Final Data Analysis - Capstone Project. Statistics with R. Duke University

Author: Jorge Álvarez de la Fuente

Background

As a statistical consultant working for a real estate investment firm, your task is to develop a model to predict the selling price of a given home in Ames, Iowa. Your employer hopes to use this information to help assess whether the asking price of a house is higher or lower than the true value of the house. If the home is undervalued, it may be a good investment for the firm.

Training Data and relevant packages

In order to better assess the quality of the model you will produce, the data have been randomly divided into three separate pieces: a training data set, a testing data set, and a validation data set. For now we will load the training data set, the others will be loaded and used later.

```
load("ames_train.Rdata")
```

Use the code block below to load any necessary packages

```
library(statsr)
```

```
## Warning in as.POSIXlt.POSIXct(Sys.time()): unable to identify current timezone 'H':
## please set environment variable 'TZ'
```

```
library(dplyr)
library(BAS)
library(pander)
library(MASS)
library(ggplot2)
library(knitr)
library(GGally)
library(gridExtra)
```

Part 1 - Exploratory Data Analysis (EDA)

When you first get your data, it's very tempting to immediately begin fitting models and assessing how they perform. However, before you begin modeling, it's absolutely essential to explore the structure of the data and the relationships between the variables in the data set.

Do a detailed EDA of the ames_train data set, to learn about the structure of the data and the relationships between the variables in the data set (refer to Introduction to Probability and Data, Week 2, for a reminder about EDA if needed). Your EDA should involve creating and reviewing many plots/graphs and considering the patterns and relationships you see.

After you have explored completely, submit the three graphs/plots that you found most informative during your EDA process, and briefly explain what you learned from each (why you found each informative).

1.1. Studying the distribution of the data set ames_train

The first step in the Exploratory Data Analysis is to find the dimensions and the summary statistics of the ames_train data set

```
# This code gets the dimensions of the ames_train data set
dim(ames_train)
```

```
## [1] 1000   81
```

The summary statistics of the ames_train data set is presented below:

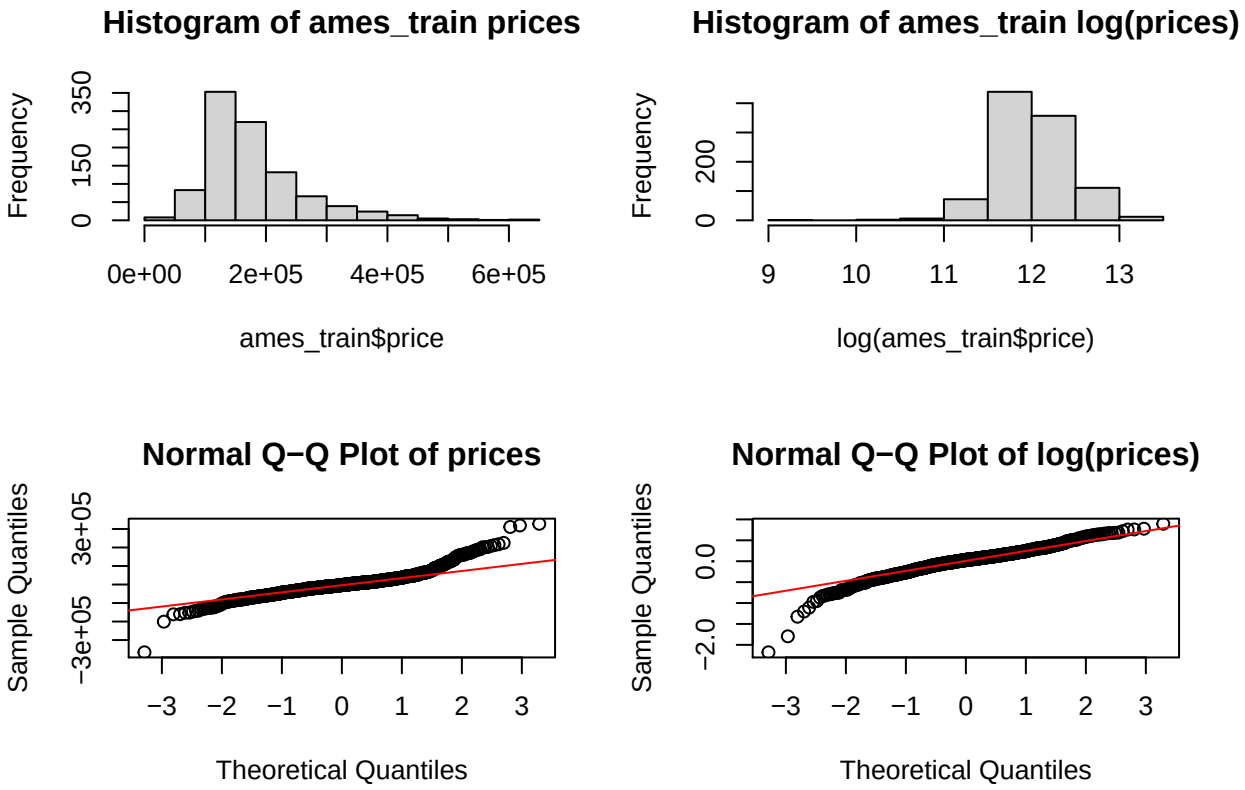
```
# This code gets the summary statistics and skewness of the ames_train data set
ames_train %>%
  summarize(Q1=quantile(price, 0.25), MEAN=mean(price), MEDIAN=median(price),
            Q3=quantile(price, 0.75), IQR=IQR(price), ST_DEVIATION=sd(price)) %>%
  mutate(SKEWNESS=ifelse(MEAN > MEDIAN, "RIGHT", "LEFT")) %>%
  pandoc.table
```

```
##
## ------------------------------------------------------------------------
##    Q1      MEAN    MEDIAN     Q3      IQR    ST_DEVIATION   SKEWNESS
## -------- -------- -------- -------- ------- -------------- ----------
##  129763   181190   159467   213000   83238      81910        RIGHT
## ------------------------------------------------------------------------
```

The summary statistics results show that the data is right skewed given the mean (181190) is larger than the median (159467).

Here below is shown the distributions and the linearity of the relationships between the quantitative explanatory variable `area` and the response variable `price`

```
# This code arranges the graphs in 2 columns and 2 rows
par(mfrow=c(2,2))

# This code plots histograms for price
hist(ames_train$price, main="Histogram of ames_train prices")
hist(log(ames_train$price), main="Histogram of ames_train log(prices)")

# This code builds the linear model area by prices
area_price_lm <- lm(formula=price ~ area, data=ames_train)
area_price_lm_log <- lm(formula=log(price) ~ area, data=ames_train)

# This code plots the normal plots
qqnorm(area_price_lm$residuals, main="Normal Q-Q Plot of prices")
qqline(area_price_lm$residuals, col="red")

qqnorm(area_price_lm_log$residuals, main="Normal Q-Q Plot of log(prices)")
qqline(area_price_lm_log$residuals, col="red")
```



The top row of plots, referred to histograms, confirms a long tail to the right, being a right skewed distribution (left figure). This is likely caused by a small number of houses being significantly more expensive than most of the other house in ames_train data set. By log-transforming this data the distribution looks more like a normal distribution (right figure).

The normal Q-Q plot of the log-transformed price (right figure) is more linear than the normal Q-Q plot of the raw price (left figure), therefore, the data is distributed more like a normal distribution with the log-transformed price.

1.2. Relationship between Overall Quality and Price

A strong relationship in ames_train data set can be found among the explanatory variable `Overall Quality` and the response variable `price`. Here below is shown the summary statistics of house prices by overall quality:

```
# This code gets the summary statistics of the response variable 'price' sorted by 'Overall Quality'
ames_train %>%
  group_by(Overall.Qual) %>%
  summarise(Q1=quantile(price, 0.25), MEAN=mean(price), MEDIAN=median(price),
            Q3=quantile(price, 0.75), IQR=IQR(price), STDEV=sd(price), .groups="drop") %>%
  mutate(SKEW=ifelse(MEAN > MEDIAN, "RIGHT", "LEFT")) %>%
  pandoc.table
```

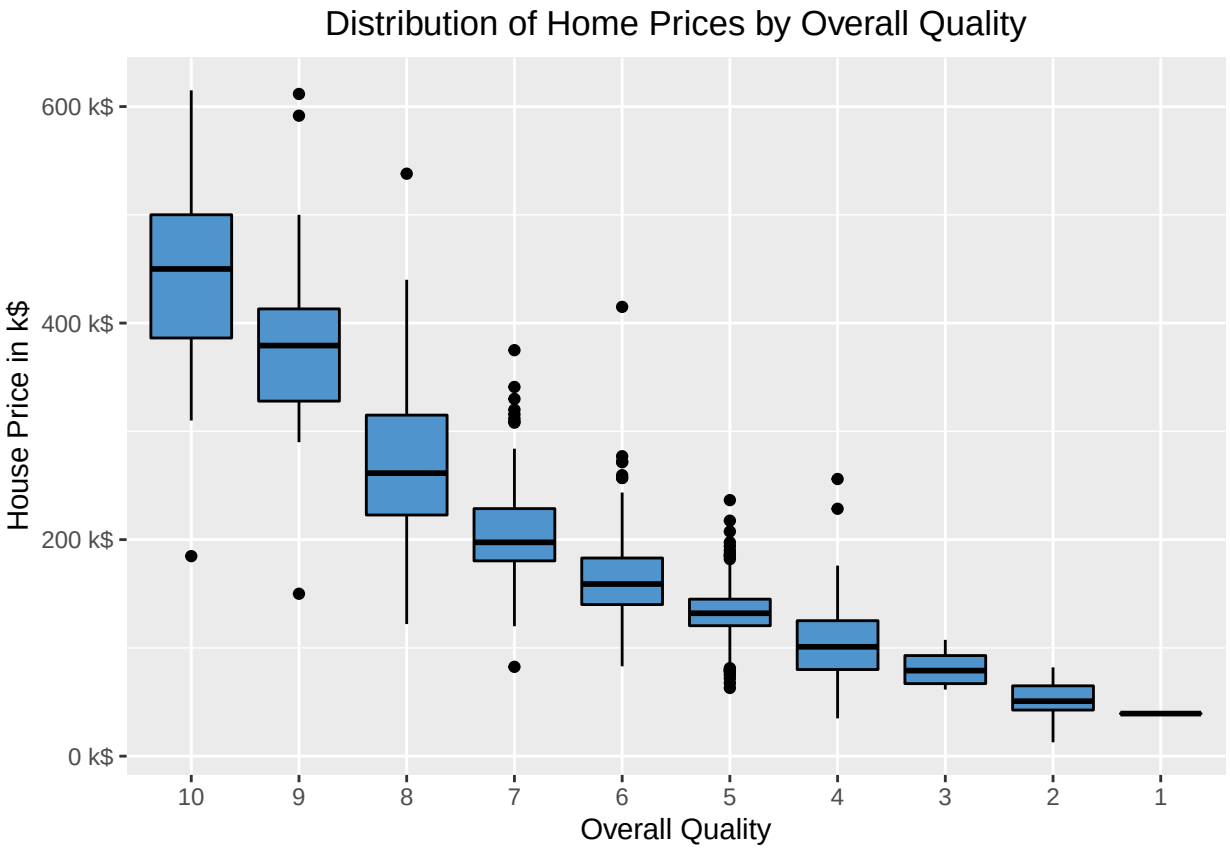
```
##
## ---------------------------------------------------------------------------
##  Overall.Qual     Q1      MEAN    MEDIAN     Q3      IQR     STDEV    SKEW
```

```
## -------------- -------- -------- -------- -------- -------- -------- -------
##       1         39300    39300    39300    39300      0        NA     LEFT
##
##       2         42578    51076    50750    64951    22373    21376    RIGHT
##
##       3         67000    81233    79000    92900    25900    16074    RIGHT
##
##       4         80000    106044   101000   125125   45125    37116    RIGHT
##
##       5         120500   132812   132000   145000   24500    24000    RIGHT
##
##       6         140000   163069   159000   183000   43000    40373    RIGHT
##
##       7         180375   205505   197500   228625   48250    43162    RIGHT
##
##       8         222725   272502   261415   315000   92275    68499    RIGHT
##
##       9         327925   382016   379250   413125   85200    81243    RIGHT
##
##       10        386250   428341   450000   500067   113817   126364   LEFT
## -------------------------------------------------------------------------------
```

The summary statistics show a relationship between `price` and `Overall Quality`. It can be observed that the highest the Overall Quality, the highest is the house price, with highest Q1, mean, median, Q3, IQR, standard deviation.

Here below the distribution of home prices by overall quality is presented:

```
# This code plots a boxplot of housing prices by overall quality in descending order by price
ggplot(data=ames_train, aes(x=reorder(Overall.Qual, -price), y=price)) +
  geom_boxplot(colour='black', fill='steelblue3') +
  ggtitle("Distribution of Home Prices by Overall Quality") +
  xlab("Overall Quality") + ylab("House Price in k$") +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  theme(plot.title=element_text(hjust=0.5))
```

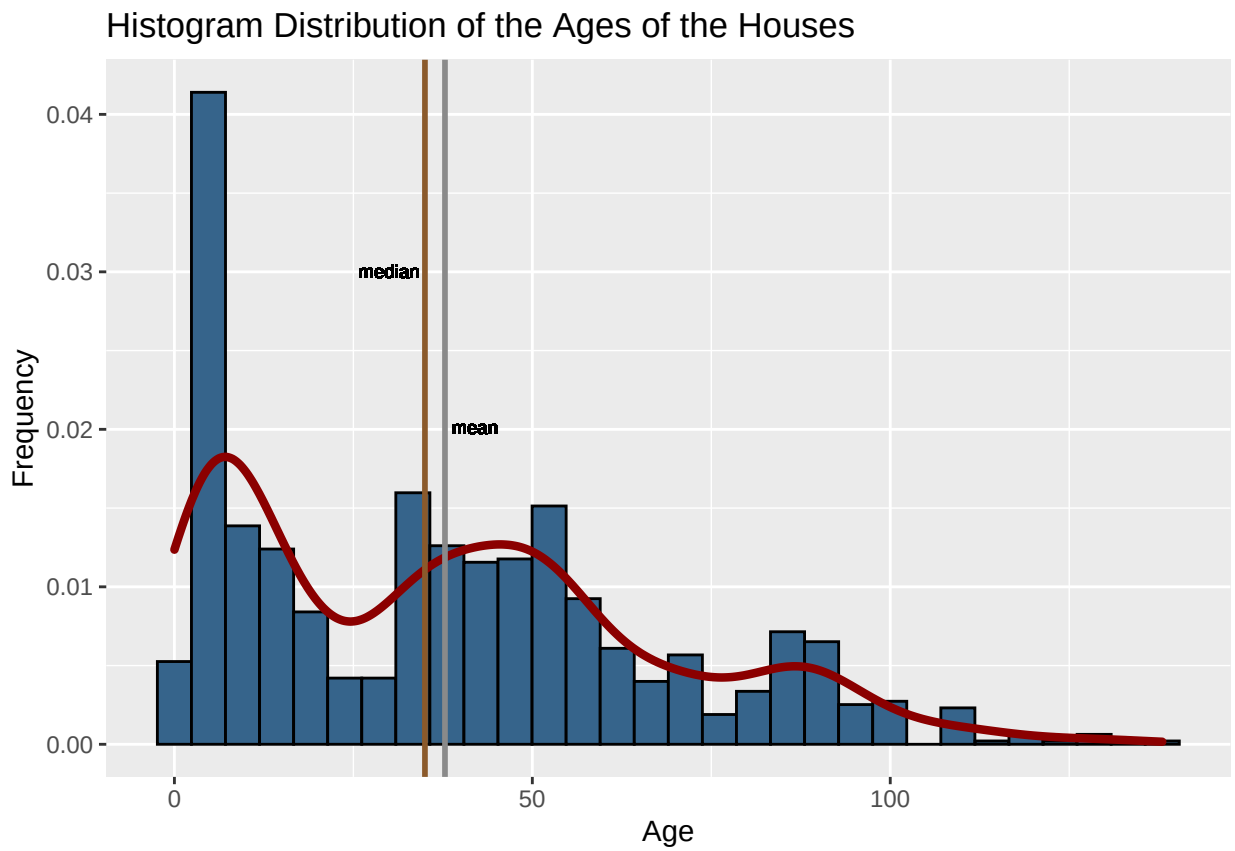


From the results of the boxplot obtained above, it can be drawn that many houses score between 4 and 10 as per Overall Quality metric. The model may vary its accuracy of predictions based on the quality level as there is not a balanced number of houses for each quality level. Other explanatory variables may represent such relationships making the final model to perform differently depending on whether the predicted house price has a more extreme value for an explanatory variable.

1.3. The distribution of house ages

Here below it can be found a labeled histogram of the ages of the houses in ames_train data set

```
# This code takes a data frame as input and gives output in vector or matrix
ames_train$age <- sapply(ames_train$Year.Built, function(x) 2010 - x)

# This code plots a histogram distribution and adds vertical lines for mean and median of ames_train da
ggplot(data=ames_train, aes(x=age, y=..density..)) +
  geom_histogram(bins=30, fill='steelblue4', colour='black') +
  geom_density(size=1.5, colour='darkred') +
  ggtitle("Histogram Distribution of the Ages of the Houses") + xlab("Age") + ylab("Frequency") +
  geom_vline(xintercept=mean(ames_train$age), colour='grey54', size=1) +
  geom_vline(xintercept=median(ames_train$age), colour='tan4', size=1) +
  geom_text(data=ames_train, aes(x=42, y=0.02, label="mean"), size=2.5, color="black", parse=T) +
  geom_text(data=ames_train, aes(x=30, y=0.03, label="median"), size=2.5, color="black", parse=T)
```



The summary statistics is presented below:

```
# This code finds the summary statistics for the variable age (Mean, Median, IQR, number of observation
ames_train %>%
  summarise(Mean_age = mean(ames_train$age), Median_age = median(ames_train$age),
            IQR = IQR(ames_train$age), Total = n())
```

```
## # A tibble: 1 x 4
##   Mean_age Median_age   IQR Total
##      <dbl>      <dbl> <dbl> <int>
## 1     37.8         35    46  1000
```

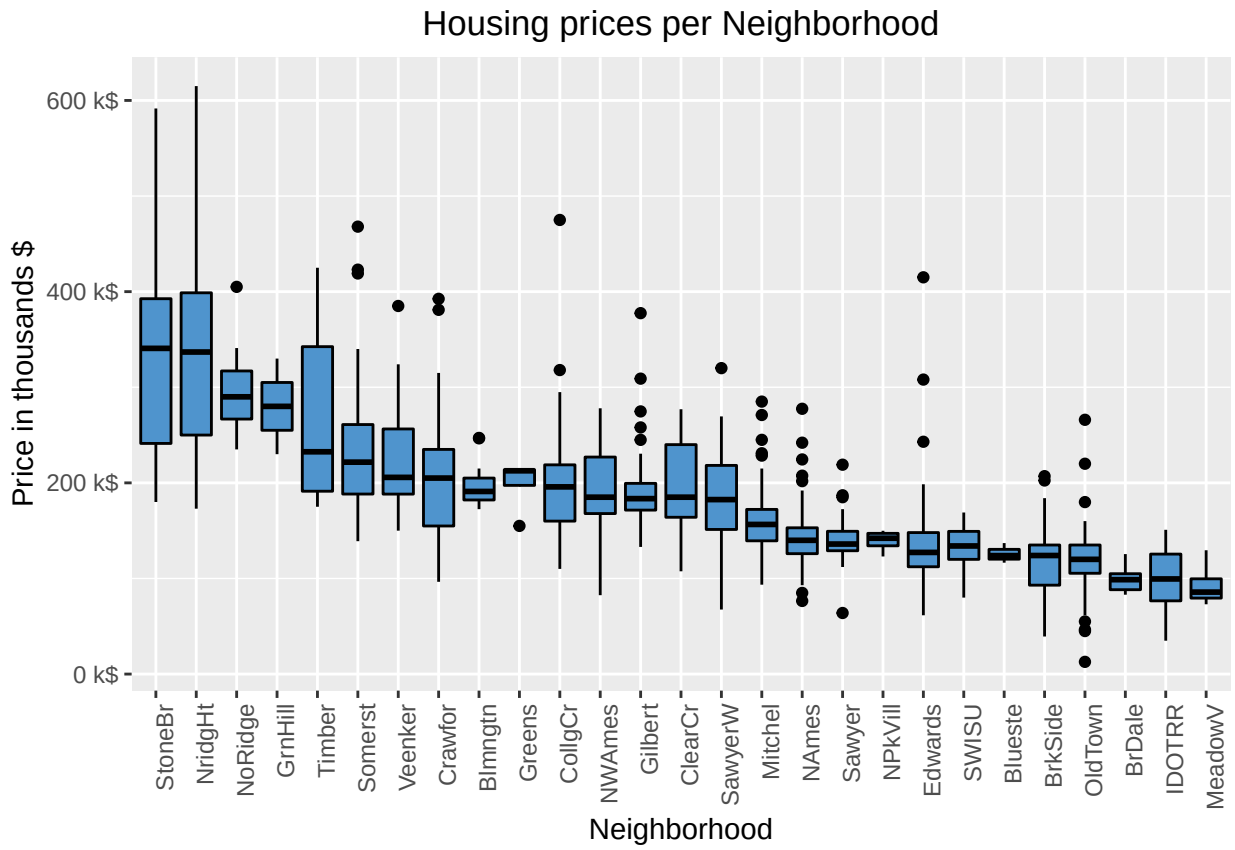
The distribution is right-skewed as it is observed there are many new houses (aged below 25) compared to old houses (aged above 25). The mean age is roughly 38 and the median is 35. The distribution is multimodal as a result of high frequencies of houses at certain ages. The IQR has a value of 46 suggesting there is a significant variability as IQR (46) is larger than Median (35).

1.4. The distribution of housing prices per neighborhood

A boxplot providing housing prices per neighborhood is shown below.

```
# This code plots a boxplot of housing prices per neighborhood in descending order by price
ggplot(data=ames_train, aes(x=reorder(Neighborhood, -price), y=price)) +
  geom_boxplot(colour='black', fill='steelblue3') +
  ggtitle("Housing prices per Neighborhood") + xlab("Neighborhood") + ylab("Price in thousands $") +
```

```
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  theme(plot.title=element_text(hjust=0.5), axis.text.x=element_text(angle=90, hjust=1))
```



Based on the results above, the most expensive neighborhood in Iowa is StoneBr and the least expensive neighborhood is MeadowV

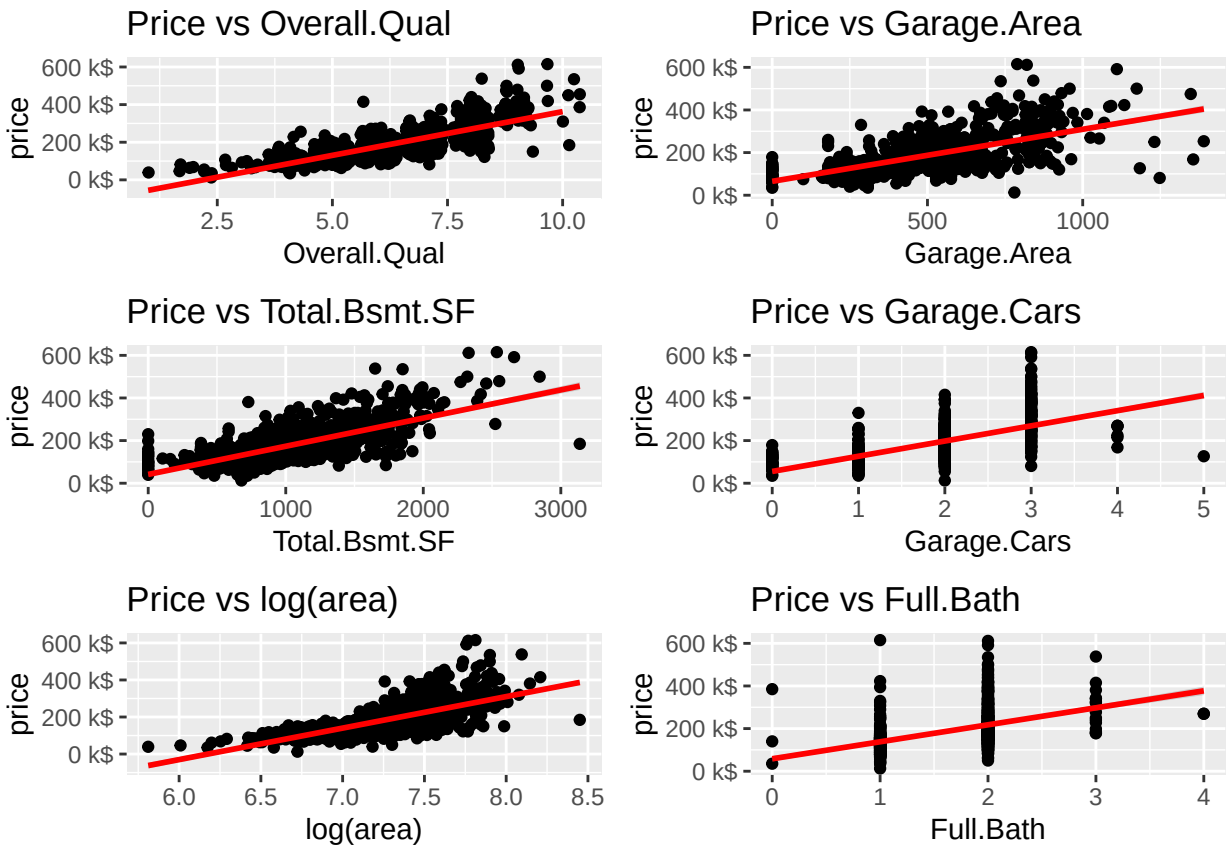
1.5. Relationship between explanatory variables and price

The following plots will give further information regarding the effect of the explanatory variables: Overall.Qual, Garage.Area, Total.Bsmt.SF, Garage.Cars, log(area), Full.Bath, Bedroom.AbvGr, Year.Built, X1st.Flr.SF, Lot.Area in the housing prices in Ames.

```
# Plot Price vs Overall.Qual
p211 <- ggplot(ames_train, aes(x = Overall.Qual, y = price)) +
  geom_jitter() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Overall.Qual") +
  stat_smooth(method = 'lm', color='red')

# Plot Price vs Garage.Area
p212 <- ggplot(ames_train, aes(x = Garage.Area, y = price)) +
  geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Garage.Area") +
  stat_smooth(method = 'lm', color='red')
```

```r
# Plot Price vs Total.Bsmt.SF
p213 <- ggplot(ames_train, aes(x = Total.Bsmt.SF, y = price)) +
  geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Total.Bsmt.SF") +
  stat_smooth(method = 'lm', color='red')

# Plot Price vs Garage.Cars
p214 <- ggplot(ames_train, aes(x = Garage.Cars, y = price)) +
  geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Garage.Cars") +
  stat_smooth(method = 'lm', color='red')

# Plot Price vs log(area)
p215 <- ggplot(ames_train, aes(x = log(area), y = price)) +
  geom_jitter() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs log(area)") +
  stat_smooth(method = 'lm', color='red')

# Plot Price vs Full.Bath
p216 <- ggplot(ames_train, aes(x = Full.Bath, y = price)) +
  geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Full.Bath") +
  stat_smooth(method = 'lm', color='red')

grid.arrange(p211, p212, p213, p214, p215, p216, ncol = 2)
```

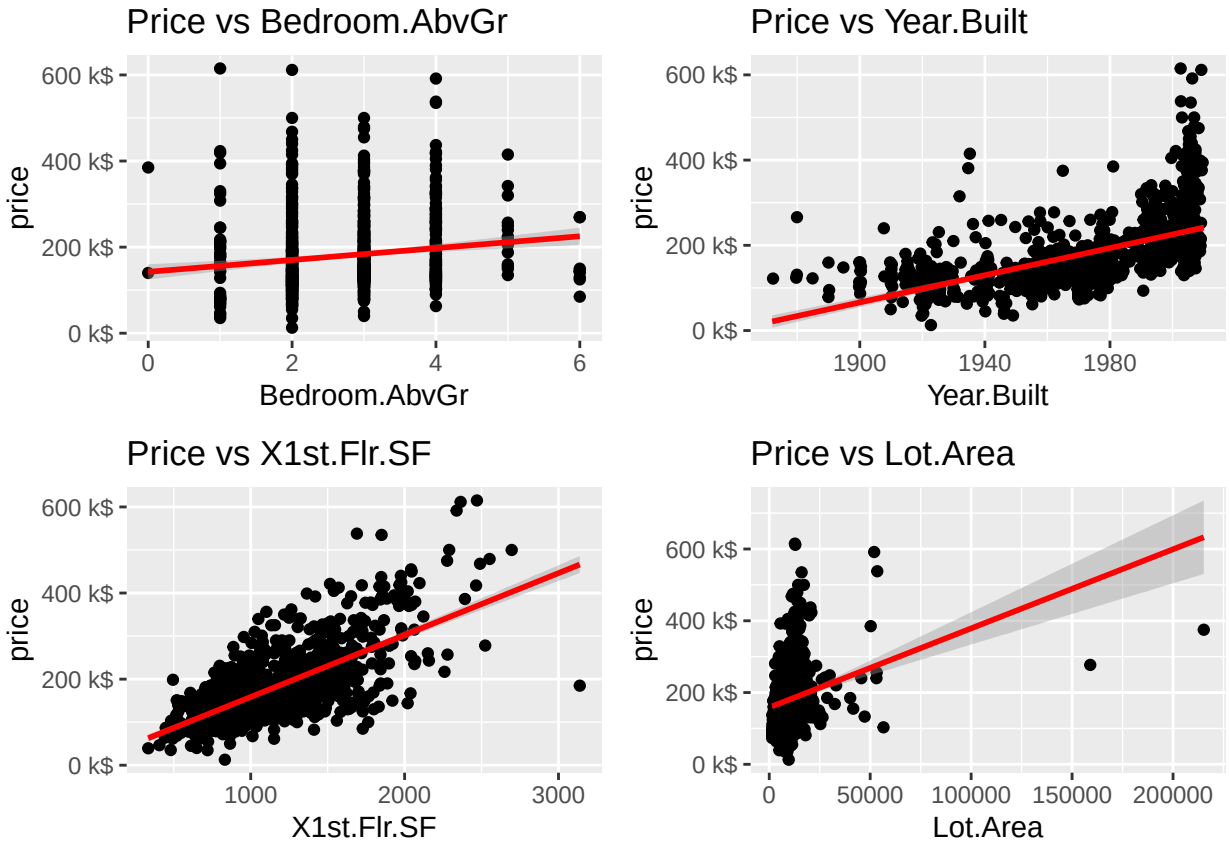
```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).

## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).

## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).

## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```




```r
# Plot Price vs Bedroom.AbvGr
p217 <- ggplot(ames_train, aes(x = Bedroom.AbvGr, y = price)) +
  geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Bedroom.AbvGr") +
  stat_smooth(method = 'lm', color='red')

# Plot Price vs Year.Built
p218 <- ggplot(ames_train, aes(x = Year.Built, y = price)) +
  geom_jitter() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Year.Built") +
  stat_smooth(method = 'lm', color='red')

# Plot Price vs X1st.Flr.SF
p219 <- ggplot(ames_train, aes(x = X1st.Flr.SF, y = price)) +
  geom_jitter() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs X1st.Flr.SF") +
  stat_smooth(method = 'lm',color='red')

# Plot Price vs Lot.Area
p2110 <- ggplot(ames_train, aes(x = Lot.Area, y = price)) +
  geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs Lot.Area") +
```

```
  stat_smooth(method = 'lm', color='red')

grid.arrange(p217, p218, p219, p2110, ncol = 2)
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



From the plots above, Overall.Qual looks the best predictor of housing price, followed by Total.Bsmt.SF, Garage.Area and log(area).

Part 2 - Development and assessment of an initial model, following a semi-guided process of analysis

Section 2.1 An Initial Model

In building a model, it is often useful to start by creating a simple, intuitive initial model based on the results of the exploratory data analysis. (Note: The goal at this stage is **not** to identify the "best" possible model but rather to choose a reasonable and understandable starting point. Later you will expand and revise this model to create your final model.

Based on your EDA, select *at most* 10 predictor variables from "ames_train" and create a linear model for `price` (or a transformed version of price) using those variables. Provide the *R code* and the *summary output table* for your model, a *brief justification* for the variables you have chosen, and a *brief discussion* of the model results in context (focused on the variables that appear to be important predictors and how they relate to sales price).

2.1.1. The initial model Using the results from the previous section the response variable log(price) will be predicted by using the following explanatory variables: Overall.Qual, Garage.Area, Total.Bsmt.SF, Garage.Cars, log(area), Full.Bath, Bedroom.AbvGr, Year.Built, X1st.Flr.SF, Lot.Area. The summary statistics of this initial model can be found down below:

```
# This code gets the summary statistics and coefficients of the initial model
initial_model <- lm(log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars
                    + log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF
                    + Lot.Area, data = ames_train)

summary(initial_model)
```

```
##
## Call:
## lm(formula = log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF +
##     Garage.Cars + log(area) + Full.Bath + Bedroom.AbvGr + Year.Built +
##     X1st.Flr.SF + Lot.Area, data = ames_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.76458 -0.07978  0.01241  0.09555  0.52496
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.122e+00  5.769e-01   3.679 0.000247 ***
## Overall.Qual   1.047e-01  6.484e-03  16.145  < 2e-16 ***
## Garage.Area    8.093e-05  5.808e-05   1.393 0.163802
## Total.Bsmt.SF  1.331e-04  2.458e-05   5.414 7.74e-08 ***
## Garage.Cars    2.995e-02  1.750e-02   1.711 0.087439 .
## log(area)      4.454e-01  3.261e-02  13.657  < 2e-16 ***
## Full.Bath     -4.869e-02  1.465e-02  -3.324 0.000920 ***
## Bedroom.AbvGr -1.979e-02  8.571e-03  -2.309 0.021130 *
## Year.Built     2.967e-03  2.621e-04  11.318  < 2e-16 ***
## X1st.Flr.SF    4.189e-05  2.808e-05   1.492 0.136115
## Lot.Area       3.011e-06  5.821e-07   5.173 2.79e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1692 on 987 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.8396, Adjusted R-squared:  0.838
## F-statistic: 516.6 on 10 and 987 DF,  p-value: < 2.2e-16
```

- Overall.Qual: This is the first feature people will look at when choosing a house or deciding to buy a house. Coefficient Value = 1.047e-01.

- Garage.Area: Coefficient Value = 8.093e-05.
- Total.Bsmt.SF: Coefficient Value = 1.331e-04.
- garage.Cars: Coefficient Value = 2.995e-02.
- log(area): A house with large area is likely to cost more than a house with a smaller area. Coefficient Value = 4.454e-01.
- Full.Bath: Coefficient Value = -4.869e-02.
- Bedroom.AbvGr: The number of bedrooms is an important factor to decide whether to buy a house or not. Coefficient Value = -1.979e-02.
- Year.Built: OVerall, a new house is more expensive than an older house. Coefficient Value = 2.967e-03.
- X1st.Flr.SF: Coefficient Value = 4.189e-05.
- Lot.Area: The position of the house will have an impact on the price. Coefficient value = 3.011e-06

The Multiple R-Squared Value is 0.8396 which indicates a strong linear relationship and the Adjusted R-Squared value is 0.838, very close to the value obtained for the Multiple R-Squared so no big penalty is observed by adding new variables to the model.

Section 2.2 Model Selection

Now either using `BAS` another stepwise selection procedure choose the "best" model you can, using your initial model as your starting point. Try at least two different model selection methods and compare their results. Do they both arrive at the same model or do they disagree? What do you think this means?

2.2.1. Model selection using Akaike Information Criterion (AIC) Akaike Information Criterion (AIC) is a method for scoring and selecting a model and compares the quality of a set of statistical models to each other model. AIC will be used in this section to choose the best model possible and to compare different model selection methods.

```
# This code applies the AIC model selection method to the initial model
initial_model_AIC <- stepAIC(initial_model, direction='backward')
```

```
## Start:  AIC=-3535.14
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area
##
##                 Df Sum of Sq    RSS     AIC
## - Garage.Area    1    0.0556 28.317 -3535.2
## <none>                       28.261 -3535.1
## - X1st.Flr.SF    1    0.0637 28.325 -3534.9
## - Garage.Cars    1    0.0838 28.345 -3534.2
## - Bedroom.AbvGr  1    0.1527 28.414 -3531.8
## - Full.Bath      1    0.3164 28.577 -3526.0
## - Lot.Area       1    0.7662 29.027 -3510.4
## - Total.Bsmt.SF  1    0.8393 29.100 -3507.9
## - Year.Built     1    3.6681 31.929 -3415.4
## - log(area)      1    5.3407 33.602 -3364.4
## - Overall.Qual   1    7.4631 35.724 -3303.3
```

```
##
## Step:  AIC=-3535.18
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF + Lot.Area
##
##                 Df Sum of Sq    RSS     AIC
## <none>                       28.317 -3535.2
## - X1st.Flr.SF    1    0.0796 28.396 -3534.4
## - Bedroom.AbvGr  1    0.1548 28.471 -3531.7
## - Full.Bath      1    0.3632 28.680 -3524.5
## - Garage.Cars    1    0.7223 29.039 -3512.0
## - Lot.Area       1    0.7761 29.093 -3510.2
## - Total.Bsmt.SF  1    0.8563 29.173 -3507.5
## - Year.Built     1    3.7019 32.018 -3414.6
## - log(area)      1    5.3694 33.686 -3363.9
## - Overall.Qual   1    7.4338 35.750 -3304.5
```

```
summary(initial_model_AIC)
```

```
##
## Call:
## lm(formula = log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area, data = ames_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.74417 -0.07983  0.01021  0.09528  0.54915
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.093e+00  5.768e-01   3.630 0.000298 ***
## Overall.Qual   1.044e-01  6.485e-03  16.105  < 2e-16 ***
## Total.Bsmt.SF  1.343e-04  2.458e-05   5.466 5.83e-08 ***
## Garage.Cars    5.001e-02  9.962e-03   5.020 6.13e-07 ***
## log(area)      4.465e-01  3.262e-02  13.687  < 2e-16 ***
## Full.Bath     -5.163e-02  1.450e-02  -3.560 0.000389 ***
## Bedroom.AbvGr -1.993e-02  8.575e-03  -2.324 0.020325 *
## Year.Built     2.979e-03  2.621e-04  11.365  < 2e-16 ***
## X1st.Flr.SF    4.650e-05  2.790e-05   1.667 0.095855 .
## Lot.Area       3.030e-06  5.823e-07   5.204 2.38e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1693 on 988 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.8393, Adjusted R-squared:  0.8378
## F-statistic: 573.2 on 9 and 988 DF,  p-value: < 2.2e-16
```

```
initial_model_AIC$anova
```

```
## Stepwise Model Path
## Analysis of Deviance Table
```

```
##
## Initial Model:
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area
##
## Final Model:
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF + Lot.Area
##
##
##             Step Df   Deviance Resid. Df Resid. Dev       AIC
## 1                                    987   28.26094 -3535.144
## 2 - Garage.Area  1 0.05559589       988   28.31654 -3535.183
```

2.2.2. Model selection using Bayesian Information Criterion (BIC) The Bayesian Information Criterion (BIC) is an index used in Bayesian statistics to choose between two or more alternative models. BIC will be used in this section to choose the best model possible and to compare different model selection methods.

```
# This code applies the BIC model selection method to the initial model
initial_model_BIC <- stepAIC(initial_model, direction='backward', k=log(nrow(ames_train)))
```

```
## Start:  AIC=-3481.16
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area
##
##                 Df Sum of Sq    RSS     AIC
## - Garage.Area    1    0.0556 28.317 -3486.1
## - X1st.Flr.SF    1    0.0637 28.325 -3485.8
## - Garage.Cars    1    0.0838 28.345 -3485.1
## - Bedroom.AbvGr  1    0.1527 28.414 -3482.7
## <none>                       28.261 -3481.2
## - Full.Bath      1    0.3164 28.577 -3477.0
## - Lot.Area       1    0.7662 29.027 -3461.4
## - Total.Bsmt.SF  1    0.8393 29.100 -3458.9
## - Year.Built     1    3.6681 31.929 -3366.3
## - log(area)      1    5.3407 33.602 -3315.3
## - Overall.Qual   1    7.4631 35.724 -3254.2
##
## Step:  AIC=-3486.11
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF + Lot.Area
##
##                 Df Sum of Sq    RSS     AIC
## - X1st.Flr.SF    1    0.0796 28.396 -3490.2
## - Bedroom.AbvGr  1    0.1548 28.471 -3487.6
## <none>                       28.317 -3486.1
## - Full.Bath      1    0.3632 28.680 -3480.3
## - Garage.Cars    1    0.7223 29.039 -3467.9
## - Lot.Area       1    0.7761 29.093 -3466.0
## - Total.Bsmt.SF  1    0.8563 29.173 -3463.3
```

```
## - Year.Built     1    3.7019 32.018 -3370.4
## - log(area)      1    5.3694 33.686 -3319.7
## - Overall.Qual   1    7.4338 35.750 -3260.4
##
## Step:  AIC=-3490.21
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Bedroom.AbvGr + Year.Built + Lot.Area
##
##                 Df Sum of Sq    RSS     AIC
## - Bedroom.AbvGr  1    0.1880 28.584 -3490.5
## <none>                       28.396 -3490.2
## - Full.Bath      1    0.3659 28.762 -3484.3
## - Garage.Cars    1    0.7652 29.161 -3470.6
## - Lot.Area       1    0.8237 29.220 -3468.6
## - Total.Bsmt.SF  1    2.9264 31.323 -3399.2
## - Year.Built     1    3.6348 32.031 -3376.9
## - log(area)      1    6.3332 34.729 -3296.2
## - Overall.Qual   1    7.3572 35.753 -3267.2
##
## Step:  AIC=-3490.53
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Year.Built + Lot.Area
##
##                 Df Sum of Sq    RSS     AIC
## <none>                       28.584 -3490.5
## - Full.Bath      1    0.4341 29.018 -3482.4
## - Lot.Area       1    0.8093 29.393 -3469.6
## - Garage.Cars    1    0.8599 29.444 -3467.9
## - Total.Bsmt.SF  1    3.0711 31.655 -3395.6
## - Year.Built     1    3.6726 32.257 -3376.8
## - log(area)      1    7.3092 35.893 -3270.2
## - Overall.Qual   1    8.6655 37.250 -3233.2
```

```
summary(initial_model_BIC)
```

```
##
## Call:
## lm(formula = log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Year.Built + Lot.Area, data = ames_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.73349 -0.08096  0.01166  0.09519  0.54051
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.271e+00  5.741e-01   3.956 8.18e-05 ***
## Overall.Qual   1.081e-01  6.239e-03  17.324  < 2e-16 ***
## Total.Bsmt.SF  1.684e-04  1.633e-05  10.313  < 2e-16 ***
## Garage.Cars    5.407e-02  9.909e-03   5.457 6.11e-08 ***
## log(area)      4.201e-01  2.641e-02  15.911  < 2e-16 ***
## Full.Bath     -5.607e-02  1.446e-02  -3.877 0.000113 ***
## Year.Built     2.955e-03  2.620e-04  11.278  < 2e-16 ***
## Lot.Area       3.083e-06  5.823e-07   5.294 1.47e-07 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1699 on 990 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.8377, Adjusted R-squared:  0.8366
## F-statistic: 730.2 on 7 and 990 DF,  p-value: < 2.2e-16
```

```
initial_model_BIC$anova
```

```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area
##
## Final Model:
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Year.Built + Lot.Area
##
##
##              Step Df   Deviance Resid. Df Resid. Dev       AIC
## 1                                    987   28.26094 -3481.159
## 2   - Garage.Area  1 0.05559589       988   28.31654 -3486.105
## 3   - X1st.Flr.SF  1 0.07963295       989   28.39617 -3490.210
## 4 - Bedroom.AbvGr  1 0.18795560       990   28.58413 -3490.534
```

2.2.3. Model selection comparison The initial and final models for AIC and BIC selection procedures are summarized below:

```
# AIC Initial model:
log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars + log(area) + Full.Bath + Bedroom
```

```
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area
```

```
# AIC Final model:
log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) + Full.Bath + Bedroom.AbvGr + Year.
```

```
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF + Lot.Area
```

```
# BIC Initial model:
log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars + log(area) + Full.Bath + Bedroom
```

```
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area
```

```
# BIC Final model:
log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) + Full.Bath + Year.Built + Lot.Area
```

```
## log(price) ~ Overall.Qual + Total.Bsmt.SF + Garage.Cars + log(area) +
##     Full.Bath + Year.Built + Lot.Area
```

As can be observed in the results above, both stepwise selection procedures AIC and BIC do not end up to the same model. The model using BIC selection procedure has a lower Adjusted R-Squared (0.8366) but with less predictors resulting in a more parsimonious model which is better to interpret the results and is a consistent estimation of the underlying data generating process. The model using AIC selection procedure has a higher Adjusted R-Squared (0.8378) but with more predictors, being the initial model with almost no changes. The AIC objective is met as is equivalent to cross-validation, but at the cost of not reaching the parsimonious model.

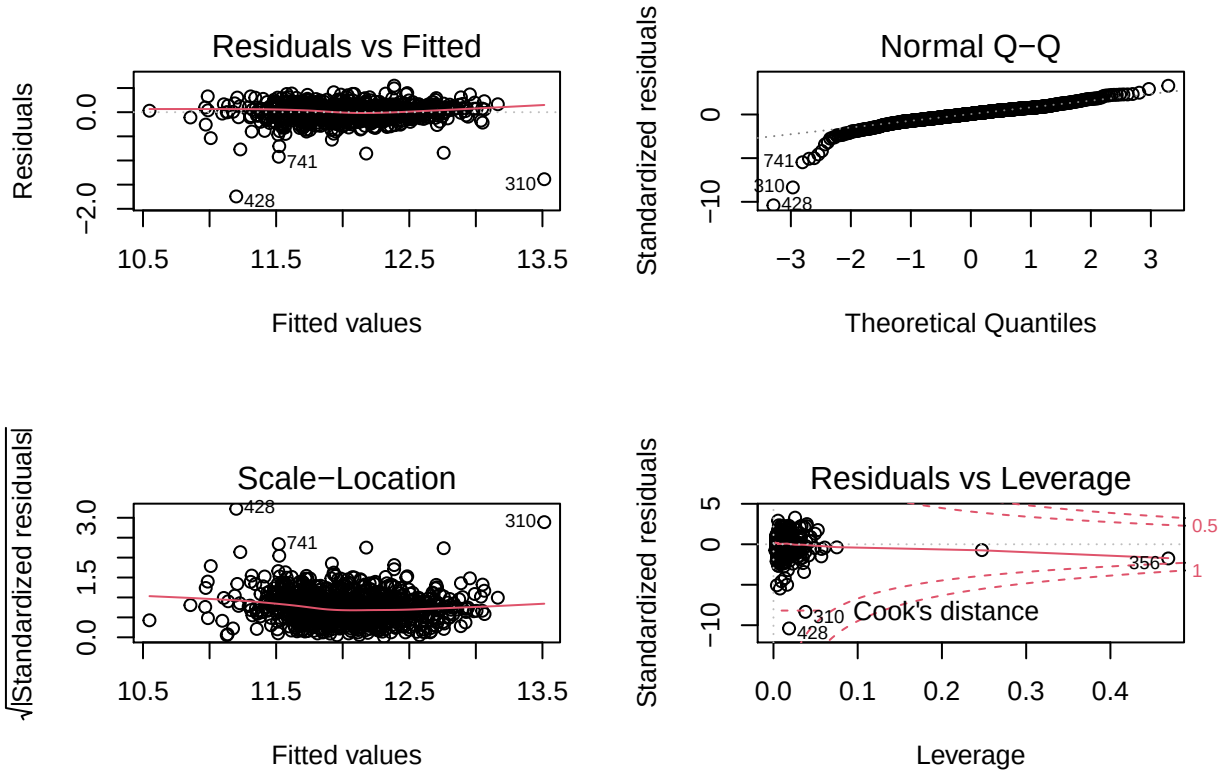
As the main objective of model selection is predicting the housing prices, the AIC final model will be selected because Adjusted R-Squared is higher than the Adjusted R-Squared got with the BIC final model.

Section 2.3 Initial Model Residuals

One way to assess the performance of a model is to examine the model's residuals. In the space below, create a residual plot for your preferred model from above and use it to assess whether your model appears to fit the data well. Comment on any interesting structure in the residual plot (trend, outliers, etc.) and briefly discuss potential implications it may have for your model and inference / prediction you might produce.

Down here, the initial model residuals plots for the selected selection procedure AIC are presented:

```
# This code plots Residuals and Normal Q-Q plots
par(mfrow=c(2,2))
plot(initial_model_AIC)
```



As can be observed from the plots obtained above, there is a constant variability of residuals with some high leverage outliers (rows 310, 428 and 356). This can be expected after using categorical variables to set up a regression model. Given the constant variability in residuals and the strong linearity as observed in Normal Q-Q plot, with a nearly normal distribution despite some outliers (rows 741, 310 and 428), it is expected a good accuracy in the model inference and predictions. Although long tails have been found, the sample is big enough to meet the Central Limit Theorem (CLT), then the distribution of the sample means will be approximately normal distributed.

Section 2.4 Initial Model RMSE

You can calculate it directly based on the model output. Be specific about the units of your RMSE (depending on whether you transformed your response variable). The value you report will be more meaningful if it is in the original units (dollars).

The RMSE (Root Mean Square Error) calculation and result of the initial model is present below:

```
# This code calculates the initial model RMSE in the training data set
pred_train <- exp(predict(initial_model_AIC, ames_train))
resid_train <- na.omit(ames_train$price - pred_train)
rmse_train <- sqrt(mean(resid_train^2))
paste('The RMSE for the within-sample initial model is', format(rmse_train, digits=6), 'dollars')
```

```
## [1] "The RMSE for the within-sample initial model is 33456.7 dollars"
```

The RMSE for the within-sample initial model is 33456.7 dollars.

Section 2.5 Overfitting

The process of building a model generally involves starting with an initial model (as you have done above), identifying its shortcomings, and adapting the model accordingly. This process may be repeated several times until the model fits the data reasonably well. However, the model may do well on training data but perform poorly out-of-sample (meaning, on a dataset other than the original training data) because the model is overly-tuned to specifically fit the training data. This is called "overfitting." To determine whether overfitting is occurring on a model, compare the performance of a model on both in-sample and out-of-sample data sets. To look at performance of your initial model on out-of-sample data, you will use the data set `ames_test`.

```
load("ames_test.Rdata")
```

Use your model from above to generate predictions for the housing prices in the test data set. Are the predictions significantly more accurate (compared to the actual sales prices) for the training data than the test data? Why or why not? Briefly explain how you determined that (what steps or processes did you use)?

The following code will load the testing data and calculate the RMSE of the ames_test data set

```
# This code calculates the RMSE of the ames_test data set in the initial model
pred_test <- exp(predict(initial_model_AIC, ames_test))
resid_test <- na.omit(ames_test$price - pred_test)
rmse_test <- sqrt(mean(resid_test^2))
paste('The RMSE for the out-of-sample initial model is', format(rmse_test, digits=6), 'dollars')
```

```
## [1] "The RMSE for the out-of-sample initial model is 25505.6 dollars"
```

The RMSE for the out-of-sample initial model is 25505.6 dollars.

The out-of-sample RMSE of the ames_test data set (25505.6 dollars) is lower than the within-sample RMSE of the ames_train data set (33456.7 dollars). Because the model is built on the training data, it would be expected that it will fit to the training data better than to the testing data. Because of these aforementioned reasons and because the difference in RMSE results is very close, overfitting is excluded.

Note to the learner: If in real-life practice this out-of-sample analysis shows evidence that the training data fits your model a lot better than the testing data, it is probably a good idea to go back and revise the model (usually by simplifying the model) to reduce this overfitting. For simplicity, we do not ask you to do this on the assignment, however.

Part 3 Development of a Final Model

Now that you have developed an initial model to use as a baseline, create a final model with *at most* 20 variables to predict housing prices in Ames, IA, selecting from the full array of variables in the dataset and using any of the tools that we introduced in this specialization.

Carefully document the process that you used to come up with your final model, so that you can answer the questions below.

Section 3.1 Final Model

Provide the summary table for your model.

Considering the initial model already have a good predictive accuracy given the Adjusted R-Squared and RMSE value obtained in the testing data, this section aims to improve its accuracy further by adding a few more predictors in order to discern between lower quality houses and higher quality houses. Thus, to the initial model, the following predictors will be added: Overall.Cond, Bsmt.Qual, Kitchen.Qual, Kitchen.AbvGr.

```
# This code builds the final model by adding to the initial model the proposed new variables
final_model <- lm(log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF +
                    Garage.Cars + log(area) + Full.Bath + Bedroom.AbvGr +
                    Year.Built + X1st.Flr.SF + Lot.Area + Overall.Cond +
                    Bsmt.Qual + Kitchen.Qual + Kitchen.AbvGr, data=ames_train)
```

The AIC model selection method will be applied to the final model using backward step selection to see whether the same variables still remain.

```
# This code applies the AIC model selection method to the final model
final_model_AIC <- stepAIC(final_model, direction='backward')
```

```
## Start:  AIC=-3700.22
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + X1st.Flr.SF +
##     Lot.Area + Overall.Cond + Bsmt.Qual + Kitchen.Qual + Kitchen.AbvGr
##
##                  Df Sum of Sq    RSS     AIC
## - X1st.Flr.SF     1    0.0366 21.240 -3700.5
## <none>                        21.203 -3700.2
## - Garage.Area     1    0.0451 21.249 -3700.1
## - Full.Bath       1    0.0858 21.289 -3698.3
## - Garage.Cars     1    0.0985 21.302 -3697.7
## - Kitchen.Qual    4    0.2342 21.438 -3697.5
## - Bedroom.AbvGr   1    0.1498 21.353 -3695.3
## - Kitchen.AbvGr   1    0.2318 21.435 -3691.6
## - Bsmt.Qual       4    0.4784 21.682 -3686.4
## - Total.Bsmt.SF   1    0.5451 21.749 -3677.4
## - Lot.Area        1    0.6792 21.883 -3671.4
## - Overall.Qual    1    2.4793 23.683 -3594.2
## - Year.Built      1    3.1025 24.306 -3568.8
## - Overall.Cond    1    4.8777 26.081 -3499.9
```

```
## - log(area)      1    5.7259 26.929 -3468.7
##
## Step:  AIC=-3700.54
## log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars +
##     log(area) + Full.Bath + Bedroom.AbvGr + Year.Built + Lot.Area +
##     Overall.Cond + Bsmt.Qual + Kitchen.Qual + Kitchen.AbvGr
##
##                 Df Sum of Sq    RSS     AIC
## <none>                       21.240 -3700.5
## - Garage.Area    1    0.0519 21.292 -3700.2
## - Full.Bath      1    0.0870 21.327 -3698.5
## - Garage.Cars    1    0.0977 21.338 -3698.1
## - Kitchen.Qual   4    0.2297 21.470 -3698.0
## - Bedroom.AbvGr  1    0.1738 21.414 -3694.6
## - Kitchen.AbvGr  1    0.2320 21.472 -3691.9
## - Bsmt.Qual      4    0.4722 21.712 -3687.1
## - Lot.Area       1    0.6970 21.937 -3671.0
## - Total.Bsmt.SF  1    2.3398 23.580 -3600.4
## - Overall.Qual   1    2.4541 23.694 -3595.7
## - Year.Built     1    3.0709 24.311 -3570.6
## - Overall.Cond   1    4.9999 26.240 -3496.0
## - log(area)      1    6.6166 27.857 -3437.6
```

```
final_model_AIC
```

```
##
## Call:
## lm(formula = log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF +
##     Garage.Cars + log(area) + Full.Bath + Bedroom.AbvGr + Year.Built +
##     Lot.Area + Overall.Cond + Bsmt.Qual + Kitchen.Qual + Kitchen.AbvGr,
##     data = ames_train)
##
## Coefficients:
##    (Intercept)    Overall.Qual     Garage.Area   Total.Bsmt.SF     Garage.Cars
##      5.328e-01       6.753e-02       7.932e-05       1.746e-04       3.319e-02
##      log(area)       Full.Bath   Bedroom.AbvGr      Year.Built        Lot.Area
##      4.908e-01      -2.759e-02      -2.275e-02       3.617e-03       2.917e-06
##   Overall.Cond     Bsmt.QualFa     Bsmt.QualGd     Bsmt.QualPo     Bsmt.QualTA
##      7.600e-02      -1.537e-01      -9.303e-02      -2.460e-01      -1.065e-01
## Kitchen.QualFa  Kitchen.QualGd  Kitchen.QualPo  Kitchen.QualTA   Kitchen.AbvGr
##     -6.401e-02      -3.780e-02       1.475e-01      -6.903e-02      -9.482e-02
```

As can be observed from the results above, the AIC model selection method simplified the final model from 14 to 13 variables. The following code aims to provide the summary statistics of the final model:

```
# This code provides the summary statistics of the final model
summary(final_model_AIC)
```

```
##
## Call:
## lm(formula = log(price) ~ Overall.Qual + Garage.Area + Total.Bsmt.SF +
##     Garage.Cars + log(area) + Full.Bath + Bedroom.AbvGr + Year.Built +
##     Lot.Area + Overall.Cond + Bsmt.Qual + Kitchen.Qual + Kitchen.AbvGr,
```

```
##     data = ames_train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.50406 -0.06447  0.00563  0.08000  0.47132
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     5.328e-01  6.886e-01   0.774 0.439246
## Overall.Qual    6.753e-02  6.422e-03  10.515  < 2e-16 ***
## Garage.Area     7.932e-05  5.186e-05   1.529 0.126490
## Total.Bsmt.SF   1.746e-04  1.700e-05  10.268  < 2e-16 ***
## Garage.Cars     3.319e-02  1.582e-02   2.098 0.036196 *
## log(area)       4.908e-01  2.843e-02  17.266  < 2e-16 ***
## Full.Bath      -2.759e-02  1.394e-02  -1.979 0.048065 *
## Bedroom.AbvGr  -2.275e-02  8.132e-03  -2.798 0.005246 **
## Year.Built      3.617e-03  3.075e-04  11.763  < 2e-16 ***
## Lot.Area        2.917e-06  5.206e-07   5.604 2.74e-08 ***
## Overall.Cond    7.600e-02  5.064e-03  15.009  < 2e-16 ***
## Bsmt.QualFa    -1.537e-01  4.226e-02  -3.637 0.000290 ***
## Bsmt.QualGd    -9.303e-02  2.176e-02  -4.276 2.09e-05 ***
## Bsmt.QualPo    -2.460e-01  1.533e-01  -1.605 0.108838
## Bsmt.QualTA    -1.065e-01  2.767e-02  -3.850 0.000126 ***
## Kitchen.QualFa -6.401e-02  4.590e-02  -1.394 0.163514
## Kitchen.QualGd -3.780e-02  2.391e-02  -1.581 0.114254
## Kitchen.QualPo  1.475e-01  1.541e-01   0.957 0.338670
## Kitchen.QualTA -6.903e-02  2.683e-02  -2.573 0.010231 *
## Kitchen.AbvGr  -9.482e-02  2.933e-02  -3.233 0.001267 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.149 on 957 degrees of freedom
##   (23 observations deleted due to missingness)
## Multiple R-squared:  0.8723, Adjusted R-squared:  0.8698
## F-statistic:   344 on 19 and 957 DF,  p-value: < 2.2e-16
```

As can be observed in the results above, the Adjusted R-Squared of the final model is higher (0.8698) than the Adjusted R-Squared of the initial model (0.8378) so all the new predictors added to the initial model are proven to be meaningful as they contribute to increase the Adjusted R-Squared value.

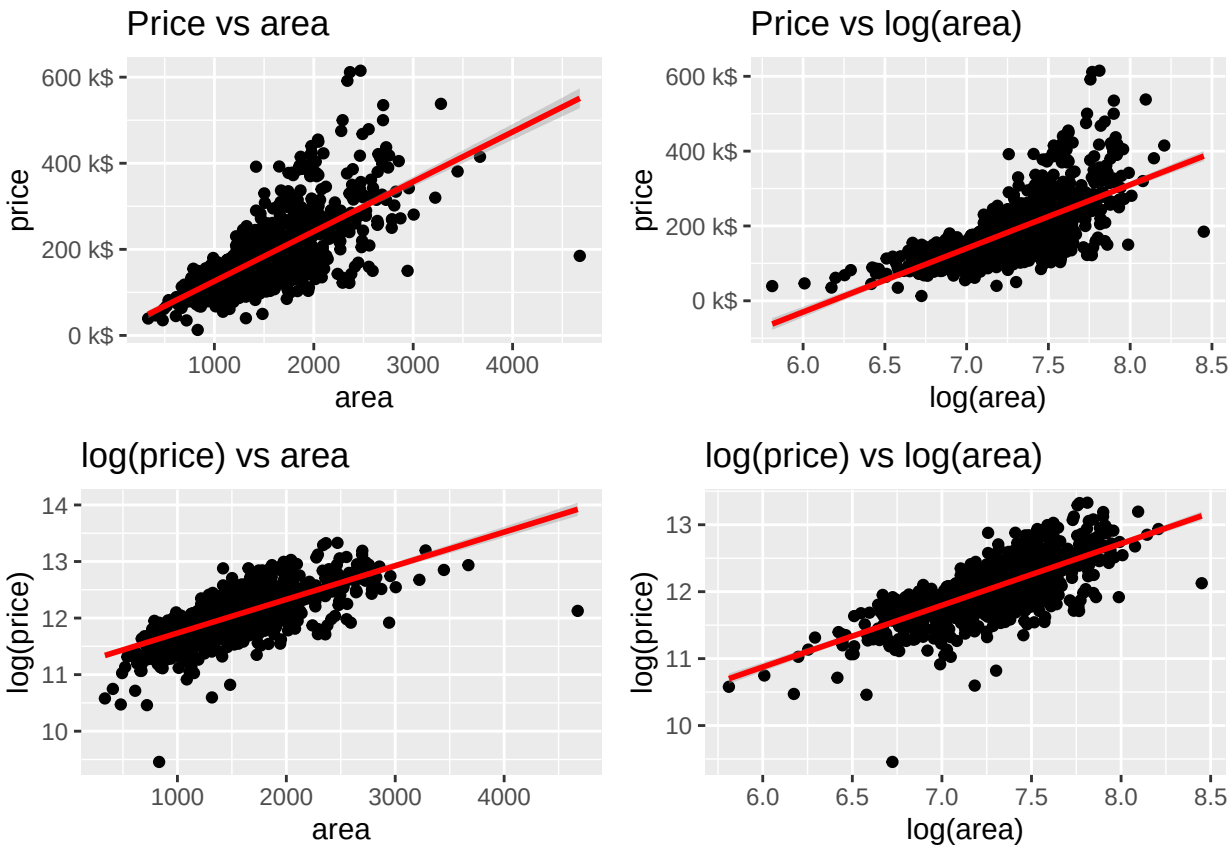
Section 3.2 Transformation

Did you decide to transform any variables? Why or why not? Explain in a few sentences.

For this model, as can be deduced from previous assignments by log-transforming the response variable `price` and the explanatory variable `area` a better linear relationship between `price` and `area` can be achieved. The following code aims to compare the linear relationship between the raw variables previously mentioned and log-transformed variables taking into consideration the possible combinations.

```r
# This code plots the different combinations between `price` and `area`
# using the raw variables and log-transformed variables

# 1) price vs area
plot321 <- ggplot(data=ames_train, aes(x=area, y=price)) + geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs area") +
  stat_smooth(method='lm', color='red')

# 2) price vs log(area)
plot322 <- ggplot(data=ames_train, aes(x=log(area), y=price)) + geom_point() +
  scale_y_continuous(labels=scales::unit_format(unit="k$", scale=1e-3)) +
  ggtitle("Price vs log(area)") +
  stat_smooth(method='lm', color='red')

# 3) log(price) vs area
plot323 <- ggplot(data=ames_train, aes(x=area, y=log(price))) + geom_point() +
  ggtitle("log(price) vs area") +
  stat_smooth(method='lm', color='red')

# 4) log(price) vs log(area)
plot324 <- ggplot(data=ames_train, aes(x=log(area), y=log(price))) + geom_point() +
  ggtitle("log(price) vs log(area)") +
  stat_smooth(method='lm', color='red')

grid.arrange(plot321, plot322, plot323, plot324, ncol=2)
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



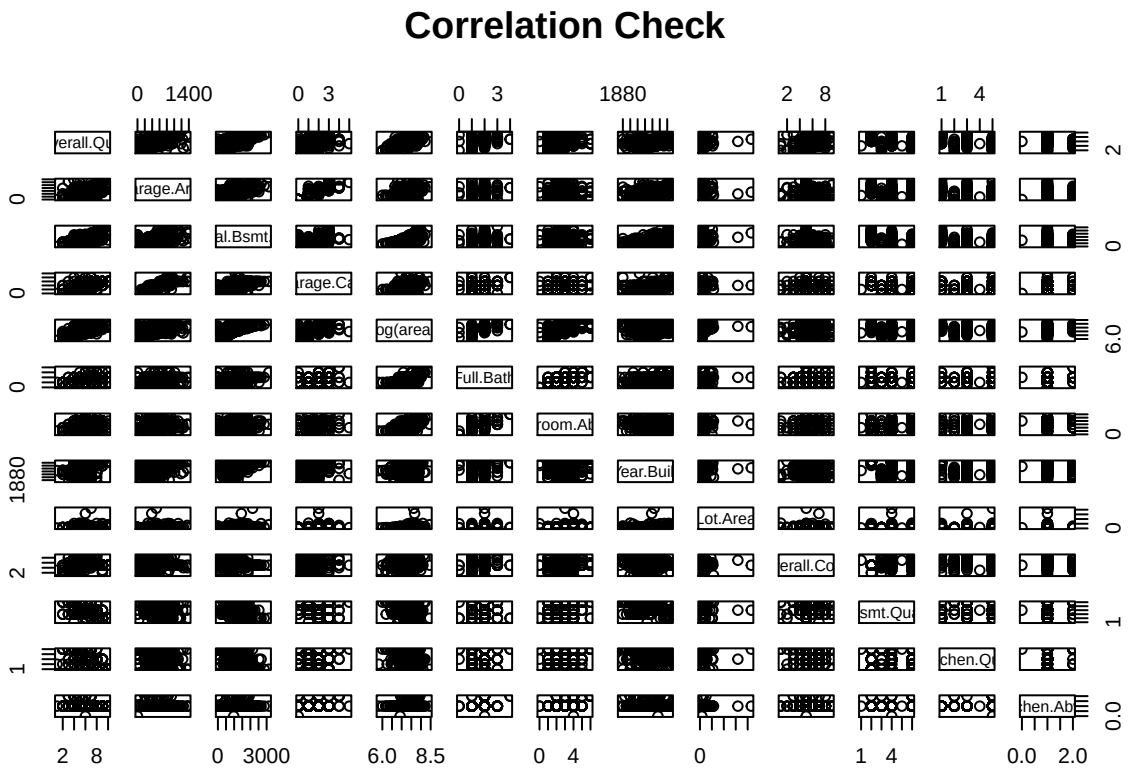
As can be observed from the plots above, by log-transforming the variables `price` and `area` stronger linear relationships can be achieved (second row of plots).

Section 3.3 Variable Interaction

Did you decide to include any variable interactions? Why or why not? Explain in a few sentences.

The R base function pairs() will be used to produce a matrix of scatter plots. This is useful to visualize the correlation of small data sets. This function will be used to check the correlation between some of the variables used in the final model:

```
# This code checks the correlation of the variables used in the final model
pairs(~ Overall.Qual + Garage.Area + Total.Bsmt.SF + Garage.Cars + log(area) + Full.Bath + Bedroom.AbvG
```



As can be observed in the results above, the variables used in the final model are weak correlated. However, as the concept of variable interactions was not fully addressed in this specialization interactions between variables have not been included.

Section 3.4 Variable Selection

What method did you use to select the variables you included? Why did you select the method you used? Explain in a few sentences.

A colinearity check was performed previously to check the correlation between predictors. The final model contains 15 variables that have a strong relationship with price. A backward step approach using AIC model selection method was performed resulting in the same variables that the final model contains and the Adjusted R-Squared obtained was higher than the Adjusted R-Squared obtained in the initial model. The colinearity check was very useful to conclude that the variables used in the final model have a good correlation. The AIC model selection method was used in the final model because it provides a higher Adjusted R-Squared compared to BIC model selection method, resulting in a more accurate housing prices prediction.

Section 3.5 Model Testing

How did testing the model on out-of-sample data affect whether or how you changed your model? Explain in a few sentences.

It would be expected that RMSE from training data will be lower than the RMSE from testing data to prove there is overfitting in the testing data. However, according to the RMSE results, overfitting was not found in the testing data. Thus, the final model resulted only from adding extra predictors to the initial model in order to enhance the model predictive power, as can be deduced from the Adjusted R-Squared values respectively.

Part 4 Final Model Assessment

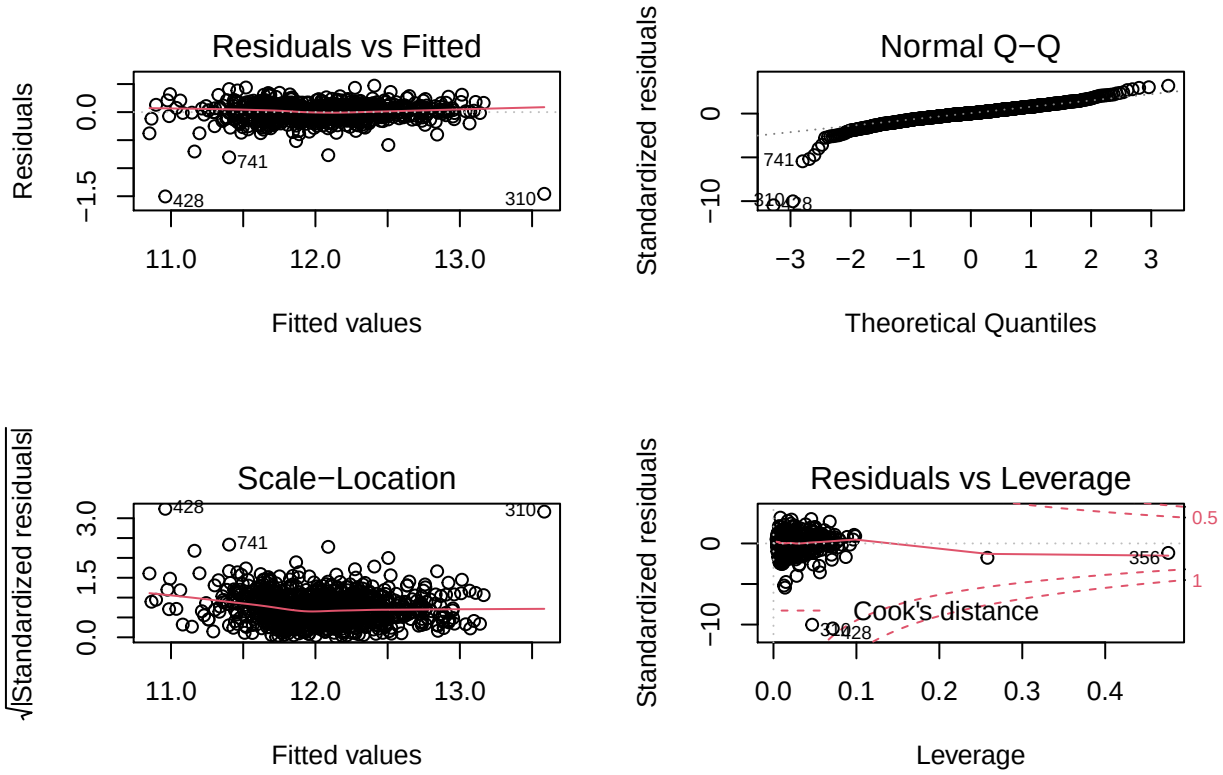
Section 4.1 Final Model Residual

For your final model, create and briefly interpret an informative plot of the residuals.

Here below the final model residuals plots for the procedure AIC are present:

```
# This code plots Residuals and Normal Q-Q plots
par(mfrow=c(2,2))
plot(final_model_AIC)
```

```
## Warning: not plotting observations with leverage one:
##   109, 126
```



As can be observed from the plots above, the Residuals vs Fitted plot shows a constant variability of residuals with some high leverage outliers (rows 428, 741 and 310). As happened with the initial model this is due to use categorical variables to set up a regression model. Given the constant variability in residuals and the strong linearity as observed in Normal Q-Q plot, with a nearly normal distribution despite some outliers (rows 741, 310 and 428), it is expected a good accuracy in predictions. Although long tails have been found, the sample is big enough to meet the Central Limit Theorem (CLT), then the distribution of the sample means will be approximately normal distributed.

Section 4.2 Final Model RMSE

For your final model, calculate and briefly comment on the RMSE.

The code below calculates the RMSE of the ames_test data set in the final model

```
# This code calculates the RMSE of the ames_test data set in the final model
pred_final_model <- exp(predict(final_model_AIC, ames_test))
resid_final_model <- na.omit(ames_test$price - pred_final_model)
RMSE_final_model <- sqrt(mean(resid_final_model^2))
paste('The RMSE for the final model in the testing data is', format(RMSE_final_model, digits=6), 'dolla
```

```
## [1] "The RMSE for the final model in the testing data is 21843.6 dollars"
```

As can be observed from the results above, the final model RMSE value is 21843.6 dollars which is lower than the initial model RMSE value in the testing data: 25505.6 dollars. The lower RMSE value in the final model indicates that this model is more accurate making predictions compared to the initial model. Then, taking RMSE as a measure of accuracy, it can be concluded that the extra predictors added to the final model are meaningful as they help to improve the Adjusted R-Squared value of the final model (0.8698 compared to 0.8378 of the initial model) as well as to reduce the value of RMSE in the final model.

Section 4.3 Final Model Evaluation

What are some strengths and weaknesses of your model?

Strengths:

1. Higher Adjusted R-Squared value compared to the initial model (final model: 0.8698 > initial model: 0.8378)
2. Lower RMSE value compared to the initial model (final model: 21843.6 < initial model: 25505.6)

Weaknesses:

1. Uses more variables compared to the initial model and BIC procedure. Furthermore, with the BIC model selection method a more parsimonious model could have been obtained.
2. As frequentist methodology has been followed to obtain the final model, no priors have been used so that the predictive power is limited due to the fact that is not based on previous knowledge.

Section 4.4 Final Model Validation

Testing your final model on a separate, validation data set is a great way to determine how your model will perform in real-life practice.

You will use the "ames_validation" dataset to do some additional assessment of your final model. Discuss your findings, be sure to mention: * What is the RMSE of your final model when applied to the validation data?
* How does this value compare to that of the training data and/or testing data? * What percentage of the 95% predictive confidence (or credible) intervals contain the true price of the house in the validation data set?
* From this result, does your final model properly reflect uncertainty?

```
load("ames_validation.Rdata")
```

4.4.1. What is the RMSE of your final model when applied to the validation data? The code below calculates the RMSE of the final model applied to validation data, following the same steps as previously done with the testing and training data.

```r
# This code calculates the RMSE of the final model from the validation data
pred_validation <- exp(predict(final_model_AIC, ames_validation))
resid_validation <- na.omit(ames_validation$price - pred_validation)
RMSE_validation <- sqrt(mean(resid_validation^2))
paste('The RMSE for the final model in the validation data is', format(RMSE_validation, digits=6), 'dol
```

```
## [1] "The RMSE for the final model in the validation data is 21264.6 dollars"
```

As can be observed in the results above, the RMSE for the final model in the validation data is 21264.6 dollars which is a great value compared to the RMSE of the final model in the testing data and the RMSE of the initial model in the testing and training data.

4.4.2. How does this value compare to that of the training data and/or testing data? A comparison of the validation and testing data RMSE values of the final model is provided below:

- Validation data RMSE is 21264.6 dollars
- testing data RMSE is 21843.6 dollars

As can be observed, the RMSE of the final model in the validation data is lower than the RMSE of the final model in the testing data. In the final model, thus, not much evidence of overfitting can be found.

4.4.3. What percentage of the 95% predictive confidence (or credible) intervals contain the true price of the house in the validation data set? The percentage of the 95% predictive confidence intervals containing the true price of the house in the validation data set is provided below:

```r
# This code gets the percentage of the 95% predictive confidence intervals containing the true price of
ci_prediction <- na.omit(exp(predict(final_model_AIC, ames_validation, interval = 'prediction', level=0
# Calculation of the proportion of observations that fall within prediction intervals
ci_percentage <- sum(ames_validation$price > ci_prediction[,"lwr"] &
                           ames_validation$price < ci_prediction[,"upr"])/nrow(ci_prediction)*100
```

```
## Warning in ames_validation$price > ci_prediction[, "lwr"]: longitud de objeto
## mayor no es múltiplo de la longitud de uno menor
```

```
## Warning in ames_validation$price < ci_prediction[, "upr"]: longitud de objeto
## mayor no es múltiplo de la longitud de uno menor
```

```r
paste('The coverage probability of this model is', format(ci_percentage, digits=3), "%")
```

```
## [1] "The coverage probability of this model is 47.2 %"
```

The coverage probability of this final model is 47.2%, thus this model properly reflects uncertainty.

Part 5 Conclusion

Provide a brief summary of your results, and a brief discussion of what you have learned about the data and your model.

The EDA was used to check out undervalued and overvalued houses using summary statistics and distributions considering age and overall quality. The distribution of Ames training data set can be considered as nearly normal due to its large size and skeweness. On the other hand, real house prices are directly determined by the willingness of households to pay for a constant-quality house, in addition, results have also demonstrated that the neighborhood plays an important role in house prices, being StoneBr the most expensive neighborhood and MeadowV the least expensive neighborhood in Iowa. A number of explanatory variables were used to find the best predictors to build the initial model afterwards, being Overall.Quality the best one.

Two stepwise selection procedures were used to determine the best model: BIC and AIC. AIC procedure was preferred according to its higher Adjusted R-Squared value compared to the lower value obtained for BIC procedure. Thanks to the higher Adjusted R-Squared obtained, a model with a better prediction power could be achieved, producing a highly effective and strong final model.

The Ames Housing data set contains more than enough variables to explore and build an accurate model aiming to predict house selling prices in Iowa. The model has been built on the training data so overfitting would be expected to occur, however, in this scenario the testing data used with this data set tends to perform better than the training data due to its lower RMSE value. Based on the results from the model considered, the RMSE in the validation data is lower compared to the RMSE in the testing data.

I learned from the data and the model that in order to achieve a strong and accurate model a good data set containing a large amount of reliable observations and variables is mandatory. Having a large data set guarantees exploring better the data to discover new findings and reveal interesting trends and insights to build a strong model with high powerful prediction capabilities and diagnose and test the model afterwards to have a good model validation. From this point forward, collecting more explanatory variables for this data set or using more advanced prediction tools, could grant building a model with a better predictive power starting with this final model as a scaffold.
