STATISTICS FOR BUSINESS ANALYTICS I

Assignment 2 on Multiple Regression

Name: Ilias Dimos

Student Number: f2822102

Professor: Ilias Leriou

Period: 2021-2022

Table of Contents

EXPLANATION OF ASSIGNMENT'S DATAFRAME 3
QUESTION I 3
QUESTION II 4
QUESTION III 5
QUESTION IV 9
QUESTION V 11
QUESTION VI 12
QUESTION VII 14
QUESTION VIII 18
QUESTION IX 24

EXPLANATION OF ASSIGNMENT'S DATAFRAME

The data for this assignment are a random sample of 63 cases from the files of a big real estate agency in USA concerning house sales from February 15 to April 30, 1993. The data was collected from many cities (and corresponding local real estate agencies) and is used as a basis for the whole company. The variables in this dataset are:

1. PRICE = Selling prices (in hundreds$)
2. SQFT = Square Feet of living space
3. AGE = Age of home (in years)
4. FEATS = Number out of 11 features (dishwasher, refrigerator, microwave, disposer, washer, intercom, skylight(s), compactor, dryer, handicap fit, cable TV access
5. NE = Located in northeast sector of city (1) or not (0)
6. COR = Corner location (1) or not (0).

QUESTION I

Read the "usdata" dataset and use str() to understand its structure.

Output

```
> str(usdata)
'data.frame':   63 obs. of  6 variables:
 $ PRICE: int  2050 2150 2150 1999 1900 1800 1560 1449 1375 1270 ...
 $ SQFT : int  2650 2664 2921 2580 2580 2774 1920 1710 1837 1880 ...
 $ AGE  : int  3 28 17 20 20 10 2 2 20 30 ...
 $ FEATS: int  7 5 6 4 4 4 5 3 5 6 ...
 $ NE   : int  1 1 1 1 1 1 1 1 1 1 ...
 $ COR  : int  0 0 0 0 0 0 0 0 0 0 ...
```

As we can see our dataset with a first look contains 6 integer type variables and 63 observations

Also, we have to check if there are NA values or NULL values in our dataset in order to fix them.

Output

```
> sum(is.na(usdata)) # no na's in the dataset
[1] 0
> sum(is.null(usdata)) # no null's in the dataset
[1] 0
>
```

Likely we have no NULL and NA values.

QUESTION II

Convert the variables PRICE, SQFT, AGE, FEATS to be numeric variables and NE, COR to be factors.

To do that we have to use the as. numeric and as. factor functions in R.

Analytically,

```
# updating the labels of the usdata for better explanation in question 2
usdata$PRICE <- as.numeric(usdata$PRICE)
usdata$SQFT <- as.numeric(usdata$SQFT)
usdata$AGE <- as.numeric(usdata$AGE)
usdata$FEATS <- as.numeric(usdata$FEATS)

usdata$NE <- as.factor(usdata$NE)
usdata$COR <- as.factor(usdata$COR)

usdata$NE <- factor(usdata$NE,levels=c(0,1),labels=c('no','yes'))
usdata$COR <- factor(usdata$COR,levels=c(0,1),labels=c('no','yes'))
```

I also turned the NE and COR variable from binary to categorical variables with “yes” or “no” values.

Output

```
> str(usdata)
'data.frame':   63 obs. of  6 variables:
 $ PRICE: num  2050 2150 2150 1999 1900 ...
 $ SQFT : num  2650 2664 2921 2580 2580 ...
 $ AGE  : num  3 28 17 20 20 10 2 2 20 30 ...
 $ FEATS: num  7 5 6 4 4 4 5 3 5 6 ...
 $ NE   : Factor w/ 2 levels "no","yes": 2 2 2 2 2 2 2 2 2 2 ...
 $ COR  : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
>
```

As you can see the type of each variable has successfully changed.

QUESTION III

Perform descriptive analysis and visualization for each variable to get an initial insight of what the data looks like. Comment on your findings.

First of all, we have to separate in one dataset the numeric variables only. After that, with the command “describe” (from the psych library in R) we can see every detail of our numeric variables (vars ,n, mean , sd ,median, trimmed , mean , mad ,min ,max ,range ,skew ,kurtosis ,se).

```
> round(t(describe(usdata1)),1)
          PRICE   SQFT  AGE FEATS
vars        1.0    2.0  3.0   4.0
n          63.0   63.0 63.0  63.0
mean     1158.4 1729.5 17.5   4.0
sd        392.7  506.7  9.6   1.3
median   1049.0 1680.0 20.0   4.0
trimmed  1106.0 1685.2 17.7   3.9
mad       262.4  392.9 11.9   1.5
min       580.0  970.0  2.0   1.0
max      2150.0 2931.0 31.0   8.0
range    1570.0 1961.0 29.0   7.0
skew        1.2    0.7 -0.2   0.5
kurtosis    0.5   -0.2 -1.5   1.1
se         49.5   63.8  1.2   0.2
```

Describe command output

```
> head(usdata1,5)
  PRICE SQFT AGE FEATS
1  2050 2650   3     7
2  2150 2664  28     5
3  2150 2921  17     6
4  1999 2580  20     4
5  1900 2580  20     4
```

The first 5 observations of the dataset that contains the numeric variables only.

Visualization analysis for the numeric and factor variables

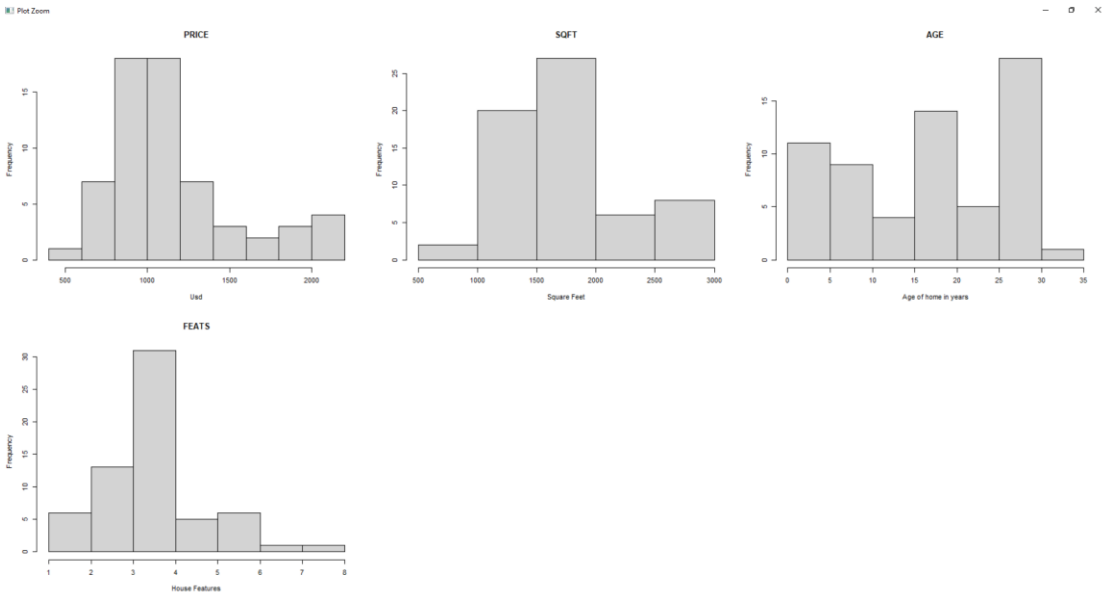
In this section we will analyze the numeric and the factor variables in our dataset by using histograms, tables, prop. tables and barplots.

First, we will start with the numeric variables.

With the commands below we were able to build the visuals we need to analyze them.

```
# Visualization for Numerics
par(mfrow=c(2,3))
n <- nrow(usdata1)
hist(usdata1[,1], main=names(usdata1)[1],xlab="Usd")
hist(usdata1[,2], main=names(usdata1)[2],xlab="Square Feet")
hist(usdata1[,3], main=names(usdata1)[3],xlab='Age of home in years')
hist(usdata1[,4], main=names(usdata1)[4],xlab="House Features")
```

Output



Explanation of each NUMERIC variable based one the histograms and the describe function above.

PRICE: WE SEE THAT THE PRICE VARIABLE HAS 1158.4 MEAN AND 1049 MEDIAN, THE SKEWNESS IS POSSITIVE BECAUSE A LOT OF THE OBSERVETAION ARE RIGHT OF THE DISTRIBUTION.

SQFT: WE SEE THAT THE MEAN SQEARE FEET OF THE REAL ESTATES IS 1729 SQ, THE MOFT OF THE REAL ESTATES HAS 1500 TO 2000 SQ FEET AS WE OBSERVE THE HISTOGRAM.

AGE: WE SEE THAT THE MEAN OF THE HOUSES'S AGE IS 17.5 AND BY OBSERVING THE DIAGRAM WE CAN SEE THAT THE MOST HOUSES ARE BETWEEN 25 TO 30 YEARS OLD, THE MEDIAN IS VERY LOW (IN COMPARISSON WITH THE MEAN) THAT MEANS THAT THE OBSERVATION ARE POSSIBLY RANDOM DISTRIBUTED.

FEATS: AS FAS AS THE FEATS OBSERVATIONS ARE CONSERNET WE SEE THAT IN 63 OBSERVATIONS THE MEAN IS 4 AND THE MEDIAN 4 THAT MEANS THAT THE FEATS ARE MORE OR LESS EVENLY DISTRIBUTED, WE SEE THAT THE MOST OF THE HOUSES HAVE 3-4 FEATURES AND A VERY SMALL NUMBER OF HOUSES HAVE 7-8 FEATURES.

Explanation of each FACTOR variable based one the histograms and the describe function above.

It's very useful to see the proportions of each factor variable and their meaning, to do this i used the "table" function in R in combination with the "prop.table" function.

```
> table(usdata$NE) # we have 39 real estates are located in northeast sector of the city and 24 arent located

 no yes
 24  39
> table(usdata$COR) # we have 14 real estates that are in cornan location and 49 that they dont

 no yes
 49  14
> round(prop.table(table(usdata$NE)),2) # finally the 62% of the real estates belong to northeast sector and 38% they dont

  no  yes
0.38 0.62
> round(prop.table(table(usdata$COR)),2) # finally the 22% of the real estates are in cornen location and the 78% they dont

  no  yes
0.78 0.22
```

The first table shows the that 39 real estates are located in northeast sector of the city and 24 the don't.

The second one shows that 14 real estates are in Corner Location and 49 they don't.

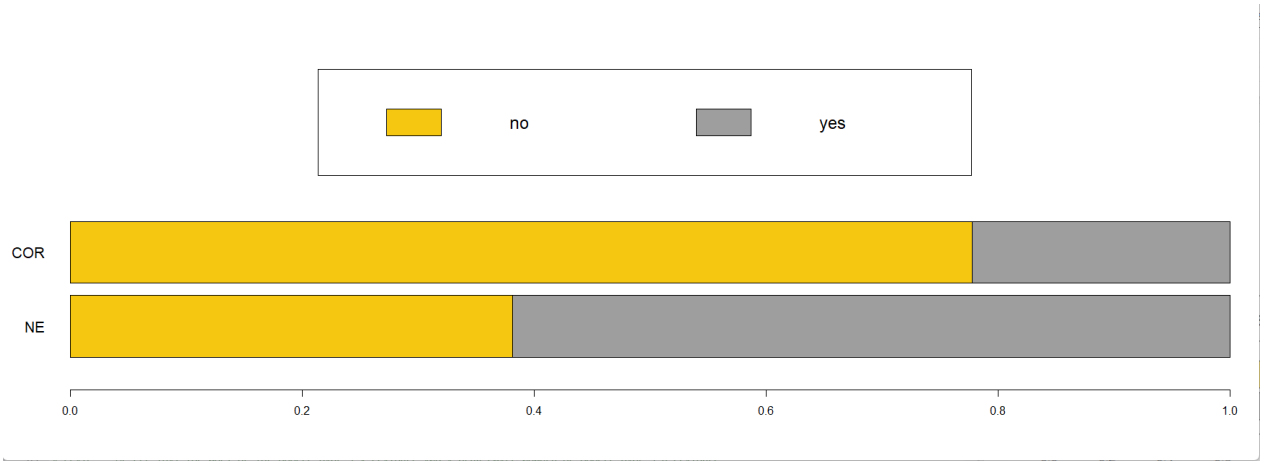
The third one shows the proportion of a real estate to be located in northeast sector that is 62% and 38% they don't.

Finally, the fourth one shows that 24% of the real estates are in corner location and 78 % they don't.

Also, in order to verify our results we will build a boxplot with the factor variables only. We the commands below I managed to make the diagram for them.

```
# visualization of the factor variables using the barplot to from the lecture

only_factors <- usdata[,!data1]
par(mfrow=c(1,1))
barplot(sapply(only_factors,table)/n, horiz=T, las=1, col=7:8, ylim=c(0,8), cex.names=1.3)
legend('center', fil=7:8, legend=c('no','yes'), ncol=2, bty='o',cex=1.5)
```

Output

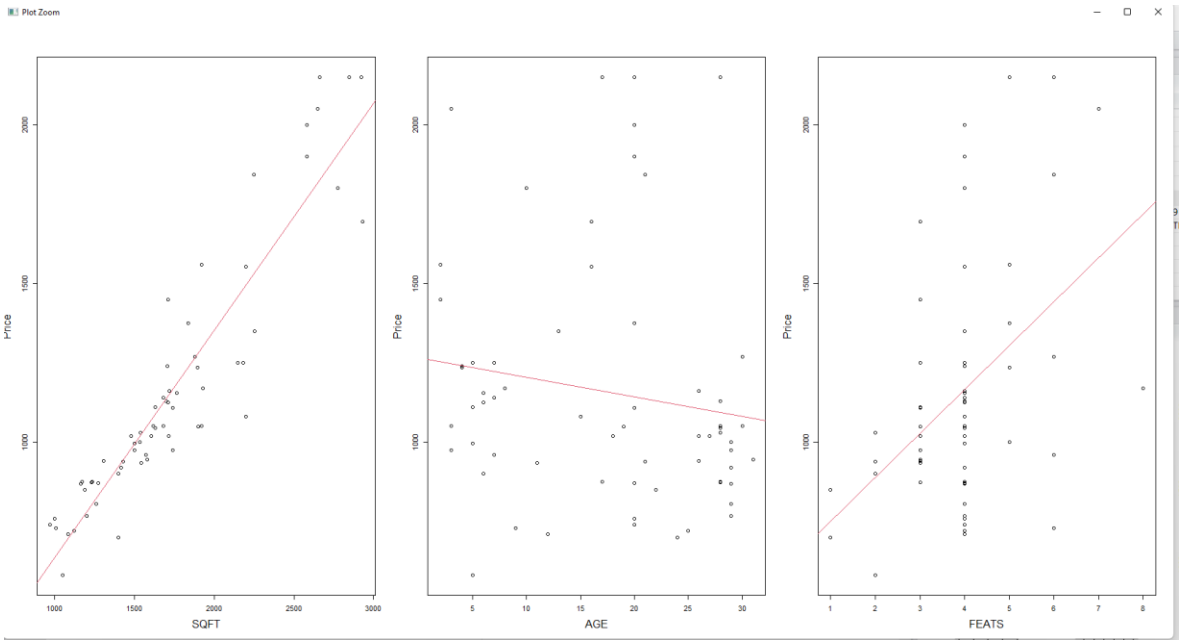


QUESTION IV

Conduct pairwise comparisons between the variables in the dataset to investigate if there are any associations implied by the dataset. (Hint: Plot variables against one another and use correlation plots and measures for the numerical variables.). Comment on your findings. Is there a linear relationship between PRICE and any of the variables in the dataset?

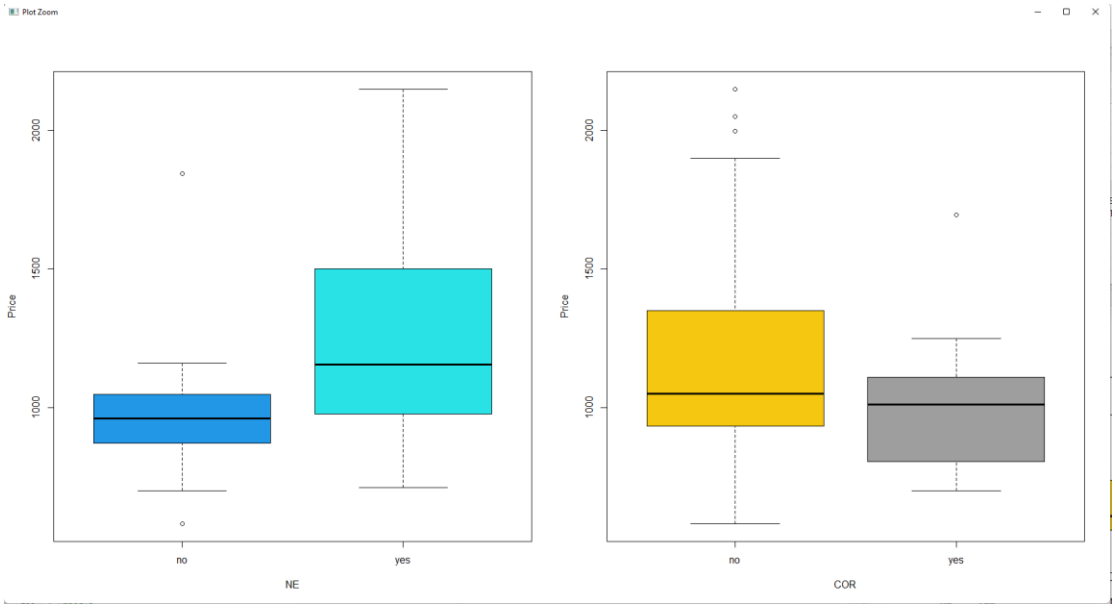
Firstly, I will build plots only for numeric variables in order to examine them.

Output



Observing the plots, we see a perfect positive linear correlation between the “PRICE” variable and the “SQFT” variable because the dots are laying in the lm line), for the variables “AGE” AND “FEATS” we can say that they don’t have a good linear correlation with the “PRICE” variable.

Now I will do the same process for factor variables but this time I will use boxplots for better explanation.



In order to finish our process with the pairwise comparisons we have to build correlation plots for numeric variables (I will use the library “corrplot”).

The result of the above graph is that:

We have strong linear correlation for “SQFT” ~ “PRICE”. Which is very logical.

Negative linear correlation for “SQFT” ~ “AGE” and for “PRICE” ~” AGE”.

Medium linear correlation for “PRICE” ~ “FEATS” and for “SQFT” ~ “FEATS”.

Finally in order to answer the last question that refers if there is any linear relationship between price and any other variable by looking the plot above, we can see yes there is a linear relationship with the “SQFT” variable.

QUESTION V

Construct a model for the expected selling prices (PRICE) according to the remaining features. (Hint: Conduct multiple regression having PRICE as a response and all the other variables as predictors). Does this linear model fit well to the data? (Hint: Comment on R^2 adj).

To find our foul Model we will use the “lm” function after that we will use the “summary” function to see the details about our foul model.

Output

```
> full_model <- lm(PRICE~.,usdata)
> summary(full_model)

Call:
lm(formula = PRICE ~ ., data = usdata)

Residuals:
    Min      1Q  Median      3Q     Max
-416.11  -71.03  -15.26   83.02  347.77

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -193.34926   94.52382  -2.046   0.0454 *
SQFT           0.67662    0.04098  16.509   <2e-16 ***
AGE            2.22907    2.28626   0.975   0.3337
FEATS         34.36573   16.27114   2.112   0.0391 *
NEyes         30.00446   47.93940   0.626   0.5339
CORyes       -53.07940   46.15653  -1.150   0.2550
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 144.8 on 57 degrees of freedom
Multiple R-squared:  0.8749,    Adjusted R-squared:  0.864
F-statistic: 79.76 on 5 and 57 DF,  p-value: < 2.2e-16
```

The first thing that we notice is that the R^2 is very high, that’s good cause the higher the R^2 the better the model fits to data also the adj R^2 square is very high that means that our variables in the model are not useless at all , the most useful variables we put to the model the higher the adj r^2 should be , all the variables are statistically significant BUT we don't like the high negative value of the intercept , maybe we have to find a way to reduce it or to get rid of it.

QUESTION VI

Find the best model for predicting the selling prices (PRICE). Select the appropriate features using stepwise methods. (Hint: Use Forward, Backward or Stepwise procedure according to AIC or BIC to choose which variables appear to be more significant for predicting selling PRICES).

In order to end up with a new model we need to select the appropriate features by using the “stepwise both” method. This method every time removes and adds variables in our model until it finds the best one. The function I used for the step wised model is the below one.

```
# QUESTION 6

final_model<-step(full_model, direction='both')
```

And the outcome is:

```
> final_model<-step(full_model, direction='both')
Start:  AIC=632.62
PRICE ~ SQFT + AGE + FEATS + NE + COR

        Df Sum of Sq     RSS    AIC
- NE     1      8218 1203977 631.05
- AGE    1     19942 1215701 631.66
- COR    1     27743 1223502 632.07
<none>               1195759 632.62
- FEATS  1     93580 1289339 635.37
- SQFT   1   5717835 6913594 741.17

Step:  AIC=631.05
PRICE ~ SQFT + AGE + FEATS + COR

        Df Sum of Sq     RSS    AIC
- AGE    1     12171 1216147 629.69
- COR    1     25099 1229076 630.35
<none>               1203977 631.05
+ NE     1      8218 1195759 632.62
- FEATS  1    106953 1310930 634.42
- SQFT   1   6288869 7492846 744.24

Step:  AIC=629.69
PRICE ~ SQFT + FEATS + COR

        Df Sum of Sq     RSS    AIC
- COR    1     22454 1238602 628.84
<none>               1216147 629.69
+ AGE    1     12171 1203977 631.05
+ NE     1       447 1215701 631.66
- FEATS  1    104259 1320407 632.87
- SQFT   1   6352036 7568184 742.87

Step:  AIC=628.84
PRICE ~ SQFT + FEATS

        Df Sum of Sq     RSS    AIC
<none>               1238602 628.84
+ COR    1     22454 1216147 629.69
+ AGE    1      9526 1229076 630.35
+ NE     1       218 1238384 630.83
- FEATS  1    138761 1377363 633.53
- SQFT   1   6389899 7628501 741.37
```

Explanation of the stepwise procedure outcome:

We start from the full model and literately we add of remove variables. Firstly, the algorithm removes the NE variable and as a result the AIC decreases , secondly the algorithm removes also the less significant variable AGE and as a result the AIC decreases even more thirdly the algorithm removes again the less significant variable COR and as a result the AIC decreases and stops its procedure we end up we the best model with a 2 COEFFICIENTS (SQFT,FEATS) and with a reduced intercept than the full model in QUESTION 5 (which is more pleasing to us).

QUESTION VII

Get the summary of your final model, (the model that you ended up having after conducting the stepwise procedure) and comment on the output. Interpret the coefficients. Comment on the significance of each coefficient and write down the mathematical formulation of the model (e.g., PRICES = Intercept + coef1*Variable1 + coef2*Variable2 +…. + ε where ε ~ N(0, ...)). Should the intercept be excluded from our model?

Output

```
Call:
lm(formula = PRICE ~ SQFT + FEATS, data = usdata)

Residuals:
    Min      1Q  Median      3Q     Max
-400.44  -71.70  -11.21   93.12  341.82

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -175.92760   74.34207  -2.366   0.0212 *
SQFT           0.68046    0.03868  17.594   <2e-16 ***
FEATS         39.83687   15.36531   2.593   0.0119 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 143.7 on 60 degrees of freedom
Multiple R-squared:  0.8705,    Adjusted R-squared:  0.8661
F-statistic: 201.6 on 2 and 60 DF,  p-value: < 2.2e-16
```

Observing the output we see that intercept is statistically significant at a 0.01 significant level.

After the stepwise procedure we end up with our final model which is: PRICE=-175.92+0.68046*SQFT+39.83687*FEATS + ε, where ε ~N(0,143.7^2)

All our coefficients are statistically significant.

Comment about the intercept and the SQFT AND FEATS.

Intercept = - 175.92, that means that if we remove all the coefficients of the house then the price of the house will be -175.92, that's not logical.

SQFT = 0.68046, that means that if the real estate will be increased by 1 square foot, the price of the house will be increased by 0.68 units (Hundred's dollars)

FEATS=39.83, that means that an addition of one feature in the house, the price of the house will be increased by 39.83 (Hundreds of Dollars)

As we said above, the negative intercept of our model doesn't make any sense, in the explanation of the intercept we assume that "if the sqft = 0" that's also a not right assumption because a house will never have a 0 sqft.

So, we will try to remove the intercept from our model to see if our model will become worst or better than the first one.

```
> summary(step_no_intercept_model)

Call:
lm(formula = PRICE ~ SQFT + FEATS + NE - 1, data = usdata)

Residuals:
    Min      1Q  Median      3Q     Max
-401.43  -71.43  -12.77   93.46  339.95

Coefficients:
        Estimate Std. Error t value Pr(>|t|)
SQFT     0.67920    0.04094  16.589   <2e-16 ***
FEATS   39.57015   15.71347   2.518   0.0145 *
NEno  -175.26995   75.24056  -2.329   0.0233 *
NEyes -171.08398   88.78132  -1.927   0.0588 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 144.9 on 59 degrees of freedom
Multiple R-squared:  0.9868,    Adjusted R-squared:  0.9859
F-statistic:  1106 on 4 and 59 DF,  p-value: < 2.2e-16
```

After running the summary in the model with no intercept we see a huge adj R-squared. That's a miscalculation by R so we will recalculate it with the proper formula below.

```
> true.r2 <- 1-sum(step_no_intercept_model$res^2)/((nrow(usdata)-1)*var(usdata$PRICE))
> true.r2 #(0.8704843)
[1] 0.8704843
```

The right ADJ R^2 is 0.8704843.

Explanation why we shouldn't remove the intercept eventually.

Firstly, we have to take in mind the significance of the intercept in our full model. Also, as we observe the adj r^2 from the model with the intercept, is getting decreased if we remove it, that's not a good indicator because we understand that the change of a good fit of our model also decreased.

So instead of removing the intercept we will try to correct it by using centered covariates).

```
> centered_covariates <- as.data.frame(scale(usdata1, center = TRUE, scale = F))
> centered_covariates$PRICE<-usdata1$PRICE
> sapply(centered_covariates,mean)
        PRICE          SQFT           AGE         FEATS
 1.158413e+03 -1.045636e-13  1.634364e-15 -1.692130e-16
> sapply(centered_covariates,sd)
     PRICE       SQFT        AGE      FEATS
392.708775 506.699435   9.599774   1.275433
> round(sapply(centered_covariates,mean),2)
  PRICE    SQFT     AGE   FEATS
1158.41    0.00    0.00    0.00
> round(sapply(centered_covariates,sd),2)
 PRICE   SQFT    AGE  FEATS
392.71 506.70   9.60   1.28
> class(centered_covariates)
[1] "data.frame"
> centered_covariates_model<-lm(PRICE~., centered_covariates)
> class(centered_covariates_model)
[1] "lm"
> summary(centered_covariates_model)

Call:
lm(formula = PRICE ~ ., data = centered_covariates)

Residuals:
    Min      1Q  Median      3Q     Max
-399.17  -66.27   -6.54   81.35  362.89

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.158e+03  1.818e+01  63.705   <2e-16 ***
SQFT        6.846e-01  3.933e-02  17.408   <2e-16 ***
AGE         1.317e+00  1.948e+00   0.676   0.5015
FEATS       4.049e+01  1.547e+01   2.618   0.0112 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 144.3 on 59 degrees of freedom
Multiple R-squared:  0.8715,    Adjusted R-squared:  0.8649
F-statistic: 133.3 on 3 and 59 DF,  p-value: < 2.2e-16
```

We see that the centered covariates seem to have a be a better fit for our model. Now we will use the stepwise procedure in order to get rid of non-significant variables from the centered covariate model.

```
> step_centered_model_final <- step(centered_covariates_model,direction = "both")
Start:  AIC=630.35
PRICE ~ SQFT + AGE + FEATS

        Df Sum of Sq     RSS    AIC
- AGE    1      9526 1238602 628.84
<none>               1229076 630.35
- FEATS  1    142786 1371862 635.28
- SQFT   1   6312487 7541564 742.65

Step:  AIC=628.84
PRICE ~ SQFT + FEATS

        Df Sum of Sq     RSS    AIC
<none>               1238602 628.84
+ AGE    1      9526 1229076 630.35
- FEATS  1    138761 1377363 633.53
- SQFT   1   6389899 7628501 741.37
> class(step_centered_model_final)
[1] "lm"
> summary(step_centered_model_final)

Call:
lm(formula = PRICE ~ SQFT + FEATS, data = centered_covariates)

Residuals:
    Min      1Q  Median      3Q     Max
-400.44  -71.70  -11.21   93.12  341.82

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.158e+03  1.810e+01  63.995   <2e-16 ***
SQFT        6.805e-01  3.868e-02  17.594   <2e-16 ***
FEATS       3.984e+01  1.537e+01   2.593   0.0119 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 143.7 on 60 degrees of freedom
Multiple R-squared:  0.8705,    Adjusted R-squared:  0.8661
F-statistic: 201.6 on 2 and 60 DF,  p-value: < 2.2e-16
```

As we can see the stepwise procedure removed the “age” variable from our model.

And we ended up in a model with a positive intercept and all the other variables statically significant, that’s the model we want!

Interpretation of the "stepwised final centered model ".

We observe that the expected price for buying a house with average characteristics will be 1.158(in Hundreds of Dollars), the average characteristics are showed below:

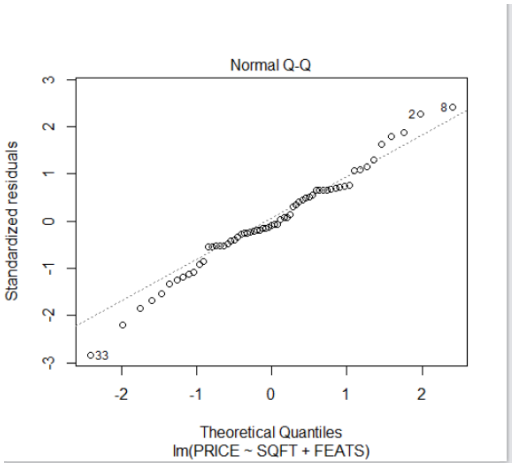
```
> mean(usdata1$FEATS) # 3.952381 ~ 4 features
[1] 3.952381
> mean(usdata1$SQFT) # 1729.54 ~ Size 1729.54 sqft
[1] 1729.54
```

QUESTION VIII

Checking the assumptions of our stepwised final centered model

Normality of residuals

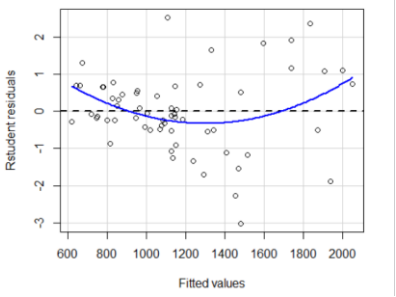
```
> lillie.test(step_centered_model_final$residuals)

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  step_centered_model_final$residuals
D = 0.10234, p-value = 0.09854

> shapiro.test(step_centered_model_final$residuals)

        Shapiro-Wilk normality test

data:  step_centered_model_final$residuals
W = 0.98483, p-value = 0.6303
```



From the test we see a vary high p value so we don't reject normality on residuals.

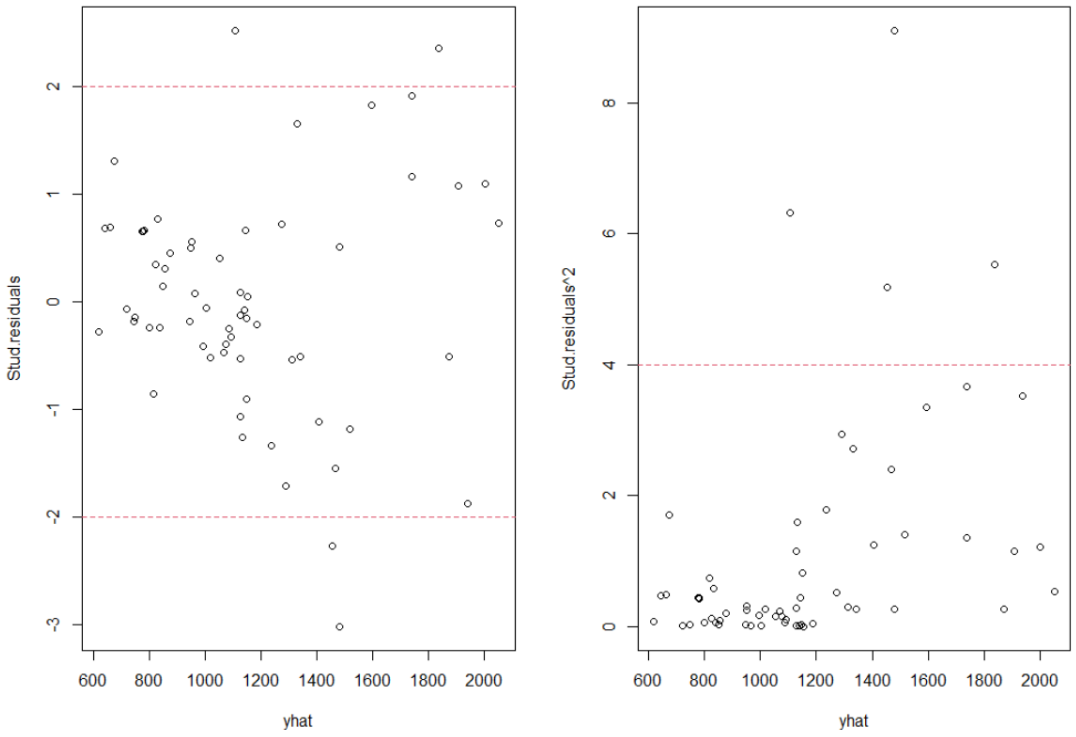
LINEARITY

```
> par(mfrow=c(1,1))
> residualPlot(step_centered_model_final,type='rstudent')
> residualPlots(step_centered_model_final,plot=F,type='rstudent')
           Test stat Pr(>|Test stat|)
SQFT          2.0388         0.045959 *
FEATS        -0.2876         0.774643
Tukey test    2.6002         0.009317 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



As we can see the linearity of our model is violated for 5% significance level.

HOMOSCEDASITY



Observing the plots, we see that out of the dotted lines there are observations, that means we probably do not have a constant variance.

```
> ncvTest(step_centered_model_final) # p value to small
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 14.99402, Df = 1, p = 0.00010785
```

We can verify that we don't have a constant variance and from the "ncvTest" which shows a very small p value and that means we reject the hypothesis of constant variance. So, the assumption of Homoscedasity is violated.

INDEPENDENCE

```
> durbinWatsonTest(step_centered_model_final)
 lag Autocorrelation D-W Statistic p-value
   1       0.2012826      1.573363   0.064
 Alternative hypothesis: rho != 0
> runs.test(step_centered_model_final$residuals)

        Runs Test

data:  step_centered_model_final$residuals
statistic = -0.25611, runs = 31, n1 = 31, n2 = 31, n = 62, p-value = 0.7979
alternative hypothesis: nonrandomness

> plot(rstudent(step_centered_model_final),type = "l")
> plot(rstudent(step_centered_model_final),type = "l")
> dwtest(step_centered_model_final)

        Durbin-Watson test

data:  step_centered_model_final
DW = 1.5734, p-value = 0.03571
alternative hypothesis: true autocorrelation is greater than 0
```

Using runs.test (library randtest) and DurbinWatsonTest(library lmtest) we ended up having independence of errors. Running the “dwtest” it seems that the independence assumption is violated so that it’s a good indicator to transform the model.

FIXING THE PROBLEMS OF HOMOSCEDASITY AND LINEARITY BY APPLYING LOG TRANSFORMATIONS.

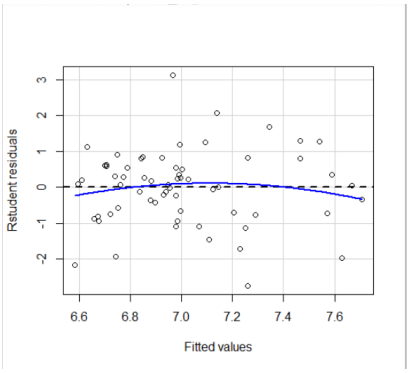
NORMALITY

```
> class(step_centered_model_final)
[1] "lm"
> step_centered_model_final

Call:
lm(formula = PRICE ~ SQFT + FEATS, data = centered_covariates)

Coefficients:
(Intercept)         SQFT        FEATS
  1158.4127       0.6805      39.8369

> logmodel <- lm(log(PRICE)~.-AGE,data=centered_covariates)
> plot(logmodel,which = 2)
> require(nortest)
> lillie.test(logmodel$residuals)

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  logmodel$residuals
D = 0.071918, p-value = 0.5784

> shapiro.test(logmodel$residuals)

        Shapiro-Wilk normality test

data:  logmodel$residuals
W = 0.98721, p-value = 0.7584
```

In both tests pvalue is very large so we don't reject normality on residuals.

LINEARITY

```
> residualPlots(logmodel,plot=F,type='rstudent')
           Test stat Pr(>|Test stat|)
SQFT         -1.1428           0.2577
FEATS        -1.1617           0.2500
Tukey test   -0.8523           0.3941
>
```



HOMOSCEDASITY

```
> ncvTest(logmodel) # p value to big enough
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 1.347195, Df = 1, p = 0.24577
>
```

The p value is large, so we don't reject the constant variance hypothesis in log model(so the assumption of the homoscedasity has been fixed).

INDEPENDENCE

```
> durbinWatsonTest(logmodel)
 lag Autocorrelation D-W Statistic p-value
   1     -0.04969166      2.097419     0.8
 Alternative hypothesis: rho != 0
> runs.test(logmodel$residuals)

        Runs Test

data:  logmodel$residuals
statistic = 0, runs = 32, n1 = 31, n2 = 31, n = 62, p-value = 1
alternative hypothesis: nonrandomness

> dwtest(logmodel)

        Durbin-Watson test

data:  logmodel
DW = 2.0974, p-value = 0.6236
alternative hypothesis: true autocorrelation is greater than 0
```

As we can see using the log model fixed the 3 outcomes of the independence assumption.

Summary of the log transformation in our model

All the assumptions ended up with a high p value which means that the log transformation fixed all the mistakes.

```
> summary(logmodel)

Call:
lm(formula = log(PRICE) ~ . - AGE, data = centered_covariates)

Residuals:
      Min        1Q    Median        3Q       Max
-0.276296 -0.075079  0.008759  0.064148  0.310828

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.006e+00  1.359e-02 515.525   <2e-16 ***
SQFT        5.402e-04  2.903e-05  18.606   <2e-16 ***
FEATS       2.850e-02  1.153e-02   2.471   0.0163 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1079 on 60 degrees of freedom
Multiple R-squared:  0.8812,    Adjusted R-squared:  0.8772
F-statistic: 222.5 on 2 and 60 DF,  p-value: < 2.2e-16
```

Interpretation of the coefficients of our final model:

Intercept: The expected price of the house when the sqft =0 and has no feats is 7.006e+00 hundred dollars.

SQFT(b1): An increase of 1 unit in the size of the house means an increase by 0.054 % in its price.

FEATS(b2): A addition of a feature to the house will increase the price of the house by 2.85 % **only when the other characteristics will remain the same.**

QUESTION IX

Conduct LASSO as a variable selection technique and compare the variables that you end up having using LASSO to the variables that you ended up having using stepwise methods in (VI). Are you getting the same results? Comment.

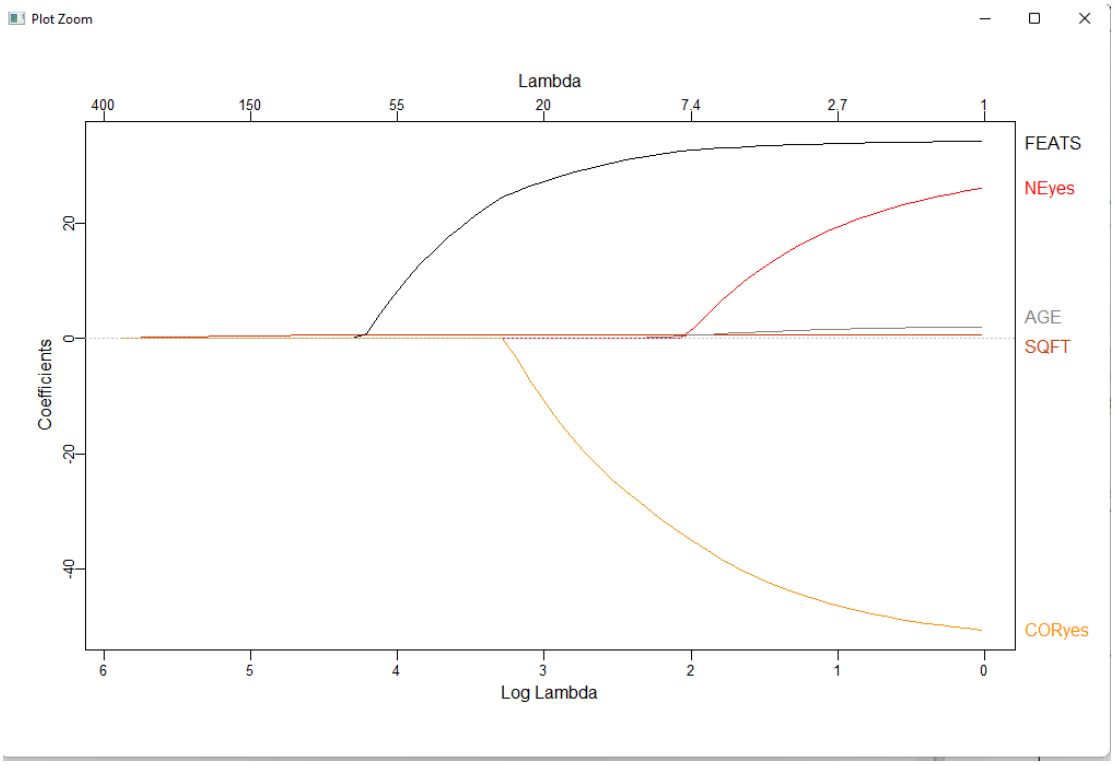
First of all, we need to create the design matrix.

```
x <- model.matrix(full_model)[,-1]
head(x,6)

lasso <-glmnet(x,centered_covariates$PRICE)
```

And now we ready to visualize it .

```
require(plotmo)

plot_glmnet(lasso)
```

Output



We see that the first coefficient that lasso aborts is the AGE for lambda ~ 2.5, the second is NEyes for lambda ~ 2.1, the third is CORyes for lambda ~ 3.2 and the last ones are the “FEATS” and the “SQFT” (this is the same order with the stepwise AIC).

Now in order to select the best lambda (that means best model) we need to do the “cross validation technique” .

```
# CROSS VALIDATION TO CHOOSE LAMBDA

lasso1 <- cv.glmnet(x,centered_covariates$PRICE, alpha = 1)

plot(lasso1)
lasso1$lambda.min
lasso1$lambda.1se
coef(lasso1, s = "lambda.min") # opou exei . exei fugei h metavliti
coef(lasso1, s = "lambda.1se")
plot(lasso1$glmnet.fit, xvar = "lambda")
abline(v=log(c(lasso1$lambda.min, lasso1$lambda.1se)), lty =2)
```

```
> lasso1$lambda.min
[1] 15.24343
> lasso1$lambda.1se
[1] 51.08993
> coef(lasso1, s = "lambda.min") # opou exei . exei fugei h metavliti
6 x 1 sparse Matrix of class "dgCMatrix"
                      s1
(Intercept) -90.2805702
SQFT          0.6576018
AGE           .
FEATS        29.2792674
NEyes         .
CORyes      -19.7009327
> coef(lasso1, s = "lambda.1se")
6 x 1 sparse Matrix of class "dgCMatrix"
                     s1
(Intercept) 69.8128873
SQFT         0.6059918
AGE          .
FEATS       10.2502735
NEyes        .
CORyes       .
```

So, we select the lambda “1se” = 51.08993 and we end up with the same model as the stepwise methods!