Log Book Entry 2

Relationships between house characteristics and sale price

Pratik Prakash Brahmapurkar

Contents

1.0 Introduction

The dataset is initially read into a variable before being analysed for outliers and missing values. Before constructing a regression model, the data is cleaned and divided into 2 files test and train. Insights are offered with the help of summary table and assumptions. Different packages are used like caret, corrplot, lmtest and car in R. 3 regression models are created out of which 1 is simple regression model (2 variables) and the other 2 are multiple regression (6 and 11 variables). The dataset is investigated by taking into consideration 12 variables. Here SalePrice is the dependent variable. The independent variables which are used are Overall.Qual, Gr.Liv.Area, Kitchen.Qual, Garage.Cars, Garage.Area, X1st.Flr.SF, Year.built, Year.Remod.Add, Lot.Area, Bsmt.Qual and Total.Bsmt.SF.

2.0 Measures of Association

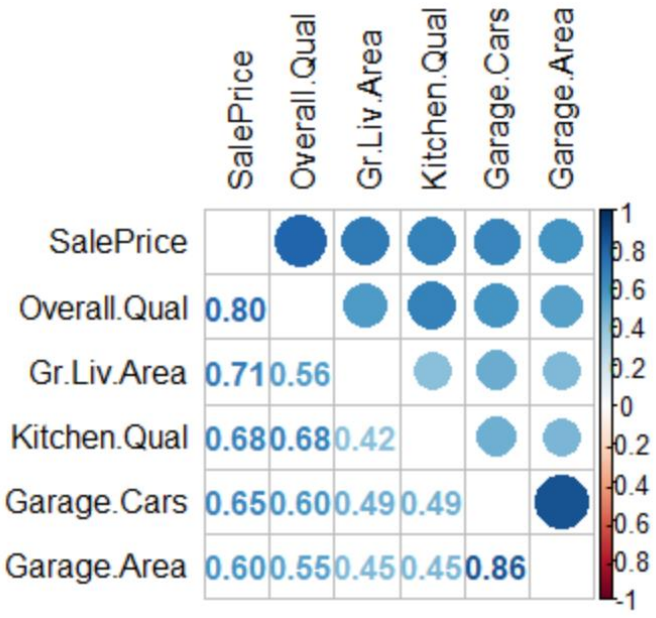

Fig 1. Visualization of Correlation Using Corrplot (Top attributes)

Overall Quality of house and Sales Price			
Correlation	**p-value <**	**t**	**df**
0.7975038	2.20E-16	71.528	2928
95 percent confidence interval: 0.7839313 0.8103144			

Table 1. Shows Pearson's product-moment correlation between overall quality and sales price.

Gound Living Area and Sales Price			
Correlation	p-value <	t	df
0.7094617	2.20E-16	54.473	2928
95 percent confidence interval: 0.6910025 0.7269962			

Table 2. Shows Pearson's product-moment correlation between living area and sales price.

Garage Cars and Sales Price			
Correlation	p-value <	t	df
0.6539314	2.20E-16	46.771	2928
95 percent confidence interval: 0.6327021 0.6741784			

Table 3. Shows Pearson's product-moment correlation between Garage Cars and sales price.

First Floor area and Sales Price			
Correlation	p-value <	t	df
0.6045819	2.20E-16	41.071	2928
95 percent confidence interval: 0.5810921 0.6270651			

Table 4. Shows Pearson's product-moment correlation between floor area and sales price.

Total basement area and Sales Price			
Correlation	p-value <	t	df
0.5707474	2.20E-16	37.611	2928
95 percent confidence interval: 0.5458166 0.5946685			

Table 5. Shows Pearson's product-moment correlation between basement area and sales price.

3.0 Regression Analysis

A. Simple Linear Regression Model

```
Call:
lm(formula = SalePrice ~ Gr.Liv.Area, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-322594  -40239   -1622   31940  385639

Coefficients:
             Estimate Std. Error t value             Pr(>|t|)
(Intercept) 18300.546   4938.921   3.705             0.000216 ***
Gr.Liv.Area   155.312      3.137  49.511 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 73730 on 2344 degrees of freedom
Multiple R-squared:  0.5112,    Adjusted R-squared:  0.511
F-statistic:  2451 on 1 and 2344 DF,  p-value: < 0.00000000000000022
```

Fig 2. Shows output of Simple Linear Regression Model

RMSE	Rsquared	MAE
78567.43	0.474681	56289.35

Table 6. Shows Root mean square error, Mean absolute error RSquared in Test data

B. **Multiple Linear Regression Model 1**

```
Call:
lm(formula = SalePrice ~ as.factor(Overall.Qual) + Gr.Liv.Area +
    as.factor(Garage.Cars) + X1st.Flr.SF + Year.Built, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-329480  -21535     129   20063  257999

Coefficients:
                             Estimate   Std. Error t value             Pr(>|t|)
(Intercept)                -1334496.834    85173.941 -15.668 < 0.0000000000000002 ***
as.factor(Overall.Qual)2      37833.737    25971.454   1.457              0.145322
as.factor(Overall.Qual)3      37973.884    23528.358   1.614              0.106671
as.factor(Overall.Qual)4      49902.855    22488.910   2.219              0.026583 *
as.factor(Overall.Qual)5      68129.960    22372.455   3.045              0.002351 **
as.factor(Overall.Qual)6      82658.059    22436.001   3.684              0.000235 ***
as.factor(Overall.Qual)7     102939.493    22551.363   4.565 0.000005263158931523 ***
as.factor(Overall.Qual)8     151623.320    22730.249   6.671 0.000000000031733847 ***
as.factor(Overall.Qual)9     231730.780    23238.820   9.972 < 0.0000000000000002 ***
as.factor(Overall.Qual)10    202081.550    24712.584   8.177 0.000000000000000472 ***
Gr.Liv.Area                      76.764        2.569  29.878 < 0.0000000000000002 ***
as.factor(Garage.Cars)2       22324.414     4440.007   5.028 0.000000533364186445 ***
as.factor(Garage.Cars)3       25077.303     4540.544   5.523 0.000000037038795025 ***
as.factor(Garage.Cars)4       59225.066     5824.054  10.169 < 0.0000000000000002 ***
as.factor(Garage.Cars)5       36511.777    13638.000   2.677              0.007476 **
as.factor(Garage.Cars)6       45724.987    44635.922   1.024              0.305753
X1st.Flr.SF                      27.758        3.006   9.235 < 0.0000000000000002 ***
Year.Built                      667.893       42.156  15.843 < 0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 44410 on 2328 degrees of freedom
Multiple R-squared:  0.8238,    Adjusted R-squared:  0.8226
F-statistic: 640.5 on 17 and 2328 DF,  p-value: < 0.0000000000000022
```

Fig 3. Shows Summary of Multiple Regression Model1

RMSE	Rsquared	MAE
47114.3	0.811762	30908.83

Table 7. Shows accuracy of the predictions of Model1

```
        Durbin-Watson test

data:  Model1
DW = 1.6948, p-value = 0.00000000000004408
alternative hypothesis: true autocorrelation is greater than 0
```

Fig 4. Shows Durbin-Watson for Multiple Regression Model1

vif(Model1)			
	GVIF	Df	GVIF^(1/(2*Df))
as.factor(Overall.Qual)	3.884991	9	1.078311
Gr.Liv.Area	1.848728	1	1.359679
as.factor(Garage.Cars)	3.013977	5	1.116642
X1st.Flr.SF	1.646658	1	1.283222
Year.Built	1.928678	1	1.388768
mean(vif(Model1))	2.369977		

Table 8. Shows assumption of no multicollnearity for Multiple Regression Model1

C. **Multiple Linear Regression Model 2**

```
Call:
lm(formula = SalePrice ~ as.factor(Overall.Qual) + Gr.Liv.Area +
    Garage.Area + X1st.Flr.SF + Year.Built + Year.Remod.Add +
    Total.Bsmt.SF + as.factor(Kitchen.Qual) + Lot.Area + as.factor(Bsmt.Qual),
    data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-337930  -18718     364   18640  270865

Coefficients:
                          Estimate    Std. Error t value             Pr(>|t|)
(Intercept)           -1436335.7441   142582.0944 -10.074 < 0.0000000000000002 ***
as.factor(Overall.Qual)2    29144.7520    24724.5288   1.179              0.23861
as.factor(Overall.Qual)3    25961.2901    22560.8061   1.151              0.24996
as.factor(Overall.Qual)4    34137.3433    21822.8505   1.564              0.11789
as.factor(Overall.Qual)5    47828.6181    21842.9114   2.190              0.02865 *
as.factor(Overall.Qual)6    58163.7457    21944.4086   2.651              0.00809 **
as.factor(Overall.Qual)7    70916.4853    22099.8203   3.209              0.00135 **
as.factor(Overall.Qual)8   113185.5749    22307.4759   5.074    0.000000420690771 ***
as.factor(Overall.Qual)9   153188.0828    23165.6220   6.613    0.000000000046673 ***
as.factor(Overall.Qual)10  136455.3886    24610.6797   5.545    0.000000032812189 ***
Gr.Liv.Area                   75.0514        2.4143  31.087 < 0.0000000000000002 ***
Garage.Area                   36.7051        5.8584   6.265    0.000000000441994 ***
X1st.Flr.SF                    3.1472        3.8555   0.816              0.41441
Year.Built                   438.7907       50.2432   8.733 < 0.0000000000000002 ***
Year.Remod.Add               285.3973       61.5626   4.636    0.000003751215953 ***
Total.Bsmt.SF                 28.5488        4.0521   7.045    0.000000000002429 ***
as.factor(Kitchen.Qual)2    -2495.8865    42318.4682  -0.059              0.95297
as.factor(Kitchen.Qual)3    -1281.6442    41988.0228  -0.031              0.97565
as.factor(Kitchen.Qual)4    10198.3212    42071.1476   0.242              0.80849
as.factor(Kitchen.Qual)5    50639.9012    42357.3288   1.196              0.23200
Lot.Area                        0.9940        0.1384   7.180    0.000000000000932 ***
as.factor(Bsmt.Qual)1       11451.3001    30338.4632   0.377              0.70587
as.factor(Bsmt.Qual)2        3156.3071     8445.5067   0.374              0.70864
as.factor(Bsmt.Qual)3        6061.8404     6859.5555   0.884              0.37695
as.factor(Bsmt.Qual)4       10331.1078     7204.3792   1.434              0.15171
as.factor(Bsmt.Qual)5       43357.2850     8462.0921   5.124    0.000000324295227 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 41770 on 2320 degrees of freedom
Multiple R-squared:  0.8447,    Adjusted R-squared:  0.8431
F-statistic: 504.9 on 25 and 2320 DF,  p-value: < 0.00000000000000022
```

Fig 5. Shows Summary of Multiple Regression Model2

RMSE	Rsquared	MAE
43512.99	0.839857	28096

Table 9. Shows accuracy of the predictions of Model2

```
        Durbin-Watson test

data:  Model2
DW = 1.6466, p-value < 0.00000000000000022
alternative hypothesis: true autocorrelation is greater than 0
```

Fig 6. Shows Durbin-Watson for Multiple Regression Model2

	GVIF	Df	GVIF^(1/(2*Df))
as.factor(Overall.Qual)	9.850125	9	1.135511
Gr.Liv.Area	1.845626	1	1.358538
Garage.Area	1.728494	1	1.314722
X1st.Flr.SF	3.062922	1	1.750121
Year.Built	3.097398	1	1.759943
Year.Remod.Add	2.241698	1	1.49723
Total.Bsmt.SF	3.276117	1	1.810005
as.factor(Kitchen.Qual)	5.06594	4	1.224849
Lot.Area	1.212688	1	1.101221
as.factor(Bsmt.Qual)	9.155959	5	1.247873
mean(vif(Model2))	2.657899		

Table 10. Shows assumption of no multicollnearity for Multiple Regression Model2

4.0 Summary of Insights

While interpreting the multiple linear regression model, the first step is analyzing what all attributes need to be taken into consideration to SalePrice in the Ames Housing data set. And the required attributes which are taken into consideration are cleaned with the outliers and missing data.

The selection of independent variables was done by checking which numerical variables have a high correlation concerning the SalePrice. The top 5 variables which have a high correlation with SalePrice are shown in Fig 1. After cleaning the Ames data set it is then split into training and testing data set that is 80-20% split.

Using a single attribute Gr.Liv.Area and the training data set simple linear regression model is executed. As shown in fig 2. the summary of the simple linear regression model, the p-value is as required which is below 0.05 but the multiple R-squared value is 0.511 (51.1%). Even though the housing price and Gr.Liv.Area correlation is 0.709(~71%) as shown in Table 2, the multiple R-squared value is still low. It can be observed from Table 6. that the comparison of the prediction with the actual value and testing data set is done which later gives us root mean square error, R-square value of testing data, and mean absolute error (MAE). The average error (RMSE) between the testing value and predicting value is 78567.43 USD, which is high as well. By observing the coefficient in Figure 2. it can be noted that the estimated standard value is positive which is 155.312 USD. It signifies that as the living area increases by 1, the cost of the house also increases by 155.312 USD.

It was also noted down that multi-collinearity attributes must not be used, For example, the correlation between attributes like Garage.Cars and Garage.Area is 0.86 as shown in Fig 1. It has a high correlation and the Garage attribute would be measured twice. So, in this case for one multiple regression model Garage.Cars was used and for other one Garage.Area was used. The categorical

attributes such as Garage.Cars, Overall.Qual, Kitchen.Qual and Basement.Qual are converted into categories.

For the multiple linear regression Model1, five attributes are considered, two of which are categorical variables. Figure 3 shows that the multiple R-Squared has risen tremendously as compared to the Simple linear regression model, which is 0.8238 (82.4 percent). Gr.Liv.Area coefficient value has similarly decreased from 155.312 to 73.88 USD. Because the p-value is less than 0.05, we may conclude that this model is likewise statistically significant. Table 9 shows that the Root mean square error for Model 1 is 47114.3, which is much smaller than the simple regression model. Table 7 shows that Durbin Watson has a value of 1.6948, which is larger than one and less than three, and the closer this value is to 2 it is better. Table 8 depicts the assumption of no multicollinearity, with a mean VIF of 2.37. If the largest VIF had been more than 10, there could have been caused for concern.

Figure 5 shows that the multiple R-squared value for Multiple Regression Model2 is 0.8447 (84.47 percent), which is higher than Model1. In Model2 10 independent variables are taken into consideration. When the estimated coefficients of Model1 and Model2 were compared, the values in Model2 decreased. The RMSE value for Model2 is similarly lower, at 43512.99. For Multiple Regression Model2, the mean assumption of no multicollinearity (VIF) is 2.65.

Model2 has the highest accuracy after confirming the Adjusted R-square for all three models. When a variable is classified as categorical, the R-square value rises. It is also possible to conclude that the more variables we include in our regression model, the higher the R-squared value and the better is the model.

Appendix 1: R Code Used

```
library(readxl)
library(dplyr)
library(caret) #to split the data
library(Hmisc) #For rcorr() function
library(corrplot)
library(lmtest)
library(car)

setwd('D:/Business Analytics/Statistics For Business/Log Book 2')
train <- read_excel('ames_train.xlsx')
test <- read_excel('ames_test.xlsx')


#Comine the train and test data into AMES
ames <- rbind(train,test)

options(scipen = 10000) #To remove 10E values

#:::::: PRE process the ames data AND Data Quality ISSUES:::::

#Change the Kitchen.Qual conditions to Number
ames$Kitchen.Qual[ames$Kitchen.Qual == 'Ex'] <- 5
ames$Kitchen.Qual[ames$Kitchen.Qual == 'Gd'] <- 4
```

```
ames$Kitchen.Qual[ames$Kitchen.Qual == 'TA'] <- 3
ames$Kitchen.Qual[ames$Kitchen.Qual == 'Fa'] <- 2
ames$Kitchen.Qual[ames$Kitchen.Qual == 'Po'] <- 1
ames$Kitchen.Qual[ames$Kitchen.Qual == 'NA'] <- 0
ames$Kitchen.Qual[is.na(ames$Kitchen.Qual)] <- 0

#Change the Bsmt.Qual conditions to Number
ames$Bsmt.Qual[ames$Bsmt.Qual == 'Ex'] <- 5
ames$Bsmt.Qual[ames$Bsmt.Qual == 'Gd'] <- 4
ames$Bsmt.Qual[ames$Bsmt.Qual == 'TA'] <- 3
ames$Bsmt.Qual[ames$Bsmt.Qual == 'Fa'] <- 2
ames$Bsmt.Qual[ames$Bsmt.Qual == 'Po'] <- 1
ames$Bsmt.Qual[ames$Bsmt.Qual == 'NA'] <- 0
ames$Bsmt.Qual[is.na(ames$Bsmt.Qual)] <- 0

#Remove the outliers which are above 4000 with the mean value
ames$Gr.Liv.Area[ames$Gr.Liv.Area > 4000] <- mean(ames$Gr.Liv.Area)

#All the NAs are converted into 0 for Lot.Frontage
ames$Lot.Frontage[is.na(ames$Lot.Frontage)] <- 0

#Neighborhood are converted into factors
ames$Neighborhood <- as.factor(ames$Neighborhood)

#Change the rating for Overall.Qual which consists of 11 to median value and factor
ames$Overall.Qual[ames$Overall.Qual == 11] <- median(ames$Overall.Qual, na.rm = TRUE)
ames$Overall.Qual <- as.factor(ames$Overall.Qual)

#Convert it into factor
ames$Overall.Cond <- as.factor(ames$Overall.Cond)

#Change Year.Built consisting 999 with median year
ames$Year.Built[ames$Year.Built == 999] <- median(ames$Year.Built)

#NA to 0
ames$Total.Bsmt.SF[is.na(ames$Total.Bsmt.SF)] <- 0
#For Total.Bsmt.SF values above 2000 are converted into mean value
ames$Total.Bsmt.SF[ames$Total.Bsmt.SF > 2000] <- mean(ames$Total.Bsmt.SF, na.rm = TRUE)

#Convert Bedroom.AbvGr into factor
ames$Bedroom.AbvGr <- as.factor(ames$Bedroom.AbvGr)

#Convert the NA in Garage.Cars to median value
ames$Garage.Cars[is.na(ames$Garage.Cars)] <- median(ames$Garage.Cars, na.rm = TRUE)
ames$Garage.Cars <- as.factor(ames$Garage.Cars)


#Garage area ABOVE 900 are changed to mean value
ames$Garage.Area[ames$Garage.Area > 900] <- mean(ames$Garage.Area , na.rm = TRUE)
ames$Garage.Area[is.na(ames$Garage.Area)] <- mean(ames$Garage.Area , na.rm = TRUE)
```

```
#TotRms.AbvGrd into factor
ames$TotRms.AbvGrd <- as.factor(ames$TotRms.AbvGrd)

#Re-summarize the Saleprice to MEAN above 780K USD
ames$SalePrice[ames$SalePrice > 780000] <- mean(ames$SalePrice, na.rm = TRUE)

#:::: RELATIONSHIPS BETWEEN DIFFERENT VARIABLES ::::::

#Change few attributes in numerical for corrplot
ames$Overall.Qual <- as.numeric(ames$Overall.Qual)
ames$Garage.Cars <- as.numeric(ames$Garage.Cars)
ames$Kitchen.Qual <- as.numeric(ames$Kitchen.Qual)
ames$Bsmt.Qual <- as.numeric(ames$Bsmt.Qual)


#Subset and Visualising Correlation Using Corrplot for multiple attributes
subdata<- ames[c("SalePrice","Overall.Qual","Gr.Liv.Area", "Kitchen.Qual", "Garage.Cars",
"Garage.Area")]
cor <- cor(subdata)
cor_sort <- as.matrix(sort(cor[,'SalePrice'], decreasing = TRUE))
corrplot.mixed(cor, tl.col="black", tl.pos="lt")

#Relationship between Overall.Qual and Sale Price with p-value and confidence interval
cor.test(x=ames$Overall.Qual, y=ames$SalePrice)

#Relationship between Living Area and Sale Price with p-value and confidence interval
cor.test(x=ames$Gr.Liv.Area, y=ames$SalePrice)

#Relationship between Garage Cars and Sale Price with p-value and confidence interval
cor.test(x=ames$Garage.Cars, y=ames$SalePrice)

#Relationship between First floor area and Sale Price with p-value and confidence interval
cor.test(x=ames$X1st.Flr.SF, y=ames$SalePrice)

#Relationship between Basement area and Sale Price with p-value and confidence interval
cor.test(x=ames$Total.Bsmt.SF, y=ames$SalePrice)


#::::: SPLIT THE AMES DATA INTO TRAINING AND TEST::::

#Delete TEST and TRAIN data first, it would set it as empty
test <- NULL
train <- NULL

#to create a partition with 80%
set.seed(123) #generate a sequence of random numbers
index <- createDataPartition(ames$SalePrice, p = 0.8, list = FALSE,)

train <- ames[index, ] #first 80% for training
test <- ames[-index, ] #bottom 20% for testing
```

```
#          ::: BUILD THE MODEL :::

#1. :::: Simple Linear Regression MODEL ::::

simple_model <- lm(SalePrice ~ Gr.Liv.Area, data = train)

#Summary of Simple Linear Regression MODEL
summary(simple_model)

#prediction using the model
simple_price_prediction <- predict(simple_model, newdata = test)

#RMSE is the difference between observed and predicted values calculated as:
sqrt(mean((simple_price_prediction - test$SalePrice)^2))
postResample(pred = simple_price_prediction, obs = test$SalePrice)

#2. ::::Multiple Linear Regression MODEL 1::::

Model1 <- lm(SalePrice ~ as.factor(Overall.Qual) + Gr.Liv.Area + as.factor(Garage.Cars) + X1st.Flr.SF +
Year.Built, data = train)

#review the model
summary(Model1)

#prediction using the model
price_prediction_1 <- predict(Model1, newdata = test)

#RMSE is the difference between observed and predicted values
#R-squared is the amount of variance in the data that is accounted for by the model
postResample(pred = simple_price_predictions, obs = test$SalePrice)

#Durbin-Watson test
dwtest(Model1)

#Cooks distance
diag1<- (train[c("SalePrice","Overall.Qual","Gr.Liv.Area","Garage.Cars","X1st.Flr.SF", "Year.Built")])
diag1$residuals <- resid(Model1)
diag1$standardized_residuals<- rstandard(Model1)
diag1$cooks_distance <- cooks.distance(Model1)
diag1$dfbeta <- dfbeta(Model1)
diag1$dffits <- dffits(Model1)
diag1$leverage <- hatvalues(Model1)
diag1$covariance_ratios <- covratio(Model1)

diag1$large_residual <- diag1$standardized_residuals > 2 | diag$standardized_residuals < -2
sum(diag1$large_residual)

#No MultiColinearity
vif(Model1)
mean(vif(Model1))
```

#3. ::::Multiple Regression Model 2::::

```
Model2 <- lm(SalePrice ~ as.factor(Overall.Qual) + Gr.Liv.Area + Garage.Area + X1st.Flr.SF +
Year.Built + Year.Remod.Add + Total.Bsmt.SF + as.factor(Kitchen.Qual) + Lot.Area +
as.factor(Bsmt.Qual) , data = train)

#review the model
summary(Model2)

#prediction using the model
price_prediction_2 <- predict(Model2, newdata = test)

#evaluate the accuracy of the predictions
#(i.e. difference between the actual sale value and the predicted sale value)
postResample(pred = price_prediction_2, obs = test$SalePrice)

#Durbin-Watson test
dwtest(Model2)

#Cooks distance
diag<- train
diag$residuals <- resid(Model2)
diag$standardized_residuals<- rstandard(Model2)
diag$cooks_distance <- cooks.distance(Model2)
diag$dfbeta <- dfbeta(Model2)
diag$dffits <- dffits(Model2)
diag$leverage <- hatvalues(Model2)
diag$covariance_ratios <- covratio(Model2)

diag$large_residual <- diag$standardized_residuals > 2 | diag$standardized_residuals < -2
sum((diag$large_residual))

#No MultiColinearity
vif(Model2)
mean(vif(Model2))
```