stepwise regression process to find best var log

Steven

2024-02-23

Data source and original descriptions

Data preparation

```
# Check if the caret package is installed
if (!requireNamespace("caret", quietly = TRUE)) {
  # If not installed, install it
  install.packages("caret")
}

# Load the caret package
library(caret)

## 载入需要的程辑包：ggplot2

## 载入需要的程辑包：lattice

# Load necessary libraries
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyve
rse 2.0.0 ──
## ✔ dplyr     1.1.1     ✔ readr     2.1.5
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ lubridate 1.9.2     ✔ tibble    3.2.1
## ✔ purrr     1.0.1     ✔ tidyr     1.3.0

## ── Conflicts ──────────────────────────────────────── tidyverse_co
nflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ✖ purrr::lift()   masks caret::lift()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to for
ce all conflicts to become errors

# Load the data
data <- read_csv(file.choose())  # open file location

## Warning: One or more parsing issues, call `problems()` on your data
frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)

## Rows: 34857 Columns: 21
## ── Column specification ──────────────────────────────────────
────────
## Delimiter: ","
## chr  (8): Suburb, Address, Type, Method, SellerG, Date, CouncilArea,
 Regionname
## dbl (13): Rooms, Price, Distance, Postcode, Bedroom2, Bathroom, Car,
 Landsiz...
##
## ℹ Use `spec()` to retrieve the full column specification for this da
ta.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet
this message.

# Correct column names
names(data)[names(data) == "Lattitude"] <- "Latitude"
names(data)[names(data) == "Longtitude"] <- "Longitude"

# Remove unnecessary columns using dplyr's select function
data_clean <- data %>%
  dplyr::select(Suburb, Rooms, Type, Price, Distance, Bedroom2, Bathroo
m, Car, Landsize, BuildingArea, YearBuilt, CouncilArea, Latitude, Longi
tude, Propertycount, Date)

# Convert 'Date' to date type
data_clean$Date <- as.Date(data_clean$Date, format = "%d/%m/%Y")

# Calculate 'YearsAfterBuilt'
data_clean$YearsAfterBuilt <- as.numeric(format(data_clean$Date, "%Y"))
 - data_clean$YearBuilt

# Calculate LogPricePerBuildingArea
data_clean$LogPricePerBuildingArea <- log(data_clean$Price / data_clean
$BuildingArea)

# Drop the "Price", "Longitude", "Latitude", "YearBuilt" and columns fr
om the dataset
# due to persistent issue of factor mis matching between training and t
esting, and since the reference models do not contain Suburb variable,
clean at the beginning
# if Date is not deleted, the result would include it
data_clean <- subset(data_clean, select = -c(Price, Longitude, Latitude
, YearBuilt, Suburb, Date))

# Remove rows with missing values
data_clean <- na.omit(data_clean)
```

```
# Convert categorical variables to factors
cat_vars <- c("Type", "CouncilArea")  # Add categorical variables here
data_clean[cat_vars] <- lapply(data_clean[cat_vars], as.factor)

# Convert non-categorical variables to numeric
non_cat_vars <- setdiff(names(data_clean), c(cat_vars, "LogPricePerBuildingArea"))
data_clean[non_cat_vars] <- lapply(data_clean[non_cat_vars], as.numeric)


# Standardize non-categorical variables
data_clean[non_cat_vars] <- scale(data_clean[non_cat_vars])

# Separate predictors and target variable
predictors <- setdiff(names(data_clean), "LogPricePerBuildingArea")

# Split data into training and testing sets
set.seed(123)
indexes <- createDataPartition(data_clean$LogPricePerBuildingArea, p = 0.8, list = FALSE)
train_data <- data_clean[indexes, ]
test_data <- data_clean[-indexes, ]

problems(data_clean)
```

Model training and AIC process

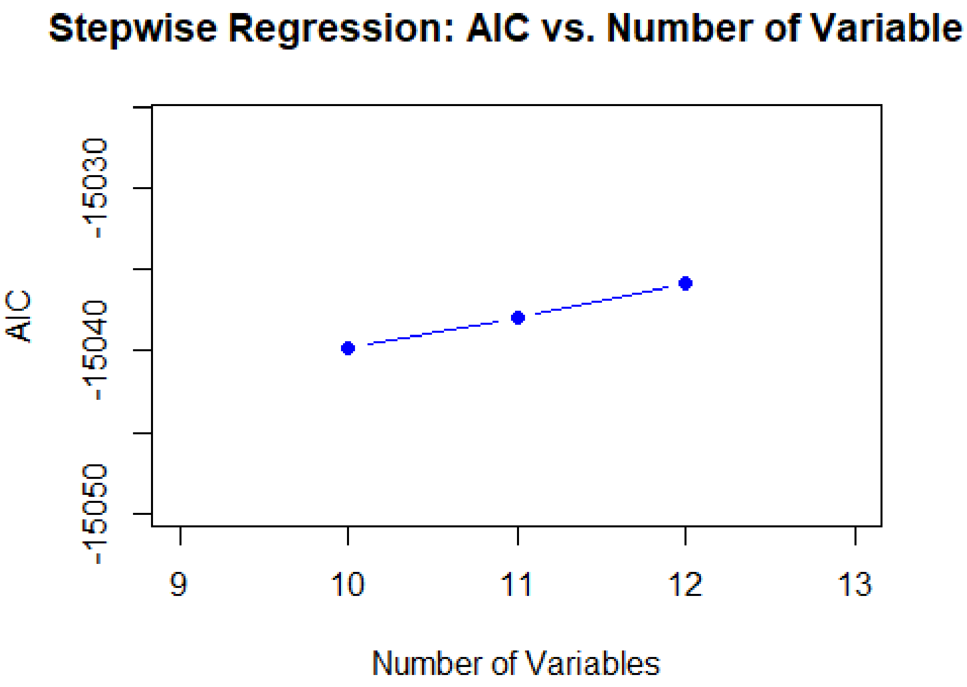
```
# Remove rows with NA, NaN, or Inf values in the target variable
train_data <- train_data[!is.na(train_data$LogPricePerBuildingArea) & !is.nan(train_data$LogPricePerBuildingArea) & !is.infinite(train_data$LogPricePerBuildingArea), ]

# Train stepwise regression model
model <- step(lm(LogPricePerBuildingArea ~ ., data = train_data[, c(predictors, "LogPricePerBuildingArea")]), direction = "backward")

## Start:  AIC=-15035.86
## LogPricePerBuildingArea ~ Rooms + Type + Distance + Bedroom2 +
##     Bathroom + Car + Landsize + BuildingArea + CouncilArea +
##     Propertycount + YearsAfterBuilt
##
##                    Df Sum of Sq    RSS    AIC
## - Rooms             1     0.003 837.70 -15038
## - Propertycount     1     0.007 837.71 -15038
## <none>                          837.70 -15036
## - Bedroom2          1     0.408 838.11 -15034
## - Landsize          1     0.803 838.50 -15031
## - Car                1     3.543  841.24 -15008
## - Bathroom           1     4.502  842.20 -15000
## - YearsAfterBuilt    1    28.311  866.01 -14802
## - Type               2    42.633  880.33 -14688
## - Distance           1    43.216  880.92 -14682
## - BuildingArea       1   248.853 1086.55 -13195
## - CouncilArea       32   293.991 1131.69 -12969
##
## Step:  AIC=-15037.84
## LogPricePerBuildingArea ~ Type + Distance + Bedroom2 + Bathroom +
##     Car + Landsize + BuildingArea + CouncilArea + Propertycount +
##     YearsAfterBuilt
##
##                     Df Sum of Sq     RSS    AIC
## - Propertycount      1     0.007  837.71 -15040
## <none>                            837.70 -15038
## - Landsize           1     0.804  838.51 -15033
## - Bedroom2           1     2.751  840.46 -15017
## - Car                1     3.551  841.26 -15010
## - Bathroom           1     4.564  842.27 -15001
## - YearsAfterBuilt    1    28.428  866.13 -14803
## - Type               2    43.024  880.73 -14687
## - Distance           1    43.213  880.92 -14684
## - BuildingArea       1   251.368 1089.07 -13181
## - CouncilArea       32   294.003 1131.71 -12971
##
## Step:  AIC=-15039.78
## LogPricePerBuildingArea ~ Type + Distance + Bedroom2 + Bathroom +
##     Car + Landsize + BuildingArea + CouncilArea + YearsAfterBuilt
##
##                     Df Sum of Sq     RSS    AIC
## <none>                            837.71 -15040
## - Landsize           1     0.805  838.52 -15035
## - Bedroom2           1     2.756  840.47 -15018
## - Car                1     3.549  841.26 -15012
## - Bathroom           1     4.563  842.27 -15003
## - YearsAfterBuilt    1    28.429  866.14 -14805
## - Type               2    43.252  880.96 -14687
## - Distance           1    44.233  881.94 -14677
## - BuildingArea       1   251.447 1089.16 -13182
## - CouncilArea       32   294.140 1131.85 -12972
```

```
# Make predictions on test data
predictions <- predict(model, newdata = test_data)

# Evaluate the model
rmse <- sqrt(mean((predictions - test_data$LogPricePerBuildingArea)^2))
print(paste("RMSE: ", rmse))
```

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

Displaying AIC value graph

```
# Update AIC values from the stepwise regression process
aic_values <- c(-15035.86, -15038, -15039.78)

# Number of variables in each step (including the intercept)
num_variables <- c(12, 11, 10)  # Adjust this based on the actual steps
 in model

# Plotting the updated AIC values
plot(num_variables, aic_values, type = "b",
     xlab = "Number of Variables",
     ylab = "AIC",
     main = "Stepwise Regression: AIC vs. Number of Variables",
     xlim = c(min(num_variables) - 1, max(num_variables) + 1),
     ylim = c(min(aic_values) - 10, max(aic_values) + 10),
     col = "blue",
     pch = 19)
```



final model summary

```
# Train the final model based on the selected predictors from the stepw
ise regression
final_model <- lm(LogPricePerBuildingArea ~ Type + Distance + Bedroom2
+ Bathroom + Car + Landsize + BuildingArea + CouncilArea + YearsAfterBu
ilt, data = train_data)
```

```r
# Print the summary of the final model
summary(final_model)
```

```
## 
## Call:
## lm(formula = LogPricePerBuildingArea ~ Type + Distance + Bedroom2 + 
##     Bathroom + Car + Landsize + BuildingArea + CouncilArea + 
##     YearsAfterBuilt, data = train_data)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -2.4916 -0.1618 -0.0275  0.1183  5.4498 
## 
## Coefficients:
##                                              Estimate Std. Error t val
ue Pr(>|t|)    
## (Intercept)                                  8.826337   0.019003 464.4
71  < 2e-16 ***
## Typet                                       -0.099754   0.017191  -5.8
03 6.82e-09 ***
## Typeu                                       -0.283054   0.014856 -19.0
53  < 2e-16 ***
## Distance                                    -0.199143   0.010327 -19.2
83  < 2e-16 ***
## Bedroom2                                     0.032268   0.006704   4.8
13 1.51e-06 ***
## Bathroom                                     0.036163   0.005839   6.1
94 6.21e-10 ***
## Car                                          0.025940   0.004749   5.4
62 4.87e-08 ***
## Landsize                                     0.010862   0.004175   2.6
02 0.009286 ** 
## BuildingArea                                -0.246039   0.005352 -45.9
75  < 2e-16 ***
## CouncilAreaBayside City Council              0.503836   0.027846  18.0
94  < 2e-16 ***
## CouncilAreaBoroondara City Council           0.404661   0.024388  16.5
92  < 2e-16 ***
## CouncilAreaBrimbank City Council            -0.258649   0.027057  -9.5
59  < 2e-16 ***
## CouncilAreaCardinia Shire Council            0.409020   0.116406   3.5
14 0.000445 ***
## CouncilAreaCasey City Council                0.218291   0.072249   3.0
21 0.002525 ** 
## CouncilAreaDarebin City Council              0.024645   0.024373   1.0
11 0.311968    
## CouncilAreaFrankston City Council            0.395573   0.058849   6.7
22 1.93e-11 ***
## CouncilAreaGlen Eira City Council            0.246480   0.025588   9.6
```

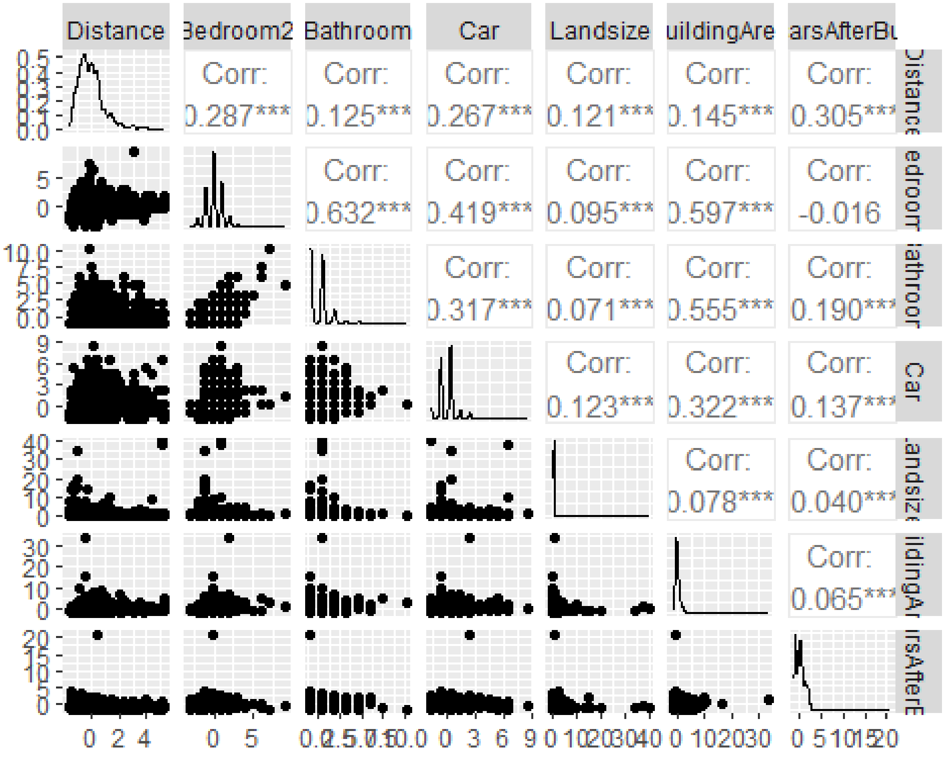
```
33  < 2e-16
## CouncilAreaGreater Dandenong City Council  0.113331   0.058574   1.9
35 0.053050
## CouncilAreaHobsons Bay City Council        0.024723   0.029806   0.8
29 0.406859
## CouncilAreaHume City Council              -0.235406   0.029167  -8.0
71 8.13e-16
## CouncilAreaKingston City Council           0.267839   0.034269   7.8
16 6.26e-15
## CouncilAreaKnox City Council               0.177618   0.044424   3.9
98 6.45e-05
## CouncilAreaMacedon Ranges Shire Council    0.492425   0.132234   3.7
24 0.000198
## CouncilAreaManningham City Council         0.203025   0.030926   6.5
65 5.58e-11
## CouncilAreaMaribyrnong City Council       -0.068159   0.026653  -2.5
57 0.010572
## CouncilAreaMaroondah City Council          0.204202   0.045207   4.5
17 6.37e-06
## CouncilAreaMelbourne City Council          0.166427   0.029536   5.6
35 1.82e-08
## CouncilAreaMelton City Council            -0.261441   0.045606  -5.7
33 1.03e-08
## CouncilAreaMitchell Shire Council          0.301946   0.180071   1.6
77 0.093623
## CouncilAreaMonash City Council             0.297117   0.029246  10.1
59  < 2e-16
## CouncilAreaMoonee Valley City Council      0.023695   0.025656   0.9
24 0.355731
## CouncilAreaMoorabool Shire Council        -0.019687   0.347555  -0.0
57 0.954830
## CouncilAreaMoreland City Council          -0.061928   0.024916  -2.4
85 0.012961
## CouncilAreaNillumbik Shire Council         0.004139   0.076393   0.0
54 0.956791
## CouncilAreaPort Phillip City Council       0.301432   0.030442   9.9
02  < 2e-16
## CouncilAreaStonnington City Council        0.408323   0.031168  13.1
01  < 2e-16
## CouncilAreaWhitehorse City Council         0.312729   0.040145   7.7
90 7.67e-15
## CouncilAreaWhittlesea City Council        -0.153068   0.033574  -4.5
59 5.22e-06
## CouncilAreaWyndham City Council           -0.382601   0.036906 -10.3
67  < 2e-16
## CouncilAreaYarra City Council              0.149312   0.031848   4.6
88 2.81e-06
## CouncilAreaYarra Ranges Shire Council      0.220659   0.091819   2.4
03 0.016279
## YearsAfterBuilt                            0.080845   0.005230  15.4
```

```
59  < 2e-16
##
## (Intercept)                               ***
## Typet                                     ***
## Typeu                                     ***
## Distance                                  ***
## Bedroom2                                  ***
## Bathroom                                  ***
## Car                                       ***
## Landsize                                  **
## BuildingArea                              ***
## CouncilAreaBayside City Council           ***
## CouncilAreaBoroondara City Council        ***
## CouncilAreaBrimbank City Council          ***
## CouncilAreaCardinia Shire Council         ***
## CouncilAreaCasey City Council             **
## CouncilAreaDarebin City Council
## CouncilAreaFrankston City Council         ***
## CouncilAreaGlen Eira City Council         ***
## CouncilAreaGreater Dandenong City Council .
## CouncilAreaHobsons Bay City Council
## CouncilAreaHume City Council              ***
## CouncilAreaKingston City Council          ***
## CouncilAreaKnox City Council              ***
## CouncilAreaMacedon Ranges Shire Council   ***
## CouncilAreaManningham City Council        ***
## CouncilAreaMaribyrnong City Council       *
## CouncilAreaMaroondah City Council         ***
## CouncilAreaMelbourne City Council         ***
## CouncilAreaMelton City Council            ***
## CouncilAreaMitchell Shire Council         .
## CouncilAreaMonash City Council            ***
## CouncilAreaMoonee Valley City Council
## CouncilAreaMoorabool Shire Council
## CouncilAreaMoreland City Council          *
## CouncilAreaNillumbik Shire Council
## CouncilAreaPort Phillip City Council      ***
## CouncilAreaStonnington City Council       ***
## CouncilAreaWhitehorse City Council        ***
## CouncilAreaWhittlesea City Council        ***
## CouncilAreaWyndham City Council           ***
## CouncilAreaYarra City Council             ***
## CouncilAreaYarra Ranges Shire Council     *
## YearsAfterBuilt                           ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3449 on 7042 degrees of freedom
## Multiple R-squared:  0.5565, Adjusted R-squared:  0.5539
## F-statistic: 215.5 on 41 and 7042 DF,  p-value: < 2.2e-16
```

variable correlation check

```
if (!requireNamespace("GGally", quietly = TRUE)) {
    install.packages("GGally")
}

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2

# Load the GGally package
library(GGally)

#lm(formula = LogPricePerBuildingArea ~ Type + Distance + Bedroom2 +
#    Bathroom + Car + Landsize + BuildingArea + CouncilArea +
#    YearsAfterBuilt, data = train_data)
# Select predictors for correlation analysis based on the final model (
excluding non-numeric variables if they are not numerically encoded)
cor_data <- train_data[, c("Distance", "Bedroom2", "Bathroom", "Car", "
Landsize", "BuildingArea", "YearsAfterBuilt")]

# Compute pairwise correlations
correlation_matrix <- cor(cor_data)

# Print pairwise correlations
print(correlation_matrix)

##                    Distance    Bedroom2    Bathroom        Car     Lan
dsize
## Distance         1.0000000  0.28744646  0.12531840  0.2668261  0.120
72475
## Bedroom2         0.2874465  1.00000000  0.63180352  0.4185207  0.094
91182
## Bathroom         0.1253184  0.63180352  1.00000000  0.3171892  0.071
18188
## Car              0.2668261  0.41852066  0.31718916  1.0000000  0.122
62196
## Landsize         0.1207248  0.09491182  0.07118188  0.1226220  1.000
00000
## BuildingArea     0.1452202  0.59721723  0.55463622  0.3219861  0.077
93123
## YearsAfterBuilt -0.3050505 -0.01601471 -0.18987369 -0.1371532 -0.039
93164
##                 BuildingArea YearsAfterBuilt
## Distance          0.14522024     -0.30505052
## Bedroom2          0.59721723     -0.01601471
## Bathroom          0.55463622     -0.18987369
## Car               0.32198611     -0.13715316
## Landsize          0.07793123     -0.03993164
## BuildingArea      1.00000000     -0.06529981
## YearsAfterBuilt  -0.06529981      1.00000000
```

```
# Create a histogram grid for visualization
ggpairs(cor_data)
```

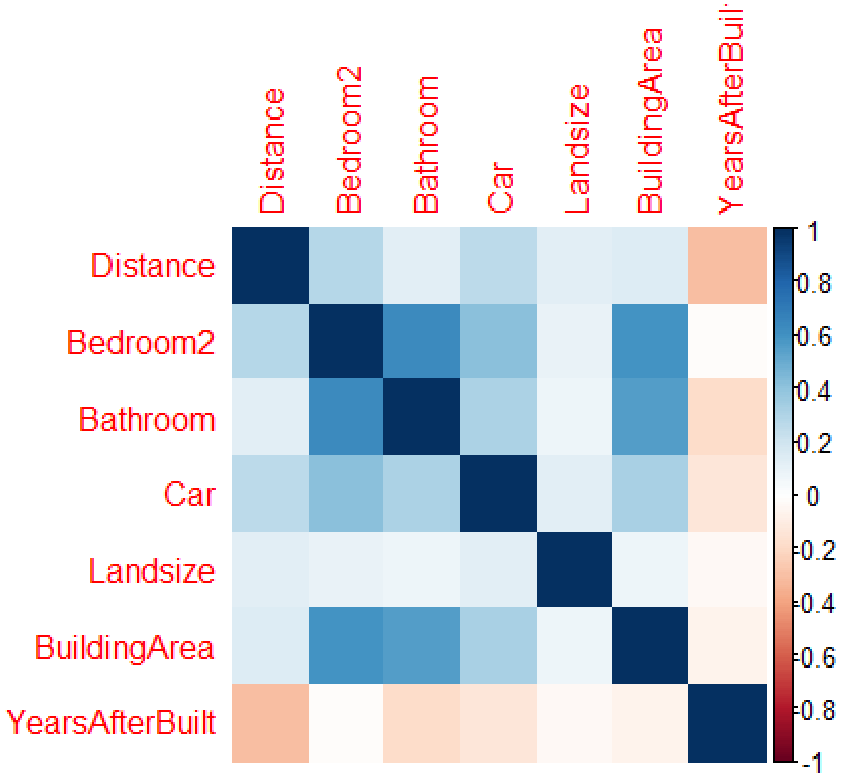


#

correlation graph display

```
# Load necessary libraries
library(corrplot)

## corrplot 0.92 loaded

# Convert non-numeric columns to numeric
cor_data_numeric <- as.data.frame(sapply(cor_data, as.numeric))

# Compute pairwise correlations
correlation_matrix <- cor(cor_data_numeric)

# Print pairwise correlations
print(correlation_matrix)

##                  Distance    Bedroom2    Bathroom        Car     Lan
dsize
## Distance        1.0000000  0.28744646  0.12531840  0.2668261  0.120
72475
## Bedroom2        0.2874465  1.00000000  0.63180352  0.4185207  0.094
91182
## Bathroom        0.1253184  0.63180352  1.00000000  0.3171892  0.071
18188
## Car             0.2668261  0.41852066  0.31718916  1.0000000  0.122
```

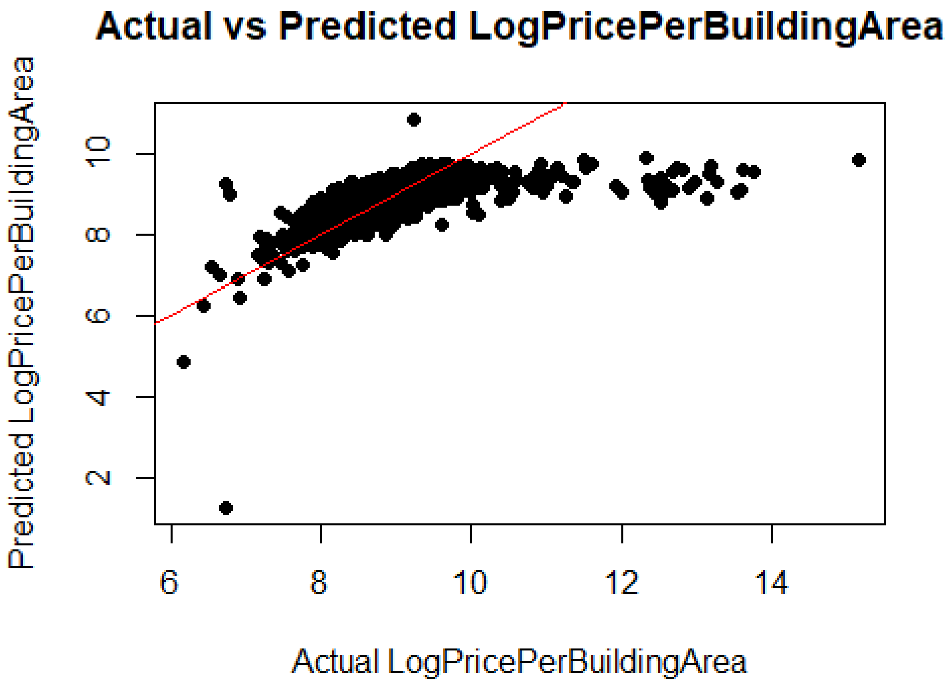
```
62196
## Landsize         0.1207248  0.09491182  0.07118188  0.1226220  1.000
00000
## BuildingArea     0.1452202  0.59721723  0.55463622  0.3219861  0.077
93123
## YearsAfterBuilt -0.3050505 -0.01601471 -0.18987369 -0.1371532 -0.039
93164
##                 BuildingArea YearsAfterBuilt
## Distance          0.14522024     -0.30505052
## Bedroom2          0.59721723     -0.01601471
## Bathroom          0.55463622     -0.18987369
## Car               0.32198611     -0.13715316
## Landsize          0.07793123     -0.03993164
## BuildingArea      1.00000000     -0.06529981
## YearsAfterBuilt  -0.06529981      1.00000000

# Create a correlation plot with color
corrplot(correlation_matrix, method = "color")
```



Actual vs predicted graph

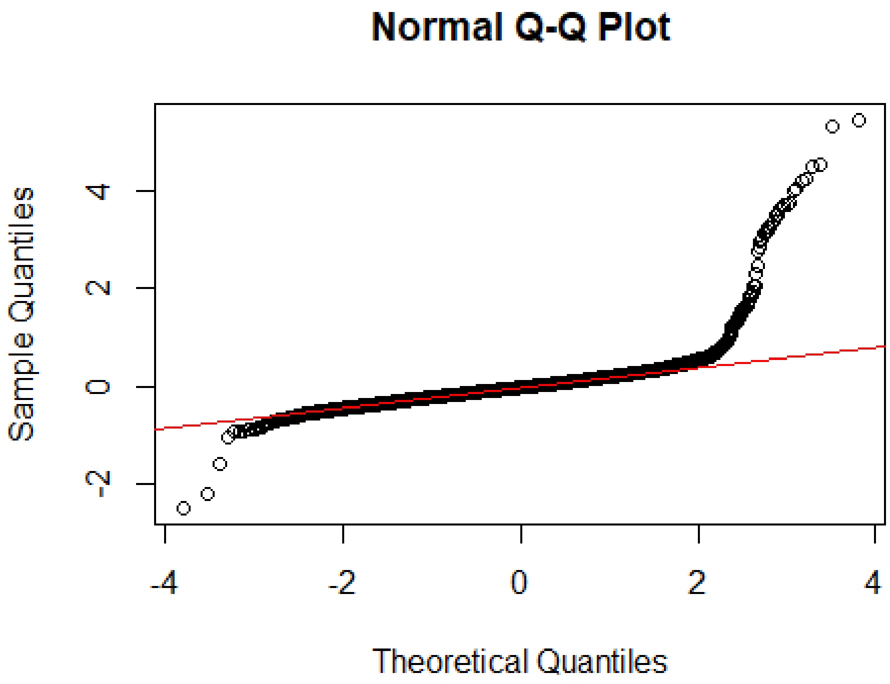
```
# Calculate predicted values
predicted_values <- predict(final_model, train_data)

# Extract actual values
actual_values <- train_data$LogPricePerBuildingArea
```

```
# Create a scatter plot of actual vs predicted values
plot(actual_values, predicted_values,
     main = "Actual vs Predicted LogPricePerBuildingArea",
     xlab = "Actual LogPricePerBuildingArea",
     ylab = "Predicted LogPricePerBuildingArea",
     pch = 19)  # pch = 19 makes the points solid

# Add a line of perfect fit for reference
abline(a = 0, b = 1, col = "red")
```

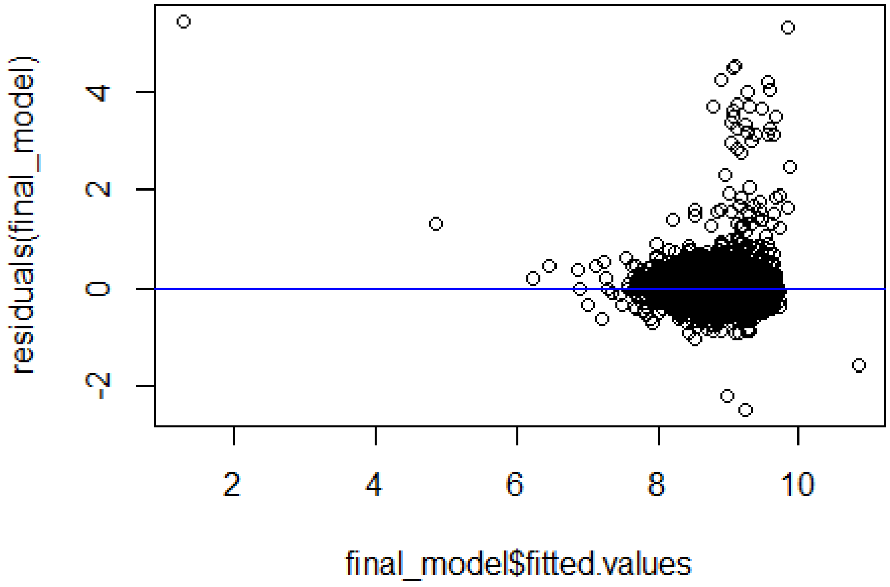


# qq and residual plots

```
# QQ plot for the first model
qqnorm(residuals(final_model))
qqline(residuals(final_model), col = "red")
```



```r
# Residual plot for the first model
plot(final_model$fitted.values, residuals(final_model))
abline(h = 0, col = "blue")
```



Comment of results

Comment of business implications