Housing_Prices

Daphney

2024-07-07

Portfolio Project: House Prices

Objective: The goal of this analysis is to predict the price of homes based on various features of housing such as number of bedrooms, house area, furnishing, etc.

Data: The Housing Prices dataset from Kaggle contains 545 observations with 13 features. These features include the median value of homes (target variable) and attributes such as average number of bedrooms, bathrooms, area square footage, etc.

```
#Load data
data <- read.csv("Housing.csv", header = TRUE)
head(data, 3)
```

```
##      price area bedrooms bathrooms stories mainroad guestroom basement
## 1 13300000 7420        4         2       3      yes        no       no
## 2 12250000 8960        4         4       4      yes        no       no
## 3 12250000 9960        3         2       2      yes        no      yes
##   hotwaterheating airconditioning parking prefarea furnishingstatus
## 1              no             yes       2      yes        furnished
## 2              no             yes       3       no        furnished
## 3              no              no       2      yes   semi-furnished
```

Exploratory Analysis:

```
summary(data)
```

```
##      price               area          bedrooms       bathrooms
##  Min.   : 1750000   Min.   : 1650   Min.   :1.000   Min.   :1.000
##  1st Qu.: 3430000   1st Qu.: 3600   1st Qu.:2.000   1st Qu.:1.000
##  Median : 4340000   Median : 4600   Median :3.000   Median :1.000
##  Mean   : 4766729   Mean   : 5151   Mean   :2.965   Mean   :1.286
##  3rd Qu.: 5740000   3rd Qu.: 6360   3rd Qu.:3.000   3rd Qu.:2.000
##  Max.   :13300000   Max.   :16200   Max.   :6.000   Max.   :4.000
##     stories        mainroad          guestroom           basement
##  Min.   :1.000   Length:545         Length:545         Length:545
##  1st Qu.:1.000   Class :character   Class :character   Class :character
##  Median :2.000   Mode  :character   Mode  :character   Mode  :character
##  Mean   :1.806
##  3rd Qu.:2.000
##  Max.   :4.000
##  hotwaterheating    airconditioning       parking         prefarea
##  Length:545         Length:545         Min.   :0.0000   Length:545
##  Class :character   Class :character   1st Qu.:0.0000   Class :character
```

```
##  Mode  :character   Mode  :character   Median :0.0000   Mode  :character
##                                        Mean   :0.6936
##                                        3rd Qu.:1.0000
##                                        Max.   :3.0000
##  furnishingstatus
##  Length:545
##  Class :character
##  Mode  :character
##
##
##
```

```
str(data)
```

```
## 'data.frame':    545 obs. of  13 variables:
##  $ price           : int  13300000 12250000 12250000 12215000 11410000
10850000 10150000 10150000 9870000 9800000 ...
##  $ area            : int  7420 8960 9960 7500 7420 7500 8580 16200 8100
5750 ...
##  $ bedrooms        : int  4 4 3 4 4 3 4 5 4 3 ...
##  $ bathrooms       : int  2 4 2 2 1 3 3 3 1 2 ...
##  $ stories         : int  3 4 2 2 2 1 4 2 2 4 ...
##  $ mainroad        : chr  "yes" "yes" "yes" "yes" ...
##  $ guestroom       : chr  "no" "no" "no" "no" ...
##  $ basement        : chr  "no" "no" "yes" "yes" ...
##  $ hotwaterheating : chr  "no" "no" "no" "no" ...
##  $ airconditioning : chr  "yes" "yes" "no" "yes" ...
##  $ parking         : int  2 3 2 3 2 2 2 0 2 1 ...
##  $ prefarea        : chr  "yes" "no" "yes" "yes" ...
##  $ furnishingstatus: chr  "furnished" "furnished" "semi-furnished"
"furnished" ...
```

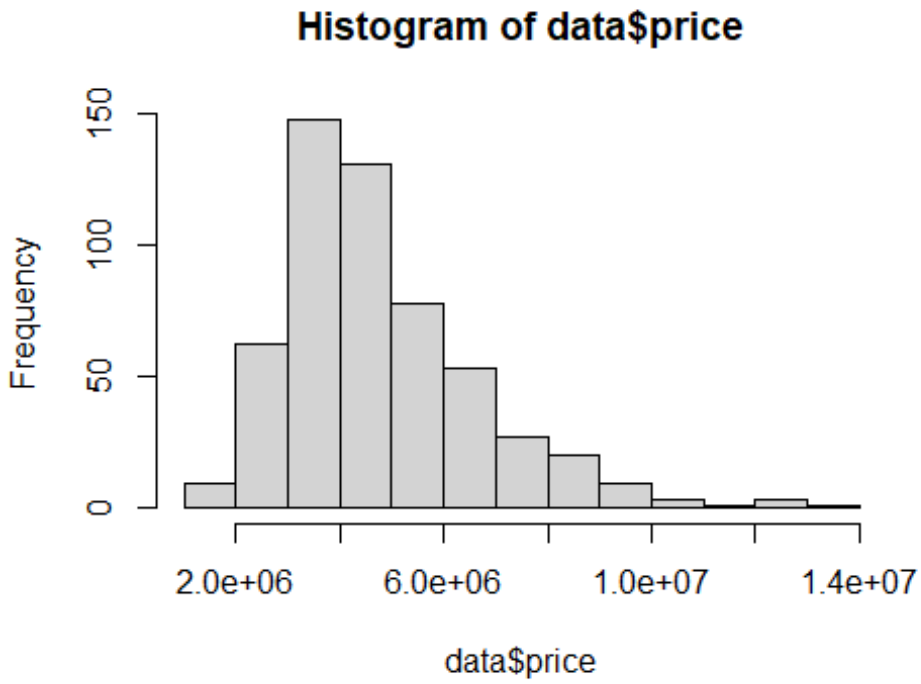
```
#Check for missing values
any(is.na(data))
```

```
## [1] FALSE
```

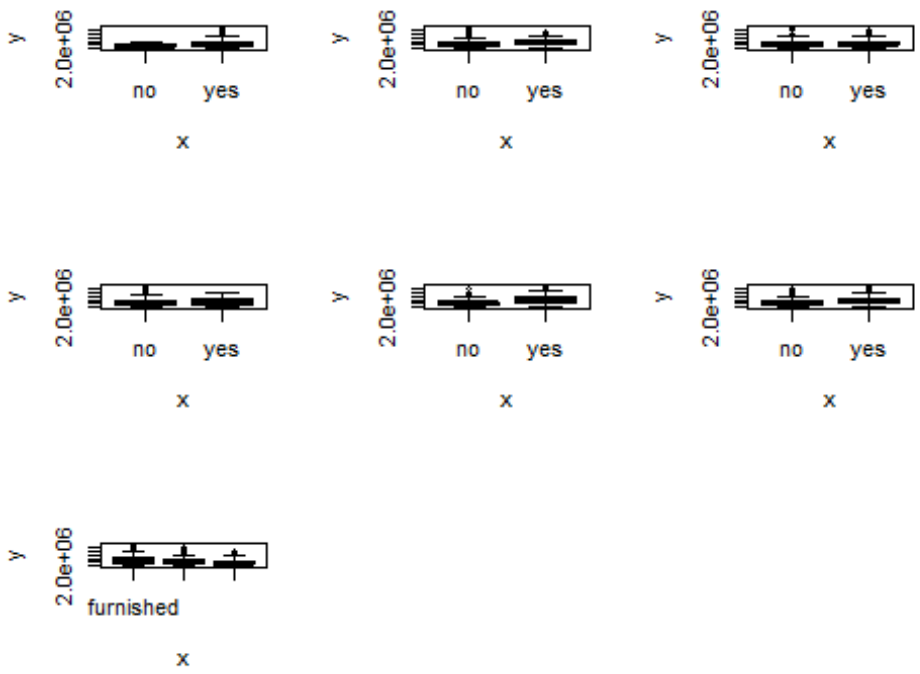
```
sum(is.na(data))
```

```
## [1] 0
```

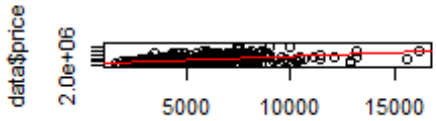
```
#Convert categorical data to numerical
data$mainroad <- as.factor(data$mainroad)
data$guestroom <- as.factor(data$guestroom)
data$basement <- as.factor(data$basement)
data$hotwaterheating <- as.factor(data$hotwaterheating)
data$airconditioning <- as.factor(data$airconditioning)
data$prefarea <- as.factor(data$prefarea)
data$furnishingstatus <- as.factor(data$furnishingstatus)

#Distribution of the target variable
hist(data$price)
```

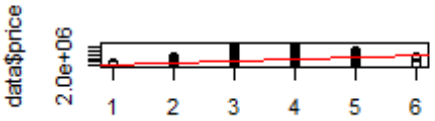


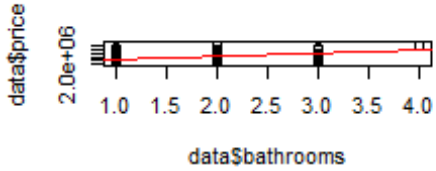

```r
#boxplot of categorical and DV
par(mfrow=c(3,3))
plot(data$mainroad, data$price)
plot(data$guestroom, data$price)
plot(data$basement, data$price)
plot(data$hotwaterheating, data$price)
plot(data$airconditioning, data$price)
plot(data$prefarea, data$price)
plot(data$furnishingstatus, data$price)
```

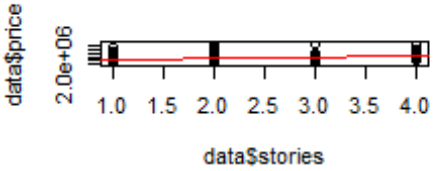


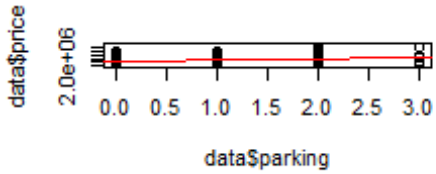

```r
#scatterplot of quantitative data and DV
par(mfrow=c(3,2))
plot(data$area, data$price)
abline(lm(price ~ area, data = data), col="red")
plot(data$bedrooms, data$price)
abline(lm(price ~ bedrooms, data = data), col="red")
plot(data$bathrooms, data$price)
abline(lm(price ~ bathrooms, data = data), col="red")
plot(data$stories, data$price)
abline(lm(price ~ stories, data = data), col="red")
plot(data$parking, data$price)
abline(lm(price ~ parking, data = data), col="red")
```











Moderate Positive trend observed for area and price. Very weak linear relationships observed with price and the other quantitative variables.

```
#Correlations of quantitative data with DV
cor(data$area, data$price)

## [1] 0.5359973

cor(data$bedrooms, data$price)

## [1] 0.366494

cor(data$bathrooms, data$price)

## [1] 0.5175453

cor(data$stories, data$price)

## [1] 0.4207124

cor(data$parking, data$price)

## [1] 0.3843936

cor(data[c(1,2,3,4,5,11)])

##               price       area  bedrooms bathrooms    stories    parking
## price     1.0000000 0.53599735 0.3664940 0.5175453 0.42071237 0.38439365
## area      0.5359973 1.00000000 0.1518585 0.1938195 0.08399605 0.35298048
```

```
## bedrooms  0.3664940 0.15185849 1.0000000 0.3739302 0.40856424 0.13926990
## bathrooms 0.5175453 0.19381953 0.3739302 1.0000000 0.32616471 0.17749582
## stories   0.4207124 0.08399605 0.4085642 0.3261647 1.00000000 0.04554709
## parking   0.3843936 0.35298048 0.1392699 0.1774958 0.04554709 1.00000000
```

Regression Analysis:

```
#Split data
library(caret)

## Warning: package 'caret' was built under R version 4.3.3

## Loading required package: ggplot2

## Loading required package: lattice

set.seed(123)
train_index <- createDataPartition(data$price, p = 0.7, list = FALSE)
train_data <- data[train_index,]
test_data <- data[-train_index,]

#Full MLR model will all variables
full_model <- lm(price ~ ., data = train_data)
summary(full_model)

##
## Call:
## lm(formula = price ~ ., data = train_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2462436  -633064   -70271   500228  4583600
##
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       25636.10  295381.07   0.087  0.93089
## area                                254.89      27.79   9.172  < 2e-16 ***
## bedrooms                          81846.53   80348.63   1.019  0.30904
## bathrooms                       1035481.32  115786.10   8.943  < 2e-16 ***
## stories                          433962.89   72590.46   5.978 5.32e-09 ***
## mainroadyes                      480845.74  162290.79   2.963  0.00325 **
## guestroomyes                     255736.60  153844.27   1.662  0.09730 .
## basementyes                      297735.83  130569.61   2.280  0.02316 *
## hotwaterheatingyes               657281.47  269190.72   2.442  0.01509 *
## airconditioningyes               811114.01  124533.11   6.513 2.41e-10 ***
## parking                          201792.58   67141.16   3.005  0.00283 **
## prefareayes                      677098.10  130831.24   5.175 3.74e-07 ***
## furnishingstatussemi-furnished    58318.26  134649.72   0.433  0.66519
## furnishingstatusunfurnished     -396970.12  144972.29  -2.738  0.00648 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 1025000 on 369 degrees of freedom
## Multiple R-squared:  0.6946, Adjusted R-squared:  0.6838
## F-statistic: 64.55 on 13 and 369 DF,  p-value: < 2.2e-16

#Reduced model with the most significant predictors
reduced_model <- lm(price ~ area + bathrooms + stories + hotwaterheating +
airconditioning + parking, data = train_data)
summary(reduced_model)

##
## Call:
## lm(formula = price ~ area + bathrooms + stories + hotwaterheating +
##     airconditioning + parking, data = train_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3145678  -678134   -81517   610802  5031341
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        2.918e+05  2.103e+05   1.388 0.166050
## area               3.227e+02  2.884e+01  11.190  < 2e-16 ***
## bathrooms          1.157e+06  1.213e+05   9.545  < 2e-16 ***
## stories            4.349e+05  7.192e+04   6.048 3.55e-09 ***
## hotwaterheatingyes 6.063e+05  2.950e+05   2.055 0.040557 *
## airconditioningyes 9.163e+05  1.350e+05   6.786 4.49e-11 ***
## parking            2.638e+05  7.334e+04   3.597 0.000365 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1131000 on 376 degrees of freedom
## Multiple R-squared:  0.6213, Adjusted R-squared:  0.6153
## F-statistic: 102.8 on 6 and 376 DF,  p-value: < 2.2e-16

#Compare the two models
anova_model <- anova(reduced_model, full_model)
anova_model

## Analysis of Variance Table
##
## Model 1: price ~ area + bathrooms + stories + hotwaterheating +
airconditioning +
##     parking
## Model 2: price ~ area + bedrooms + bathrooms + stories + mainroad +
guestroom +
##     basement + hotwaterheating + airconditioning + parking +
##     prefarea + furnishingstatus
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1    376 4.8092e+14
## 2    369 3.8786e+14  7 9.3059e+13 12.648 1.456e-14 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(reduced_model)$r.squared
```

```
## [1] 0.6212968
```

```
summary(full_model)$r.squared
```

```
## [1] 0.694577
```

```
cat("Reduced Model adj R^2:", summary(reduced_model)$adj.r.squared)
```

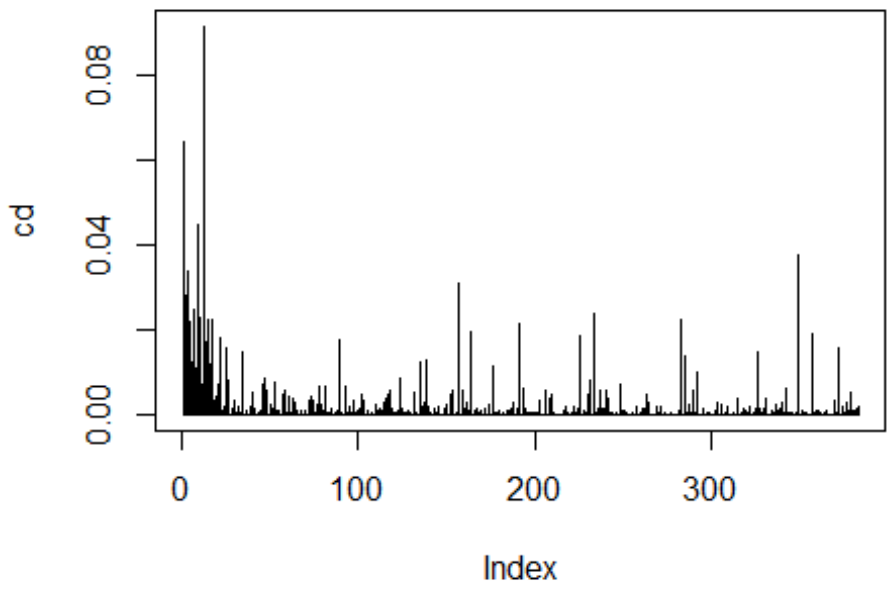
```
## Reduced Model adj R^2: 0.6152536
```

```
cat("Full Model adj R^2:", summary(full_model)$adj.r.squared)
```

```
## Full Model adj R^2: 0.6838168
```

P-value of the f-statistic is 2.2e-16, which is less than the alpha level so we can reject the null. Concluding that, the additional predictors in the full model significantly improves the model. R^2 is the proportion of variance explained by the predictors. Adj r^2 considers the other variables. The full model has a higher adj r^2 value than the reduced model. It explains 68.3% of the variability in price. That's about 7% more than the reduced model.

```
#Checking for outliers
cd <- cooks.distance(full_model)
plot(cd, type ="h")
abline(h=1, col="red")
```



No significant outliers using threshold of 1.

```
#Checking for multicolinearity among predictors
library(car)

## Loading required package: carData

vif <- (1/(1- summary(full_model)$r.squared))
vif(full_model)

##                       GVIF Df GVIF^(1/(2*Df))
## area              1.327118  1        1.152006
## bedrooms          1.314742  1        1.146622
## bathrooms         1.291509  1        1.136446
## stories           1.457275  1        1.207176
## mainroad          1.198059  1        1.094559
## guestroom         1.185244  1        1.088689
## basement          1.360067  1        1.166219
## hotwaterheating   1.056965  1        1.028088
## airconditioning   1.226661  1        1.107547
## parking           1.184569  1        1.088379
## prefarea          1.163305  1        1.078566
## furnishingstatus 1.121952  2        1.029185
```

No evidence of multicollinearity.

```
#Prediction
predict_reduced <- predict(reduced_model, newdata = test_data)
```

```
predict_full <- predict(full_model, newdata = test_data)
predict_reduced
```

```
##       1       4       5       7       9      14      17      18      25
27
## 7750181 7604873 6157743 9716975 6377193 5740140 6786779 8533667 7760588
7199219
##      29      32      34      38      52      54      55      56      57
58
## 7176250 6892095 6821441 8695028 6410436 7452551 6593146 5435656 6274974
8431206
##      60      62      75      78      85      86      88      90      95
100
## 7463041 5930146 4493096 6925632 4666557 7490394 3689893 7930794 6546755
4690495
##     105     112     114     115     116     118     119     131     132
136
## 5344374 6899921 5516496 4606421 5909973 4429574 4872429 3433334 5069935
7199219
##     145     146     151     155     157     158     164     166     170
175
## 5016118 4802721 4893002 5182287 4111049 4649638 5003131 5729575 6503593
3809383
##     176     180     182     183     186     192     195     201     203
208
## 7167024 5030608 4906635 4335985 3287383 5772338 5671941 4694206 3477706
4203669
##     212     216     218     219     225     228     235     236     238
239
## 6575022 4096688 5977664 4354461 6632869 5000708 5256513 5332385 3936053
4932183
##     245     247     250     251     252     259     265     267     268
269
## 4036097 3020250 5087089 3886919 3696430 3886836 3900554 3986796 4170831
4832976
##     272     273     274     276     282     283     285     289     298
301
## 2934003 3727006 3976389 3185484 4780448 3937424 4655639 4862021 4651415
4070547
##     304     305     307     308     311     317     320     328     333
335
## 3336517 5338190 4145013 4163567 3815759 5644588 5166262 5590562 4094186
3816108
##     339     342     346     348     350     351     354     355     356
361
## 3539106 4595660 2969502 3400336 3874736 4293009 3675859 4858953 5074367
3188066
##     362     365     369     374     379     384     386     394     401
405
## 3827054 3932370 3704420 4203669 5010904 5456600 3058977 4280155 3281491
```

```
3900554
##     406     409     410     413     419     425     426     427     428
432
## 2871799 3175157 3439784 3161522 4892675 3319655 3439784 2755619 3011456
4261759
##     436     441     452     453     457     459     462     463     464
466
## 3188066 3493925 4062640 5751356 2658157 3561696 4401255 3016297 2881480
3110612
##     467     469     470     473     475     476     479     481     485
493
## 3316428 2799186 3368790 4282165 3987525 3287383 3744838 3706112 2865344
3438253
##     495     505     506     512     516     521     522     523     524
525
## 4078777 3828425 4526391 2910525 3355155 4369226 3057363 3117954 4376134
2937634
##     534     539
## 3093750 3061881

predict_full

##       1       4       5       7       9      14      17      18      25
27
## 7989837 8075794 6396767 9813277 7247193 5725755 7301570 7940141 7148684
8188208
##      29      32      34      38      52      54      55      56      57
58
## 6842650 6580636 6603860 8149435 5751501 7144562 6547241 4800734 6888217
8880477
##      60      62      75      78      85      86      88      90      95
100
## 7356848 5474310 4842014 7269902 4733018 7038870 3990489 7480667 6430162
5534208
##     105     112     114     115     116     118     119     131     132
136
## 6293025 6041100 6051973 4829962 6764213 4747237 5643839 3601463 5404783
6502349
##     145     146     151     155     157     158     164     166     170
175
## 5501660 4797829 5609283 5083180 5269129 4765215 6061126 5714803 5828355
4659422
##     176     180     182     183     186     192     195     201     203
208
## 6988366 4963379 5923035 3913207 3320879 5994594 5946131 4974721 3747633
4587281
##     212     216     218     219     225     228     235     236     238
239
## 5913195 3752784 6296314 4913468 6244570 4979631 5439541 5430193 4288506
5278386
```

```
##     245     247     250     251     252     259     265     267     268
269
## 5300358 3854112 6275252 4042391 4092663 3945315 4178621 4149066 4169622
4868437
##     272     273     274     276     282     283     285     289     298
301
## 3179905 4275538 4032992 3465091 4501345 3696595 4321945 4651978 4519493
4187426
##     304     305     307     308     311     317     320     328     333
335
## 3666228 5285780 4149231 4157303 4103066 4898349 5506676 4897526 4750988
3837297
##     339     342     346     348     350     351     354     355     356
361
## 4148875 4745305 4202194 3112356 3942340 4362715 4494469 4482529 4727934
3227713
##     362     365     369     374     379     384     386     394     401
405
## 3814251 3983190 3154699 4149381 5873089 5621104 2670466 3235989 3516730
3026598
##     406     409     410     413     419     425     426     427     428
432
## 2604474 2762229 2958074 3898568 4447136 3320811 3652780 2428836 4177012
4259721
##     436     441     452     453     457     459     462     463     464
466
## 2772424 3565858 3918478 5411386 3568120 3239804 4115777 3081210 3083747
2711250
##     467     469     470     473     475     476     479     481     485
493
## 3020526 2920564 3312136 4326239 3170554 3396585 2897027 3321728 2036684
3433458
##     495     505     506     512     516     521     522     523     524
525
## 3475934 3365603 4089152 2154215 3374407 3705340 2188346 3286293 4483148
2574626
##     534     539
## 2870206 2672760

#calculate mse and rsme
residuals1 <- test_data$price - predict_reduced
mse <- mean(residuals1^2)
rmse <- sqrt(mse)
print(paste("MSE:", mse))

## [1] "MSE: 1619839246511.61"

print(paste("RMSE:", rmse))

## [1] "RMSE: 1272729.05463481"
```

```r
residuals2 <- test_data$price - predict_full
mse2 <- mean(residuals2^2)
rmse2 <- sqrt(mse)
print(paste("MSE full model:", mse2))
```

```
## [1] "MSE full model: 1382686385937.49"
```

```r
print(paste("RMSE full model:", rmse2))
```

```
## [1] "RMSE full model: 1272729.05463481"
```

```r
#Adj r^2
cat("Reduced Model adj R^2:", summary(reduced_model)$adj.r.squared)
```

```
## Reduced Model adj R^2: 0.6152536
```

```r
cat("Full Model adj R^2:", summary(full_model)$adj.r.squared)
```

```
## Full Model adj R^2: 0.6838168
```

```r
#Predict price for a specific scenario
new_data = data.frame(
  area = 7420,
  bedrooms = 4,
  bathrooms = 2,
  stories = 3,
  mainroad = factor("yes", levels = levels(data$mainroad)),
  guestroom = factor("no", levels = levels(data$guestroom)),
  basement = factor("no", levels = levels(data$basement)),
  hotwaterheating = factor("no", levels = levels(data$hotwaterheating)),
  airconditioning = factor("yes", levels = levels(data$airconditioning)),
  parking = 2,
  prefarea = factor("yes", levels = levels(data$prefarea)),
  furnishingstatus = factor("furnished", levels =
levels(data$furnishingstatus))
)

predict(reduced_model, newdata =  new_data)
```

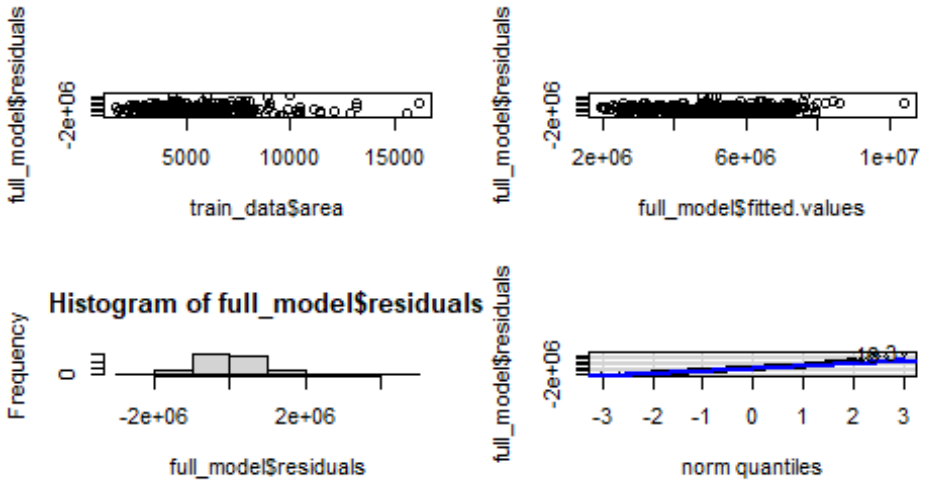
```
##       1
## 7750181
```

```r
predict(full_model, newdata = new_data)
```

```
##       1
## 7989837
```

```r
#Testing Assumptions
#Linearity & Variance: residuals vs IV
library(car)
par(mfrow =c(3,2))
plot(train_data$area, full_model$residuals)
```

```
#Independence : residuals vs fitted
plot(full_model$fitted.values, full_model$residuals)
abline(full_model)

## Warning in abline(full_model): only using the first two of 14 regression
## coefficients

#Normality
hist(full_model$residuals)
qqPlot(full_model$residuals)

##  3 16
##  2 10
```



The full model that includes all of the predictors was able to predict 67% of the variability in the data. It is a better model than the reduced model, by about 7% since it explains more of the variability in price.