ALY6015 – Intermediate Analytics – Week 1 – R Assignment

Regression Diagnostics with R

Name: Hari Priya Ramamoorthy
NUID: 002324226
Date: November 7, 2024

Introduction

This report performs exploratory data analysis (EDA) and regression analysis on the Ames Housing dataset to identify variables influencing sale price. The dataset contains 2,930 observations and 82 variables, sourced from the Ames Assessor’s Office, which provides detailed housing attributes, including both structural and locational characteristics that impact residential property prices.

Data Analysis

Step-1 : Data Cleaning and EDA

The first step involved pre-processing, including standardizing column names, categorizing data where necessary, and addressing missing values to ensure data quality for accurate analysis. Given the right-skewed distributions in Figures 1, 2, and 3, a log transformation was applied to normalize the values, reducing skewness and improving model predictability.

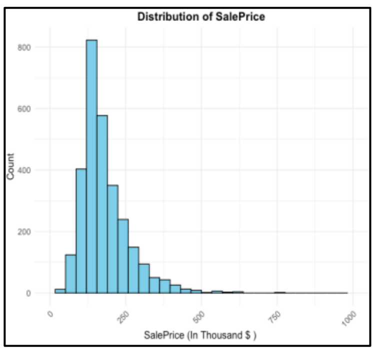


Figure 1: Histogram of sale price shows Right skewed Distribution

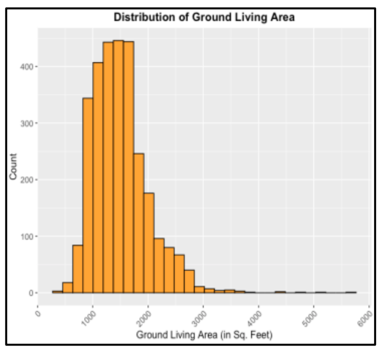


Figure 2: Histogram of Living Area shows Right skewed

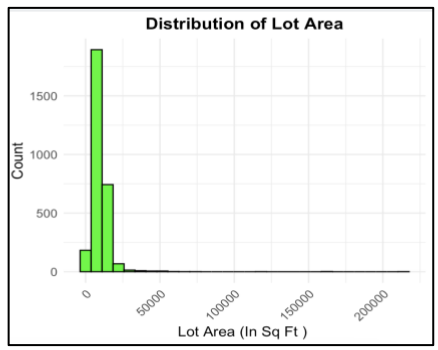


Figure 3: Histogram of Lot Area shows Right skewed Distribution

Figures 4 and 5 demonstrate that sale price varies significantly with both overall quality and neighborhood. This suggests that these factors play an essential role in determining home values.

Figure 4 : Saleprice Positively varies by Overall Quality

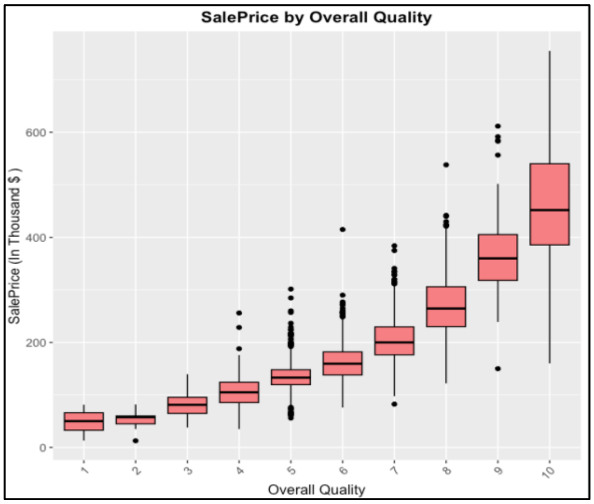
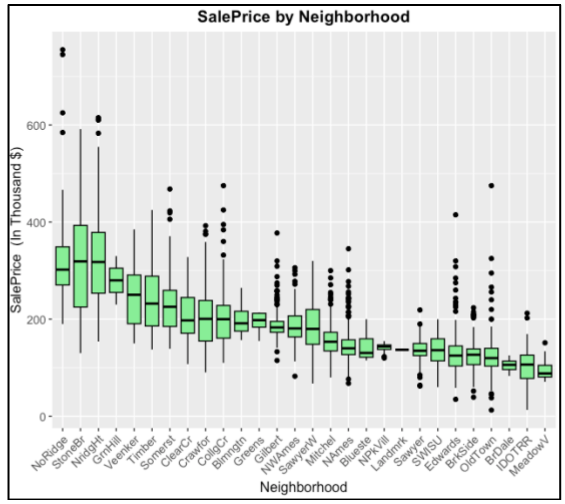


Figure 5: Saleprice Positively varies by Neighborhood

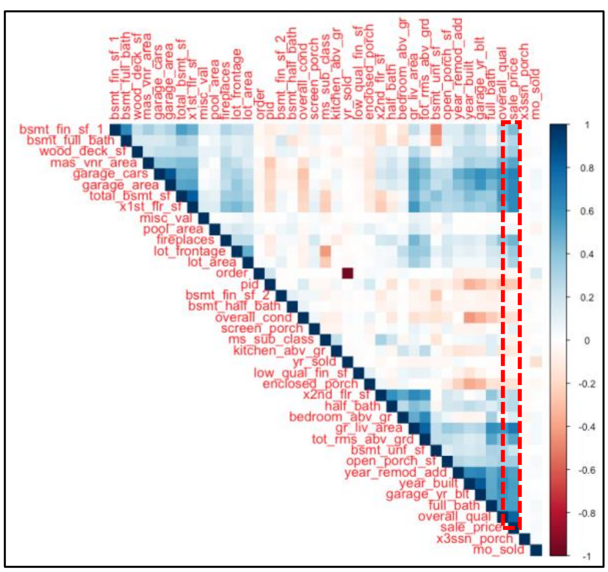


Further analysis is performed through correlation plots to investigate the relationships between sale price and other variables.

Step-3: Correlation Analysis With Corrplot() And Scatter Plot on Numerical Variables

Figure 6 presents the correlation between sale price and other numerical variables. The blue shading indicates positive correlations, while orange represents negative correlations, and white suggests no correlation. The intensity of the color indicates the strength of the correlation, with the red dotted line highlighting correlations with the sale price.

Figure 6 : Correlation Plot Numerical Variables. Red-dashed line highlights our focus for sale_price column correlations



Figures 7, 8, and 9 illustrate the correlations between selected variables and sale price, with fitted regression lines. A strong positive correlation is observed between sale price and living area size (Figure 8), while a moderate correlation is noted between the number of rooms and sale price (Figure 9). Additionally, Figure 10 shows a negligible correlation between pool area and sale price.

Figure 7 : High Correlation (Living Area Size Vs Sale price)

Figure 8 : Medium Correlation (Total Rooms Vs Sale price)

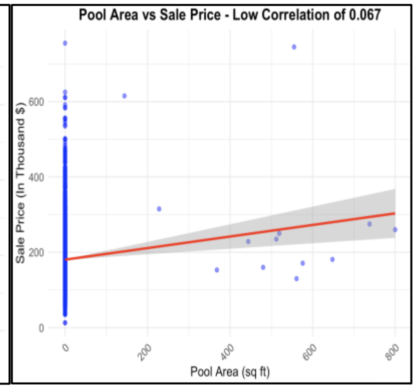


Figure 9: Low Correlation (Pool Area Vs Sale price)

Based on EDA and domain knowledge, Several key variables were identified as potentially influential predictors of sale price. The key predictors of sale price include living area above grade (gr_liv_area), overall quality (overall_qual), garage area (garage_area), lot area (lot_area), total rooms above grade (tot_rms_abv_grd), and total basement area (total_bsmt_sf), with each contributing to price based on space, quality, utility, and potential for expansion.

Together, these variables provide a comprehensive view of the property's size, quality, and functionality, all of which are key determinants of its market value.

Step-4: Linear Model with No Feature Engineering

Figure 10 shows the summary of data of Linear Model Fitted without Feature Engineering

Figure 10: Summary of Linear Model Fitted without Feature Engineering

```
> summary(model1)

Call:
lm(formula = sale_price ~ gr_liv_area + overall_qual + garage_area +
    lot_area + tot_rms_abv_grd + total_bsmt_sf + x1st_flr_sf,
    data = housing)

Residuals:
    Min      1Q  Median      3Q     Max
-551528  -19293    -470   16698  272143

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -1.043e+05  4.053e+03 -25.742  < 2e-16 ***
gr_liv_area      4.813e+01  2.839e+00  16.957  < 2e-16 ***
overall_qual     2.586e+04  6.966e+02  37.132  < 2e-16 ***
garage_area      5.483e+01  4.121e+00  13.306  < 2e-16 ***
lot_area         6.389e-01  9.432e-02   6.773 1.52e-11 ***
tot_rms_abv_grd -2.182e+03  7.547e+02  -2.891  0.00386 **
total_bsmt_sf    2.360e+01  2.783e+00   8.481  < 2e-16 ***
x1st_flr_sf      1.045e+01  3.236e+00   3.230  0.00125 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 37120 on 2922 degrees of freedom
Multiple R-squared:  0.7846,    Adjusted R-squared:  0.7841
F-statistic:  1520 on 7 and 2922 DF,  p-value: < 2.2e-16
```

Equation obtained:

sale_price = −104,300+48.13* *gr_liv_area* + 25,860**overall_qual* + 54.83**garage_area* + 0.639**lot_area* − 2,182**tot_rms_abv_grd* + 23.60* *total_bsmt_sf* + 10.45* x1st_flr_sf

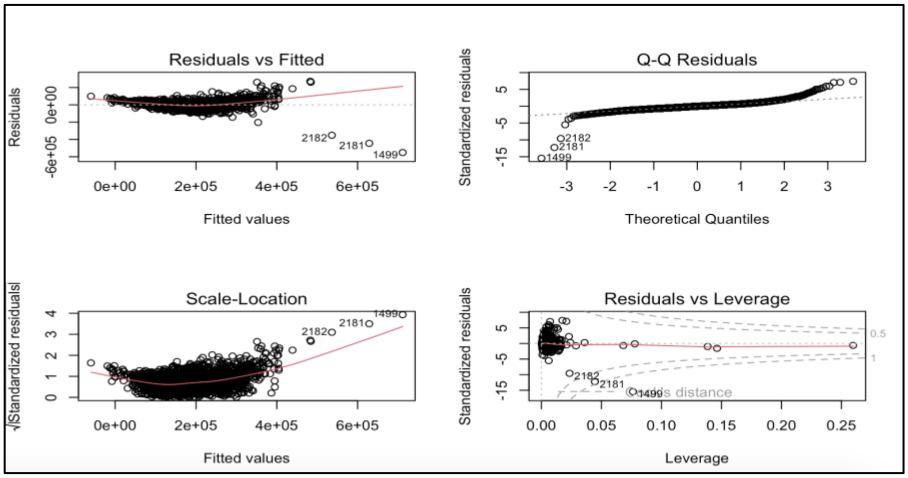
Adjusted R-Square: 0.7841 = 78.41% accuracy

Interpretation:

Figure 10 provides the summary of the linear model fitted without feature engineering. The equation obtained indicates that the intercept (-104,300) represents the estimated sale price when all predictors are zero. The coefficients represent the change in sale price for a one-unit increase in each variable. Notably, living area (gr_liv_area) and overall quality (overall_qual) have the highest impact on sale price, while other variables influence the price to a lesser extent. The adjusted R-squared of 0.7841 suggests that the model explains 78.41% of the variance in the sale price.

Step-5: Residual Plot Analysis

Figure 12: Residual Plot of Model without Feature Engineering



Insights from Figure 12:

Figure 12 presents the residual plot for the model without feature engineering, revealing several key insights:

1. **The linear assumption is violated**, as indicated by the non-flat residual vs. fitted plot.
2. The **residuals appear to be normally distributed**, as the Q-Q plot shows the residuals aligning with a straight line.
3. The **homoscedasticity test fails**, with the scale-location plot indicating non-constant variance.
4. **Multicollinearity exists**, as shown by outliers in the residuals vs. leverage plot.

Hence, the next step is to do scale, handle Multi Collinearity, Outliers to refine the model.

Step-6 : Handle Homoscedasticity with Normalizing the variables

To address the skewness identified in the scale-location plot, log transformation was applied to continuous variables like sale_price and lot_area. This normalization helped mitigate the skewness and will improve model stability.

Step-7 : Examine Outliers To Remove or Retain in the dataset

Figure 13: Outlier Test – Shows outliers with small p-values

```
> print(outlier_result)
        rstudent unadjusted p-value Bonferroni p
1499 -16.128761          3.9671e-56   1.1624e-52
2181 -12.585949          2.0588e-35   6.0323e-32
2182  -9.736805          4.5317e-22   1.3278e-18
1768   7.464695          1.0957e-13   3.2103e-10
1761   7.158503          1.0273e-12   3.0101e-09
45     7.101896          1.5391e-12   4.5096e-09
434    6.005882          2.1381e-09   6.2646e-06
1064   5.980603          2.4931e-09   7.3049e-06
433    5.538845          3.3153e-08   9.7140e-05
1183  -5.533878          3.4097e-08   9.9903e-05
```

Figure 14: Outlier rows – Not removed as they contain Information

```
> # Show the outlier rows with their values
> print(outlier_data[new_model_columns])
     overall_qual tot_rms_abv_grd garage_area lot_area_normalized
182             2               5         780            9.175335
1554            1               4         487            9.587680
1499           10              12        1418           11.064871
2181           10              15        1154           10.578725
2182           10              11         884           10.598982
1183            9               9         864           10.109363
1556            4               6         250            9.047821
373             7               6         544            9.384294
727             4               4           0            8.971956
```

The outlier test results (Figure 13) reveal the presence of outliers in the dataset, with specific rows highlighted in Figure 14. These outliers stem from significant variations in garage area, quality, and total room count in certain houses. Despite this, the inclusion of these values is relevant, as they offer insights into home sales trends across properties with both typical and high values for garage area and room count.

Step-8: Identify Multi Collinearity with VIF Test

Based on VIF Test in Figure -15, **gr_liv_area** and **x1st_flr_sf** are highly multicollinear variables with VIF of 4.37 and 3.41 respectively. Hence, these variables could be removed.

Figure 15: VIF of Model of Model without Feature Engineering

	Variance_Inflation_Factor
gr_liv_area	4.376503
x1st_flr_sf	3.418457
total_bsmt_sf	3.194123
tot_rms_abv_grd	2.995168
overall_qual	2.053302
garage_area	1.668443
lot_area	1.174085

Step-9: Build Model 2: After Feature Engineering:

After scaling, removing multicollinearity and examining outlier checks, new linear model was fitted to understand the variables influencing the normalized salesprice. The summary of model after feature engineering is shown in Figure-16.

Figure 16: Summary of Model After Feature Engineering

```
> summary(model2)

Call:
lm(formula = sale_price_normalized ~ overall_qual + tot_rms_abv_grd +
    garage_area + lot_area_normalized, data = housing)

Residuals:
     Min       1Q   Median       3Q      Max
-1.87675 -0.09760  0.01078  0.10994  0.76420

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         9.284e+00  6.700e-02  138.57   <2e-16 ***
overall_qual        1.866e-01  3.149e-03   59.24   <2e-16 ***
tot_rms_abv_grd     3.230e-02  2.566e-03   12.59   <2e-16 ***
garage_area         3.631e-04  2.082e-05   17.44   <2e-16 ***
lot_area_normalized 1.342e-01  7.650e-03   17.54   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1907 on 2925 degrees of freedom
Multiple R-squared:  0.7814,	Adjusted R-squared:  0.7811
F-statistic:  2613 on 4 and 2925 DF,  p-value: < 2.2e-16
```

Equation obtained:

Sale Price (Normalized)=9.284+(0.1866× overall_qual)+(0.0323× tot_rms_abv_grd*)+(0.0003631×garage_area)+(0.1342×lot_area_normalized)*

***Adjusted R-Square:** 0.7811 = 78.11%*

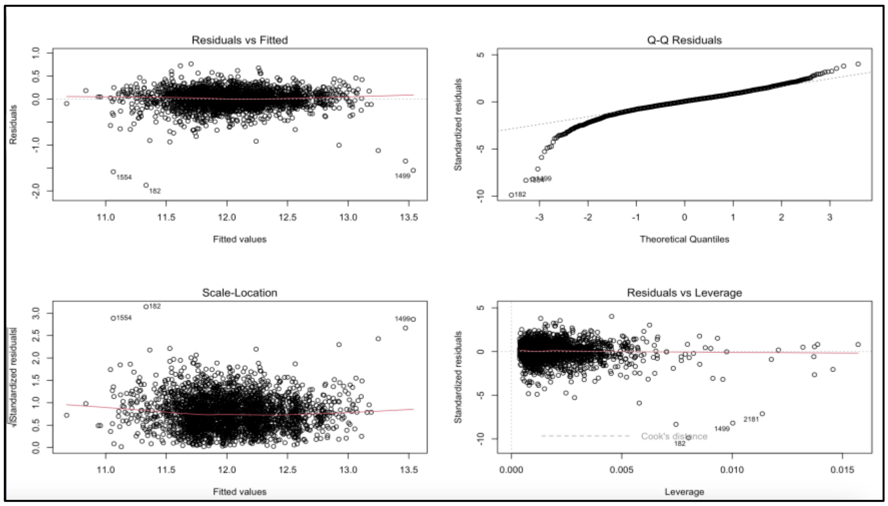
Interpretation:

The equation reveals that overall quality has the strongest positive impact on sale price, followed by lot area, total rooms above grade, and garage area. These variables contribute to price based on size, quality, and functionality. The adjusted R-squared of 0.7811 indicates that the model explains **78.11% of the variance in sale price, which is nearly identical to the previous model, but with only four significant predictors after feature engineering.**

Step-10: Residual Plot After Feature Engineering

Figure 17 shows the residual plot for the model after feature engineering. The plots indicate that linearity, no multicollinearity, homoscedasticity, and outliers have been appropriately addressed, confirming the assumptions of a well-fitting linear regression model.

Figure 17 : Residual Plot of Model After Feature Engineering



Step-11: Subset Regression with StepWise Regression

The stepwise regression model was used to identify the best set of predictor variables for **sale_price_normalized**. The results, displayed in **Figure 18**, indicate that each variable included in the final model contributes to reducing the **Akaike Information Criterion (AIC)**, which suggests that all variables have predictive value for the normalized sale price. The step model's best equation is as below:

Figure 18 : Stepwise Regression Results

Table: Stepwise Regression: Steps, AIC, and Variables Added/Removed

Step	AIC	Variable Added/Removed
1	-5258.357	
2	-8610.411	+ overall_qual
3	-9256.435	+ lot_area_normalized
4	-9552.337	+ garage_area
5	-9704.907	+ tot_rms_abv_grd

Sale Price (Normalized)=9.284+(0.1866× overall_qual)+(0.0323× tot_rms_abv_grd*)+(0.0003631×garage_area)+(0.1342×lot_area_normalized)*

***Adjusted R-Square:** 0.7811 = 78.11%*

Insights

1. **Significant Predictors**: The model indicates that **overall quality**, **lot area**, **garage area**, and **total rooms above grade** are all significant predictors of the sale price.
2. **Model Fit**: With an **R-squared value of 0.7811**, the model explains approximately 78% of the variance in sale price, providing a good fit for predicting the normalized sale price.
3. **AIC Reduction**: The stepwise approach, which selects predictors based on their contribution to minimizing AIC, has resulted in a set of variables that collectively contribute to the best model identified in Step-9 of this analysis.
4. **Model Comparison:** The model with 7 variables and the model with 4 variables predict similarly, emphasizing the importance of identifying the most effective predictors.

Conclusion

The Stepwise Regression Model identifies key predictors—*overall_qual*, *lot_area_normalized*, *garage_area*, and *tot_rms_abv_grd*—with an adjusted R-squared of 0.7811, indicating a strong fit for predicting normalized sale prices as identified by model in Step-9. After performing feature engineering and addressing issues like multicollinearity and homoscedasticity, the revised model showed only slight improvement in fit, with fewer predictors but similar R-squared values. The stepwise model, being simpler and more interpretable, provides a clearer understanding of the most influential factors in determining sale prices, while both models underscore the importance of property quality, size, and functionality.

References:

1. Stack Overflow. (2018, April 5). *Using ggpairs on a large dataset with many variables*. Stack Overflow. https://stackoverflow.com/questions/48123611/using-ggpairs-on-a-large-dataset-with-many-variables
2. Stack Overflow. (2019, May 31). *R step function not writing out complete model in result report*. Stack Overflow. https://stackoverflow.com/questions/53531517/r-step-function-not-writing-out-complete-model-in-result-report

Appendix A

R code

```
### ALY6015 - Week 1 - Regression Ananlysis
## Created By: Hari Priya Ramamoorthy
## Dataset Details:Ames Housing
## Aim of Analysis - Perform EDA and Regression Analysis on
Housing data to find the numeric variables that are influencing
sales price.


################################################################
####### Load Packages
################################################################
###
install.packages('GGally','tibble','knitr','tidyr','janitor','mo
ments','car','leaps')
library(GGally)
library(dplyr)
library(ggplot2)
library(tidyr)
library(tibble)
library(knitr)
library(corrplot)
library(janitor)

library(car)
library(moments)
library(leaps)
################################################################
####### Load Packages
################################################################
###

################################################################
####### Data cleaning - Step-1,2
################################################################
###

##Load Ameshousing Dataset
housing <- read.table(file.choose(), sep=",",header=TRUE,
stringsAsFactors = FALSE)
```

```
##Standardize column names with janitor package
housing<-janitor::clean_names(housing)
names(housing)

create_glimpse_table <- function(df) {
  tibble(
    Column_Name = names(df),
    Data_Type = sapply(df, class),
    Example_Value = sapply(df, function(x) if (length(x) > 0)
x[1] else NA)
  )
}
```

```
raw_data_glimpse<-create_glimpse_table(housing)
summary(housing)
str(housing)


### check missing values  Check
missing_values <- sapply(housing, function(x) sum(is.na(x)))
print(missing_values[missing_values > 0])

#Based on data dictionary, replace values for Columns where NA
means not present.
not_present_columns <- c('alley', 'garage_finish',
'garage_type', 'garage_qual', 'garage_cond',

'bsmt_fin_type_1','bsmt_fin_type_2','bsmt_exposure','bsmt_cond',
'bsmt_qual',
                         'fireplace_qu',  'pool_qc','fence',
'misc_feature')
# Replace NA values with 'Not Present' in the specified columns
housing[not_present_columns] <-
lapply(housing[not_present_columns], function(x)
ifelse(is.na(x), 'Not Present', x))

# for numerical columns let's replace NA values with Median
num_missing_cols <- sapply(housing, function(x) is.numeric(x) &&
any(is.na(x)))
num_missing_cols <- names(num_missing_cols[num_missing_cols ==
TRUE])

# Replace NA with the median in these numerical columns
housing[num_missing_cols] <- lapply(housing[num_missing_cols],
function(x) {
  x[is.na(x)] <- median(x, na.rm = TRUE)
```

```
  return(x)
})
### check missing values Again
missing_values <- sapply(housing, function(x) sum(is.na(x)))
# Print the result
print(missing_values[missing_values > 0])
#################################################################
####### Data cleaning - Step-1,2
#################################################################
###

#################################################################
####### EDA Visualizations - Step3
#################################################################
############


##  Histogram on Sales Price Distribution - Right Skewed
ggplot(housing, aes(x = sale_price/1000)) +
  geom_histogram(bins = 30, fill = "skyblue", color = "black") +
  labs(title = "Distribution of SalePrice", x = "SalePrice (In
Thousand $ ) ", y = "Count")+
  theme_minimal()+
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust =
1),
    axis.text.y = element_text(size = 10)
  ) +
  scale_x_continuous(limits = c(0,1000, 100))

##  Histogram on Sales Price Distribution - Right Skewed
ggplot(housing, aes(x = lot_area)) +
  geom_histogram( fill = "green", color = "black") +
  labs(title = "Distribution of Lot Area", x = "Lot Area (In Sq
Ft ) ", y = "Count")+

  theme_minimal()+
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust =
```

```
1),
    axis.text.y = element_text(size = 10)
  )


# Histogram of Living Area
ggplot(housing, aes(x = gr_liv_area)) +
  geom_histogram(bins = 30, fill = "orange", color = "black") +
  labs(title = "Distribution of Ground Living Area", x = "Ground
Living Area (in Sq. Feet)", y = "Count")+
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust =
1),
    axis.text.y = element_text(size = 10)
  )

# Boxplot comparing Saleprice variation by Overall Quality of
the house
ggplot(housing, aes(x =
reorder(factor(overall_qual),desc(sale_price/1000)), y =
sale_price/1000)) +
  geom_boxplot(fill = "lightcoral", color = "black") +
  labs(title = "SalePrice by Overall Quality", x = "Overall
Quality", y = "SalePrice (In Thousand $ )") +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust =
1),
    axis.text.y = element_text(size = 10)
  )

# Box plot for SalePrice vs. Neighborhood
ggplot(housing, aes(x =
reorder(neighborhood,desc(sale_price/1000)), y =
sale_price/1000)) +
  geom_boxplot(fill = "lightgreen", color = "black") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "SalePrice by Neighborhood", x = "Neighborhood",
y = "SalePrice  (In Thousand $)")+
    theme(
```

```
      plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
      axis.title.x = element_text(size = 12),
      axis.title.y = element_text(size = 12),
      axis.text.x = element_text(size = 10, hjust = 1),
      axis.text.y = element_text(size = 10)
    )
################################################################
####### EDA - Step-3
################################################################
############

################################################################
####### Correlarion And Scatter Plot Analysis For Numeric
Variables -Step-4,5,6 #####################################
numeric_vars <- sapply(housing, is.numeric)
corr_matrix <- cor(housing[, numeric_vars], use =
"complete.obs")

# Plot correlation matrix showing only the upper half
#### Insights from Plot
# High Correlation : gr_liv_area,overallquality; Low/No
Correlation:pool_area,year ; Medium correlation-
tot_rmsabv_grd,full_bath
corrplot(corr_matrix, method = "color", order = "hclust", tl.cex
= 1.1, type = "upper")
title(main = "Correlation Heat Map")

# scatter plot-1 Between Highly Correlated Ground living Area vs
SalePrice
ggplot(housing, aes(x = gr_liv_area, y = sale_price/1000)) +
  geom_point(color = "blue", alpha = 0.5) +
  labs(title = "Ground Living Area vs Sale Price - High
Correlation of 0.714",
       x = "Ground Living Area (sq ft)",
       y = "Sale Price (In Thousand $)") +
  geom_smooth(method = "lm", col = "red", size = 1) +  # Linear
regression line
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust =
1),
    axis.text.y = element_text(size = 10)
```

```
  )

# scatter plot-2 Between Least Correlated pool_area vs SalePrice
ggplot(housing, aes(x = pool_area, y = sale_price/1000)) +
  geom_point(color = "blue", alpha = 0.5) +
  labs(title = "Pool Area vs Sale Price - Low Correlation of
0.067",
       x = "Pool Area (sq ft)",
       y = "Sale Price (In Thousand $)") +
  theme_minimal() +
  geom_smooth(method = "lm", col = "red", size = 1) +  # Linear
regression line

  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust =
1),
    axis.text.y = element_text(size = 10)
  )

ggplot(housing, aes(x = tot_rms_abv_grd, y = sale_price/1000)) +
  geom_point(color = "blue", alpha = 0.5) +
  labs(title = "Total Rooms vs Sale Price - Medium Correlation
of 0.523",
       x = "Total Rooms ",
       y = "Sale Price (In Thousand $)") +
  theme_minimal() +
  geom_smooth(method = "lm", col = "red", size = 1) +  # Linear
regression line

  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face =
"bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust =
1),
    axis.text.y = element_text(size = 10)
  )

# Scatter plot matrix for numeric variables with high,low,
medium correlation
# Reference: https://stackoverflow.com/questions/68638725/how-
to-address-overplotting-in-ggallyggpairs
```

```
ggpairs(housing[, c("sale_price", "gr_liv_area",
"overall_qual","tot_rms_abv_grd", "pool_area",'x1st_flr_sf')])
################################################################
####### Correlarion And Scatter Plot Analysis For Numeric
Variables -Step-4,5,6 ####################################


################################################################
####### LR model 1 - No Feature Engineering
################################################################
#######
########## Select Highly correlated variables to model the sales
price
model_columns <- c("gr_liv_area", "overall_qual", "garage_area",
'lot_area','tot_rms_abv_grd','total_bsmt_sf','x1st_flr_sf')
model1 <- lm(sale_price ~ gr_liv_area + overall_qual +
garage_area + lot_area + tot_rms_abv_grd + total_bsmt_sf+
x1st_flr_sf, data=housing)
summary(model1)
### Residual Plot
par(mfrow=c(2,2))
plot(model1)

## Normalize continuous variables sale_price lot_area to remove
skewness,
housing$sale_price_normalized <- log(housing$sale_price)
housing$lot_area_normalized =log(housing$lot_area)


######### Multi-collinearity Check: correlation Matrix and VIF
Test , Outlier Check : car package #########
##### Insights from correlation Matrix and VIF Test
## 1. Remove gr_liv_area - correlated with Overall_quality,1st
floor square feet
#corr_matrix_model_data <- cor(housing[, model_columns], use =
"complete.obs")
#corrplot(corr_matrix_model_data, method = "color", order =
"hclust", tl.cex = 1.1, title = "Correlation Heatmap")
######### Identify Multi Collinearity with VIF Test
vif1<-data.frame("Variance_Inflation_Factor"=vif(model1))

########## Outlier Test on Model 2 #################

outlier_result <- outlierTest(model1)
print(outlier_result)
# Let's get the indices of the outlier observations based on the
row numbers from the outlierTest result
```

```
outlier_indices <- c(182, 1554, 1499, 2181, 2182, 1183, 1556,
373, 727)
# Extract the outliers from the dataset
outlier_data <- housing[outlier_indices, ]
# Show the outlier rows with their values
print(outlier_data[model_columns])
#### Insights 1. Outlier Present - Because of the room size and
garage, which has meaningful data, so no action taken
########## Outlier Test #################

#################################################################
####### LR model 1 - No Feature Engineering
#################################################################
#######


#################################################################
####### LR model 2 - After Feature Engineering
#################################################################
#######

new_model_columns = c("overall_qual", "tot_rms_abv_grd",
"garage_area", "lot_area_normalized")
model2 <- lm(sale_price_normalized ~ overall_qual +
tot_rms_abv_grd + garage_area + lot_area_normalized ,
data=housing)
summary(model2)



### Residual Plot
par(mfrow=c(2,2))
plot(model2)

## Component +Residual Plot for each predictor
### Residual Plot
par(mfrow=c(1,1))
crPlots(model2)
#################################################################
####### LR model 2 - After Feature Engineering
#################################################################
#######

################################ subset Regression
#################################################################

nullModel <- lm(sale_price_normalized ~ 1,data=housing )
```

```
fullModel <- lm(sale_price_normalized ~ overall_qual +
tot_rms_abv_grd + garage_area + lot_area_normalized
,data=housing )

# step-wise regression in both directions
step_model <-
step(nullModel,scope=list(lower=nullModel,upper=fullModel),direc
tion = "both")

# Extract the anova table from the stepwise model, which
contains the step information.
step_info <- step_model$anova

# Create a data frame to store step information for plotting
stepwise_df <- data.frame(
  Step = 1:nrow(step_info),
  AIC = step_info$AIC,
  Variable = step_info$Step
)

# Use kable to display the table in a report-friendly format
kable(stepwise_df,
      caption = "Stepwise Regression: Steps, AIC, and Variables
Added/Removed",
      col.names = c("Step", "AIC", "Variable Added/Removed"),
      format = "markdown")

# Print the table for reporting
print(stepwise_df)
summary(step_model)
#################################################################
stepwise Regression
#################################################################
```