Housing Data Analysis

Sarah Theriot

2024-10-18

Housing Data Analysis

Step 1: Set Up Packages

Step 2: Load and Explore Data

```
# Load and Explore the Data

# Read the Dataset
housing_data <- read.csv("C:/Users/sarah/Desktop/MSDS/Statistics for Data Science/Week 8/week-6-housing

# View the first few rows of the data
head(housing_data)
```

```
##   Sale.Date Sale.Price sale_reason sale_instrument sale_warning sitetype
## 1  1/3/2006     698000           1               3                    R1
## 2  1/3/2006     649990           1               3                    R1
## 3  1/3/2006     572500           1               3                    R1
## 4  1/3/2006     420000           1               3                    R1
## 5  1/3/2006     369900           1               3           15       R1
## 6  1/3/2006     184667           1              15        18 51       R1
##             addr_full  zip5 ctyname postalctyn       lon      lat building_grade
## 1  17021 NE 113TH CT 98052 REDMOND    REDMOND -122.1124 47.70139              9
## 2  11927 178TH PL NE 98052 REDMOND    REDMOND -122.1022 47.70731              9
## 3 13315 174TH AVE NE 98052            REDMOND -122.1085 47.71986              8
## 4  3303 178TH AVE NE 98052 REDMOND    REDMOND -122.1037 47.63914              8
## 5  16126 NE 108TH CT 98052 REDMOND    REDMOND -122.1242 47.69748              7
## 6   8101 229TH DR NE 98053            REDMOND -122.0341 47.67545              7
##   square_feet_total_living bedrooms bath_full_count bath_half_count
## 1                     2810        4               2               1
## 2                     2880        4               2               0
## 3                     2770        4               1               1
## 4                     1620        3               1               0
## 5                     1440        3               1               0
## 6                     4160        4               2               1
##   bath_3qtr_count year_built year_renovated current_zoning sq_ft_lot prop_type
## 1               0       2003              0             R4      6635         R
## 2               1       2006              0             R4      5570         R
## 3               1       1987              0             R6      8444         R
```

```
## 4               1       1968               0            R4      9600           R
## 5               1       1980               0            R6      7526           R
## 6               1       2005               0         URPSO      7280           R
##   present_use
## 1           2
## 2           2
## 3           2
## 4           2
## 5           2
## 6           2
```

```r
# Get a summary of the data to understand its structure
summary(housing_data)
```

```
##   Sale.Date           Sale.Price       sale_reason    sale_instrument
##  Length:12865       Min.   :    698   Min.   : 0.00   Min.   : 0.000
##  Class :character   1st Qu.: 460000   1st Qu.: 1.00   1st Qu.: 3.000
##  Mode  :character   Median : 593000   Median : 1.00   Median : 3.000
##                     Mean   : 660738   Mean   : 1.55   Mean   : 3.678
##                     3rd Qu.: 750000   3rd Qu.: 1.00   3rd Qu.: 3.000
##                     Max.   :4400000   Max.   :19.00   Max.   :27.000
##  sale_warning         sitetype           addr_full              zip5
##  Length:12865       Length:12865       Length:12865       Min.   :98052
##  Class :character   Class :character   Class :character   1st Qu.:98052
##  Mode  :character   Mode  :character   Mode  :character   Median :98052
##                                                           Mean   :98053
##                                                           3rd Qu.:98053
##                                                           Max.   :98074
##    ctyname           postalctyn             lon               lat
##  Length:12865       Length:12865       Min.   :-122.2   Min.   :47.46
##  Class :character   Class :character   1st Qu.:-122.1   1st Qu.:47.67
##  Mode  :character   Mode  :character   Median :-122.1   Median :47.69
##                                        Mean   :-122.1   Mean   :47.68
##                                        3rd Qu.:-122.0   3rd Qu.:47.70
##                                        Max.   :-121.9   Max.   :47.73
##  building_grade  square_feet_total_living    bedrooms      bath_full_count
##  Min.   : 2.00   Min.   :  240            Min.   : 0.000   Min.   : 0.000
##  1st Qu.: 8.00   1st Qu.: 1820            1st Qu.: 3.000   1st Qu.: 1.000
##  Median : 8.00   Median : 2420            Median : 4.000   Median : 2.000
##  Mean   : 8.24   Mean   : 2540            Mean   : 3.479   Mean   : 1.798
##  3rd Qu.: 9.00   3rd Qu.: 3110            3rd Qu.: 4.000   3rd Qu.: 2.000
##  Max.   :13.00   Max.   :13540            Max.   :11.000   Max.   :23.000
##  bath_half_count  bath_3qtr_count   year_built   year_renovated
##  Min.   :0.0000   Min.   :0.000   Min.   :1900   Min.   :   0.00
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:1979   1st Qu.:   0.00
##  Median :1.0000   Median :0.000   Median :1998   Median :   0.00
##  Mean   :0.6134   Mean   :0.494   Mean   :1993   Mean   :  26.24
##  3rd Qu.:1.0000   3rd Qu.:1.000   3rd Qu.:2007   3rd Qu.:   0.00
##  Max.   :8.0000   Max.   :8.000   Max.   :2016   Max.   :2016.00
##  current_zoning       sq_ft_lot        prop_type          present_use
##  Length:12865       Min.   :    785   Length:12865       Min.   :  0.000
##  Class :character   1st Qu.:   5355   Class :character   1st Qu.:  2.000
##  Mode  :character   Median :   7965   Mode  :character   Median :  2.000
##                     Mean   :  22229                      Mean   :  6.598
```

```
##                     3rd Qu.:  12632                     3rd Qu.:  2.000
##                     Max.   :1631322                     Max.   :300.000
```

```r
# Check for missing values
missing_values <- sum(is.na(housing_data))
print(paste("Total missing values:", missing_values))
```

```
## [1] "Total missing values: 0"
```

Step 3: Data Transformations

```r
# Clean the Data by removing rows with missing values
housing_data <- na.omit(housing_data)
# Cleaned the Data to assist with an easier analysis

# Example Transformation: Create a new variable for price per square foot
housing_data$price_per_sq_ft <- housing_data$Sale.Price / housing_data$square_feet_total_living
```

Step 4: Create a Linear Regression Model

```r
# Create a linear regression model where 'sq_ft_lot' predicts Sale Price
model1 <- lm(Sale.Price ~ sq_ft_lot, data = housing_data)
```

Step 5: Analyze the Model

```r
# Get a summary of the first model
summary_model1 <- summary(model1)
print(summary_model1)
```

```
##
## Call:
## lm(formula = Sale.Price ~ sq_ft_lot, data = housing_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2016064  -194842   -63293    91565  3735109
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.418e+05  3.800e+03  168.90   <2e-16 ***
## sq_ft_lot   8.510e-01  6.217e-02   13.69   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 401500 on 12863 degrees of freedom
## Multiple R-squared:  0.01435,    Adjusted R-squared:  0.01428
## F-statistic: 187.3 on 1 and 12863 DF,  p-value: < 2.2e-16
```

```r
# Explain the results (R², adj. R²)
r_squared_model1 <- summary_model1$r.squared
adj_r_squared_model1 <- summary_model1$adj.r.squared
print(paste("R²:", r_squared_model1))
```

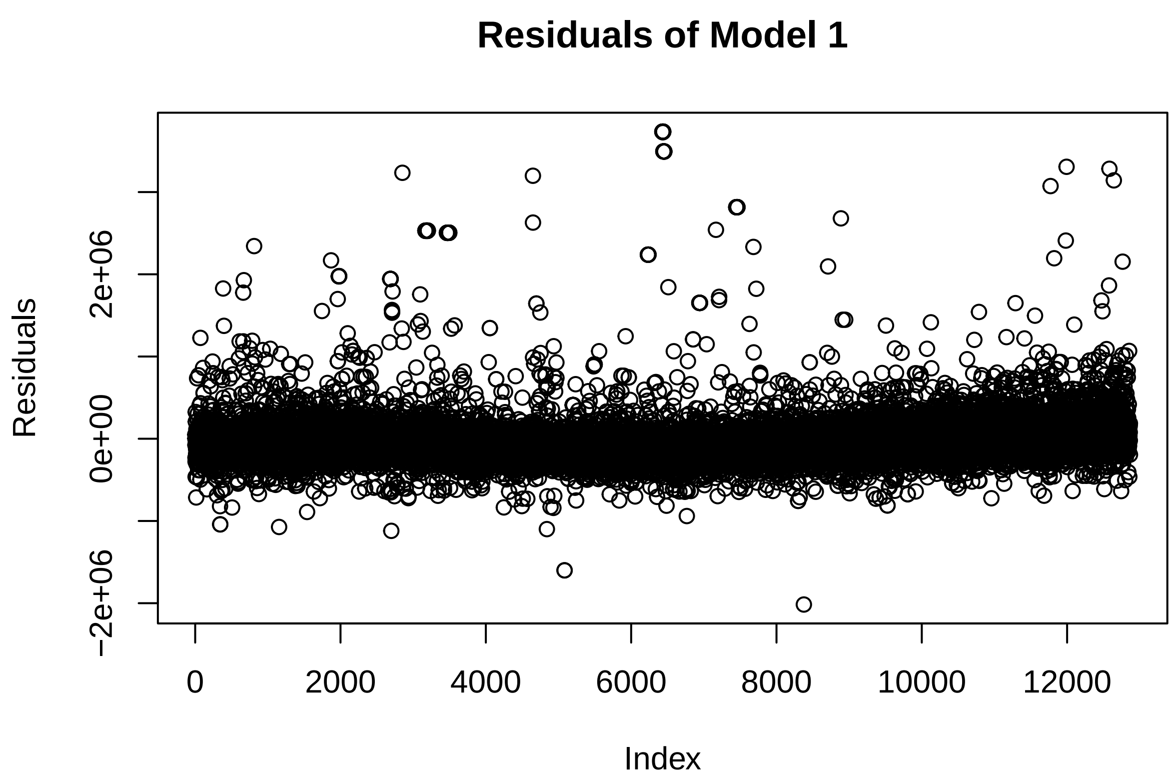
```
## [1] "R²: 0.0143549714063911"
```

```r
print(paste("Adjusted R²:", adj_r_squared_model1))
```

```
## [1] "Adjusted R²: 0.014278345033959"
```

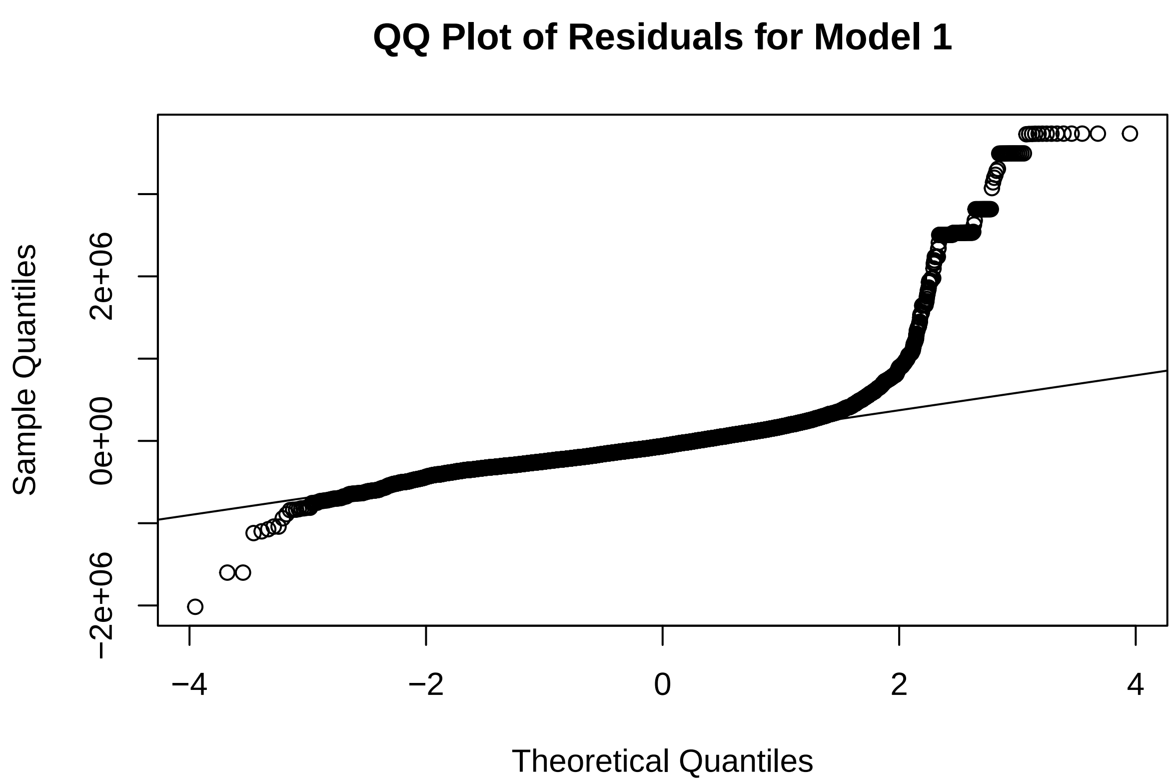
```r
# Get Residuals
residuals_model1 <- resid(model1)

# Plot Residuals
plot(residuals_model1, main="Residuals of Model 1", ylab="Residuals", xlab="Index")
```



The first model shows an R^2 value of 0.0144, which means that only about 1.4% of the changes in Sale Price can be explained by the lot size, this is a pretty weak connection. The lot size coefficient tells us that for each extra square foot, the Sale Price goes up by about $0.85, and this result is significant. Overall, this low R^2 suggests that other factors are likely more important in determining Sale Price. ## Step 6: QQ Plot for Residuals

```r
# Create a QQ Plot
qqnorm(residuals_model1, main="QQ Plot of Residuals for Model 1")
qqline(residuals_model1)
```



The residuals plot shows a mostly straight line with a slight upward incline, indicating that the model's predictions are fairly consistent across most values. # Step 7: Multiple Linear Regression Model

```r
# Create a multiple regression model using available predictors
model2 <- lm(Sale.Price ~ square_feet_total_living + bedrooms + bath_full_count + bath_half_count, data

# Get a summary of the second model
summary_model2 <- summary(model2)
print(summary_model2)
```

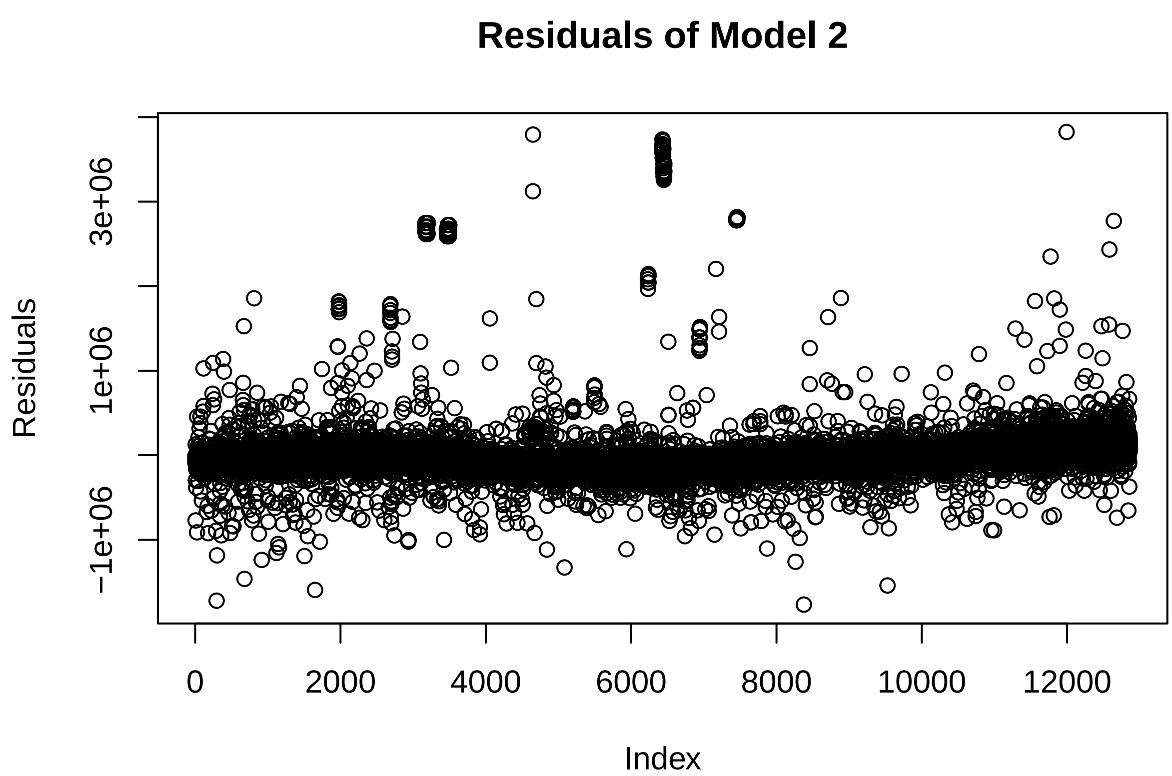
```
##
## Call:
## lm(formula = Sale.Price ~ square_feet_total_living + bedrooms +
##     bath_full_count + bath_half_count, data = housing_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1766785  -118681   -41745    43659  3823860
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              202179.839   14052.978  14.387  < 2e-16 ***
```

```
## square_feet_total_living    181.839      4.466  40.712  < 2e-16 ***
## bedrooms                 -24937.119   4418.206  -5.644 1.69e-08 ***
## bath_full_count           41444.194   5696.926   7.275 3.67e-13 ***
## bath_half_count           14655.841   6356.188   2.306   0.0211 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 359000 on 12860 degrees of freedom
## Multiple R-squared:  0.2123, Adjusted R-squared:  0.2121
## F-statistic: 866.6 on 4 and 12860 DF,  p-value: < 2.2e-16
```

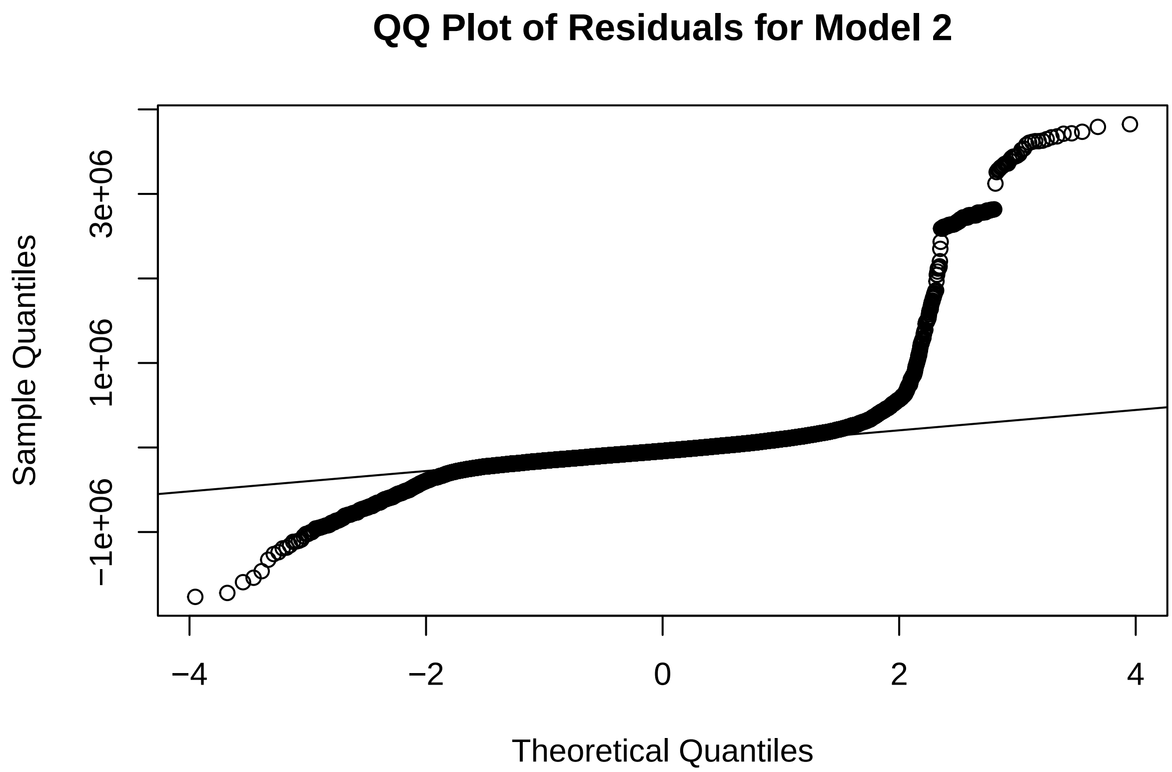
Step 8: Analyze the Residuals of the Second Model

```r
# Get Residuals for Model 2
residuals_model2 <- resid(model2)

# Plot Residuals for Model 2
plot(residuals_model2, main="Residuals of Model 2", ylab="Residuals", xlab="Index")
```




```r
# QQ Plot for Model 2 Residuals
qqnorm(residuals_model2, main="QQ Plot of Residuals for Model 2")
qqline(residuals_model2)
```



Step 9: Compare Models with ANOVA

```r
# Compare the two models using ANOVA
anova_results <- anova(model1, model2)
print(anova_results)
```

```
## Analysis of Variance Table
##
## Model 1: Sale.Price ~ sq_ft_lot
## Model 2: Sale.Price ~ square_feet_total_living + bedrooms + bath_full_count +
##     bath_half_count
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1  12863 2.0734e+15
## 2  12860 1.6570e+15  3 4.1642e+14 1077.3 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Step 10: Assess Model Bias

From the ANOVA results, Model 2 shows a significant improvement over Model 1, with a much lower Residual Sum of Squares (RSS) and a high F-value (1077.3) with a p-value less than 2.2e-16. This suggests that the additional predictors in Model 2 help explain the variability in Sale Price much better than Model 1. When examining the residuals, they appear to be randomly scattered around zero, which indicates no obvious bias in the predictions. The QQ plot also shows that the residuals closely follow the diagonal line, suggesting they meet the normality assumption. Overall, these results indicate that neither model exhibits significant bias, but Model 2 is more reliable due to its improved fit and better handling of variability in the data.

Step 11: Calculate RMSE

```r
# Make Predictions for Model 1
preds_model1 <- predict(model1, newdata = housing_data)
rmse_model1 <- rmse(housing_data$Sale.Price, preds_model1)

# Calculate RMSE for Model 2
preds_model2 <- predict(model2, newdata = housing_data)
rmse_model2 <- rmse(housing_data$Sale.Price, preds_model2)
```

Step 12: Compare RMSE

```r
print(paste("RMSE for Model 1:", rmse_model1))
```

```
## [1] "RMSE for Model 1: 401452.546946963"
```

```r
print(paste("RMSE for Model 2:", rmse_model2))
```

```
## [1] "RMSE for Model 2: 358880.953658268"
```

The RMSE for Model 1 is approximately 401,453, while for Model 2, it is around 358,881. This indicates that Model 2 has a lower RMSE, suggesting it makes more accurate predictions than Model 1, improving by about 42,572.

Step 13: Evaluate Improvement

```r
improvement <- rmse_model1 - rmse_model2
print(paste("Improvement in RMSE:", improvement))
```

```
## [1] "Improvement in RMSE: 42571.5932886947"
```