MODULE 1 - R PRACTICE ASSIGNMENTREPORT: REGRESSION DIAGNOSTICS WITH R

Ayush Anand

Northeastern University: College of Professional Studies

ALY 6015: Intermediate Analytics

Professor – Valeriy Shevchenko

7th November 2024

Table of Contents:

- About the Dataset
- Data Exploration and Handling Missing Values
- Data Cleaning
- oxplot of Sale price
- Data Visualization with respect to the Sale Price
- Correlation Analysis Using Heat Map
- Scatterplots of Total Basement Area VS variable with Highest / Lowest / 0.5correlations with Sale Price
- Regression Model
- Calculating MRSE, AIC, BIC
- Checking for multicollinearity in the Model
- Checking for Outliers in the Model
- After Removing Outliers from the Model
- Does Removing Outliers Improve the Model?
- Subsets regression method to identify the "best" model
- Comparison between Model 2 and Subsets regression Model
- Report Summary
- References
- Appendix

About the Dataset:

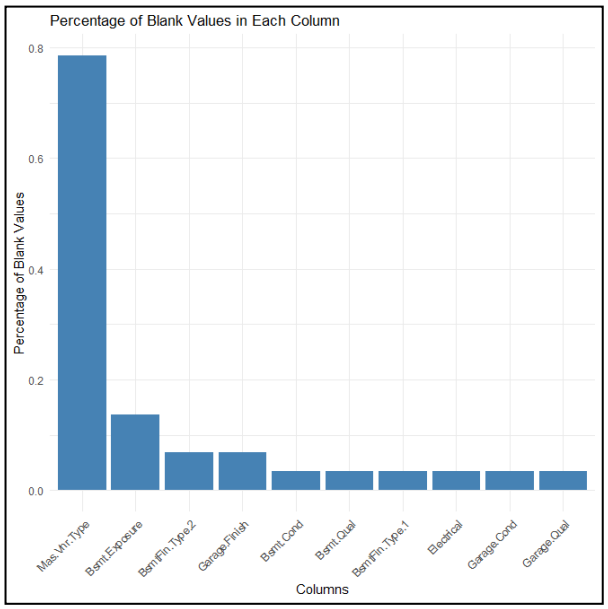
Data Source: Collected from the Ames Assessor's Office in Iowa
Time Frame: 2006-2010
Dataset Size: 2,930 records and 82 variables
Variable Types: Includes 23 nominal, 23 ordinal, 14 discrete, 20 continuous variables, plus two identifiers for each observation.

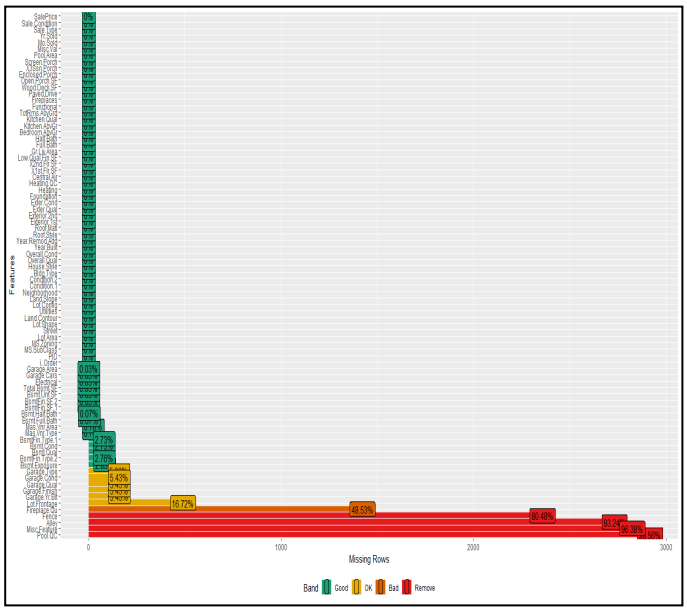
Dataset Overview: The Ames Housing dataset offers an in-depth view of real estate data from Ames, Iowa. It's gained popularity as a comprehensive alternative to the Boston Housing dataset due to its larger size, a broader range of features, and the added advantage of minimal missing data.

Project Goal: The objective is to develop a model that predicts home sale prices in Ames based on various factors like property size, age, location, and condition. This process will involve data cleaning, exploratory data analysis, handling missing values, data transformation, and model building. The final model can assist stakeholders in estimating property values, supporting informed real estate decisions.

Data Exploration and Handling Missing Values:

We started by loading and exploring the dataset with libraries like readr and dplyr. Viewing the data structure (str(ames_data)) helped us understand the types and dimensions of each variable. Additionally, summary statistics (summary(ames_data)) gave us a clearer picture of variable ranges, central tendencies, and variability..

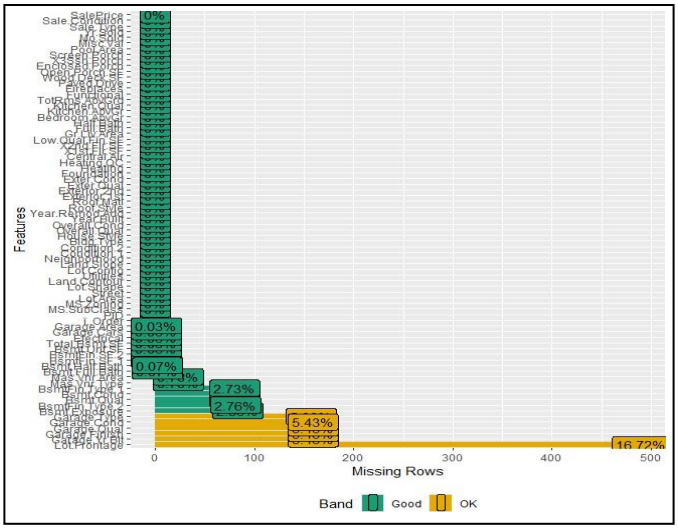


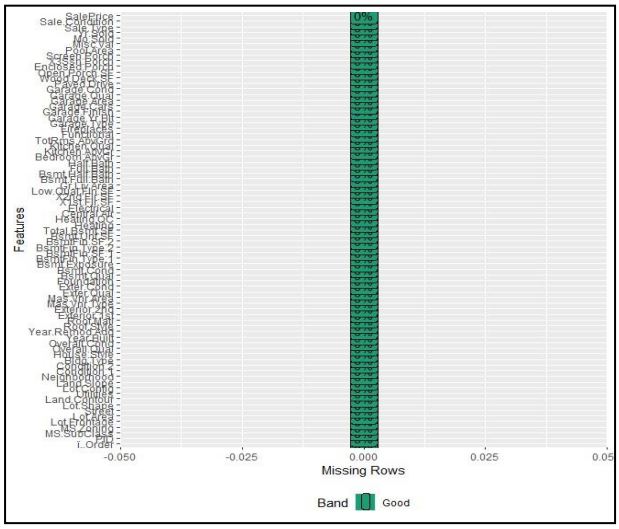


The dataset was analyzed for blank values, which were visualized with a bar chart to identify the columns with higher missing values. These blanks were then converted to NA values for consistent data handling.

Data Cleaning:

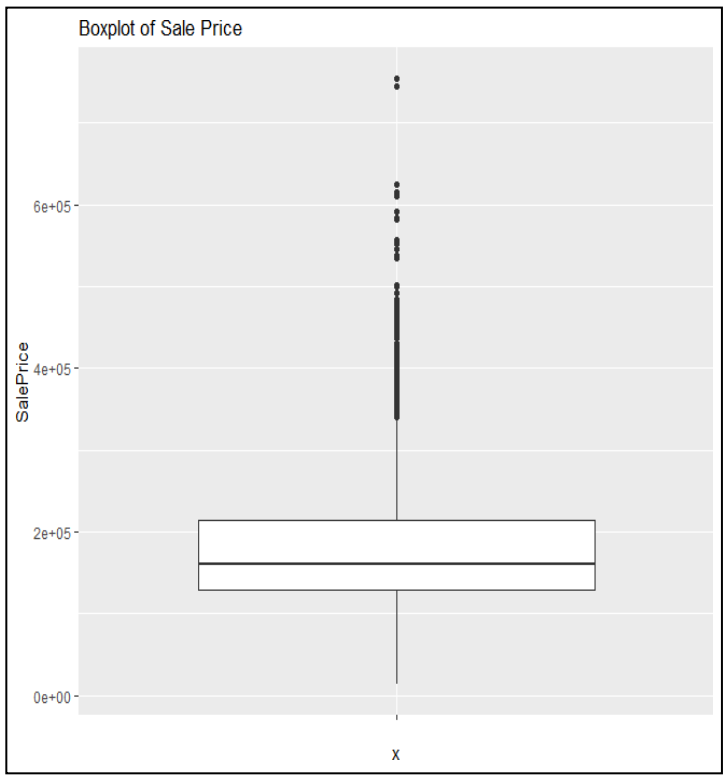
We investigated the extent of missing data within the Ames Housing dataset. Columns with **missing value percentages exceeding 40%** were identified as potentially unreliable due to the high proportion of missing information. These identified **columns were then removed** from the data frame to provide better analysis.





Finally**, all the NA values of Numerical columns were replaced by the median value** of that column and a custom function get_mode replaced missing values in **categorical columns with the mode value** of that column.

Boxplot of Sale price:



Key Insights: The boxplot of the sale price indicates that the data is positively skewed, with a long tail towards higher values. This suggests that there are some houses with significantly higher sale prices compared to the majority.

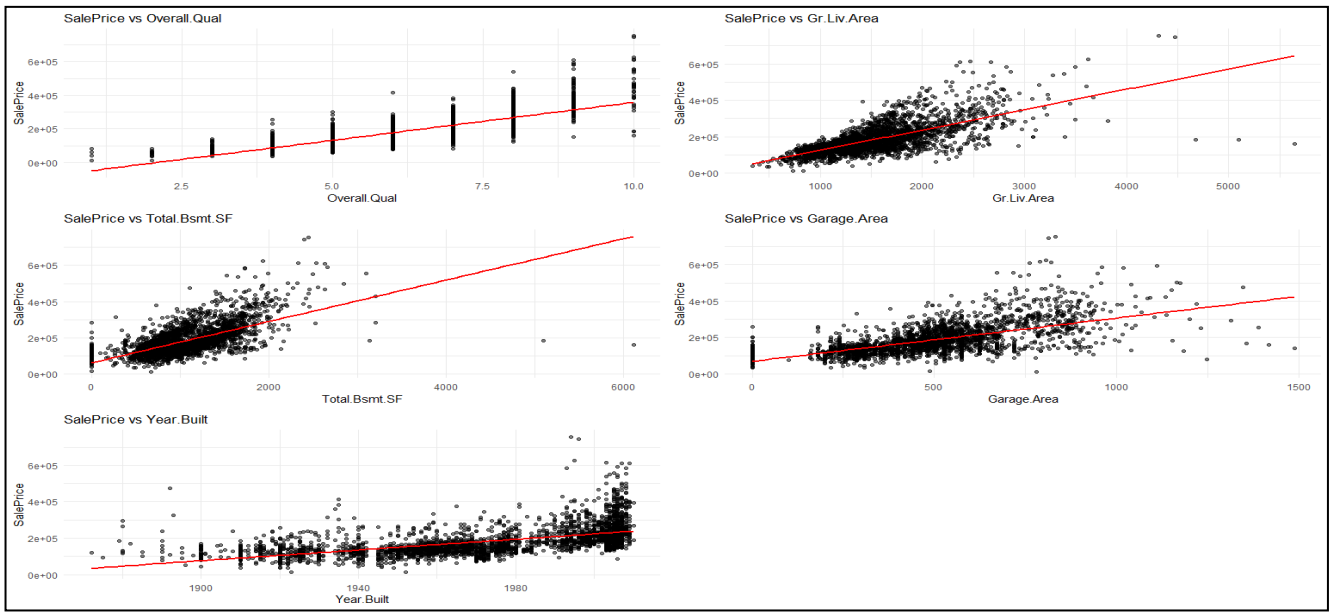
Median: The median sale price is approximately $200,000.

Quartiles: The first quartile (Q1) is around $150,000, and the third quartile (Q3) is around $250,000.

Interquartile Range (IQR): The IQR, the difference between Q3 and Q1, is about $100,000.

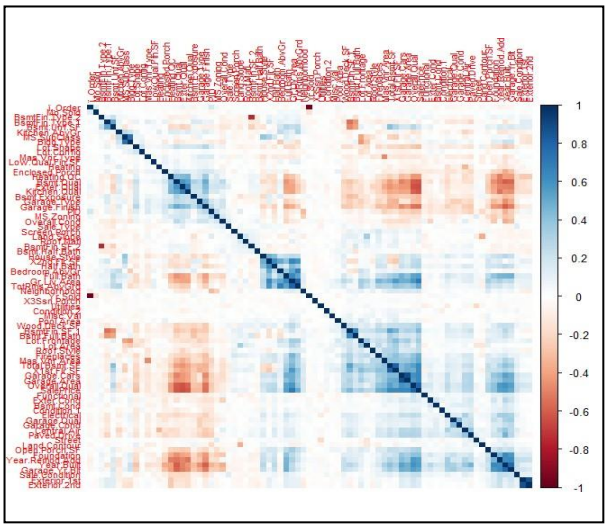
Outliers: There are several outliers above the upper whisker, indicating houses with exceptionally high sale prices.

Data Visualization with respect to the Sale Price:



- **Sale Price vs Overall Quality (Overall Quality):** There's a strong positive relationship, indicating thathouses with higher overall quality ratings tend to have higher sale prices
- **Sale Price vs Above Ground Living Area (Gr Liv Area):** There's a clear positive relationship, suggesting thatlarger houses generally sell for higher prices.
- **Sale Price vs Total Basement Square Feet (Total.Bsmt.SF):** There's a positive relationship, but it's not as strong as with the above-ground living area. Houses with larger basements tend to have higher sale prices,but there's more variability.
- **Sale Price vs Garage Area (Garage.Area):** There's a positive relationship, indicating that houses with largergarages tend to have higher sale prices, but the relationship isn't as strong as with overall quality or living area.
- **Sale Price vs Year Built (Year.Built):** There's a slight positive trend, suggesting that newer houses tend tosell for somewhat higher prices, but the relationship isn't as strong as with the other variables.

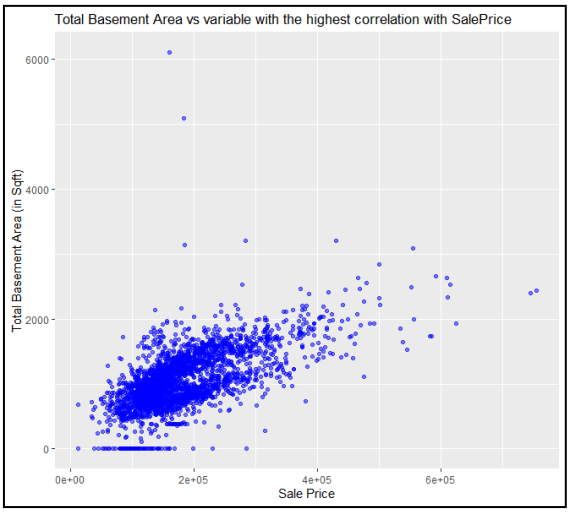
Correlation Analysis Using HeatMap:



The plot displays a symmetric matrix in which each cell reflects the correlation between two variables. The color scale spans from dark blue (indicating a strong negative correlation) to white (no correlation) and dark red (strong positive correlation). Variables are organized hierarchically, clustering those with similar correlation patterns together.

Strong Positive Correlations: The variables most strongly correlated with **'SalePrice'** include **'Overall Qual'**, **'Gr Liv Area'**, **'Total Bsmt SF'**, **'Garage Area'**, and **'1st Flr SF'**. This indicates that factors such as the quality of the house, its total living area, and the sizes of specific areas (basement, garage, and first floor) are key predictors of sale price.

Scatterplots of Total Basement Area Vs variable with Highest / Lowest / 0.5 correlations with SalePrice:



Total Basement Area Vs Variable with Highest Correlation to Sale Price:

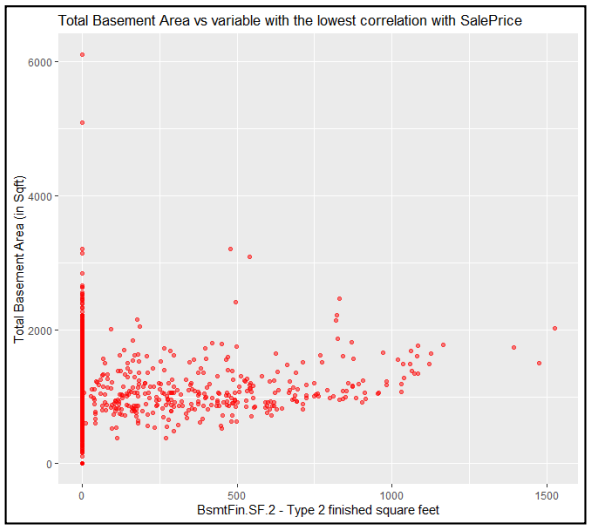
Variable with Highest Correlation with SalePrice – SalePrice

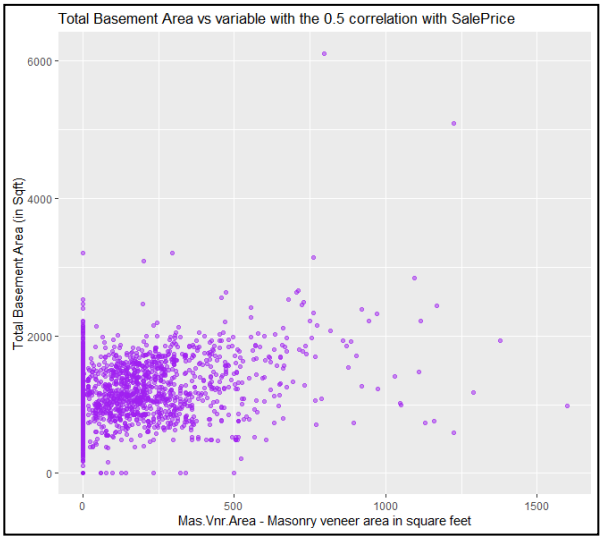
This scatter plot reveals a strong positive relationship between Total Basement Area and SalePrice, the variable most highly correlated with it. As **SalePrice** increases, Total Basement Area tends to increase as well, suggesting that houses with larger basements are often of higher quality or offer more living space, factors that contribute to higher sale prices.

Total Basement Area Vs Variable with Lowest Correlation to Sale Price:

Variable with Lowest Correlation with SalePrice – BsmtFin.SF.2

This scatter plot shows almost no relationship between Total Basement Area and a variable that has little to no correlation with sale price. The points are likely scattered randomly without any clear trend, indicating that this variable does not significantly affect either basement size or sale price.





Total Basement Area Vs Variable with 0.5 Correlation to Sale Price:

Variable with 0.5 Correlation with **SalePrice** – **Mas.Vnr.Area**

This scatter plot reveals a moderate positive relationship between Total Basement Area and a variable with a mid-level correlation to SalePrice (around 0.5). While the trend is weaker than in the first plot, it remains evident. For instance, houses with more recent construction dates or larger garage spaces may also feature larger basements, which contributes moderately to higher sale prices.

Regression Model:

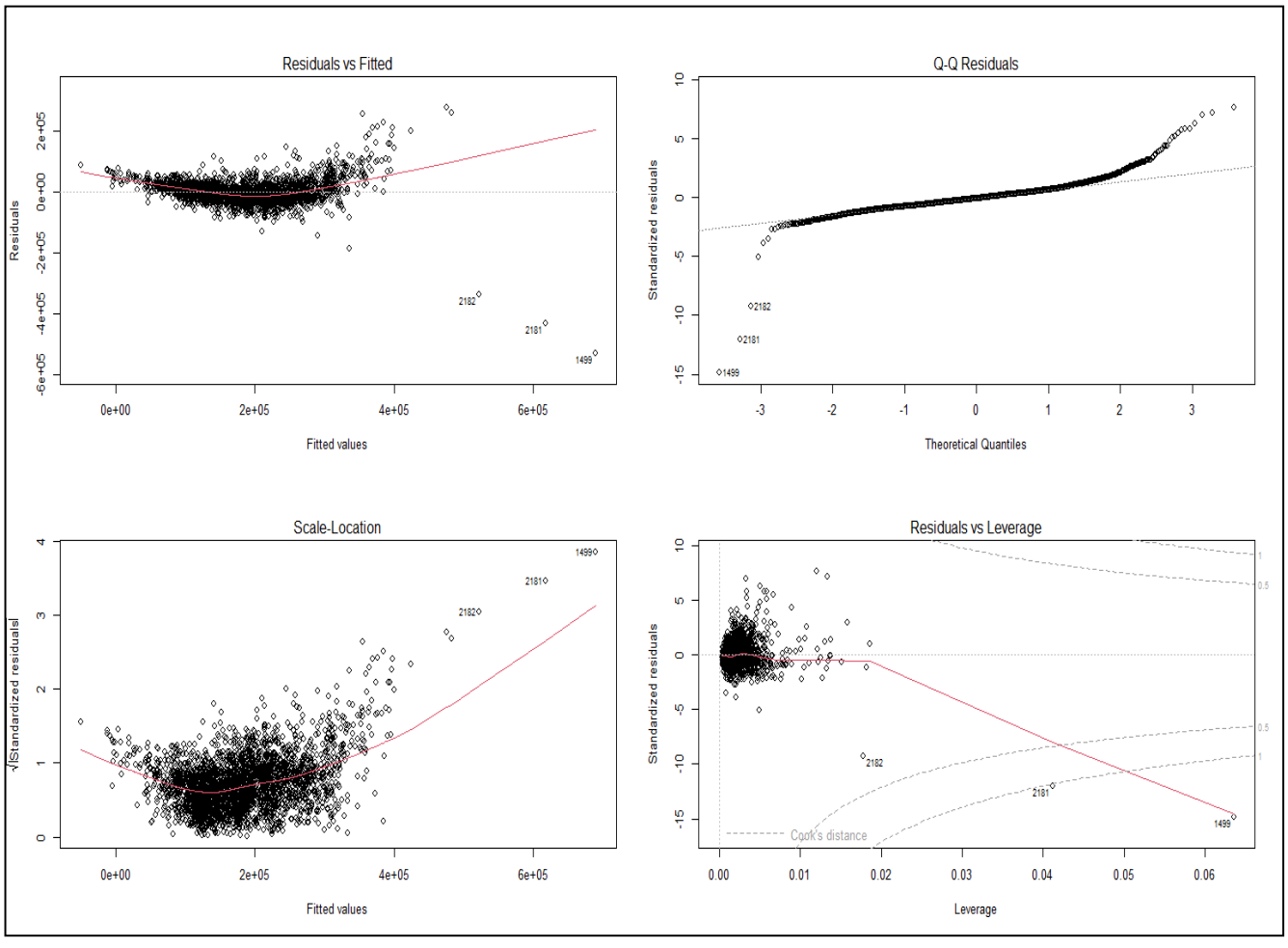
Model 1:

Outcome Variable: Sale Price
Predictor Variables: Overall.Qual, Gr.Liv.Area, Year.Built, Total.Bsmt.SF, Garage.Area
Equation of the model:

$$SalePrice = \beta_0 + \beta_1(Overall.Qual) + \beta_2(Gr.Liv.Area) + \beta_3(Year.Built) + \beta_4(Total.Bsmt.SF) + \beta_5(Garage.Area) + \varepsilon$$

Where,
Intercept (β_0): -7.361e+05
Overall.Qual coefficients (β_1): 2.109e+04
Gr.Liv.Area coefficients (β_2): 5.145e+01
Year.Built coefficients (β_3): 3.328e+02
Total.Bsmt.SF coefficients (β_4): 3.067e+01
Garage.Area coefficients (β_5): 4.834e+01

R^2	Adjusted R^2	Residual Std. Error	F Statistic
0.7884	**0.788**	**36780 (DF = 2924)**	**2178 on 5 and 2924 DF**



1. **Residuals vs Fitted (Top Left)**:
    - **Interpretation**: On the x-axis, the fitted values (predicted SalePrice) are shown against the residuals (errors) on the y-axis.
    - **Insight**: Ideally, residuals should appear randomly scattered around zero without any discernible pattern. In this plot, however, a slight curve is visible, hinting at some non-linearity in the model. This suggests that the model may not fully capture the relationship between the predictors and SalePrice, indicating possible issues with model fit.

2. **Normal Q-Q Plot (Top Right)**:
    - **Interpretation**: This plot checks if the residuals follow a normal distribution. The points should ideally fall along the 45-degree line.
    - **Insight**: Residuals should ideally be randomly distributed around zero, showing no specific pattern. However, in this plot, a slight curve suggests the presence of non-linearity in the model. This indicates that the model might not be fully capturing the relationship between the predictors and SalePrice, pointing to potential issues with model fit.

3. **Scale-Location Plot (Bottom Left)**:
    - **Interpretation**: The square root of the standardized residuals is plotted against the fitted values in order to determine if the residuals are homoscedastic (constant variance).
    - **Insight**: The red line shows an upward trend, indicating heteroscedasticity (uneven variance). As fitted values increase, the spread of residuals widens, suggesting that error variability grows with higher SalePrice predictions. This pattern violates a core assumption of linear regression, which assumes constant variance of errors across all levels of the predicted values.

4. **Residuals vs Leverage Plot (Bottom Right)**:
    - **Interpretation**: This plot helps identify influential data points. Points with high leverage or high standardized residuals can have a large impact on the regression model.
    - **Insight**: Certain points, including observations 2182 and 1499, exhibit high leverage and appear to be influential outliers. These points fall outside Cook's distance lines, suggesting they may disproportionately impact the model's predictions. Further investigation of these points is advisable to determine whether they should be removed or adjusted.

Calculating MRSE, AIC, BIC:

Mean Residual Standard Error (MRSE): Mean Residual Standard Error (MRSE) is essential for assessing a model's accuracy and reliability. A lower MRSE indicates a better fit, as it means the model's predictions are closer to the actual values, enhancing the model's precision.
The MRSE value of the model is **679.5209**

AIC (Akaike Information Criterion): The criterion penalizes models with additional parameters, though less harshly than the Bayesian Information Criterion (BIC). Its goal is to strike a balance between model fit and complexity, making it particularly useful when aiming to find a model that performs well in predicting future observations.
The AIC value of the model is **69927.79**

BIC (Bayesian Information Criterion): This criterion penalizes models with additional parameters more strongly than the Akaike Information Criterion (AIC), generally favoring simpler models, even if they don't fit the data as closely as more complex ones. It's often used when the aim is to identify the "true" model that most accurately represents the data generation process.
The BIC value of the model is **69969.67**

Checking for multicollinearity in the Model:

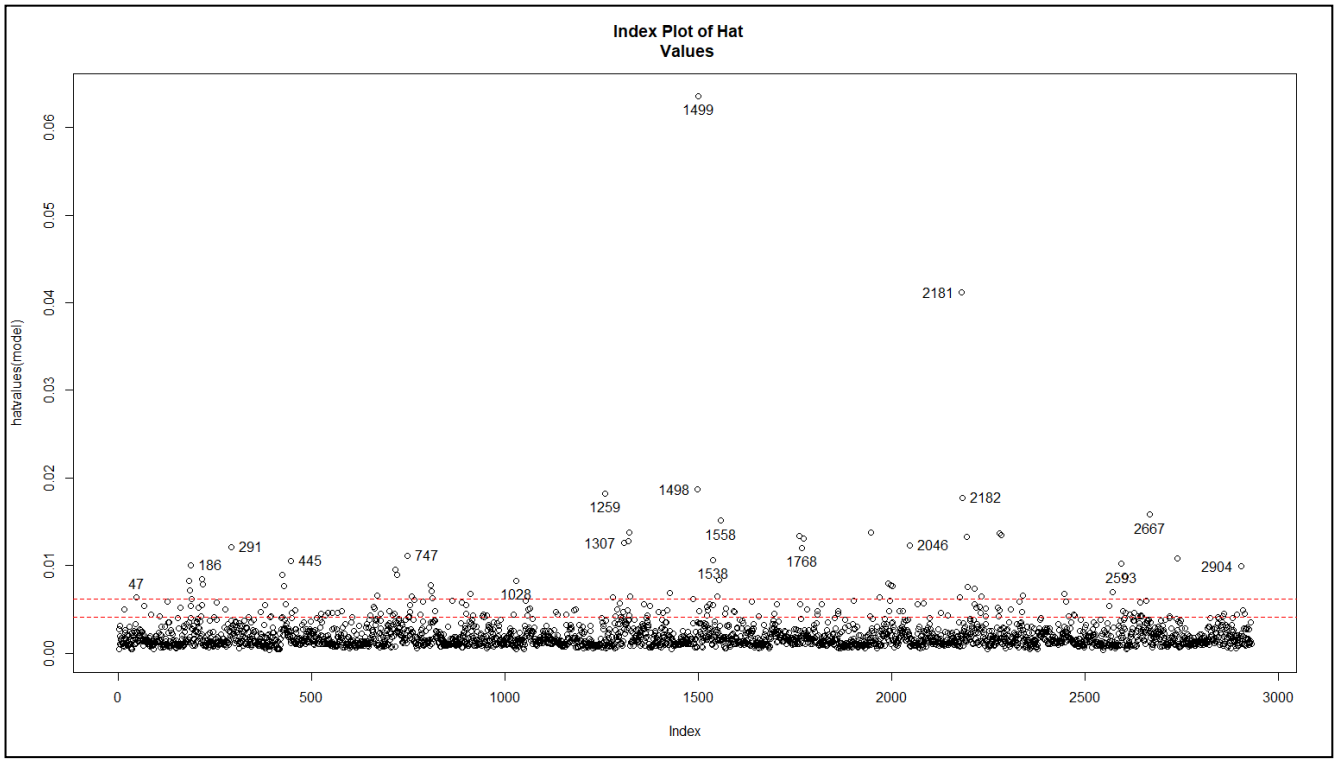
A measure of how much multicollinearity inflates a regression coefficient's variance is called the Variance Inflation Factor (VIF). The variable is significantly linked with other variables in the model if its VIF is high.

Overall.Qual	Gr.Liv.Area	Year.Built	Total.Bsmt.SF	Garage.Area
2.440698	1.691470	1.726083	1.574292	1.732841

Values of the predictors < 10: Not concerning
Values of the predictors > 10: high level of multicollinearity

Since **all the values of the predictors are lying between 1 and 2** hence we can say there is **No multicollinearityin our Model.**

Checking for Outliers in the Model:



Hat Values: Each observation's leverage on the model is indicated by the hat values on the y-axis of this graphic. The fit of the model is more significantly impacted by observations with higher hat values. The index of observations in the dataset is shown on the x-axis.

High-Leverage Points: Several observations fall above the red dashed lines, marking them as high-leverage points. Key high-leverage observations include 1499, 2181, 2182, 1768, and 1761, among others. These points exert considerable influence on the regression model and warrant further investigation to assess their impact.

Potential Influence on Model: High-leverage points can heavily impact the regression coefficients, which may result in a biased or unstable model. It's important to examine these observations closely to determine if they are valid data points or if they represent outliers or errors that might need to be removed or handled differently.

After Removing Outliers from the Model:

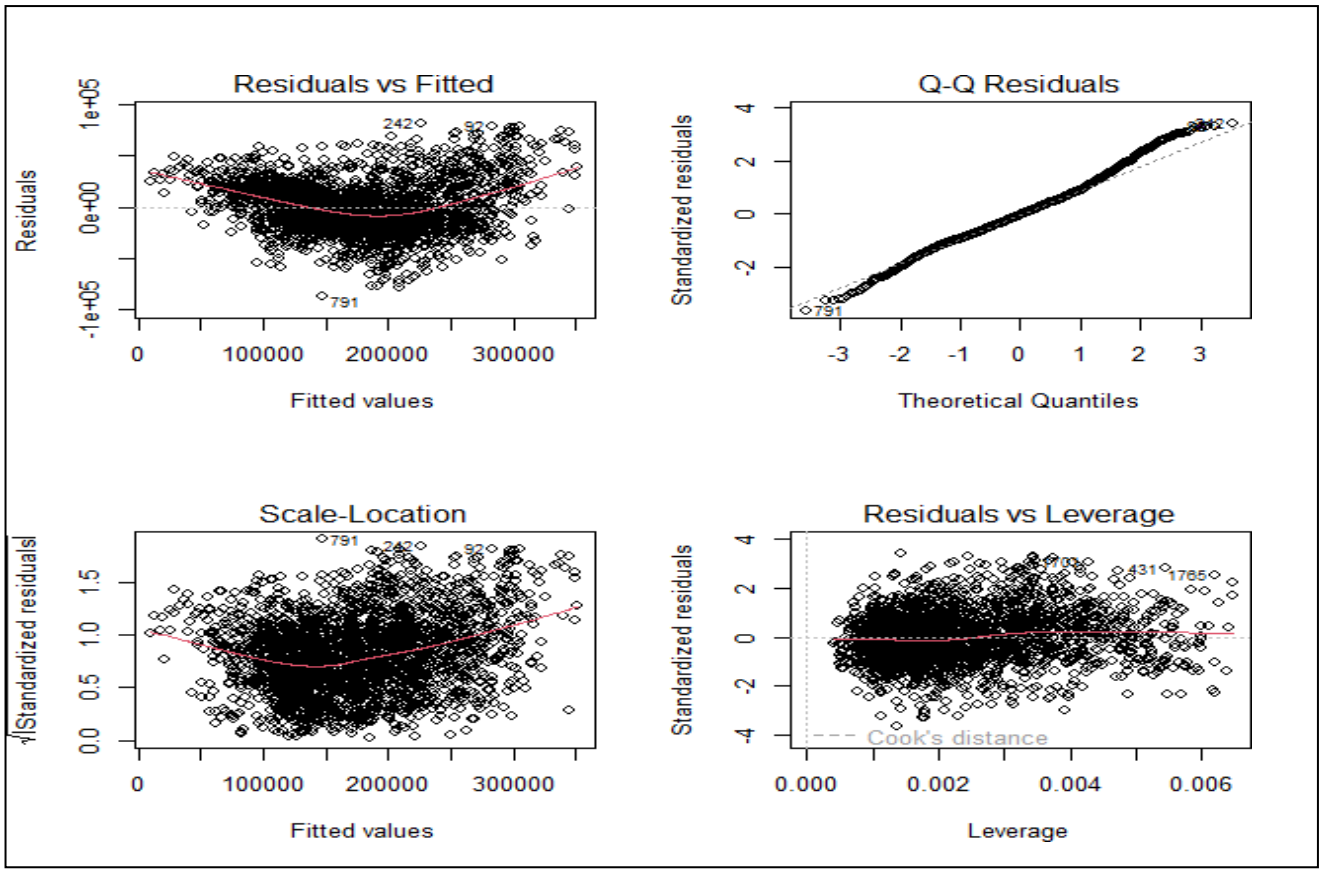
Model 2:

Outcome Variable: Sale Price
Predictor Variables: Overall.Qual, Gr.Liv.Area, Year.Built, Total.Bsmt.SF, Garage.Area
Equation of the model:

$$SalePrice = \beta_0 + \beta_1(Overall.Qual) + \beta_2(Gr.Liv.Area) + \beta_3(Year.Built) + \beta_4(Total.Bsmt.SF) + \beta_5(Garage.Area) + \varepsilon$$

Where,
Intercept (β_0): -6.660e+05
Overall.Qual coefficients (β_1): 1.633e+04
Gr.Liv.Area coefficients (β_2): 5.525e+01
Year.Built coefficients (β_3): 3.064e+02
Total.Bsmt.SF coefficients (β_4): 3.485e+01
Garage.Area coefficients (β_5): 4.633e+01

R^2	Adjusted R^2	Residual Std. Error	F Statistic
0.8604	**0.8602**	**23900 (DF = 2633)**	**3246 on 5 and 2633 DF**



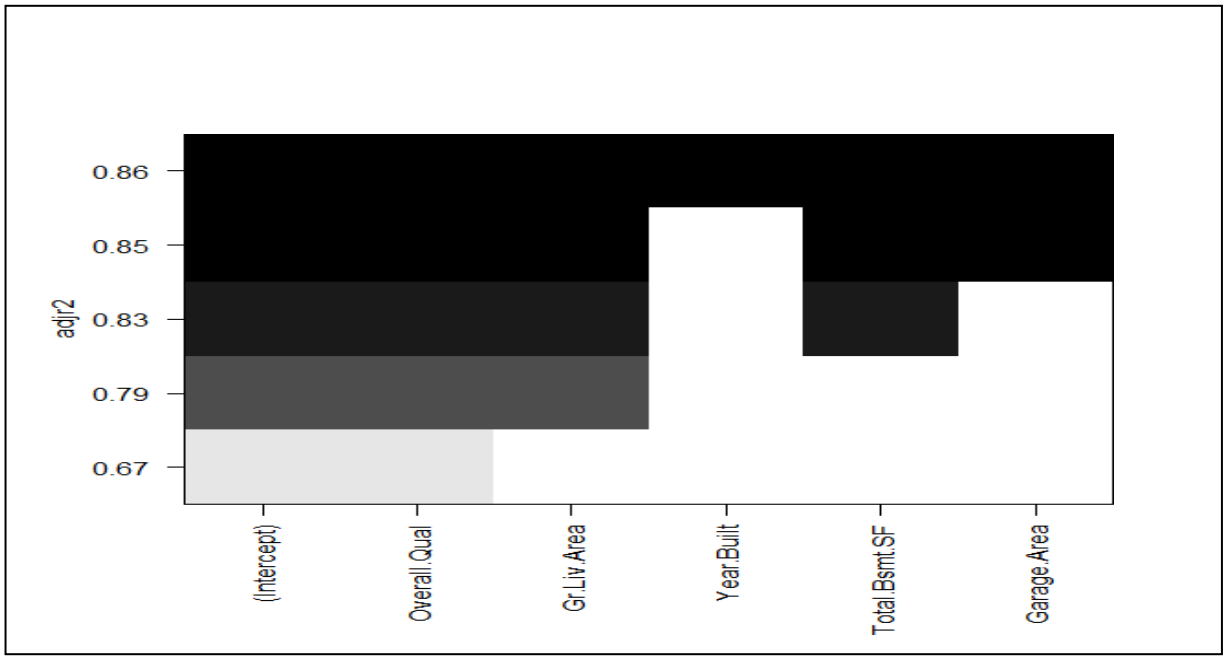
Does Removing Outliers Improve the Model?

Yes, after removing the outliers it improved the model.
Lower Residual Standard Error (MRSE):
Model 2 shows a notably lower Mean Residual Standard Error (MRSE) of 441.499 compared to Model 1's 679.5209. This suggests that Model 2's predictions are, on average, closer to the actual values, providing a better fit to the data.

Higher Adjusted R-squared:
Model 1's adjusted R-squared is 0.788, whereas Model 2's is 0.8602. After controlling for the number of predictors, this indicates that Model 2 accounts for a higher percentage of the variance in the dependent variable.

Lower AIC and BIC:
Model 2 has lower AIC and BIC values. These information criteria penalize models with more parameters.Lower values indicate a better balance between model fit and complexity.

Subsets regression method to identify the "best" model:



The regression model's adjusted R-squared values for different subsets of variables are shown in the plot. The model's fit to the data after accounting for the number of predictors is shown by adjusted R-squared. A model that better fits the data, balancing predictive power and model complexity, is indicated by a higher adjusted R-squared.

Best Model: The model achieving the highest adjusted R-squared includes all variables except "Year.Built." Among the subsets evaluated, this model offers the best fit to the data.

Variable Importance:

- "**Overall.Qual**" and "**Gr.Liv.Area**" appear to be the most influential variables, contributing substantially to the adjusted R-squared even in simpler models.
- "**Total.Bsmt.SF**" and "**Garage.Area**" also positively affect the model's fit, though their impact is somewhat less pronounced compared to the top two variables.

Comparison between Model 2 and Subsets regression Model:

Both Model 2 and the subset regression model demonstrate strong predictive performance, with each achieving an adjusted R-squared of 0.86. This similarity suggests that the extra variables included in the subset regression model may not add significant incremental value to the prediction accuracy.

Report Summary:

In this analysis, we built a linear regression model to predict house prices based on five continuous variables: Overall Quality, Above Ground Living Area, Year Built, Total Basement Area, and Garage Area. Diagnostic plots were used to evaluate the model's performance, identifying potential issues with high-leverage points (such as observations 1499, 2181, and 2182), as well as signs of non-linearity and heteroscedasticity. Scatter plots between SalePrice and the predictors showed strong positive correlations, particularly for Overall Quality and Living Area. To refine the model, we examined and removed outliers and influential points, as they could potentially distort model accuracy.

References:

GeeksforGeeks. (n.d.). *Detecting multicollinearity with VIF in Python*. Retrieved from https://www.geeksforgeeks.org/detecting-multicollinearity-with-vif-python/

Statistics By Jim. (n.d.). *Multicollinearity in regression analysis: Problems, detection, and solutions*. Retrieved from https://statisticsbyjim.com/regression/multicollinearity-in-regression-analysis/

Appendix:

```
# ================================================================================
# Loading Necessary Libraries
# ================================================================================

library(readr)
library(dplyr)
library(tidyr)
library(ggplot2)
library(scales)
library(corrplot)
library(DataExplorer)
library(gridExtra)
library(car)
library(leaps)

# ================================================================================
# Step 1: Reading the Dataset
# ================================================================================

# Load the Ames housing dataset
ames_data <- read.csv("C:/Users/ayush/Downloads/Intermediate_Analytics/AmesHousing.csv")
head(ames_data)

# ================================================================================
# Step 2: Exploratory Data Analysis
# ================================================================================

# Display column names
column_names <- names(ames_data)
print(column_names)

# Get the structure and summary of the dataset
str(ames_data)
summary(ames_data)

# ================================================================================
# Step 3: Displaying Columns with Blank ("") Values in Plot
# ================================================================================

# Calculate the percentage of blank ("") values in each column
blank_percentage <- sapply(ames_data, function(x) sum(x == "", na.rm = TRUE)) / nrow(ames_data) * 100
blank_df <- data.frame(Column = names(blank_percentage), Blank_Percentage = blank_percentage)
blank_df <- subset(blank_df, Blank_Percentage > 0)

# Plot the percentage of blank values for each column
ggplot(blank_df, aes(x = reorder(Column, -Blank_Percentage), y = Blank_Percentage)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  theme_minimal() +
  labs(title = "Percentage of Blank Values in Each Column",
       x = "Columns",
       y = "Percentage of Blank Values") +
```

```
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

# Convert blank ("") values to NA in the entire dataframe
ames_data[ames_data == ""] <- NA

# Checking for NA values in each column
colSums(is.na(ames_data))

# View missing data patterns
plot_missing(ames_data)

# ================================================================================
# Step 4: Removing Columns that Have Missing Values > 40%
# ================================================================================

# Calculate the percentage of missing values for each column
missing_percent <- sapply(ames_data, function(x) sum(is.na(x)) / length(x) * 100)

# Print the names of columns that were dropped
dropping_columns <- names(missing_percent[missing_percent > 40])
print("Dropping columns:")
print(dropping_columns)

# Filter out columns where the missing percentage is greater than 40%
ames_data <- ames_data[, missing_percent <= 40]

# View missing data patterns
plot_missing(ames_data)

# ================================================================================
# Step 5: Handling Missing Values (Categorical - Mode, Numerical - Median)
# ================================================================================

# Function to calculate mode
get_mode <- function(x) {
  unique_values <- unique(x)
  unique_values[which.max(tabulate(match(x, unique_values)))]
}

# Handling missing values in ames_data
ames_data <- ames_data %>%
  # Replace NA in numeric columns with median
  mutate(across(where(is.numeric), ~ ifelse(is.na(.), median(., na.rm = TRUE), .))) %>%
  # Convert character columns to factor to ensure compatibility with get_mode
  mutate(across(where(is.character), as.factor)) %>%
  # Replace NA in factor (categorical) columns with mode
  mutate(across(where(is.factor), ~ ifelse(is.na(.), get_mode(na.omit(.)), .)))

# View missing data patterns
plot_missing(ames_data)

# ================================================================================
# Step 6: Visualizations
```

```
# ============================================================================

# Visualizations
# View histograms of variables
hist(ames_data$SalePrice, main = 'Sale Price')

# View boxplots of variables
ggplot(ames_data, aes(x = "", y = SalePrice)) +
  geom_boxplot() +
  labs(title = "Boxplot of Sale Price")

# Select variables to plot against SalePrice
variables <- c("Overall.Qual", "Gr.Liv.Area", "Total.Bsmt.SF", "Garage.Area", "Year.Built")

# Create a list to store individual plots
plots <- list()

# Create scatter plots for each variable
for (var in variables) {
  p <- ggplot(ames_data, aes_string(x = var, y = "SalePrice")) +
    geom_point(alpha = 0.5) +
    geom_smooth(method = "lm", se = FALSE, color = "red") +
    labs(title = paste("SalePrice vs", var),
         x = var,
         y = "SalePrice") +
    theme_minimal()
  plots[[var]] <- p
}

# Arrange plots in a grid
library(gridExtra)
grid.arrange(grobs = plots, ncol = 2)

# ============================================================================
# Step 7: Correlation Analysis
# ============================================================================

# Producing a correlation matrix for numeric values
numeric_ames_data <- ames_data[, sapply(ames_data, is.numeric)]
cor_matrix <- cor(numeric_ames_data, use = "complete.obs")

# Plotting the correlation matrix
corrplot(cor_matrix, method = "color", order = "hclust", tl.cex = 0.7)

# ============================================================================
# Step 8: Making Scatter Plots for Total Basement Area vs Sale Price
# ============================================================================

# Find highest correlation with SalePrice
highest_cor <- names(which.max(abs(cor_matrix["SalePrice", ])))
print(highest_cor)
ggplot(ames_data, aes_string(x = highest_cor, y = "Total.Bsmt.SF")) +
  geom_point(color = 'blue', alpha = 0.5) +
```

```
  labs(title = "Total Basement Area vs variable with the highest correlation with SalePrice",
       x = "Sale Price",
       y = "Total Basement Area (in Sqft)")

# Find lowest correlation with SalePrice
lowest_cor <- names(which.min(abs(cor_matrix["SalePrice", ])))
print(lowest_cor)
ggplot(ames_data, aes_string(x = lowest_cor, y = "Total.Bsmt.SF")) +
  geom_point(color = 'red', alpha = 0.5) +
  labs(title = "Total Basement Area vs variable with the lowest correlation with SalePrice",
       x = "BsmtFin.SF.2 - Type 2 finished square feet",
       y = "Total Basement Area (in Sqft)")

# Find correlation closest to 0.5 with SalePrice
mid_cor <- names(which.min(abs(abs(cor_matrix["SalePrice", ]) - 0.5)))
print(mid_cor)
ggplot(ames_data, aes_string(x = mid_cor, y = "Total.Bsmt.SF")) +
  geom_point(color = 'purple', alpha = 0.5) +
  labs(title = "Total Basement Area vs variable with the 0.5 correlation with SalePrice",
       x = "Mas.Vnr.Area - Masonry veneer area in square feet",
       y = "Total Basement Area (in Sqft)")

# ============================================================================
# Step 9: Fitting a Regression Model
# ============================================================================

# Fitting a regression model using 5 continuous variables
model1 <- lm(SalePrice ~ Overall.Qual + Gr.Liv.Area + Year.Built + Total.Bsmt.SF + Garage.Area, data =
       ames_data)
summary(model1)

# Plotting the regression model
par(mfrow = c(2, 2))
plot(model1)

# Calculating the Mean Residual Standard Error (MRSE) for the model
rse <- summary(model1)$sigma
n1 <- nrow(ames_data)
mrse <- rse / sqrt(n1)
mrse

# Calculating AIC and BIC of the model:
AIC(model1)
BIC(model1)

# ============================================================================
# Step 10: Checking for Multicollinearity and Outliers
# ============================================================================

# Checking for multicollinearity
library(car)
vif(model1)
```

```
# Checking for outliers using Bonferroni test
outliers <- outlierTest(model1)
print(outliers)

# Plotting high-leverage observations
hat.plot <- function(model1) {
  p <- length(coefficients(model1))
  n <- length(fitted(model1))

  plot(hatvalues(model1), main = "Index Plot of Hat Values")
  abline(h = c(2, 3) * p / n, col = "red", lty = 2)
  identify(1:n, hatvalues(model1), names(hatvalues(model1)))
}
hat.plot(model1)

# Influential observations
cutoff <- 4 / (nrow(ames_data) - length(model1$coefficients) - 2)
plot(model1, which = 4, cook.levels = cutoff)
abline(h = cutoff, lty = 2, col = "red")

# ==============================================================================
# Step 11: Using All Subsets Regression to Identify the "Best" Model
# ==============================================================================

# Using all subsets regression to identify the "best" model
subsets <- regsubsets(SalePrice ~ Overall.Qual + Gr.Liv.Area + Year.Built + Total.Bsmt.SF + Garage.Area,
                      data = ames_data, nvmax = 5)
summary(subsets)

# Plotting subsets regression:
plot(subsets, scale = "adjr2")
```