Prediction of Price Per Unit for Houses

Tushar Khatri

31/08/2022

# Context

This project was made because we were intrigued and we wanted to gain hands-on experience with the Regression Project .

# Objective

Addressed the assumption regression model such as heteroscedasity , multicollinearity , normality of residual.   
Fitted Multiple Regression Model on House Price dataset using the set of significant predictor variables.  
Calculating Model fit to check whether model fits data well or not

# Problem Statement

Create a Regression Model to estimate the price of houses per unit area.

# Data

The data is the most important aspect of a assignment, to which special attention should be paid. Indeed, the data will heavily affect the findings depending on where we found them, how they are presented, if they are consistent,if there is an outlier, and so on. To obtain, clean, and convert the data, many sub steps are required. We will go through these steps to understand how they've been used in my project

# Dataset

Data set : https://www.kaggle.com/datasets/quantbruce/real-estate-price-prediction

# Importing Library

library(readxl)

## Warning: package 'readxl' was built under R version 4.1.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.1.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(moments)

## Warning: package 'moments' was built under R version 4.1.3

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.1.3

library(psych)

## Warning: package 'psych' was built under R version 4.1.3

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

# Reading data using read.csv

data<-read.csv("C:/Users/TUSHAR/Downloads/PROJECT (CV) EXCEL SHEET/Real estate.csv")  
print(head(data))

## No X1.transaction.date X2.house.age X3.distance.to.the.nearest.MRT.station  
## 1 1 2012.917 32.0 84.87882  
## 2 2 2012.917 19.5 306.59470  
## 3 3 2013.583 13.3 561.98450  
## 4 4 2013.500 13.3 561.98450  
## 5 5 2012.833 5.0 390.56840  
## 6 6 2012.667 7.1 2175.03000  
## X4.number.of.convenience.stores X5.latitude X6.longitude  
## 1 10 24.98298 121.5402  
## 2 9 24.98034 121.5395  
## 3 5 24.98746 121.5439  
## 4 5 24.98746 121.5439  
## 5 5 24.97937 121.5425  
## 6 3 24.96305 121.5125  
## Y.house.price.of.unit.area Bedroom Bathroom  
## 1 37.9 NA NA  
## 2 42.2 NA NA  
## 3 47.3 NA NA  
## 4 54.8 NA NA  
## 5 43.1 NA NA  
## 6 32.1 NA NA

# Exploratory Data Analysis

##### Dropping some columns which will be of no use.

data = data[-c(1,6,7)]  
print(head(data))

## X1.transaction.date X2.house.age X3.distance.to.the.nearest.MRT.station  
## 1 2012.917 32.0 84.87882  
## 2 2012.917 19.5 306.59470  
## 3 2013.583 13.3 561.98450  
## 4 2013.500 13.3 561.98450  
## 5 2012.833 5.0 390.56840  
## 6 2012.667 7.1 2175.03000  
## X4.number.of.convenience.stores Y.house.price.of.unit.area Bedroom Bathroom  
## 1 10 37.9 NA NA  
## 2 9 42.2 NA NA  
## 3 5 47.3 NA NA  
## 4 5 54.8 NA NA  
## 5 5 43.1 NA NA  
## 6 3 32.1 NA NA

##### Renaming Columns

colnames(data)<-c("transaction\_date","house\_age","distance\_MRT\_station","num\_conv\_stores",  
 "price\_perunit\_area","Bedroom.","Bathroom." )

data$Bedroom.[data$price\_perunit\_area<=30]=1  
data$Bedroom.[data$price\_perunit\_area>30 & data$price\_perunit\_area<=55]=2  
data$Bedroom.[data$price\_perunit\_area>55 & data$price\_perunit\_area<=80]=3  
data$Bedroom.[data$price\_perunit\_area>80]=4  
  
data$Bathroom.[data$price\_perunit\_area<=45]=1  
data$Bathroom.[data$price\_perunit\_area>45 & data$price\_perunit\_area<=80]=2  
data$Bathroom.[data$price\_perunit\_area>80]=3

##### Checking Null Values

lapply(data,is.null)

## $transaction\_date  
## [1] FALSE  
##   
## $house\_age  
## [1] FALSE  
##   
## $distance\_MRT\_station  
## [1] FALSE  
##   
## $num\_conv\_stores  
## [1] FALSE  
##   
## $price\_perunit\_area  
## [1] FALSE  
##   
## $Bedroom.  
## [1] FALSE  
##   
## $Bathroom.  
## [1] FALSE

#No null values found in any coloumn of the Dataset

### Data type and Data Description of Coloumn

Transaction\_Date: This coloumn gives the year of the transaction.  
House\_Age : This coloumn represnets the age of House.  
Distance\_MRT\_station : This coloumn describes about the Distance of the property   
 from the MRT station  
Num\_Conv\_Stores: This coloumn gives the number of convienience store near   
 that house.  
Price\_perunit\_area : This coloumn describes about the price per unit of that   
 property.  
Bedroom: This coloumn describes about the number of bedroom in that house.  
Bathroom: This coloumn describes about the number of bathroom in that house.

### Data Structure

str(data)

## 'data.frame': 414 obs. of 7 variables:  
## $ transaction\_date : num 2013 2013 2014 2014 2013 ...  
## $ house\_age : num 32 19.5 13.3 13.3 5 7.1 34.5 20.3 31.7 17.9 ...  
## $ distance\_MRT\_station: num 84.9 306.6 562 562 390.6 ...  
## $ num\_conv\_stores : int 10 9 5 5 5 3 7 6 1 3 ...  
## $ price\_perunit\_area : num 37.9 42.2 47.3 54.8 43.1 32.1 40.3 46.7 18.8 22.1 ...  
## $ Bedroom. : num 2 2 2 2 2 2 2 2 1 1 ...  
## $ Bathroom. : num 1 1 2 2 1 1 1 2 1 1 ...

### Data Cleaning

#1.) Extracting useful info from “transaction\_date” column. Like Year, Quarter.

#2.) Transaction date column has 2 parts Year.MonthCode

#3.) Each month equals to 83.33 units additional to their previous month.

#4.) Eg-> 2013.250 means -> Year is 2013 and month is 250/83.33 (i.e 3rd)

#5.) Converting Transaction month in the form of Quarters.

data$transaction\_year<-as.numeric(substr(data$transaction\_date,1,4))  
data$transaction\_month<-as.numeric((substr(data$transaction\_date\*1000,5,7)))  
data$transaction\_month[data$transaction\_month<=250]=1  
data$transaction\_month[data$transaction\_month<=500 & data$transaction\_month>250]=2  
data$transaction\_month[data$transaction\_month<=750 & data$transaction\_month>500]=3  
data$transaction\_month[data$transaction\_month>=750]=4  
  
print(head(data))

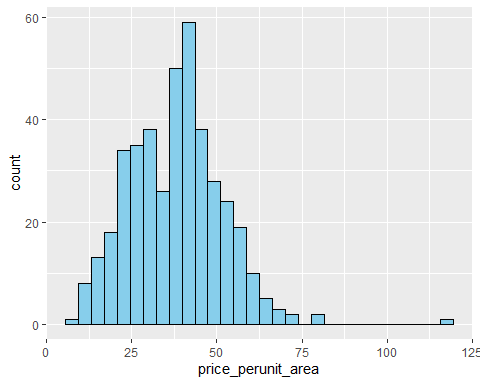
## transaction\_date house\_age distance\_MRT\_station num\_conv\_stores  
## 1 2012.917 32.0 84.87882 10  
## 2 2012.917 19.5 306.59470 9  
## 3 2013.583 13.3 561.98450 5  
## 4 2013.500 13.3 561.98450 5  
## 5 2012.833 5.0 390.56840 5  
## 6 2012.667 7.1 2175.03000 3  
## price\_perunit\_area Bedroom. Bathroom. transaction\_year transaction\_month  
## 1 37.9 2 1 2012 4  
## 2 42.2 2 1 2012 4  
## 3 47.3 2 2 2013 3  
## 4 54.8 2 2 2013 2  
## 5 43.1 2 1 2012 4  
## 6 32.1 2 1 2012 3

# Verifying the assumption of Regression

### Histogram Plots

library(ggplot2)  
p<-ggplot(data, aes(x=price\_perunit\_area)) +   
 geom\_histogram(color="black", fill="skyblue")  
p

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

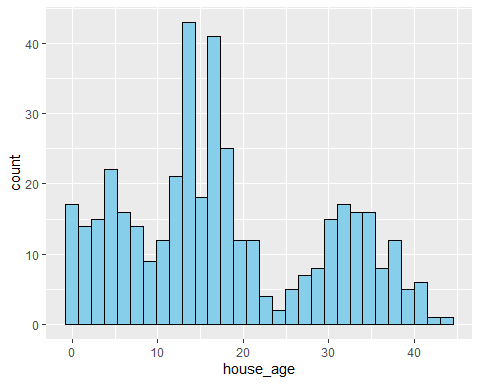


### Interpretation

1) From the plot we can see it is approximately normal distributed and have unique mode  
2) A the values of price per unit area increases count also increases uoto certain level then decreases after certain point.  
having maximum frequency around 42 approximately.

p2<-ggplot(data, aes(x=house\_age)) +   
 geom\_histogram(color="black", fill="skyblue")  
p2

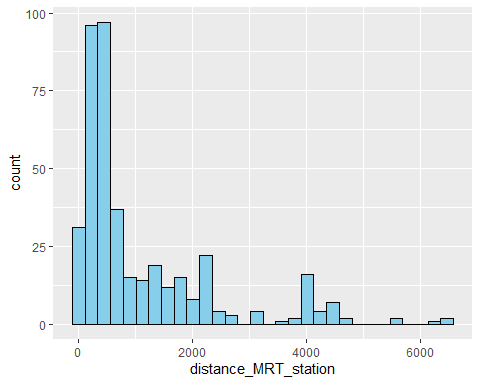
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

 ### Interpretation

1) From the plot we can see it doesn't have unique mode.  
2) From the plot we can see it not distributed uniformly according to the age variable

p3<-ggplot(data, aes(x=distance\_MRT\_station)) +   
 geom\_histogram(color="black", fill="skyblue")  
p3

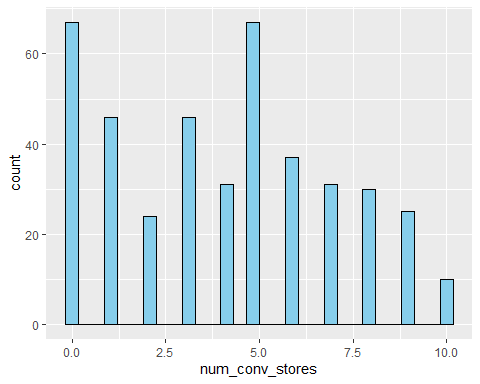
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

 ### Interpretation

1) From the plot we can see it is skewed towards right showing as the distance from mrt station icreases the value of count   
 variables go on decreasing.  
2) After certain distance >5000 we can see the number of houses are very low as compared to before that showing   
 negative correlation between distance and count variable.

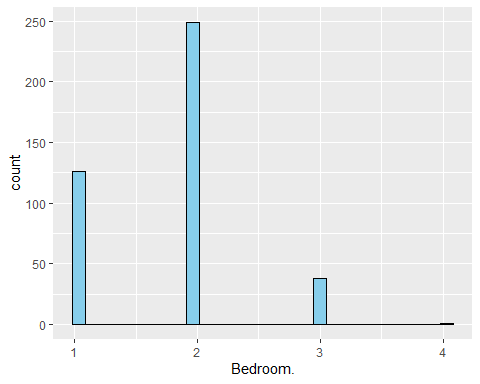
p4<-ggplot(data, aes(x=num\_conv\_stores)) +   
 geom\_histogram(color="black", fill="skyblue")  
p4

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



p5<-ggplot(data, aes(x=Bedroom.)) +   
 geom\_histogram(color="black", fill="skyblue")  
p5

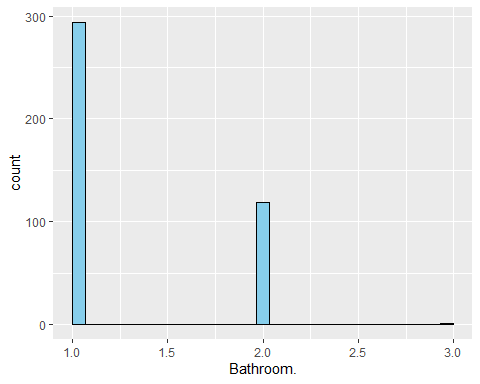
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

 ### Interpretation

1) From the plot we can see the count of houses haveing 2 bedroom is maximum as comapred to the houses having 1 and 3 bedroom.  
2) From the plot we can interpret than mode is 2 .

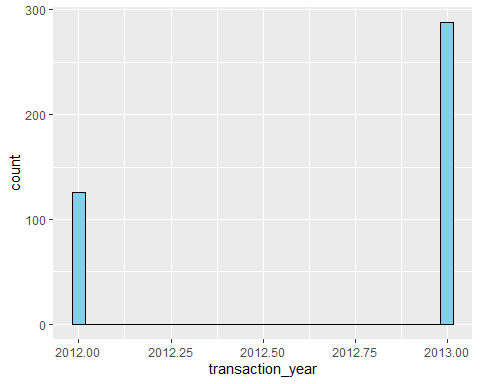
p6<-ggplot(data, aes(x=Bathroom.)) +   
 geom\_histogram(color="black", fill="skyblue")  
p6

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



p7<-ggplot(data, aes(x=transaction\_year)) +   
  
 geom\_histogram(color="black", fill="skyblue")  
p7

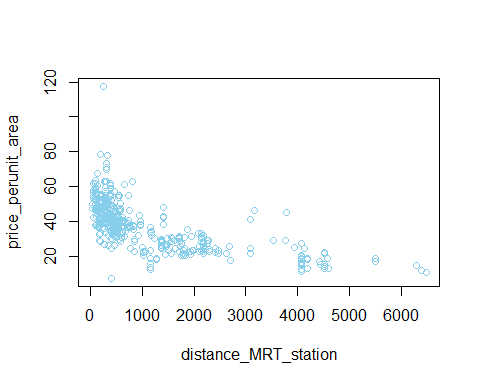
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



# Scatter Plots

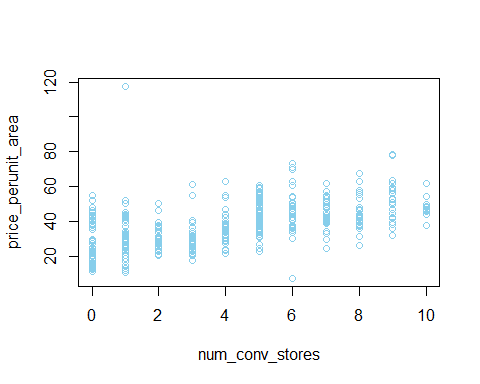
##### These Plots help to explain the values and how they are scattered

plot(data$distance\_MRT\_station,data$price\_perunit\_area,col="skyblue",  
 xlab="distance\_MRT\_station",ylab="price\_perunit\_area")

 ### Interpretation

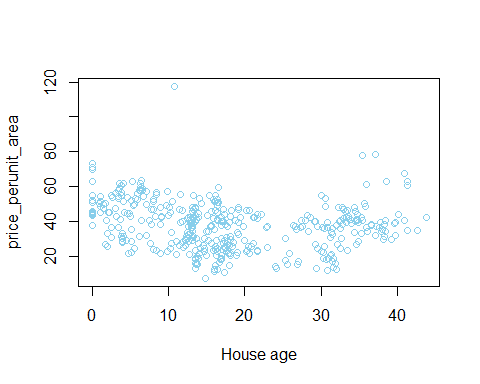
1) From the scatter plot we can see that as distance from MRT station icreases price per unit decreases for houses.  
2) We can see there is oulier showing price per unit area is 120 when distance is closest to station.

plot(data$num\_conv\_stores,data$price\_perunit\_area,col="skyblue",xlab="num\_conv\_stores",ylab="price\_perunit\_area")

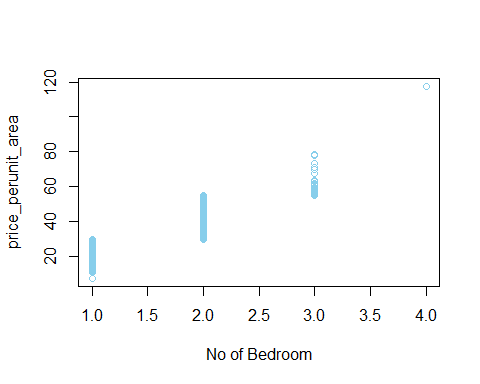
 ### Interpretation

1) From the plot we can see that as the number of conv store increaes the variability between the price of houses decreases   
2) And as the number of stores goes on increasing price also increases showing positive correlation (relationship) between theses two variable

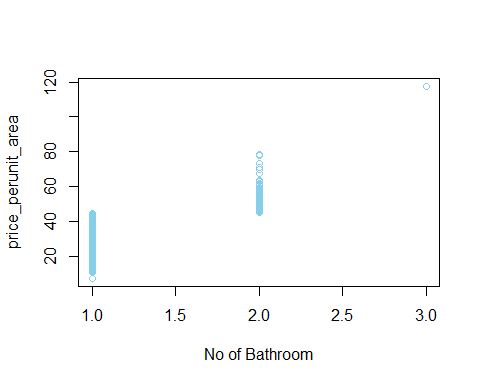
plot(data$house\_age,data$price\_perunit\_area,col="skyblue",xlab="House age",ylab="price\_perunit\_area")



plot(data$Bedroom.,data$price\_perunit\_area,col="skyblue",xlab="No of Bedroom",ylab="price\_perunit\_area")



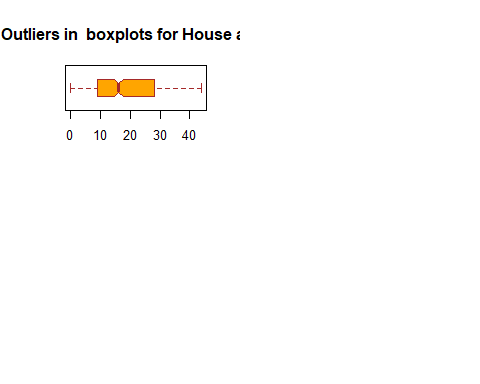
plot(data$Bathroom.,data$price\_perunit\_area,col="skyblue",xlab="No of Bathroom",ylab="price\_perunit\_area")



# Boxplot

#### Looking for Outliers

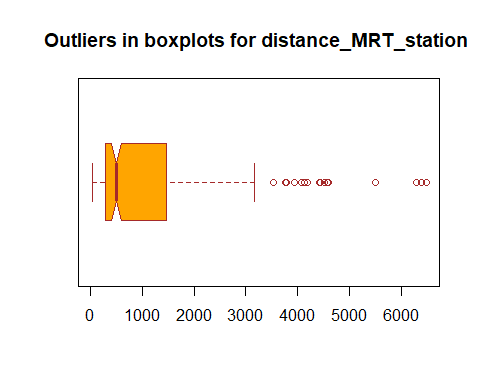
par(mfrow=c(2,2))  
boxplot(data$house\_age,main = "Outliers in boxplots for House age ",col = "orange",  
 border = "brown",horizontal = T,notch = T)



### Interpretation

1) From the boxplot in house age we can see that median is shifted more towards first quartile showing highly skewed data,  
2) We can there there is almost no outlier in house age data.  
3) It is skewed towards right

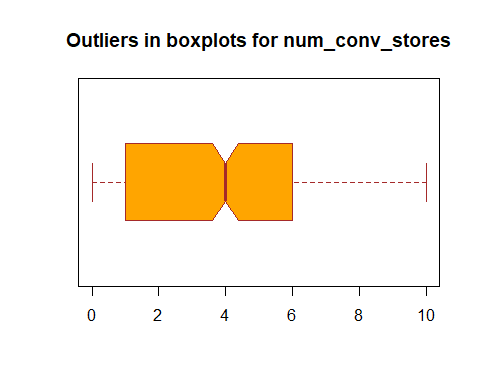
boxplot(data$distance\_MRT\_station,main = "Outliers in boxplots for distance\_MRT\_station "  
 ,col = "orange",border = "brown",horizontal = T,notch = T)



### Interpretation

1) From the boxplot in distance mrt staion we can see that median is shifted more towards first quartile showing highly skewed data,  
2) We can there there are many ouliers in house age data.  
3) It is highly skewed towards right.

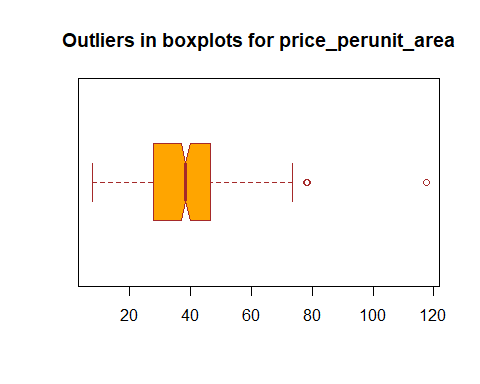
boxplot(data$num\_conv\_stores,main = "Outliers in boxplots for num\_conv\_stores",  
 col = "orange",border = "brown",horizontal = T,notch = T)



### Interpretation

1) From the boxplot in distance mrt staion we can see that median is shifted more towards third quartile showing highly skewed data,  
2) We can there there are almost no outliers in house age data.  
3) It is highly skewed towards left.

boxplot(data$price\_perunit\_area,main = "Outliers in boxplots for price\_perunit\_area",  
 col = "orange",border = "brown",horizontal = T,notch = T)



# Fitting Multiple Regression Model

# In Mutiple Regression, the Null Hypothesis is that the coefficients associated with the variables is equal to zero.   
# The alternate hypothesis is that the coefficients are not equal to zero   
# (i.e. there exists a relationship between the independent variable in question and the dependent variable).  
# P value has 3 stars which means x is of very high statistical significance.  
# P value is less than 0. Genraaly below 0.05 is considered good.  
# R-Squared tells us is the proportion of variation in the dependent (response) variable   
 that has been explained by this model.

model = lm(data$price\_perunit\_area~data$house\_age+data$distance\_MRT\_station+  
 data$num\_conv\_stores+data$Bedroom.+data$Bathroom.,data)   
#Create the Multiple regression model  
print(model)

##   
## Call:  
## lm(formula = data$price\_perunit\_area ~ data$house\_age + data$distance\_MRT\_station +   
## data$num\_conv\_stores + data$Bedroom. + data$Bathroom., data = data)  
##   
## Coefficients:  
## (Intercept) data$house\_age   
## 4.090132 -0.031876   
## data$distance\_MRT\_station data$num\_conv\_stores   
## -0.001845 0.143415   
## data$Bedroom. data$Bathroom.   
## 12.293575 10.704993

# Checking Heteroscedasticty

library(lmtest)

## Warning: package 'lmtest' was built under R version 4.1.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.1.3

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

bptest(model)

##   
## studentized Breusch-Pagan test  
##   
## data: model  
## BP = 37.564, df = 5, p-value = 4.616e-07

### Interpretation

Since p-value of the Breush Pagan test is less than 0.01 we reject the null hypothesis at 1% level of significance

# Checking for Dependence of Residual Using Durbin Watson test

library(car)

## Warning: package 'car' was built under R version 4.1.3

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.1.3

##   
## Attaching package: 'car'

## The following object is masked from 'package:psych':  
##   
## logit

## The following object is masked from 'package:dplyr':  
##   
## recode

durbinWatsonTest(model)

## lag Autocorrelation D-W Statistic p-value  
## 1 0.03990953 1.919742 0.394  
## Alternative hypothesis: rho != 0

### Interpretation

From the output we can see that the test statistic is 1.91 and the corresponding p-value is 0.42. Since this p-value is less than 0.42, we can accept the null hypothesis and conclude that the residuals in this regression model are not autocorrelated.

# Using Pricipal Component analysis to detect Multicollinearity

results <- prcomp(data, scale = TRUE)  
  
#reverse the signs  
results$rotation <- -1\*results$rotation  
  
#display principal components  
results$rotation

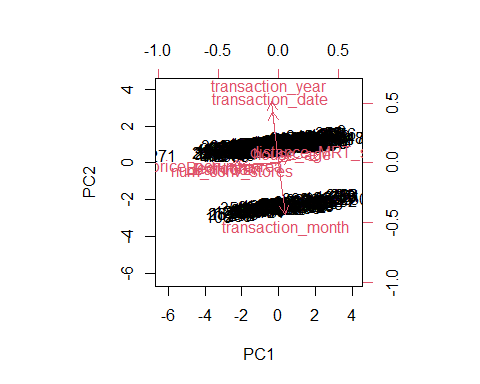
## PC1 PC2 PC3 PC4  
## transaction\_date -0.06666922 0.53226019 -0.07743340 0.61143745  
## house\_age 0.11154650 0.06659942 0.81364201 0.28259993  
## distance\_MRT\_station 0.41394768 0.08935134 -0.28057678 0.29898071  
## num\_conv\_stores -0.38265633 -0.07229235 0.35390902 0.04001680  
## price\_perunit\_area -0.51039024 -0.03061958 -0.07370432 0.07196318  
## Bedroom. -0.47765276 -0.03121597 0.04010773 0.03861140  
## Bathroom. -0.40818257 -0.04049283 -0.34313992 0.21589734  
## transaction\_year -0.07512746 0.63908252 -0.01108391 -0.02581933  
## transaction\_month 0.06653226 -0.53579439 -0.05574056 0.63351628  
## PC5 PC6 PC7 PC8  
## transaction\_date -0.30199941 -0.182837469 -0.08325276 -0.019591242  
## house\_age 0.47891774 0.024684061 -0.10488319 0.015217544  
## distance\_MRT\_station 0.29524586 0.438151967 0.59622173 0.133714046  
## num\_conv\_stores -0.51407097 0.644599523 0.20399480 -0.001157477  
## price\_perunit\_area 0.20941152 -0.134197962 0.10137934 0.809762536  
## Bedroom. 0.19162664 -0.352214876 0.60877259 -0.486197200  
## Bathroom. 0.44598371 0.424669640 -0.44621037 -0.298179896  
## transaction\_year -0.03278198 0.005482773 -0.02804377 -0.018328335  
## transaction\_month -0.21781872 -0.189833322 -0.06482510 -0.012392068  
## PC9  
## transaction\_date 0.447721557  
## house\_age 0.006503473  
## distance\_MRT\_station -0.006618411  
## num\_conv\_stores -0.010113843  
## price\_perunit\_area -0.009889608  
## Bedroom. -0.002462804  
## Bathroom. 0.011399223  
## transaction\_year -0.763487270  
## transaction\_month -0.464984297

#reverse the signs of the scores  
results$x <- -1\*results$x  
  
#display the scores  
head(results$x)

## PC1 PC2 PC3 PC4 PC5 PC6  
## [1,] -0.5774822 -2.4201402 2.1708254 0.6742276 -0.94997288 0.40974644  
## [2,] -0.6585411 -2.4626514 1.0853540 0.4258338 -1.18287897 0.19838571  
## [3,] -1.5762667 0.6653383 -0.8219935 1.5673394 -0.21901866 0.02810579  
## [4,] -1.9004257 0.9947259 -0.7875038 0.8323513 0.18977024 0.18614850  
## [5,] -0.2672329 -2.6038903 -0.4312992 -0.1456914 -0.97087168 -0.63363738  
## [6,] 0.9878323 -2.2018538 -0.7608236 -0.7108650 0.09685606 -0.05294706  
## PC7 PC8 PC9  
## [1,] 0.2997222 -0.04919563 -0.01157597  
## [2,] 0.4823238 0.21389586 -0.01956612  
## [3,] -0.7443043 -0.18397955 -0.15296807  
## [4,] -0.6030667 0.27976687 0.14626258  
## [5,] 0.4099726 0.26439390 -0.14858306  
## [6,] 1.1230146 -0.17443724 0.03101083

# Birplot to Display the results

biplot(results, scale = 0)

 ### calculate total variance explained by each principal component

results$sdev^2 / sum(results$sdev^2)

## [1] 0.388215125 0.263967851 0.129079832 0.070275059 0.059523124 0.046945825  
## [7] 0.032925524 0.007434412 0.001633247

# Interpretation from Principal Component Analysis

From the results we can observe the following:  
  
The first principal component explains 38.8% of the total variance in the dataset.  
The second principal component explains 26.3% of the total variance in the dataset.  
The third principal component explains 12.9% of the total variance in the dataset.  
The fourth principal component explains 7.02% of the total variance in the dataset.

Thus, the first two principal components explain a majority of the total variance in the data.  
  
This is a good sign because the previous biplot projected each of the observations from the original data onto a scatterplot that only took into account the first two principal components.

### Review the summary of the model fitted

model = lm(data$price\_perunit\_area~data$house\_age+data$distance\_MRT\_station+  
 data$num\_conv\_stores+data$Bedroom.+data$Bathroom.,data) #Create the linear regression  
print(model)

##   
## Call:  
## lm(formula = data$price\_perunit\_area ~ data$house\_age + data$distance\_MRT\_station +   
## data$num\_conv\_stores + data$Bedroom. + data$Bathroom., data = data)  
##   
## Coefficients:  
## (Intercept) data$house\_age   
## 4.090132 -0.031876   
## data$distance\_MRT\_station data$num\_conv\_stores   
## -0.001845 0.143415   
## data$Bedroom. data$Bathroom.   
## 12.293575 10.704993

# Identifying Set of Significant Predictor Variables

### Summary Description

model = lm(data$price\_perunit\_area~data$house\_age+data$distance\_MRT\_station+  
 data$num\_conv\_stores+data$Bedroom.+data$Bathroom.,data) #Create the linear regression  
summary(model)

##   
## Call:  
## lm(formula = data$price\_perunit\_area ~ data$house\_age + data$distance\_MRT\_station +   
## data$num\_conv\_stores + data$Bedroom. + data$Bathroom., data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.152 -2.840 0.055 2.689 32.787   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.0901321 1.2137216 3.370 0.000824 \*\*\*  
## data$house\_age -0.0318755 0.0202258 -1.576 0.115804   
## data$distance\_MRT\_station -0.0018449 0.0002389 -7.722 8.95e-14 \*\*\*  
## data$num\_conv\_stores 0.1434147 0.0963135 1.489 0.137249   
## data$Bedroom. 12.2935752 0.5354549 22.959 < 2e-16 \*\*\*  
## data$Bathroom. 10.7049926 0.6091331 17.574 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.352 on 408 degrees of freedom  
## Multiple R-squared: 0.8989, Adjusted R-squared: 0.8977   
## F-statistic: 725.8 on 5 and 408 DF, p-value: < 2.2e-16

### Looking at some key statistics from the summary

The values we are concerned with are -  
  
1)The coefficients and significance (p-values)  
2)R-squared  
3)F statistic and its significance  
  
###  
1) The p-value for house age is 0.11580 In other words, there’s 11.58% chance  
that this predictor is not meaningful for the regression.  
  
2) The p-value for Distance MRT station is < 8.95e-14 A very small value means that   
age is probably an excellent addition to our model.  
  
3) The p-value for num\_conv\_stores is 0.137249.In other words, there’s 13.72% chance  
that this predictor is not meaningful for the regression.  
  
4) The p-value for No of Bedroom is < 2e-16. A very small value means that   
age is probably an excellent addition to our model.  
  
5) The p-value for the number of Bathroom is < 2e-16. A very small value means that   
age is probably an excellent addition to our model.  
  
  
###  
2) R - squared is 0.8989  
Meaning that 89.89% of the variance in House price prediction is explained by explainatory variable  
  
###  
3) F statistic has a very low p value (practically low)  
Meaning that the model fit is statistically significant, and the explained variance isn't purely by chance.  
  
###  
4) Transacation Year and Transaction month hav every high p value showing they are not meaningful to be Predictor Variable.  
  
  
So far we have seen how to build a linear regression model using the whole dataset. If we   
build it that way, there is no way to tell how the model will perform with new data. So   
the preferred practice is to split your dataset into a 80:20 sample (training:test), then,  
build the model on the 80% sample and then use the model thus built to predict the dependent  
variable on test data.  
  
Doing it this way, we will have the model predicted values for the 20% data (test) as well   
as the actuals (from the original dataset). By calculating accuracy measures (like min\_max  
accuracy) and error rates (MAPE or MSE), we can find out the prediction accuracy of the model.

## Value of Adjusted R^2 from model is 0.897

# Buiding Model on Training Data

### Traning Dataset

set.seed(100)   
trainingRowIndex <- sample(1:nrow(data), 0.8\*nrow(data))  
trainingData <- data[trainingRowIndex, ] # model training data  
testData <- data[-trainingRowIndex, ] # test data

set.seed(100)   
trainingRowIndex <- sample(1:nrow(data), 0.8\*nrow(data))  
trainingData <- data[trainingRowIndex, ] # model training data  
testData <- data[-trainingRowIndex, ] # test data  
lmMod <- lm(price\_perunit\_area ~ house\_age + distance\_MRT\_station +num\_conv\_stores+Bedroom.+Bathroom.,data=trainingData)   
# build the model  
distPred <- predict(lmMod, testData) # predict price per unit area

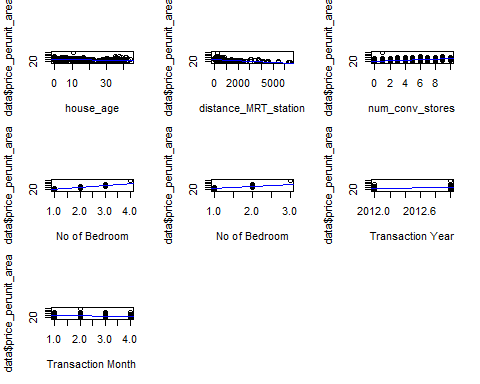
##### Interpretation

1) We have applied t-test to know whether the particular variable is correlated with   
 variable which has to be predicted or not(response variable).  
   
2)From the model summary, the model p value and predictor’s p value are less than the   
significance level, so we know we have a statistically significant model.

# Regression Plots

### Plot of Line of Best fits

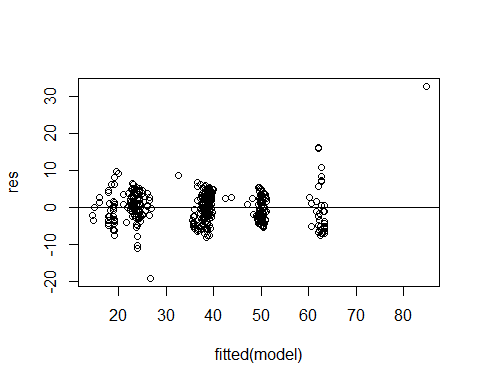
par(mfrow=c(3,3))  
plot(y=data$price\_perunit\_area, x=data$house\_age,xlab="house\_age")  
abline(lm(data$price\_perunit\_area~data$house\_age),col="blue")  
  
plot(y=data$price\_perunit\_area, x=data$distance\_MRT\_station,xlab="distance\_MRT\_station")  
abline(lm(data$price\_perunit\_area~data$distance\_MRT\_station),col="blue")  
  
plot(y=data$price\_perunit\_area, x=data$num\_conv\_stores,xlab="num\_conv\_stores")  
abline(lm(data$price\_perunit\_area~data$num\_conv\_stores),col="blue")  
  
plot(y=data$price\_perunit\_area, x=data$Bedroom.,xlab="No of Bedroom")  
abline(lm(data$price\_perunit\_area~data$Bedroom.),col="blue")  
  
plot(y=data$price\_perunit\_area, x=data$Bathroom.,xlab="No of Bedroom")  
abline(lm(data$price\_perunit\_area~data$Bathroom.),col="blue")  
  
plot(y=data$price\_perunit\_area, x=data$transaction\_year,xlab="Transaction Year")  
abline(lm(data$price\_perunit\_area~data$transaction\_year),col="blue")  
  
plot(y=data$price\_perunit\_area, x=data$transaction\_month,xlab="Transaction Month")  
abline(lm(data$price\_perunit\_area~data$transaction\_month),col="blue")



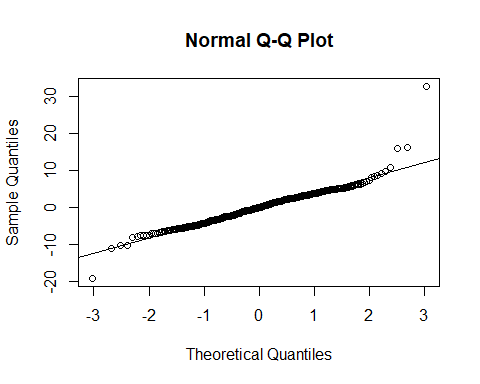
# Checking Normality of Error Terms using Q-Q Plot

### Residual Plot

res <- resid(model)  
  
# Looking for patterns in the residuals  
  
  
plot(fitted(model), res)  
abline(0,0)



qqnorm(res)  
qqline(res)



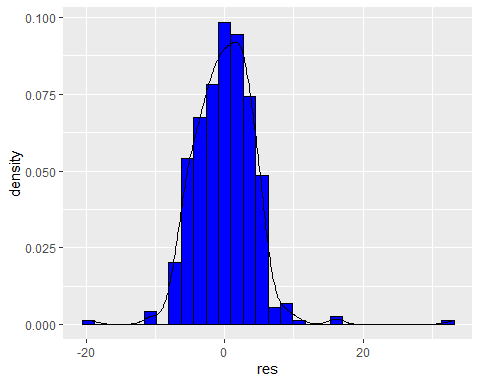
##### Interpretation

1)We have produced a Q-Q plot in plot 1, which is useful for determining if the residuals  
follow a normal distribution. If the data values in the plot fall along a roughly straight  
line at a 45-degree angle then the data is normally distributed.Here it is normally distributed

### Density plot of Residual

# We need to check if the error terms are also normally distributed (which is infact, one of  
  
# the major assumptions of linear regression), let us plot the histogram of the error terms   
  
# and see what it looks like.  
  
res <- resid(model)  
p<-ggplot(data, aes(x=res)) +   
 geom\_histogram(aes(y=..density..),color="black", fill="blue")+geom\_density()  
p

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



sd(res)

## [1] 4.325714

mean(res)

## [1] 3.621667e-17

##### Interpretation

1) From the residual plot we can see the density curve of the residual is approximately normally distributed.  
2) It is symmtric about 0 (approximately) so it can be said it is normally distributed   
approximately with mean 0 and standard deviation approximately equals to 4.325714/

# Conclusions

With several characteristics, the suggested method predicts the property price. We dvided data into 80:20 to test   
Data on remaining data(20%) to get the best model. Value of R^2 for this model was found to be descent showing model  
fits well to the data and explains greatest variablility in data.

# Insights of the analysis done on above data

1) Analyzed the data to detect problems such as heteroscedasticity, dependence & non-normality of errors   
2) Fitted Multiple Regression Model on House Price dataset using the set of significant predictor variables  
3) Addressed multicollinearity using Principal Component Analysis and achieved adjusted R2 = 0.897