mean   :307
##  07/14/2016:   575   3rd Qu.:40.76   3rd Qu.:-73.86   3rd Qu.:408
##  06/28/2018:   572   Max.   :40.91   Max.   :-73.70   Max.   :503
##  (Other)   :341002   NA's   :10950   NA's   :10950    NA's   :10950
##  Council.District  Census.Tract         BIN               BBL
##  Min.   : 1.00    Min.   :     1   Min.   :1000000   Min.   :0.000e+00
##  1st Qu.:13.00    1st Qu.:   149   1st Qu.:2049511   1st Qu.:2.038e+09
##  Median :28.00    Median :   394   Median :3231406   Median :3.062e+09
##  Mean   :26.69    Mean   :  9933   Mean   :3143269   Mean   :3.031e+09
##  3rd Qu.:40.00    3rd Qu.:  1118   3rd Qu.:4214870   3rd Qu.:4.066e+09
##  Max.   :51.00    Max.   :157903   Max.   :5516445   Max.   :5.081e+09
##  NA's   :10950    NA's   :10950    NA's   :12150     NA's   :12150
##                                          NTA
##                                            : 10950
##  Upper West Side                           :  6194
##  Turtle Bay-East Midtown                   :  5768
##  Forest Hills                              :  5195
##  Upper East Side-Carnegie Hill             :  5111
##  Hudson Yards-Chelsea-Flatiron-Union Square:  4958
##  (Other)                                   :306883
```

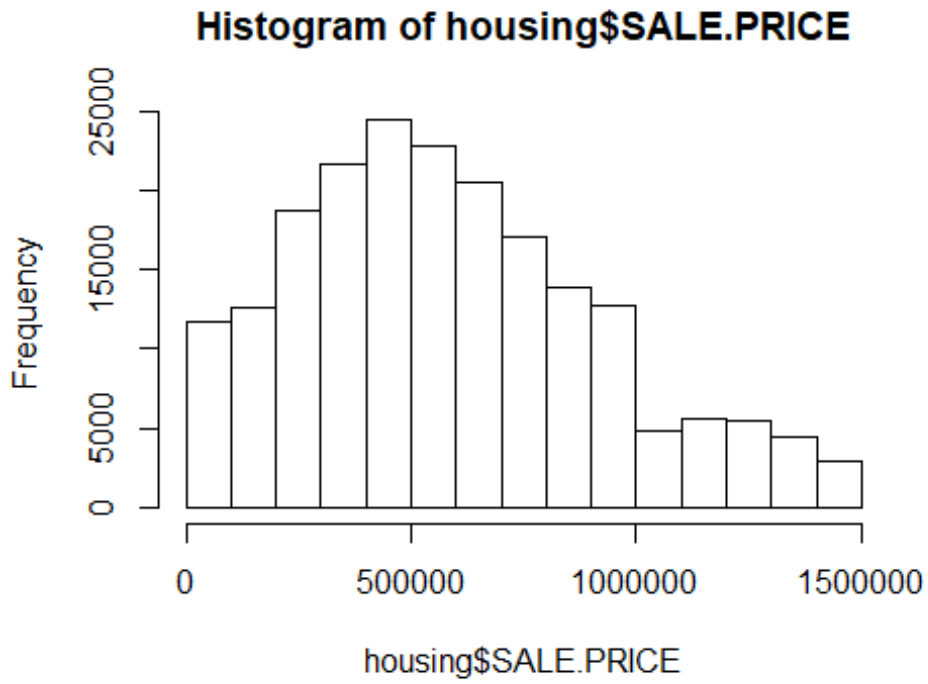
```
str(housing)
```

```
## 'data.frame':    345059 obs. of  30 variables:
##  $ X                             : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ BOROUGH                       : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ NEIGHBORHOOD                  : Factor w/ 261 levels "AIRPORT JFK",..:
83 115 115 156 156 157 157 157 157 165 ...
##  $ BUILDING.CLASS.CATEGORY       : Factor w/ 91 levels "01  ONE FAMILY
DWELLINGS",..: 28 1 13 26 26 26 45 45 86 26 ...
##  $ TAX.CLASS.AS.OF.FINAL.ROLL    : Factor w/ 12 levels
"","1","1A","1B",..: 8 2 7 7 7 7 12 12 12 7 ...
##  $ BLOCK                         : int  7 1643 1643 1320 1365 1042 1009
1009 1042 869 ...
##  $ LOT                           : int  38 122 123 4247 1526 1314 37 37
1316 1066 ...
##  $ EASE.MENT                     : logi  NA NA NA NA NA NA ...
##  $ BUILDING.CLASS.AS.OF.FINAL.ROLL: Factor w/ 190 levels
"","A0","A1","A2",..: 159 6 21 137 137 137 61 61 150 137 ...
##  $ ADDRESS                       : Factor w/ 267866 levels "-00 136TH
AVENUE",..: 209897 60824 62349 99058 171503 144643 4995 4995 144643 182808
...
##  $ APARTMENT.NUMBER              : Factor w/ 9540 levels
"","#4","#9","#PHC",..: 1 1 1 2887 2698 7982 1 1 9145 1132 ...
##  $ ZIP.CODE                      : int  10004 10029 10029 10017 10022
10019 10019 10019 10019 10016 ...
##  $ RESIDENTIAL.UNITS             : Factor w/ 288 levels
"","0","1","1,092",..: 149 3 32 3 3 3 2 2 2 3 ...
##  $ COMMERCIAL.UNITS              : Factor w/ 124 levels
"","0","1","1,132",..: 37 2 3 2 2 2 37 37 2 2 ...
##  $ TOTAL.UNITS                   : Factor w/ 324 levels
"","0","1","1,092",..: 246 3 44 3 3 3 110 110 3 3 ...
##  $ LAND.SQUARE.FEET              : Factor w/ 12601 levels "","-
0","0","1",..: 498 351 351 2 2 2 10975 10975 2 2 ...
##  $ GROSS.SQUARE.FEET             : Factor w/ 12569 levels "","-
0","0","1",..: 11047 4503 6407 2 2 2 1657 1657 2 2 ...
##  $ YEAR.BUILT                    : Factor w/ 193 levels
"","0","1,018",..: 73 73 73 152 145 178 180 180 178 159 ...
##  $ TAX.CLASS.AT.TIME.OF.SALE     : int  2 1 2 2 2 2 4 4 4 2 ...
##  $ BUILDING.CLASS.AT.TIME.OF.SALE : Factor w/ 189 levels
"A0","A1","A2",..: 158 5 20 136 136 136 60 60 149 136 ...
##  $ SALE.PRICE                    : Factor w/ 28299 levels
"0","1","1,000",..: 15380 1 1 1 1 1 1 1 1 1 ...
##  $ SALE.DATE                     : Factor w/ 1459 levels
"01/01/2016","01/01/2017",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Latitude                      : num  40.7 40.8 40.8 40.8 40.8 ...
##  $ Longitude                     : num  -74 -73.9 -73.9 -74 -74 ...
##  $ Community.Board               : int  101 111 111 106 106 104 105 105
104 105 ...
##  $ Council.District              : int  1 8 8 4 4 3 4 4 3 4 ...
##  $ Census.Tract                  : int  9 182 182 90 8603 133 137 137 133
82 ...
##  $ BIN                           : int  1000014 1052276 1052277 1037599
```

```
1076281 1087538 1089420 1089420 1087538 1017603 ...
##  $ BBL                           : num  1.00e+09 1.02e+09 1.02e+09
1.01e+09 1.01e+09 ...
##  $ NTA                           : Factor w/ 193 levels
"","Airport","Allerton-Pelham Gardens",..: 10 52 52 171 171 36 109 109 36 116
...

housing <- subset(housing,select = -c(X))

#Change Sale price variable from factor to numeric
housing$SALE.PRICE <- gsub(",","",housing$SALE.PRICE)
housing$SALE.PRICE <- as.numeric(as.character(housing$SALE.PRICE))
housing <- subset(housing,housing$SALE.PRICE != 0) #Remove records with no
sales price data
housing <- subset(housing,housing$SALE.PRICE != 1) #Remove records with sales
price equal to 1 as it suggests there is no information
housing <- subset(housing,housing$SALE.PRICE < 4000000) #Remove records that
are equal to and over 4,000,000
housing <- subset(housing,housing$SALE.PRICE < 1500000) #Remove records that
are over 1,500,000

#Box plot and histogram to investigate the variable
boxplot(housing$SALE.PRICE)
```




```
hist(housing$SALE.PRICE)
```




```r
#Change the format of other variables
housing$BOROUGH <- as.factor(housing$BOROUGH)
housing$TAX.CLASS.AT.TIME.OF.SALE <-
as.factor(housing$TAX.CLASS.AT.TIME.OF.SALE)
housing$SALE.DATE <- as.Date(housing$SALE.DATE)
housing$Community.Board <- as.factor(housing$Community.Board)
housing$Council.District <- as.factor(housing$Council.District)
housing$ADDRESS <- as.character(housing$ADDRESS)

#Did not change the format of apartment number field as it does not have
provide useful information
housing$TOTAL.UNITS <- gsub(",","",housing$TOTAL.UNITS)
housing$TOTAL.UNITS <- as.numeric(as.character(housing$TOTAL.UNITS))

housing$RESIDENTIAL.UNITS <- gsub(",","",housing$RESIDENTIAL.UNITS)
housing$RESIDENTIAL.UNITS <-
as.numeric(as.character(housing$RESIDENTIAL.UNITS))

housing$COMMERCIAL.UNITS <- gsub(",","",housing$COMMERCIAL.UNITS)
housing$COMMERCIAL.UNITS <-
as.numeric(as.character(housing$COMMERCIAL.UNITS))

housing$LAND.SQUARE.FEET <- gsub("- 0","0",housing$LAND.SQUARE.FEET)
housing$LAND.SQUARE.FEET <- gsub("","0",housing$LAND.SQUARE.FEET)
housing$LAND.SQUARE.FEET <- gsub(",","",housing$LAND.SQUARE.FEET)
housing$LAND.SQUARE.FEET <- as.numeric(housing$LAND.SQUARE.FEET)
```

```r
housing$GROSS.SQUARE.FEET <- gsub("- 0","0",housing$GROSS.SQUARE.FEET)
housing$GROSS.SQUARE.FEET <- gsub("","0",housing$GROSS.SQUARE.FEET)
housing$GROSS.SQUARE.FEET <- gsub(",","",housing$GROSS.SQUARE.FEET)
housing$GROSS.SQUARE.FEET <- as.numeric(housing$GROSS.SQUARE.FEET)

housing$YEAR.BUILT <- gsub(",","",housing$YEAR.BUILT)

#Transformed Data Summary
summary(housing)
```

```
##  BOROUGH                     NEIGHBORHOOD
##  1:35027   FLUSHING-NORTH         :  6615
##  2:20462   UPPER EAST SIDE (59-79):  3356
##  3:50037   MIDTOWN WEST           :  3176
##  4:69431   BAYSIDE                :  3132
##  5:24382   FOREST HILLS           :  3129
##            UPPER EAST SIDE (79-96):  3044
##            (Other)                :176887
##                     BUILDING.CLASS.CATEGORY TAX.CLASS.AS.OF.FINAL.ROLL
##  01 ONE FAMILY DWELLINGS        :38511      1      :96724
##  10 COOPS - ELEVATOR APARTMENTS :29436      2      :77655
##  02 TWO FAMILY DWELLINGS        :27879      4      : 7560
##  13 CONDOS - ELEVATOR APARTMENTS:18446      2C     : 4584
##  01  ONE FAMILY DWELLINGS       :12548      1A     : 4380
##  10  COOPS - ELEVATOR APARTMENTS:10651      2A     : 3279
##  (Other)                        :61868      (Other): 5157
##      BLOCK            LOT       EASE.MENT
BUILDING.CLASS.AS.OF.FINAL.ROLL
##  Min.   :    1   Min.   :   1   Mode:logical   D4     :39738
##  1st Qu.: 1484   1st Qu.:  20   NA's:199339    R4     :23333
##  Median : 3943   Median :  48                  A1     :19278
##  Mean   : 4671   Mean   : 337                  A5     :16553
##  3rd Qu.: 6826   3rd Qu.: 226                  B2     :12701
##  Max.   :16350   Max.   :9087                  B1     :10896
##                                                (Other):76840
##    ADDRESS          APARTMENT.NUMBER    ZIP.CODE     RESIDENTIAL.UNITS
##  Length:199339              :162502   Min.   :    0   Min.   :   0.00
##  Class :character   3A      :   717   1st Qu.:10309   1st Qu.:   0.00
##  Mode  :character   2B      :   697   Median :11211   Median :   1.00
##                     3B      :   677   Mean   :10803   Mean   :   1.22
##                     2A      :   652   3rd Qu.:11364   3rd Qu.:   2.00
##                     TIMES   :   615   Max.   :11697   Max.   :1844.00
##                     (Other): 33479   NA's   :8       NA's   :11417
##  COMMERCIAL.UNITS    TOTAL.UNITS       LAND.SQUARE.FEET
GROSS.SQUARE.FEET
##  Min.   :   0.000   Min.   :   0.000   Min.   :0.000e+00   Min.
:0.000e+00
##  1st Qu.:   0.000   1st Qu.:   0.000   1st Qu.:0.000e+00   1st
Qu.:0.000e+00
```

```
##  Median :   0.000   Median :   1.000   Median :1.006e+08   Median
:1.001e+07
##  Mean   :   0.078   Mean   :   1.336   Mean   :5.835e+11   Mean
:4.465e+10
##  3rd Qu.:   0.000   3rd Qu.:   2.000   3rd Qu.:2.006e+08   3rd
Qu.:1.008e+08
##  Max.   :2261.000   Max.   :2261.000   Max.   :1.090e+17   Max.
:3.007e+15
##  NA's   :11417      NA's   :11417
##   YEAR.BUILT        TAX.CLASS.AT.TIME.OF.SALE
BUILDING.CLASS.AT.TIME.OF.SALE
##  Length:199339      1:104017                 D4     :39739
##  Class :character   2: 87681                 R4     :24442
##  Mode  :character   3:     2                 A1     :19331
##                     4:  7639                 A5     :16605
##                                              B2     :12700
##                                              B1     :10900
##                                              (Other):75622
##    SALE.PRICE         SALE.DATE             Latitude       Longitude
##  Min.   :      2   Min.   :0001-01-20   Min.   :40.50   Min.   :-74.25
##  1st Qu.: 335000   1st Qu.:0004-02-20   1st Qu.:40.64   1st Qu.:-73.98
##  Median : 550000   Median :0007-01-20   Median :40.71   Median :-73.93
##  Mean   : 590353   Mean   :0007-01-29   Mean   :40.71   Mean   :-73.92
##  3rd Qu.: 800000   3rd Qu.:0010-02-20   3rd Qu.:40.76   3rd Qu.:-73.84
##  Max.   :1499999   Max.   :0012-12-20   Max.   :40.91   Max.   :-73.70
##                    NA's   :128285       NA's   :5518    NA's   :5518
##  Community.Board  Council.District  Census.Tract         BIN
##  407    :  9694   51     :  8934   Min.   :     1   Min.   :1000000
##  503    :  8860   50     :  7902   1st Qu.:   169   1st Qu.:2085874
##  412    :  8210   4      :  7330   Median :   450   Median :3396713
##  501    :  7839   19     :  7014   Mean   : 11536   Mean   :3306011
##  413    :  7408   49     :  6916   3rd Qu.:  1277   3rd Qu.:4271408
##  (Other):151810   (Other):155725   Max.   :157903   Max.   :5516445
##  NA's   :  5518   NA's   :  5518   NA's   :5518     NA's   :6139
##       BBL                                                          NTA
##  Min.   :0.000e+00                                                  :  5518
##  1st Qu.:2.050e+09   Forest Hills                                   :  3892
##  Median :3.080e+09   Turtle Bay-East Midtown                        :  3268
##  Mean   :3.183e+09   Flushing                                       :  3043
##  3rd Qu.:4.090e+09   Upper West Side                                :  2752
##  Max.   :5.081e+09   Sheepshead Bay-Gerritsen Beach-Manhattan Beach:  2679
##  NA's   :6139        (Other)                                        :178187
```

```
str(housing)
```

```
## 'data.frame':    199339 obs. of  29 variables:
##  $ BOROUGH                       : Factor w/ 5 levels "1","2","3","4",..:
2 2 2 2 2 2 2 2 2 2 ...
##  $ NEIGHBORHOOD                  : Factor w/ 261 levels "AIRPORT JFK",..:
188 188 188 211 211 211 211 247 247 250 ...
```

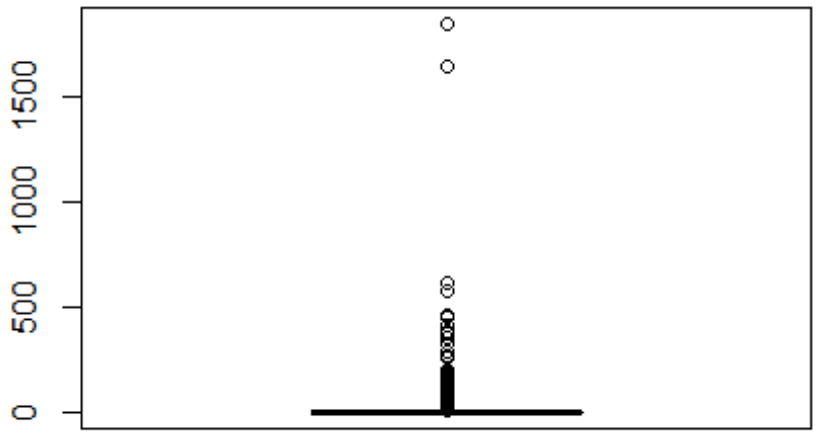
```
##  $ BUILDING.CLASS.CATEGORY        : Factor w/ 91 levels "01  ONE FAMILY
DWELLINGS",..: 39 39 39 37 37 63 63 39 39 39 ...
##  $ TAX.CLASS.AS.OF.FINAL.ROLL     : Factor w/ 12 levels
"","1","1A","1B",..: 12 12 12 12 12 12 12 12 12 12 ...
##  $ BLOCK                          : int  4320 4320 4320 4196 4196 4196
4196 3988 3988 4835 ...
##  $ LOT                            : int  1 1 1 7 7 9 9 34 34 6 ...
##  $ EASE.MENT                      : logi  NA NA NA NA NA NA ...
##  $ BUILDING.CLASS.AS.OF.FINAL.ROLL: Factor w/ 190 levels
"","A0","A1","A2",..: 87 87 87 113 113 126 126 87 87 87 ...
##  $ ADDRESS                        : chr  "2140 HOLLAND AVENUE" "2140
HOLLAND AVENUE" "2140 HOLLAND AVENUE" "3049 BUHRE AVENUE" ...
##  $ APARTMENT.NUMBER               : Factor w/ 9540 levels
"","#4","#9","#PHC",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ ZIP.CODE                       : int  10462 10462 10462 10461 10461
10461 10461 10461 10461 10466 ...
##  $ RESIDENTIAL.UNITS              : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ COMMERCIAL.UNITS               : num  1 1 1 2 2 1 1 1 1 1 ...
##  $ TOTAL.UNITS                    : num  1 1 1 2 2 1 1 1 1 1 ...
##  $ LAND.SQUARE.FEET               : num  1.02e+10 1.02e+10 1.02e+10
7.01e+08 7.01e+08 ...
##  $ GROSS.SQUARE.FEET              : num  2.02e+10 2.02e+10 2.02e+10
8.01e+08 8.01e+08 ...
##  $ YEAR.BUILT                     : chr  "1932" "1932" "1932" "1925" ...
##  $ TAX.CLASS.AT.TIME.OF.SALE      : Factor w/ 4 levels "1","2","3","4": 4
4 4 4 4 4 4 4 4 4 ...
##  $ BUILDING.CLASS.AT.TIME.OF.SALE : Factor w/ 189 levels
"A0","A1","A2",..: 86 86 86 112 112 125 125 86 86 86 ...
##  $ SALE.PRICE                     : num  384000 384000 384000 336000
336000 480000 480000 288000 288000 320000 ...
##  $ SALE.DATE                      : Date, format: "0001-01-20" "0001-01-
20" ...
##  $ Latitude                       : num  40.9 40.9 40.9 40.8 40.8 ...
##  $ Longitude                      : num  -73.9 -73.9 -73.9 -73.8 -73.8 ...
##  $ Community.Board                : Factor w/ 60 levels
"101","102","103",..: 23 23 23 22 22 22 22 22 22 24 ...
##  $ Council.District               : Factor w/ 51 levels
"1","2","3","4",..: 13 13 13 13 13 13 13 13 13 12 ...
##  $ Census.Tract                   : int  22403 22403 22403 26602 26602
26602 26602 200 200 420 ...
##  $ BIN                            : int  2049411 2049411 2049411 2046602
2046602 2046603 2046603 2041966 2041966 2063283 ...
##  $ BBL                            : num  2.04e+09 2.04e+09 2.04e+09
2.04e+09 2.04e+09 ...
##  $ NTA                            : Factor w/ 193 levels
"","Airport","Allerton-Pelham Gardens",..: 137 137 137 136 136 136 136 176
176 187 ...

#Tax class 1 and 2 are residential and building class category 3 and 4 are
commercial
```

```r
#Handle missing values in # of apartments - replace all 0 number of units
with 1 res where class is 1 and 2, and 1 commercial where class is 3 and 4
#Get the count of rows with 0 residential unitls and tax class a 1:
nrow(housing[housing$RESIDENTIAL.UNITS == "0" &
housing$TAX.CLASS.AT.TIME.OF.SALE == "1",])
```

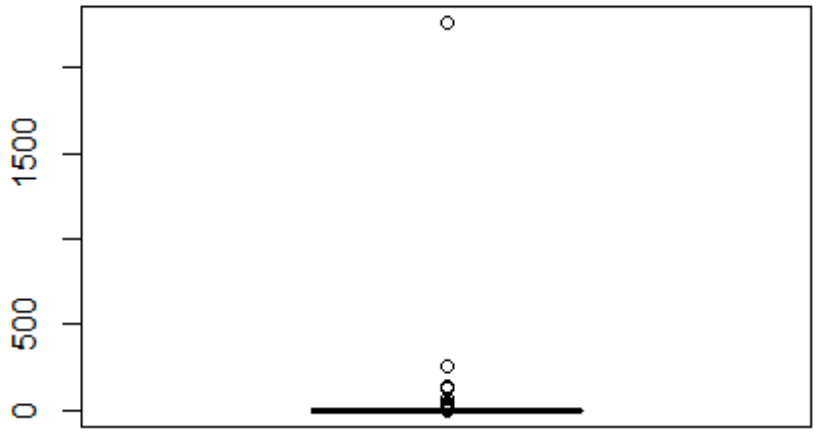
```
## [1] 2807
```

```r
housing[is.na(housing$RESIDENTIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "1","RESIDENTIAL.UNITS"] <- 1 #Replace all NA where Tax class is 1 with 1
residential
housing[is.na(housing$RESIDENTIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "2","RESIDENTIAL.UNITS"] <- 1 #Replace all NA where Tax class is 2 with 1
residential
housing[is.na(housing$RESIDENTIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "3","RESIDENTIAL.UNITS"] <- 0 #Replace all NA where Tax class is 3 with 0
residential
housing[is.na(housing$RESIDENTIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "4","RESIDENTIAL.UNITS"] <- 0 #Replace all NA where Tax class is 4 with 0
residential

housing[is.na(housing$COMMERCIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "1","COMMERCIAL.UNITS"] <- 0 #Replace all NA where Tax class is 1 with 0
Commercial
housing[is.na(housing$COMMERCIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "2","COMMERCIAL.UNITS"] <- 0 #Replace all NA where Tax class is 2 with 0
Commercial
housing[is.na(housing$COMMERCIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "3","COMMERCIAL.UNITS"] <- 1 #Replace all NA where Tax class is 3 with 1
Commercial
housing[is.na(housing$COMMERCIAL.UNITS) & housing$TAX.CLASS.AT.TIME.OF.SALE
== "4","COMMERCIAL.UNITS"] <- 1 #Replace all NA where Tax class is 4 with 1
Commercial

housing[housing$RESIDENTIAL.UNITS == 0 & housing$TOTAL.UNITS == 0 &
housing$TAX.CLASS.AT.TIME.OF.SALE == "1","RESIDENTIAL.UNITS"] <- 1 #Replace
all 0 where Tax class is 1 with 1 residential
housing[housing$RESIDENTIAL.UNITS == 0 & housing$TOTAL.UNITS == 0 &
housing$TAX.CLASS.AT.TIME.OF.SALE == "2","RESIDENTIAL.UNITS"] <- 1 #Replace
all 0 where Tax class is 2 with 1 residential

housing[housing$COMMERCIAL.UNITS == 0 & housing$TOTAL.UNITS == 0 &
housing$TAX.CLASS.AT.TIME.OF.SALE == "3","COMMERCIAL.UNITS"] <- 1 #Replace
all 0 where Tax class is 3 with 1 Commercial
housing[housing$COMMERCIAL.UNITS == 0 & housing$TOTAL.UNITS == 0 &
housing$TAX.CLASS.AT.TIME.OF.SALE == "4","COMMERCIAL.UNITS"] <- 1 #Replace
all 0 where Tax class is 4 with 1 Commercial

#recalculate Total units after the 0s have been updated correctly
housing$TOTAL.UNITS <- housing$RESIDENTIAL.UNITS + housing$COMMERCIAL.UNITS
```

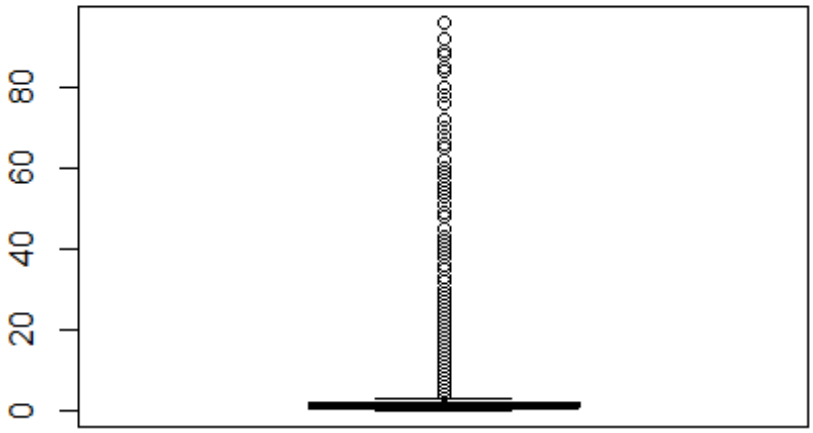
```r
#Investigate residential unit and Commercial units using box plot
boxplot(housing$RESIDENTIAL.UNITS)
```



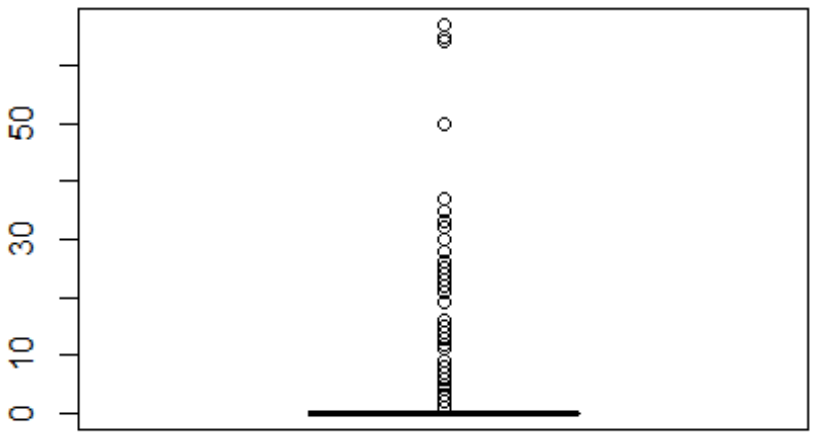

```r
boxplot(housing$COMMERCIAL.UNITS)
```



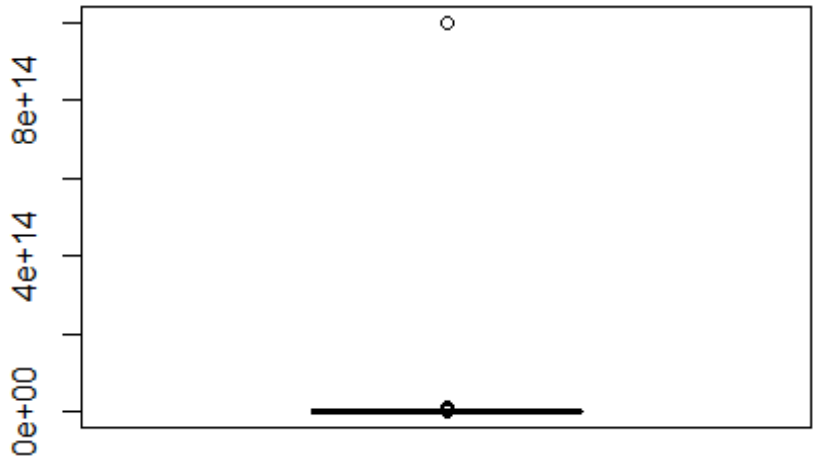

```
#remove records with 500 or more residential units and records with more than
400 commercail units to remove outliers
housing <- housing[housing$RESIDENTIAL.UNITS < 100,]
housing <- housing[housing$COMMERCIAL.UNITS < 80,]
summary(housing$RESIDENTIAL.UNITS)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   1.000   1.349   2.000  96.000

summary(housing$COMMERCIAL.UNITS)

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.00000  0.00000  0.00000  0.06212  0.00000 67.00000

summary(housing$TOTAL.UNITS)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   1.000   1.411   2.000  97.000

boxplot(housing$RESIDENTIAL.UNITS)
```



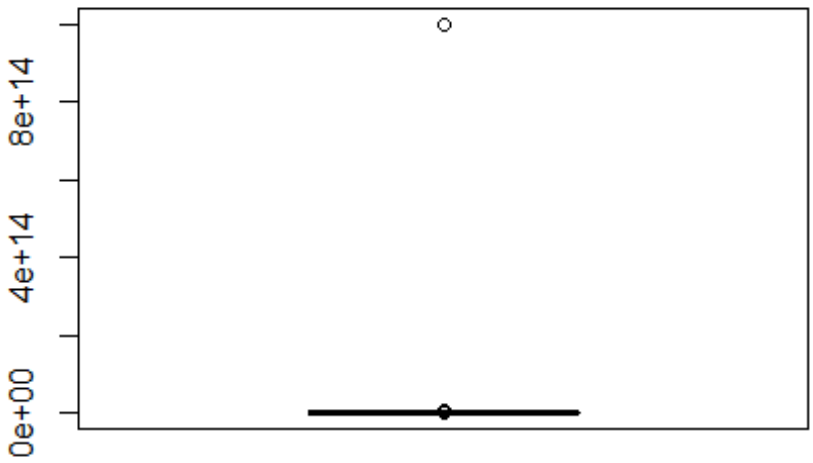

```r
boxplot(housing$COMMERCIAL.UNITS)
```



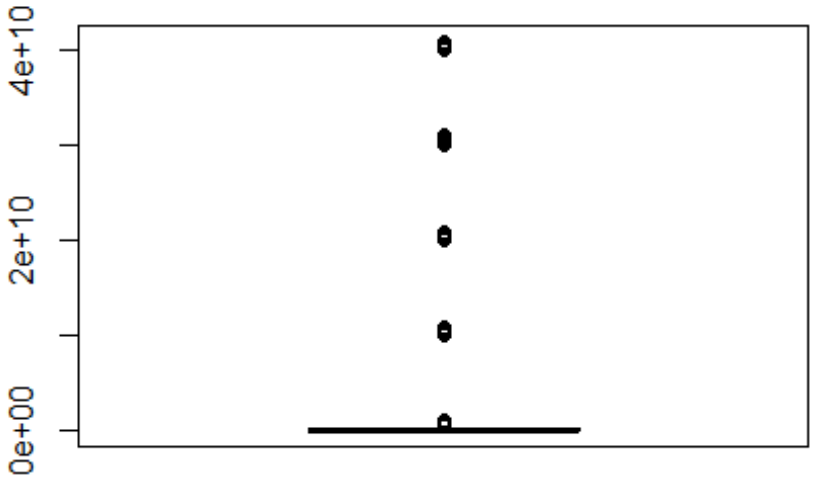
```r
#Investigate and remove outliers from square feet
boxplot(housing$GROSS.SQUARE.FEET)
```




```r
housing <- housing[housing$GROSS.SQUARE.FEET < 50000000000,]
boxplot(housing$LAND.SQUARE.FEET)
```




```
housing <- housing[housing$LAND.SQUARE.FEET < 50000000000,]
boxplot(housing$GROSS.SQUARE.FEET)
```



```r
boxplot(housing$LAND.SQUARE.FEET)

#Dropping 10 Variables that are not useful for the analysis
CleanHousing <-
housing[,c(1,2,3,4,5,6,8,11,12,13,14,15,16,17,18,19,20,21,29)]
str(CleanHousing)
```

```
## 'data.frame':    197563 obs. of  19 variables:
##  $ BOROUGH                       : Factor w/ 5 levels "1","2","3","4",..:
2 2 2 2 2 2 2 2 2 ...
##  $ NEIGHBORHOOD                  : Factor w/ 261 levels "AIRPORT JFK",..:
188 188 188 211 211 211 211 247 247 250 ...
##  $ BUILDING.CLASS.CATEGORY       : Factor w/ 91 levels "01  ONE FAMILY
DWELLINGS",..: 39 39 39 37 37 63 63 39 39 39 ...
##  $ TAX.CLASS.AS.OF.FINAL.ROLL    : Factor w/ 12 levels
"","1","1A","1B",..: 12 12 12 12 12 12 12 12 12 12 ...
##  $ BLOCK                         : int  4320 4320 4320 4196 4196 4196
4196 3988 3988 4835 ...
##  $ LOT                           : int  1 1 1 7 7 9 9 34 34 6 ...
##  $ BUILDING.CLASS.AS.OF.FINAL.ROLL: Factor w/ 190 levels
"","A0","A1","A2",..: 87 87 87 113 113 126 126 87 87 87 ...
##  $ ZIP.CODE                      : int  10462 10462 10462 10461 10461
10461 10461 10461 10461 10466 ...
##  $ RESIDENTIAL.UNITS             : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ COMMERCIAL.UNITS              : num  1 1 1 2 2 1 1 1 1 1 ...
##  $ TOTAL.UNITS                   : num  1 1 1 2 2 1 1 1 1 1 ...
##  $ LAND.SQUARE.FEET              : num  1.02e+10 1.02e+10 1.02e+10
7.01e+08 7.01e+08 ...
##  $ GROSS.SQUARE.FEET             : num  2.02e+10 2.02e+10 2.02e+10
8.01e+08 8.01e+08 ...
##  $ YEAR.BUILT                    : chr  "1932" "1932" "1932" "1925" ...
##  $ TAX.CLASS.AT.TIME.OF.SALE     : Factor w/ 4 levels "1","2","3","4": 4
4 4 4 4 4 4 4 4 4 ...
##  $ BUILDING.CLASS.AT.TIME.OF.SALE : Factor w/ 189 levels
"A0","A1","A2",..: 86 86 86 112 112 125 125 86 86 86 ...
##  $ SALE.PRICE                    : num  384000 384000 384000 336000
336000 480000 480000 288000 288000 320000 ...
##  $ SALE.DATE                     : Date, format: "0001-01-20" "0001-01-
20" ...
##  $ NTA                           : Factor w/ 193 levels
"","Airport","Allerton-Pelham Gardens",..: 137 137 137 136 136 136 136 176
176 187 ...
```

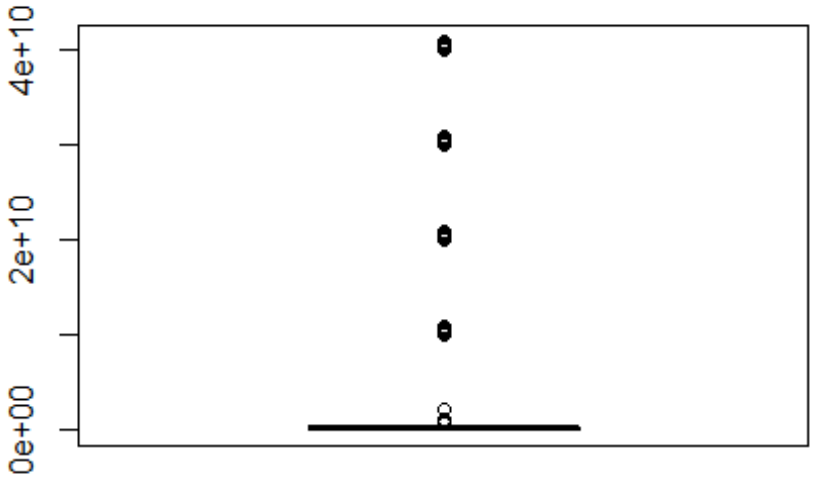
```r
#Moving sales price to be the first column
col_idx <- grep("SALE.PRICE", names(CleanHousing))
CleanHousing <- CleanHousing[, c(col_idx, (1:ncol(CleanHousing))[-col_idx])]
names(CleanHousing)
```

```
##  [1] "SALE.PRICE"                    "BOROUGH"
##  [3] "NEIGHBORHOOD"                  "BUILDING.CLASS.CATEGORY"
```

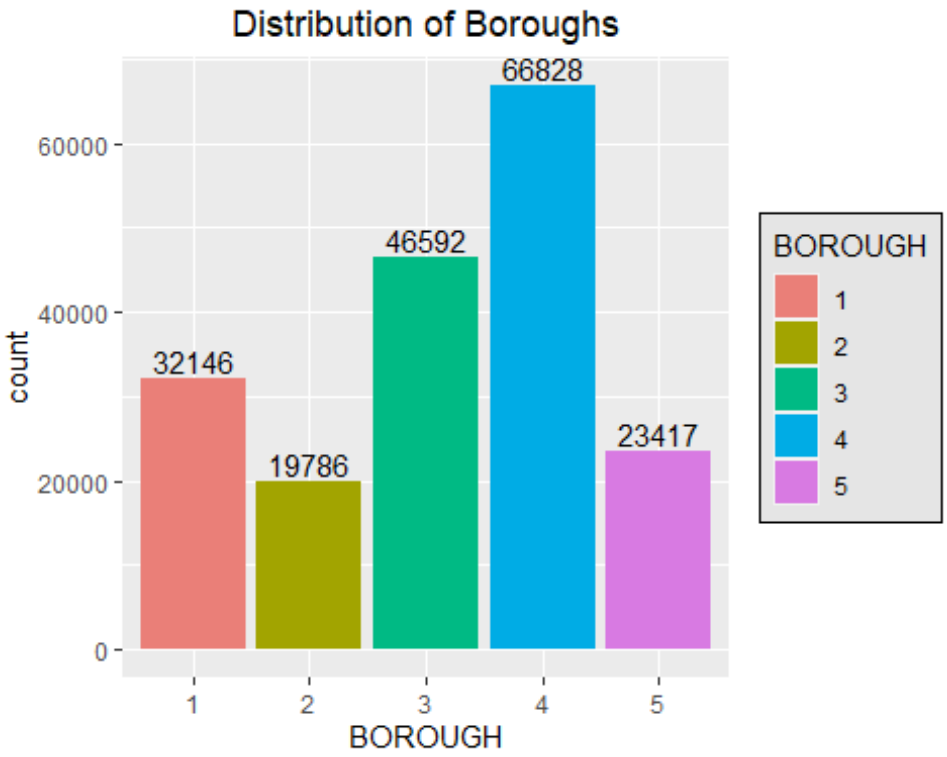
```
##  [5] "TAX.CLASS.AS.OF.FINAL.ROLL"      "BLOCK"
##  [7] "LOT"                             "BUILDING.CLASS.AS.OF.FINAL.ROLL"
##  [9] "ZIP.CODE"                        "RESIDENTIAL.UNITS"
## [11] "COMMERCIAL.UNITS"                "TOTAL.UNITS"
## [13] "LAND.SQUARE.FEET"                "GROSS.SQUARE.FEET"
## [15] "YEAR.BUILT"                      "TAX.CLASS.AT.TIME.OF.SALE"
## [17] "BUILDING.CLASS.AT.TIME.OF.SALE"  "SALE.DATE"
## [19] "NTA"
```

```
#Removing records with Tax calss as of final roll as 3 and 4 as they are not residential units
CleanHousing <- CleanHousing[CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL !="4",]
CleanHousing <- CleanHousing[CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL !="3",]
CleanHousing <- CleanHousing[CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL !="",]
# Tax class roll value 1 relates to most residential which are less than 3 storey, and 2 relates to all other properties including mix use so putting into categories

#change 1a,b,c,d to 1 and 2a,b,c to 2
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[3] <- "1"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[3] <- "1"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[3] <- "1"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[3] <- "1"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[4] <- "2"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[4] <- "2"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[4] <- "2"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[1] <- "1"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[3] <- "2"
levels(CleanHousing$TAX.CLASS.AS.OF.FINAL.ROLL)[3] <- "2"

#Bar graph Plot to explore Categorical varibales:
#install.packages("ggplot2")
#install.packages("ggplot2",repos = "http://cran.us.r-project.org")
library(ggplot2)
```

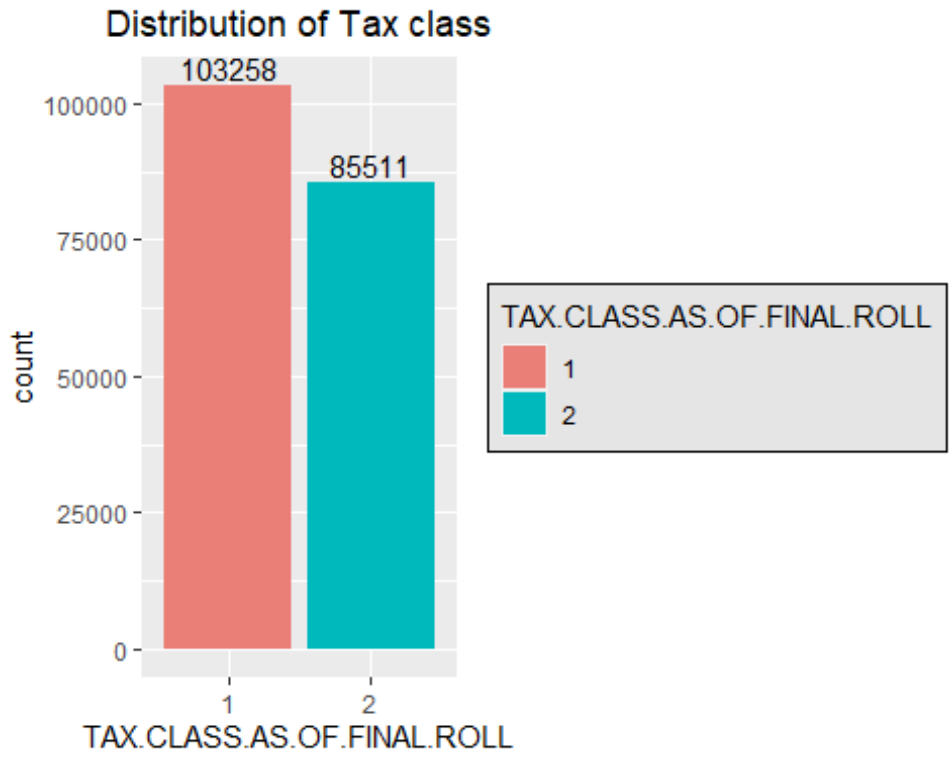
```
## Warning: package 'ggplot2' was built under R version 3.6.3
```




```
options(repr.plot.width=5, repr.plot.height=4)
ggplot(CleanHousing, aes(x = BOROUGH, fill = BOROUGH )) +
  geom_bar()+
  scale_fill_hue(c = 80)+
  ggtitle("Distribution of Boroughs")+
  theme(plot.title = element_text(hjust = 0.5),legend.position="right",
  legend.background = element_rect(fill="grey90",size=0.5,
linetype="solid",colour ="black"))+
  geom_text(stat='count',aes(label=..count..),vjust=-0.25)
```




```
options(repr.plot.width=5, repr.plot.height=4)
ggplot(CleanHousing, aes(x = TAX.CLASS.AS.OF.FINAL.ROLL, fill =
TAX.CLASS.AS.OF.FINAL.ROLL )) +
  geom_bar()+
  scale_fill_hue(c = 80)+
  ggtitle("Distribution of Tax class")+
  theme(plot.title = element_text(hjust = 0.5),legend.position="right",
        legend.background = element_rect(fill="grey90",size=0.5,
linetype="solid",colour ="black"))+
  geom_text(stat='count',aes(label=..count..),vjust=-0.25)
```




```
#average number of units per borough:
library(plyr)

## Warning: package 'plyr' was built under R version 3.6.3

ddply(CleanHousing, .(BOROUGH), summarize,  size=mean(TOTAL.UNITS))

##   BOROUGH     size
## 1       1 1.105270
## 2       2 1.738451
## 3       3 1.622403
## 4       4 1.354836
## 5       5 1.283128

#get a summary of price based on building type or location
library(plyr)
ddply(CleanHousing, .(BOROUGH), summarize,Total =
length(BOROUGH),Max_price=max(SALE.PRICE),Min_price=min(SALE.PRICE))

##   BOROUGH Total Max_price Min_price
## 1       1 32146   1499999         2
## 2       2 19786   1496000         2
## 3       3 46592   1499999         2
## 4       4 66828   1499077         3
## 5       5 23417   1490000        10
```

```
ddply(CleanHousing, .(TAX.CLASS.AT.TIME.OF.SALE), summarize,Total =
length(TAX.CLASS.AT.TIME.OF.SALE),Max_price=max(SALE.PRICE),Min_price=min(SAL
E.PRICE))
```

```
##   TAX.CLASS.AT.TIME.OF.SALE  Total Max_price Min_price
## 1                         1 103252   1499999         2
## 2                         2  85495   1499999         2
## 3                         4     22   1200000     10000
```

```
ddply(CleanHousing, .(BUILDING.CLASS.AT.TIME.OF.SALE), summarize,Total =
length(BUILDING.CLASS.AT.TIME.OF.SALE),Max_price=max(SALE.PRICE),Min_price=mi
n(SALE.PRICE))
```

```
##    BUILDING.CLASS.AT.TIME.OF.SALE Total Max_price Min_price
## 1                              A0  1222   1305000        10
## 2                              A1 19324   1499000         3
## 3                              A2  7944   1496827        10
## 4                              A3   595   1499000        10
## 5                              A4   215   1499999        10
## 6                              A5 16592   1485000        10
## 7                              A6   289    699000        10
## 8                              A7     7   1425000        10
## 9                              A8   383   1325000      3600
## 10                             A9  3731   1499000        10
## 11                             B1 10891   1499500         3
## 12                             B2 12684   1498000        10
## 13                             B3  9288   1499000        10
## 14                             B9  3525   1499900        10
## 15                             C0  8232   1499000         3
## 16                             C1   310   1487398         3
## 17                             C2   920   1493016        10
## 18                             C3  1643   1499000        10
## 19                             C4    98   1450000         2
## 20                             C5   160   1460000        10
## 21                             C6  9326   1499000        10
## 22                             C7    91   1475000         2
## 23                             C8     8    255000    100000
## 24                             C9    17   1100000        10
## 25                             D0   348   1490000     18888
## 26                             D1    17   1433862        10
## 27                             D3     1    699300    699300
## 28                             D4 39679   1499000        10
## 29                             D5     1    849956    849956
## 30                             D6     3    576299    270000
## 31                             D7     6   1290568      1000
## 32                             D9     2    121659        10
## 33                             F9     1    350000    350000
## 34                             G0   232   1480000        10
## 35                             G7     2     10000     10000
## 36                             HR     1    725000    725000
```

```
## 37                                K2     1    850000    850000
## 38                                K4     3    710000    290000
## 39                                M9     2    800000    600000
## 40                                O8     1    625000    625000
## 41                                P7     1    560000    560000
## 42                                R0     1    600000    600000
## 43                                R1  2931   1499999        10
## 44                                R2  2710   1495000        10
## 45                                R3  3798   1475444        10
## 46                                R4 22879   1499999        10
## 47                                R5     1     20000     20000
## 48                                R6   267   1495000        10
## 49                                R7     2    565000    499000
## 50                                R8    89   1425550        10
## 51                                R9  3535   1499000        10
## 52                                RB     1     20000     20000
## 53                                RG     6    625000     95000
## 54                                RR    19   1399000    210000
## 55                                S0    19   1100000        10
## 56                                S1   710   1480000        10
## 57                                S2  1242   1490000        10
## 58                                S3   226   1475000         3
## 59                                S4   132   1490000         7
## 60                                S5   121   1495000         7
## 61                                S9   222   1457330         2
## 62                                V0  2009   1480000         2
## 63                                V1     2   1200000    450000
## 64                                V2     7    865000    225000
## 65                                V3    14    950000         3
## 66                                Z0    30   1150000     40000
```

```
ddply(CleanHousing, .(TAX.CLASS.AS.OF.FINAL.ROLL), summarize,Total =
length(TAX.CLASS.AS.OF.FINAL.ROLL),Max_price=max(SALE.PRICE),Min_price=min(SA
LE.PRICE))
```

```
##   TAX.CLASS.AS.OF.FINAL.ROLL  Total Max_price Min_price
## 1                          1 103258   1499999         2
## 2                          2  85511   1499999         2
```

```
#Remove rows where the frequency of a specific categorical data is less than
25
CleanHousing <- CleanHousing[CleanHousing$BUILDING.CLASS.AT.TIME.OF.SALE %in%
names(which(table(CleanHousing$BUILDING.CLASS.AT.TIME.OF.SALE) > 25)), ]

#Dropping more varibales before the heat map:
cleanhousing1 <- CleanHousing[,c(1,2,5,6,7,9,10,11,12,13,14)]

#Changing factors into numbers:
cleanhousing1$BOROUGH <- as.numeric(factor(cleanhousing1$BOROUGH),levels =
c("1","2","3","4","5"), labels =c(1,2,3,4,5), ordered  = TRUE)
```

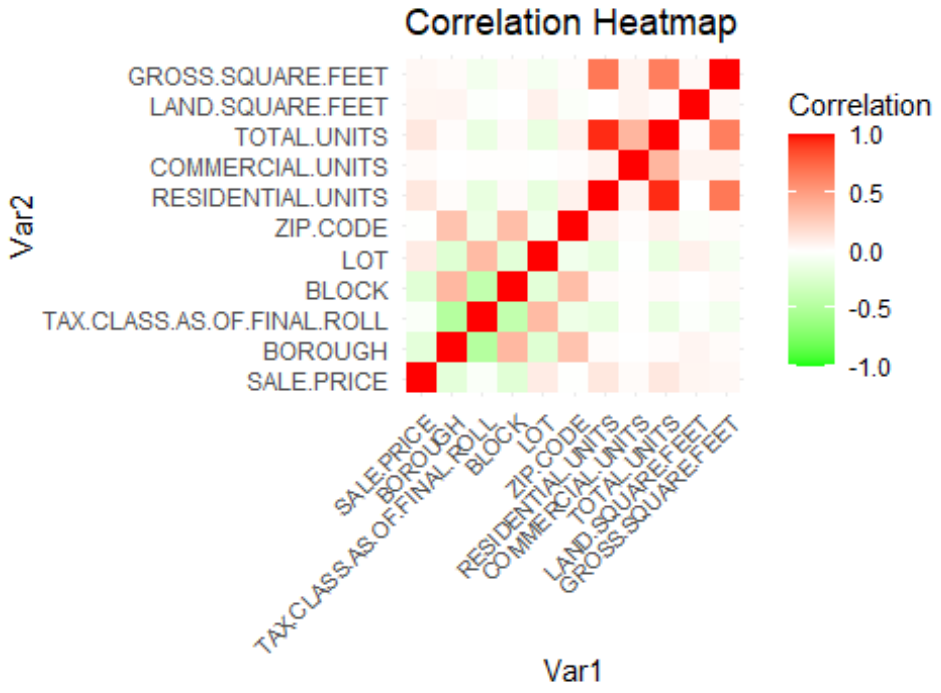
```r
cleanhousing1$TAX.CLASS.AS.OF.FINAL.ROLL <-
as.numeric(factor(cleanhousing1$TAX.CLASS.AS.OF.FINAL.ROLL),levels =
c("1","2"), labels =c(1,2), ordered  = TRUE)
str(cleanhousing1)
```

```
## 'data.frame':    188623 obs. of  11 variables:
##  $ SALE.PRICE                : num  180000 10 386000 499999 150000 ...
##  $ BOROUGH                   : num  3 3 3 3 4 4 4 1 1 1 ...
##  $ TAX.CLASS.AS.OF.FINAL.ROLL: num  2 2 1 2 1 1 2 2 2 2 ...
##  $ BLOCK                     : int  1241 5320 7024 2263 6441 6381 45 402
720 751 ...
##  $ LOT                       : int  1009 1 42 1741 17 13 1019 1205 76 1
...
##  $ ZIP.CODE                  : int  11216 11218 11224 11205 11355 11355
11101 10009 10011 10001 ...
##  $ RESIDENTIAL.UNITS         : num  1 1 1 1 1 2 1 1 1 1 ...
##  $ COMMERCIAL.UNITS          : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ TOTAL.UNITS               : num  1 1 1 1 1 2 1 1 1 1 ...
##  $ LAND.SQUARE.FEET          : num  0e+00 0e+00 7e+08 0e+00 4e+08 ...
##  $ GROSS.SQUARE.FEET         : num  0.00 0.00 2.00e+08 0.00 1.01e+08 ...
```

```r
#Creating a correlation heat map
options(repr.plot.width=8, repr.plot.height=6)
library(ggplot2)
library(reshape2)
```

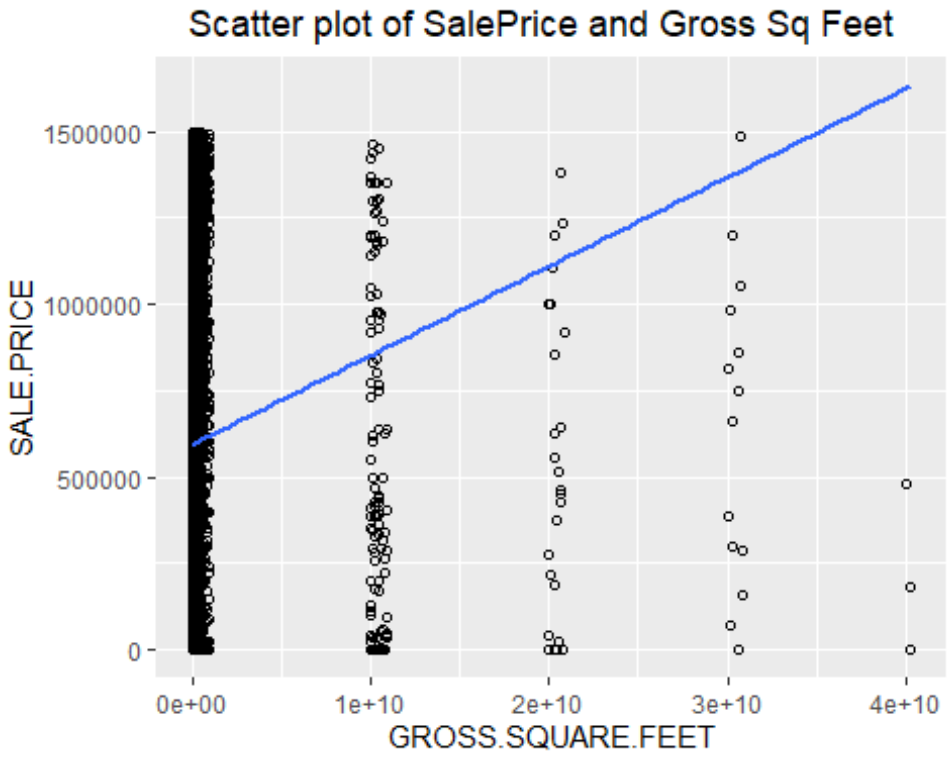
```
## Warning: package 'reshape2' was built under R version 3.6.3
```

```r
qplot(x=Var1, y=Var2, data=melt(cor(cleanhousing1, use="p")), fill=value,
geom="tile") +
  scale_fill_gradient2(low = "green", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1,1), space = "Lab",
                       name="Correlation") +
  theme_minimal()+
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 8, hjust =
1))+
  coord_fixed()+
  ggtitle("Correlation Heatmap") +
  theme(plot.title = element_text(hjust = 0.4))
```

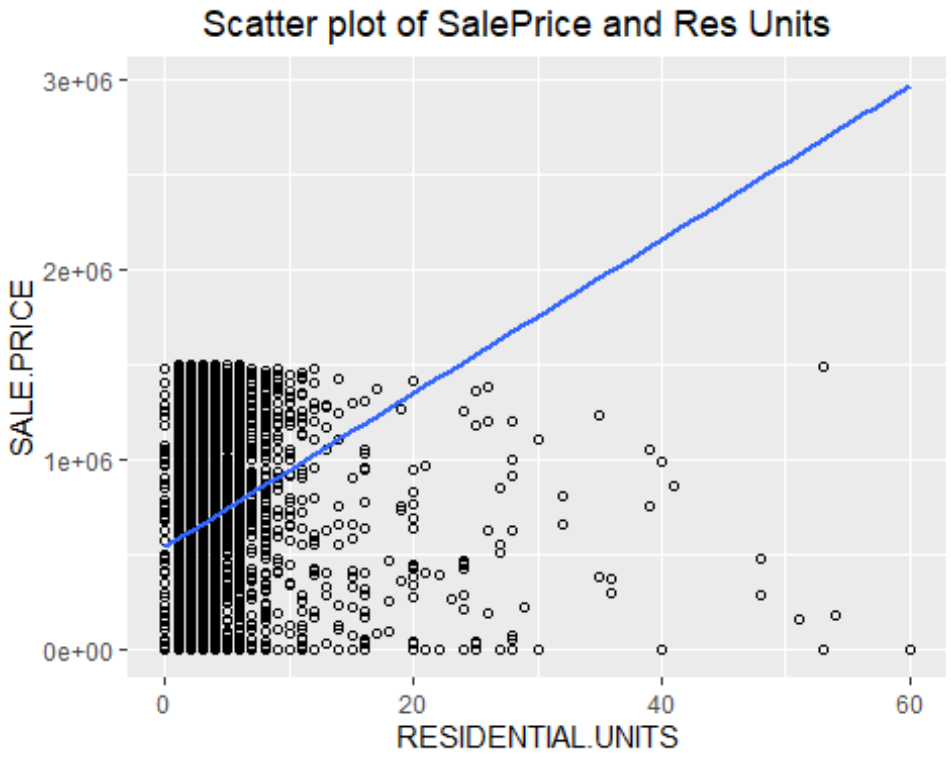



```
#Scatter plot for numeric variables vs sales price
options(repr.plot.width=9, repr.plot.height=6)
ggplot(cleanhousing1, aes(x=GROSS.SQUARE.FEET, y=SALE.PRICE)) +
  geom_point(shape=1) +
  geom_smooth(method=lm , se=FALSE)+
  ggtitle("Scatter plot of SalePrice and Gross Sq Feet") +
  theme(plot.title = element_text(hjust = 0.4))
```

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

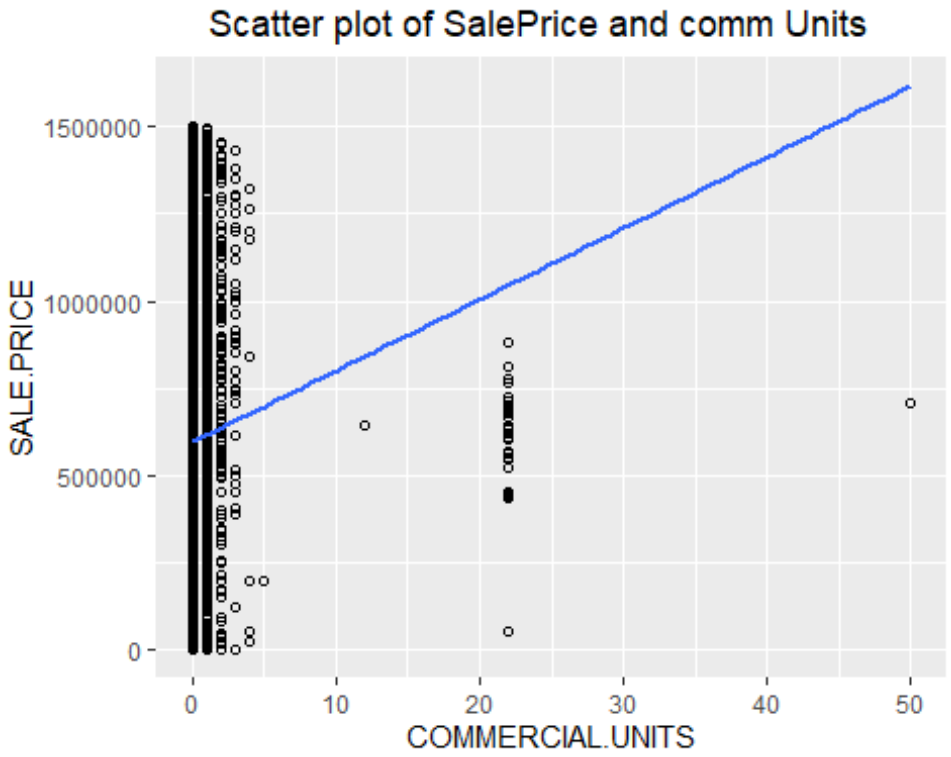



```
ggplot(cleanhousing1, aes(x=RESIDENTIAL.UNITS, y=SALE.PRICE)) +
  geom_point(shape=1) +
  geom_smooth(method=lm , se=FALSE)+
  ggtitle("Scatter plot of SalePrice and Res Units") +
  theme(plot.title = element_text(hjust = 0.4))

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



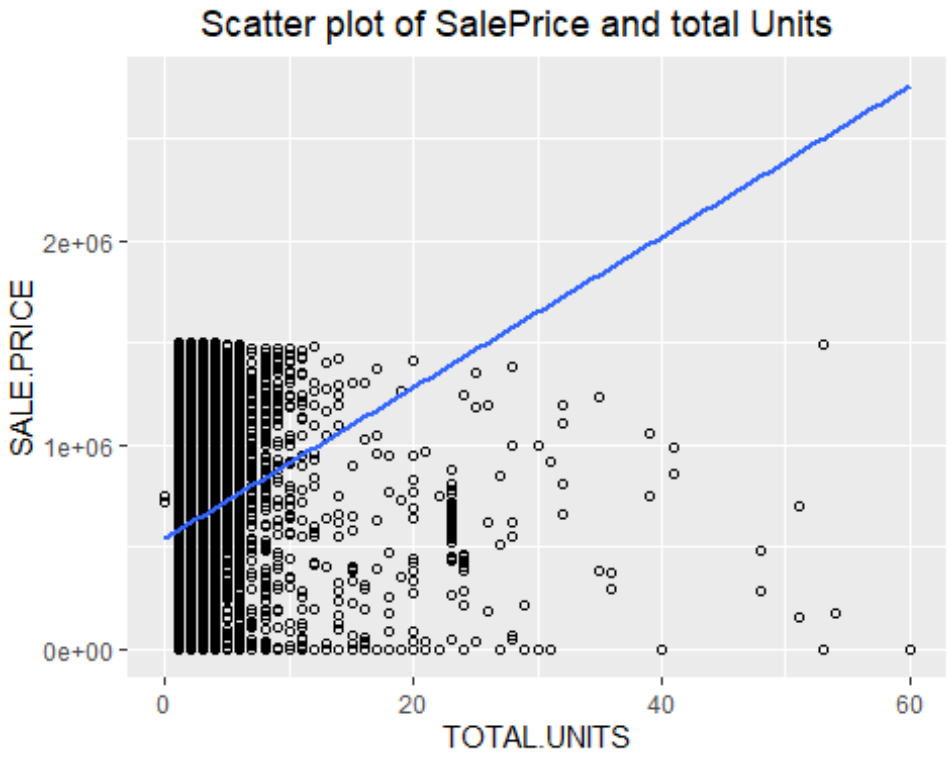

```
ggplot(cleanhousing1, aes(x=COMMERCIAL.UNITS, y=SALE.PRICE)) +
  geom_point(shape=1) +
  geom_smooth(method=lm , se=FALSE)+
  ggtitle("Scatter plot of SalePrice and comm Units") +
  theme(plot.title = element_text(hjust = 0.4))
```

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



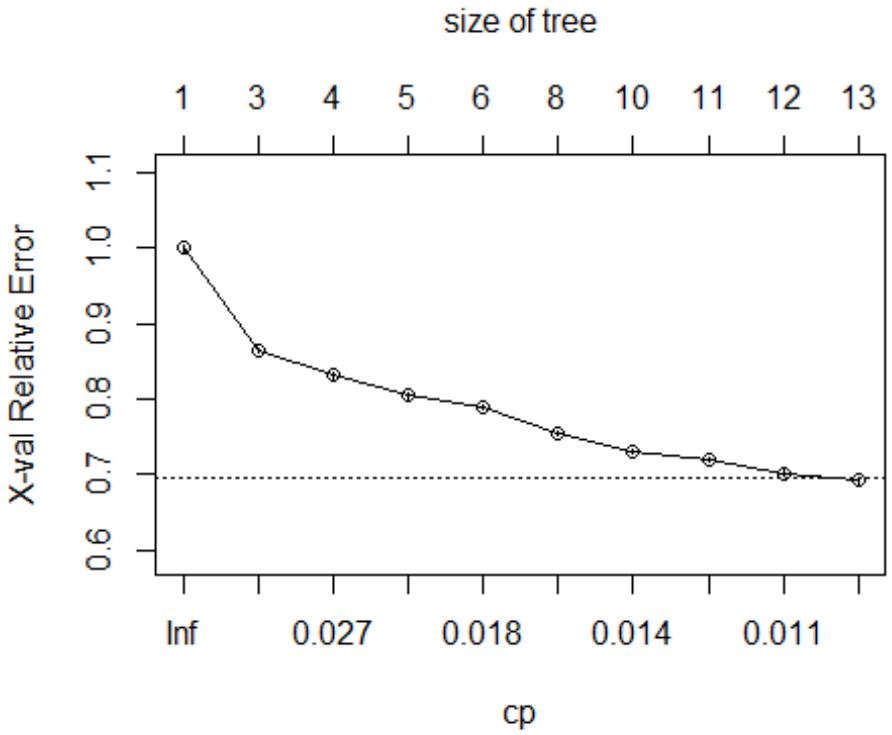

```
ggplot(cleanhousing1, aes(x=TOTAL.UNITS, y=SALE.PRICE)) +
  geom_point(shape=1) +
  geom_smooth(method=lm , se=FALSE)+
  ggtitle("Scatter plot of SalePrice and total Units") +
  theme(plot.title = element_text(hjust = 0.4))
```

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




```
#Linear regression:
#Set training and test set
set.seed(10000)
train.index <- sample(c(1:dim(cleanhousing1)[1]), dim(cleanhousing1)[1]*0.8)
train <- cleanhousing1[train.index,]
valid <- cleanhousing1[-train.index,]
model <- lm(SALE.PRICE ~ ., data = train)
summary(model)

##
## Call:
## lm(formula = SALE.PRICE ~ ., data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2586868  -205082   -27769   187996  1380280
##
## Coefficients: (1 not defined because of singularities)
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    5.460e+05  1.102e+04   49.53  < 2e-16 ***
## BOROUGH                       -6.093e+04  7.699e+02  -79.14  < 2e-16 ***
## TAX.CLASS.AS.OF.FINAL.ROLL -1.703e+05  2.027e+03  -84.01  < 2e-16 ***
## BLOCK                         -2.190e+01  2.545e-01  -86.02  < 2e-16 ***
## LOT                            6.467e+01  1.440e+00   44.92  < 2e-16 ***
## ZIP.CODE                       4.645e+01  1.038e+00   44.76  < 2e-16 ***
## RESIDENTIAL.UNITS              5.181e+04  1.113e+03   46.55  < 2e-16 ***
## COMMERCIAL.UNITS               1.452e+04  2.287e+03    6.35 2.16e-10 ***
```

```
## TOTAL.UNITS                          NA          NA       NA       NA
## LAND.SQUARE.FEET            8.418e-06  3.998e-07   21.06  < 2e-16 ***
## GROSS.SQUARE.FEET          -5.333e-05  2.205e-06  -24.19  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 313300 on 150884 degrees of freedom
##   (4 observations deleted due to missingness)
## Multiple R-squared:  0.1306, Adjusted R-squared:  0.1306
## F-statistic:  2519 on 9 and 150884 DF,  p-value: < 2.2e-16

#Test model
library(forecast)

## Warning: package 'forecast' was built under R version 3.6.3

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

#use predict() to make prediction on a new set
pred1 <- predict(model,valid,type = "response")

## Warning in predict.lm(model, valid, type = "response"): prediction from a
rank-
## deficient fit may be misleading

residuals <- valid$SALE.PRICE - pred1
linreg_pred <- data.frame("Predicted" = pred1, "Actual" = valid$SALE.PRICE,
"Residual" = residuals)
accuracy(pred1, valid$SALE.PRICE)

##                 ME     RMSE      MAE       MPE     MAPE
## Test set -1946.963 313607.7 245535.1 -102400.3 102427.4

#Classification tree:
#install.packages("rpart.plot")
install.packages("rpart.plot",repos = "http://cran.us.r-project.org")

## package 'rpart.plot' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\hrandhaw\AppData\Local\Temp\Rtmp65mhOG\downloaded_packages

library(rpart)
library(rpart.plot)

## Warning: package 'rpart.plot' was built under R version 3.6.3

tree.classfication <- rpart(SALE.PRICE~.,data = train,control =
rpart.control(cp = 0.01))
plotcp(tree.classfication)
```




```
printcp(tree.classfication)

##
## Regression tree:
## rpart(formula = SALE.PRICE ~ ., data = train, control = rpart.control(cp =
0.01))
##
## Variables actually used in tree construction:
## [1] BLOCK             BOROUGH           GROSS.SQUARE.FEET LOT
## [5] ZIP.CODE
##
## Root node error: 1.7034e+16/150898 = 1.1288e+11
##
## n= 150898
##
##          CP nsplit rel error  xerror      xstd
## 1  0.068073      0   1.00000 1.00001 0.0034083
## 2  0.033233      2   0.86385 0.86464 0.0032281
## 3  0.021371      3   0.83062 0.83127 0.0032114
## 4  0.018461      4   0.80925 0.80479 0.0031444
## 5  0.018071      5   0.79079 0.78955 0.0030900
## 6  0.013948      7   0.75465 0.75465 0.0030398
## 7  0.013442      9   0.72675 0.72984 0.0030524
## 8  0.012066     10   0.71331 0.71945 0.0030333
## 9  0.010652     11   0.70124 0.70240 0.0030013
## 10 0.010000     12   0.69059 0.69186 0.0029946
```

```
rpart.plot(tree.classfication, box.palette = "BuOr")
```