FinalPaper

Himanshu Saxena

April 25, 2019

1 (Bivariate Regression)

```r
#install.packages('tinytex')
#tinytex::install_tinytex()

#install.packages("Ecdat")
library(Ecdat)
```

```
## Warning: package 'Ecdat' was built under R version 3.5.3

## Loading required package: Ecfun

## Warning: package 'Ecfun' was built under R version 3.5.3

##
## Attaching package: 'Ecfun'

## The following object is masked from 'package:base':
##
##     sign

##
## Attaching package: 'Ecdat'

## The following object is masked from 'package:datasets':
##
##     Orange
```

```r
data(Housing)
summary(Housing)
```

```
##      price          lotsize         bedrooms        bathrms
##  Min.   : 25000   Min.   : 1650   Min.   :1.000   Min.   :1.000
##  1st Qu.: 49125   1st Qu.: 3600   1st Qu.:2.000   1st Qu.:1.000
##  Median : 62000   Median : 4600   Median :3.000   Median :1.000
##  Mean   : 68122   Mean   : 5150   Mean   :2.965   Mean   :1.286
##  3rd Qu.: 82000   3rd Qu.: 6360   3rd Qu.:3.000   3rd Qu.:2.000
##  Max.   :190000   Max.   :16200   Max.   :6.000   Max.   :4.000
##     stories       driveway  recroom   fullbase  gashw     airco
##  Min.   :1.000   no : 77   no :449   no :355   no :521   no :373
##  1st Qu.:1.000   yes:469   yes: 97   yes:191   yes: 25   yes:173
##  Median :2.000
##  Mean   :1.808
```

```
##   3rd Qu.:2.000
##   Max.   :4.000
##      garagepl      prefarea
##   Min.   :0.0000   no :418
##   1st Qu.:0.0000   yes:128
##   Median :0.0000
##   Mean   :0.6923
##   3rd Qu.:1.0000
##   Max.   :3.0000
```

we can observe that the variables price, lotsize, bedrooms, bathrms, stories and garagepl are numeric, while driveway, recroom, fullbase, gashw, arico and prefarea are factors with yes and no. We will encode yes as 1 and no as 0

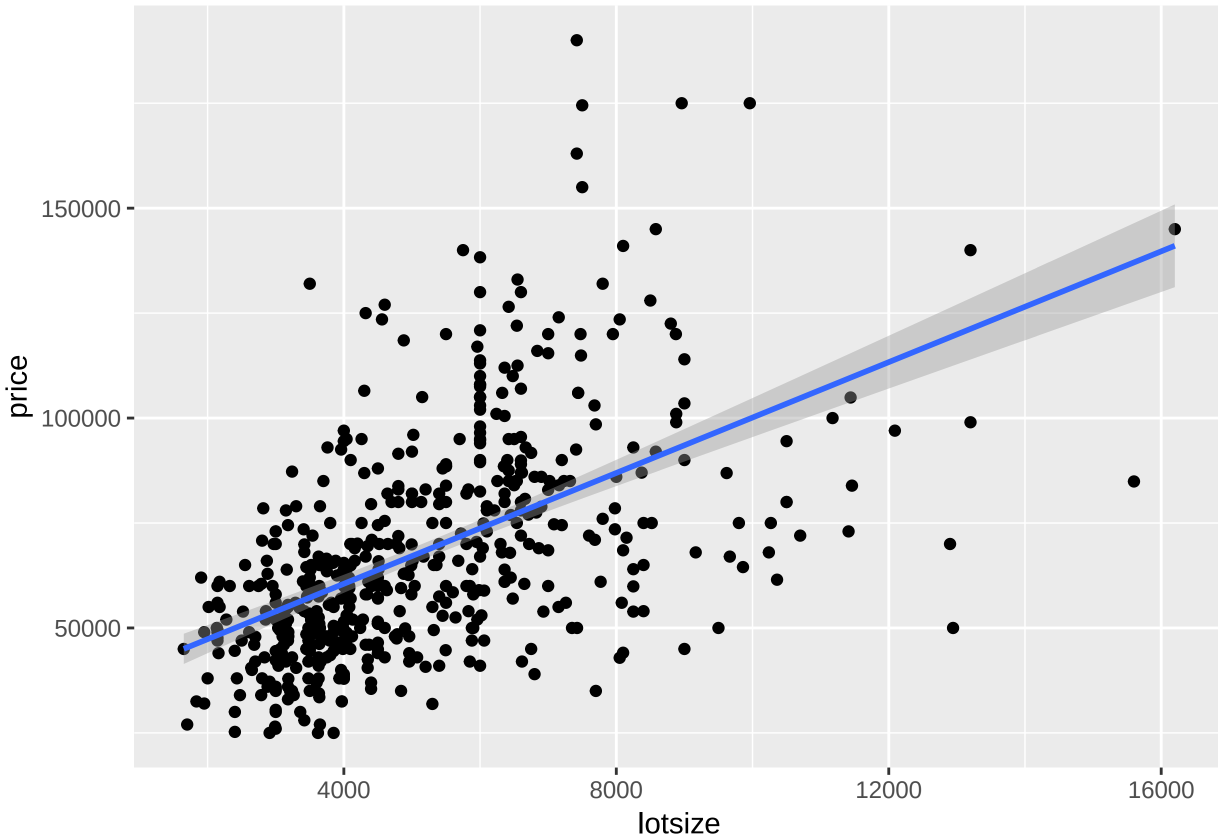
```r
Housing$driveway=ifelse(Housing$driveway=="yes",1,0)
Housing$recroom=ifelse(Housing$recroom=="yes",1,0)
Housing$fullbase=ifelse(Housing$fullbase=="yes",1,0)
Housing$gashw=ifelse(Housing$gashw=="yes",1,0)
Housing$airco=ifelse(Housing$airco=="yes",1,0)
Housing$prefarea=ifelse(Housing$prefarea=="yes",1,0)
```

Question 1

1. Using the Housing dataset, create a scatter plot of sale price of a house (y-axis) and the lot size of the property (x-axis). Use the ggplot function and include a regression line. Using the graph, describe the relation between the two variables.

Ans.

```r
library(ggplot2)

ggplot(Housing,aes(x=lotsize, y=price))+geom_point()+geom_smooth(method=lm)
```



The graph shows there is a positive corelation between lotsize and price. If lotsize will increase price will increase and vice versa.

Question 2

2. Estimate a bivariate regression of the sale price of a house on the lot size of the property. Interpret the estimated beta parameters, the statistical significance and R2.

Ans.

```
binary=lm(Housing$price~Housing$lotsize, data=Housing)
binary$coefficients
```

```
(Intercept) Housing$lotsize
```

34136.191565 6.598768

```
suppressMessages(library(stargazer))
stargazer(binary, type="latex", title = "Summary of binary regression",
          dep.var.labels = "price", header = FALSE, float=FALSE)
```

	Dependent variable:
	price
lotsize	6.599***
	(0.446)
Constant	34,136.190***
	(2,491.064)
Observations	546
R^2	0.287
Adjusted R^2	0.286
Residual Std. Error	22,567.050 (df = 544)
F Statistic	219.056*** (df = 1; 544)
Note:	*p<0.1; **p<0.05; ***p<0.01

The value of beta_0 is 34136.19 and value of beta_1 is 6.598 which means that for a unit increase in lotsize the price will increase by 6.598

The F-statistic is 219.1, Degree of freedom is 544. The lotsize variable is statistically significant as its p value is very low

The value of adjusted R square is 0.286. Thus only about 28.6% of the variation in price can be attributed to variation in lotsize

Question 3

3. Is there any reason to believe that the estimated slope parameter in the previous regression is biased? (Explain)

Ans.

Yes, the slope in question 2 is biased because of following reasons: 1. The bivariate regression is linear while the points are not arranged in linear fashion. 2. The bivariate regression explains only 28.58% variance in the model. 3. The points are arranged in a non linear fashion, so a bivariate linear regression model is a bad fit to the data.

2 (Multivariate Regression)

Question 4

4. Using the rest of the variables in the dataset, construct a correlation matrix and use it to check if the assumption of exogeneity is valid in the estimated model in question (2). (Explain)

Ans.

```
# creating a correlation matrix to check for exogeneity


#Computing correlation with price and lotsize only
corMat = cor(Housing, cbind(Housing$price, Housing$lotsize))
```

```r
corMat = cbind(corMat, corMat[,1]*corMat[,2])
colnames(corMat) = c("Price", "LotSize", "PriceXLotSize")

#Sorting data from more to less correlated
corMat = corMat[order(-abs(corMat[,3])),]
# correlation table
corMat
```

```
##               Price      LotSize PriceXLotSize
## price    1.00000000  0.535795672  0.5357956724
## lotsize  0.53579567  1.000000000  0.5357956724
## garagepl 0.38330199  0.352871658  0.1352564095
## airco    0.45334656  0.221764888  0.1005363493
## bathrms  0.51671925  0.193833484  0.1001574933
## driveway 0.29716682  0.288777751  0.0858151653
## prefarea 0.32907432  0.234782230  0.0772608029
## bedrooms 0.36644736  0.151851492  0.0556455782
## recroom  0.25495955  0.140327323  0.0357777908
## stories  0.42119023  0.083674995  0.0352430904
## fullbase 0.18621767  0.047486731  0.0088428685
## gashw    0.09283654 -0.009200907 -0.0008541804
```

- The exogeneity assumption will not be satisfied if there are variables which violate the exogeneity assumption. That is if they are highly correlated with price and lotsize. Therefore I checked if there are any variables which are highly correlated. I can see from the matrix that variable garagepl, airco, bathrms, driveway, prefarea, bedrooms, recroom are correlated with both price and lotsize. Hence, we can say that the previous regression is biased
- Also, stories is not correlated to lotsize, fullbase is not correlated to lotsize, and gashw is not correlated to both price and lotsize.

Question 5

5. Estimate a set of multivariate models to address the potential issue of OVB, adding at most one additional variable each time. Display all the estimated models side-by-side (you may need two or more stargazer tables here). Using the multivariate models, do you think there is evidence that the estimated parameter in (2) was biased? which of the estimated models you consider the least bias (from now own, we'll call this model the best model)?

Ans. I added variables in order displayed in the product of price and lot size in the previous table.

```r
suppressMessages(library(stargazer))

binary=lm(Housing$price~Housing$lotsize, data=Housing)

lm2=lm(Housing$price~Housing$lotsize+Housing$garagepl,
       data=Housing)

lm3=lm(Housing$price~Housing$lotsize+Housing$garagepl+
         Housing$airco, data=Housing)

lm4=lm(Housing$price~Housing$lotsize+Housing$garagepl+
```

```r
         Housing$airco+Housing$bathrms, data=Housing)

lm5=lm(Housing$price~Housing$lotsize+Housing$garagepl+
         Housing$airco+Housing$bathrms+Housing$driveway,
       data=Housing)

lm6=lm(Housing$price~Housing$lotsize+Housing$garagepl+
         Housing$airco+Housing$bathrms+Housing$driveway+
         Housing$prefarea, data=Housing)

lm7=lm(Housing$price~Housing$lotsize+Housing$garagepl+
         Housing$airco+Housing$bathrms+Housing$driveway+
         Housing$prefarea+Housing$bedrooms, data=Housing)

lm8=lm(Housing$price~Housing$lotsize+Housing$garagepl+
         Housing$airco+Housing$bathrms+Housing$driveway+
         Housing$prefarea+Housing$bedrooms+Housing$recroom, data=Housing)

lm9=lm(Housing$price~Housing$lotsize+Housing$garagepl+
         Housing$airco+Housing$bathrms+Housing$driveway+
         Housing$prefarea+Housing$bedrooms+Housing$recroom+
         Housing$stories, data=Housing)

lm10=lm(Housing$price~Housing$lotsize+Housing$garagepl+
          Housing$airco+Housing$bathrms+Housing$driveway+
          Housing$prefarea+Housing$bedrooms+Housing$recroom+
          Housing$stories+ Housing$fullbase, data=Housing)

lm11=lm(Housing$price~Housing$lotsize+Housing$garagepl+
          Housing$airco+Housing$bathrms+Housing$driveway+
          Housing$prefarea+Housing$bedrooms+Housing$recroom+
          Housing$stories+ Housing$fullbase+ Housing$gashw, data=Housing)


stargazer(list(binary, lm2, lm3) , type = "latex", title =
            "Multiple variate regression of price (1/4)", header=FALSE,
          column.sep.width = "-15pt",font.size = "small",
          dep.var.labels = "price", float=FALSE)
```

	Dependent variable:		
	price		
	(1)	(2)	(3)
lotsize	6.599***	5.635***	4.847***
	(0.446)	(0.462)	(0.431)
garagepl		6,878.237***	5,946.030***
		(1,163.740)	(1,072.100)
airco			19,268.380***
			(1,902.763)
Constant	34,136.190***	34,340.150***	32,934.040***
	(2,491.064)	(2,417.072)	(2,222.856)
Observations	546	546	546
R^2	0.287	0.330	0.437
Adjusted R^2	0.286	0.328	0.434
Residual Std. Error	22,567.050 (df = 544)	21,894.510 (df = 543)	20,095.930 (df = 542)
F Statistic	219.056*** (df = 1; 544)	133.827*** (df = 2; 543)	140.085*** (df = 3; 542)
Note:			*p<0.1; **p<0.05; ***p<0.01

```r
stargazer(list(lm4, lm5, lm6) , type = "latex", title =
            "Multiple variate regression of price (2/4)", header=FALSE,
          column.sep.width = "-15pt",font.size = "small",
          dep.var.labels = "price", float=FALSE)
```

	Dependent variable:		
	price		
	(1)	(2)	(3)
lotsize	4.287***	3.885***	3.496***
	(0.382)	(0.386)	(0.379)
garagepl	4,651.574***	4,168.203***	4,236.784***
	(949.676)	(938.945)	(908.370)
airco	16,298.270***	15,993.610***	15,402.600***
	(1,692.126)	(1,663.580)	(1,612.137)
bathrms	19,671.880***	19,911.410***	19,782.560***
	(1,565.171)	(1,538.420)	(1,488.359)
driveway		10,220.790***	8,328.955***
		(2,249.915)	(2,197.989)
prefarea			10,911.740***
			(1,768.942)
Constant	12,364.070***	5,781.983**	7,157.513**
	(2,551.472)	(2,895.059)	(2,809.440)
Observations	546	546	546
R^2	0.564	0.580	0.608
Adjusted R^2	0.561	0.576	0.603
Residual Std. Error	17,696.170 (df = 541)	17,383.490 (df = 540)	16,816.170 (df = 539)
F Statistic	174.983*** (df = 4; 541)	149.195*** (df = 5; 540)	139.201*** (df = 6; 539)
Note:			*p<0.1; **p<0.05; ***p<0.01

```
stargazer(list(lm7, lm8, lm9) , type = "latex", title =
            "Multiple variate regression of price (3/4)", header=FALSE,
          column.sep.width = "-15pt",font.size = "small",
          dep.var.labels = "price", float = FALSE)
```

	Dependent variable:		
	price		
	(1)	(2)	(3)
lotsize	3.400***	3.316***	3.440***
	(0.373)	(0.370)	(0.358)
garagepl	4,009.048***	4,121.579***	4,559.991***
	(893.837)	(885.966)	(857.955)
airco	14,814.050***	14,349.720***	11,906.240***
	(1,589.164)	(1,580.061)	(1,572.891)
bathrms	17,433.230***	17,013.920***	15,175.730***
	(1,551.794)	(1,542.058)	(1,516.323)
driveway	9,048.952***	8,759.434***	6,840.416***
	(2,165.250)	(2,146.379)	(2,093.685)
prefarea	10,554.680***	9,854.542***	10,115.210***
	(1,739.668)	(1,735.582)	(1,675.765)
bedrooms	4,734.667***	4,671.830***	2,440.751**
	(1,047.159)	(1,037.370)	(1,061.108)
recroom		6,364.557***	6,846.258***
		(1,886.790)	(1,822.794)
stories			5,781.239***
			(909.938)
Constant	−3,556.794	−3,049.459	−3,187.969
	(3,637.783)	(3,606.332)	(3,481.065)
Observations	546	546	546
R^2	0.622	0.630	0.656
Adjusted R^2	0.617	0.624	0.650
Residual Std. Error	16,520.830 (df = 538)	16,363.750 (df = 537)	15,795.040 (df = 536)
F Statistic	126.540*** (df = 7; 538)	114.281*** (df = 8; 537)	113.515*** (df = 9; 536)
Note:			*p<0.1; **p<0.05; ***p<0.01

```
stargazer(list(lm10, lm11) , type = "latex", title =
            "Multiple variate regression of price (4/4)", header=FALSE,
          column.sep.width = "-15pt",font.size = "small",
          dep.var.labels = "price", float=FALSE)
```

	Dependent variable:	
	price	
	(1)	(2)
lotsize	3.536***	3.546***
	(0.355)	(0.350)
garagepl	4,512.089***	4,244.829***
	(849.458)	(840.544)
airco	11,693.250***	12,632.890***
	(1,558.328)	(1,555.021)
bathrms	14,677.400***	14,335.560***
	(1,508.028)	(1,489.921)
driveway	6,638.478***	6,687.779***
	(2,073.499)	(2,045.246)
prefarea	9,007.644***	9,369.513***
	(1,689.676)	(1,669.091)
bedrooms	1,919.541*	1,832.003*
	(1,061.250)	(1,047.000)
recroom	4,519.340**	4,511.284**
	(1,926.238)	(1,899.958)
stories	6,678.946***	6,556.946***
	(937.576)	(925.290)
fullbase	5,558.221***	5,452.386***
	(1,609.766)	(1,588.024)
gashw		12,831.410***
		(3,217.597)
Constant	−4,115.163	−4,038.350
	(3,456.578)	(3,409.471)
Observations	546	546
R^2	0.663	0.673
Adjusted R^2	0.657	0.666
Residual Std. Error	15,636.530 (df = 535)	15,423.190 (df = 534)
F Statistic	105.437*** (df = 10; 535)	99.968*** (df = 11; 534)
Note:		*p<0.1; **p<0.05; ***p<0.01

- All multivariate regressions show that the estimated paramter in the bivariate model was biased.
- Regressions lm9-lm11 don't correct much OVB, the price is virtually the same for those models. Therefore, we are confident that after correcting for OVB the estimated paramter is about 3.4 (0.35). Therefore, if the lotsize increase by 1 unit, the expected price will increase by 3.4 units
- The bivariate model greatly exaggerates the effect of lotsize; in the original estimation if lot size increases by 1 unit the price goes up by 6.59 units. That is, in the bivariate model the effect of lotsize is more important than what it actually is after controlling for other relevant factors
- The adj-R2 greatly improves after adding more controls.
- Adding gaswh didn't cause any change in the estimated value of beta 1. Which means that we can

probably exclude this variable from the regression. Then, we'll continue our analysis assuming that regression (10) lm10 correctly controlled for OVB. We will call lm10 our best model

Question 6

6. Check if the best model suffers from multicollinearity (if it does, don't try to fix it, just explain? what problems it may cause).

Ans.

```
# Computing VIF for model (10) lm10

# Running auxiliary regressions
aux1_lm10 = lm(lotsize~garagepl+airco+bathrms+driveway+prefarea+
                 bedrooms+recroom+stories+fullbase, data=Housing)

aux2_lm10 = lm(garagepl~lotsize+airco+bathrms+driveway+prefarea+
                 bedrooms+recroom+stories+fullbase, data=Housing)

aux3_lm10 = lm(airco~lotsize+garagepl+bathrms+driveway+prefarea+
                 bedrooms+recroom+stories+fullbase, data=Housing)

aux4_lm10 =lm(bathrms~lotsize+garagepl+airco+driveway+prefarea+
                bedrooms+recroom+stories+fullbase, data=Housing)

aux5_lm10 = lm(driveway~lotsize+garagepl+airco+bathrms+prefarea+
                 bedrooms+recroom+stories+fullbase, data=Housing)

aux6_lm10 = lm(prefarea~lotsize+garagepl+airco+bathrms+driveway+
                 bedrooms+recroom+stories+fullbase, data=Housing)

aux7_lm10 = lm(bedrooms~lotsize+garagepl+airco+bathrms+driveway+
                 prefarea+recroom+stories+fullbase, data=Housing)

aux8_lm10 = lm(recroom~lotsize+garagepl+airco+bathrms+driveway+
                 prefarea+bedrooms+stories+fullbase, data=Housing)

aux9_lm10 = lm(stories~lotsize+garagepl+airco+bathrms+driveway+
                 prefarea+bedrooms+recroom+fullbase, data=Housing)

aux10_lm10 = lm(fullbase~lotsize+garagepl+airco+bathrms+driveway+
                  prefarea+bedrooms+recroom+stories, data=Housing)

# Getting r2
aux1_r2 = summary(aux1_lm10)$r.squared
aux2_r2 = summary(aux2_lm10)$r.squared
aux3_r2 = summary(aux3_lm10)$r.squared
aux4_r2 = summary(aux4_lm10)$r.squared
aux5_r2 = summary(aux5_lm10)$r.squared
aux6_r2 = summary(aux6_lm10)$r.squared
aux7_r2 = summary(aux7_lm10)$r.squared
aux8_r2 = summary(aux8_lm10)$r.squared
aux9_r2 = summary(aux9_lm10)$r.squared
```

```r
aux10_r2 = summary(aux10_lm10)$r.squared

# Computing VIF
aux1_vif = 1 / (1 - aux1_r2)
aux2_vif = 1 / (1 - aux2_r2)
aux3_vif = 1 / (1 - aux3_r2)
aux4_vif = 1 / (1 - aux4_r2)
aux5_vif = 1 / (1 - aux5_r2)
aux6_vif = 1 / (1 - aux6_r2)
aux7_vif = 1 / (1 - aux7_r2)
aux8_vif = 1 / (1 - aux8_r2)
aux9_vif = 1 / (1 - aux9_r2)
aux10_vif = 1 / (1 - aux10_r2)

vifs = c(aux1_vif, aux2_vif, aux3_vif, aux4_vif,aux5_vif,aux6_vif,
         aux7_vif, aux8_vif, aux9_vif,aux10_vif)
vifs
```

```
##  [1] 1.321558 1.193205 1.173814 1.278248 1.163049 1.144247 1.365032
##  [8] 1.210499 1.476968 1.316175
```

```r
# Testing if VIF are greater than 10
vifs > 10
```

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

```r
# Testing if VIF are greater than 5
vifs > 5
```

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

Because for all regressors VIF is less than 5 we can be confident that imperfect multicollienarity is not an issue in regression (10) lm10

3 (Non-linear Functional Forms)

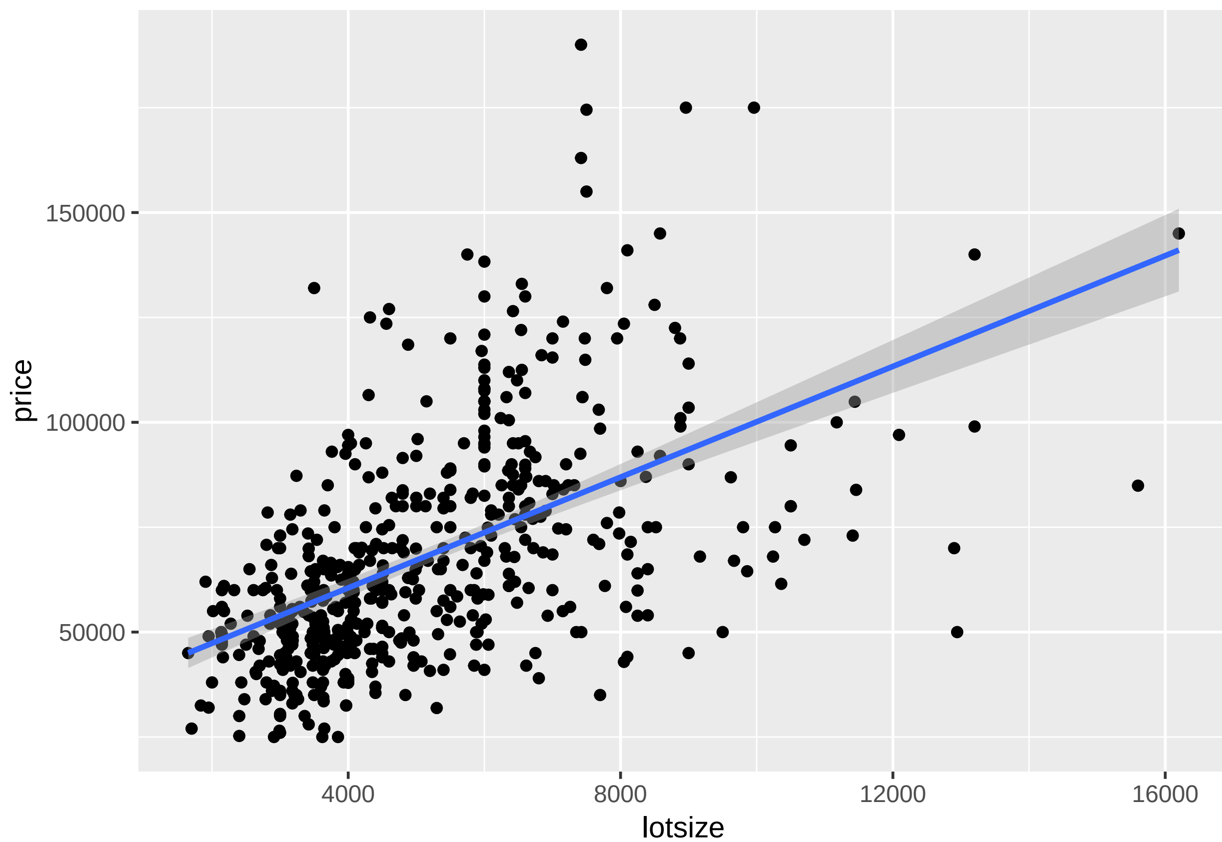
Question 7

7. Take a look at the graph from part (1), do you think there is any reason to believe that the effect of lot size on price is not the same for all the domain of lot size? if yes, is the effect increasing or decreasing?

Ans.

Looking at the graph again:

```r
ggplot(Housing,aes(x=lotsize, y=price))+geom_point()+geom_smooth(method=lm)
```



- We can see that the effect of lot size on price is not the same for all the domain of lot sizes. The slope of the regression line is not constant for all values of lotsizes.
- The effect of lot sizes from 0 to 6000 on price is linear with a slope, lets call it 's1'. But the effect of lotsize on price increases the slope (s2) after lotsize of 6000. Thus slop tends to increase (s2>s1)
- Also, if the other lower points are considered, the effect of lot size on price tends to decrease, i.e. s3<s1.
- This confirms the presence of non linearity in the regression model.

Question 8

8. Estimate the best model again, but this time transform the lot size variable to natural logarithms. Interpret the estimated parameter for log of the lot size.

Ans.

```
#lets look at the bet model again:

lm10=lm(Housing$price~Housing$lotsize+Housing$garagepl+Housing$airco+
          Housing$bathrms+Housing$driveway+Housing$prefarea+
          Housing$bedrooms+Housing$recroom+Housing$stories+
          Housing$fullbase, data=Housing)

# transformation

reg10=lm(Housing$price~I(log(Housing$lotsize))+Housing$garagepl+
```

```
           Housing$airco+Housing$bathrms+Housing$driveway+
           Housing$prefarea+Housing$bedrooms+Housing$recroom+
           Housing$stories+ Housing$fullbase, data=Housing)

#output of reg10

stargazer(reg10, title =
            "Output of reg10)", header=FALSE, dep.var.labels = "price",
          type = "latex", float=FALSE)
```

	Dependent variable:
	price
lotsize))	20,275.100***
	(1,973.839)
garagepl	4,423.724***
	(845.955)
airco	11,005.240***
	(1,560.052)
bathrms	14,610.110***
	(1,500.763)
driveway	5,456.694***
	(2,087.240)
prefarea	9,442.816***
	(1,673.835)
bedrooms	1,982.153*
	(1,055.310)
recroom	3,552.636*
	(1,926.557)
stories	6,610.003***
	(932.302)
fullbase	5,844.643***
	(1,603.837)
Constant	−156,277.100***
	(15,997.270)
Observations	546
R^2	0.667
Adjusted R^2	0.661
Residual Std. Error	15,558.410 (df = 535)
F Statistic	107.037*** (df = 10; 535)
Note:	*p<0.1; **p<0.05; ***p<0.01

A 1% increase in lotsizes causes a change in price of 0.01 X 20275.1 = 202.75

Question 9

9. Estimate the best model twice: (a) first, adding a quadratic term for lot size, and, (b) second, adding a quadratic and cubic terms. Using the change in lot size as a one standard deviation change from the mean, compare the effect of lot size in the original model, model (a), and, model (b). Can you reject the hypothesis that the relation between lot size and price is linear? quadratic? cubic? (Explain)

```r
#(a) first, adding a quadratic term for lot size

quad= lm(Housing$price~Housing$lotsize+Housing$garagepl+Housing$airco+
           Housing$bathrms+Housing$driveway+Housing$prefarea+
           Housing$bedrooms+Housing$recroom+Housing$stories+
           Housing$fullbase +I(Housing$lotsize^2), data=Housing)

#(b) second, adding a quadratic and cubic terms
quad_cubic=lm(Housing$price~Housing$lotsize+Housing$garagepl+
                Housing$airco+Housing$bathrms+Housing$driveway+
                Housing$prefarea+Housing$bedrooms+Housing$recroom+
                Housing$stories+ Housing$fullbase+I(Housing$lotsize^2)
              +I(Housing$lotsize^3), data=Housing)

#change in lot size as a one standard deviation change from the mean
mean(Housing$lotsize)
```

[1] 5150.266

```r
sd(Housing$lotsize)
```

[1] 2168.159

```r
initial_lotsize=mean(Housing$lotsize)
final_lotsize=initial_lotsize+sd(Housing$lotsize)

#output
stargazer(list(lm10, quad, quad_cubic), title =
            "Output", header=FALSE, dep.var.labels = "price",
          type = "latex", float=FALSE)
```

	Dependent variable:		
	price		
	(1)	(2)	(3)
lotsize	3.536***	5.807***	13.067***
	(0.355)	(1.246)	(3.605)
garagepl	4,512.089***	4,372.290***	4,573.452***
	(849.458)	(850.577)	(852.894)
airco	11,693.250***	11,249.060***	10,972.360***
	(1,558.328)	(1,572.003)	(1,572.029)
bathrms	14,677.400***	14,633.540***	14,493.390***
	(1,508.028)	(1,504.536)	(1,500.910)
driveway	6,638.478***	6,046.883***	5,364.004**
	(2,073.499)	(2,091.739)	(2,108.884)
prefarea	9,007.644***	8,966.316***	9,631.885***
	(1,689.676)	(1,685.705)	(1,708.457)
bedrooms	1,919.541*	1,975.094*	1,842.208*
	(1,061.250)	(1,059.071)	(1,057.333)
recroom	4,519.340**	3,872.181**	3,621.405*
	(1,926.238)	(1,951.481)	(1,948.443)
stories	6,678.946***	6,570.846***	6,698.696***
	(937.576)	(937.022)	(935.778)
fullbase	5,558.221***	5,659.918***	5,928.450***
	(1,609.766)	(1,606.740)	(1,606.234)
lotsizeˆ2)		−0.0002*	−0.001**
		(0.0001)	(0.001)
lotsizeˆ3)			0.00000**
			(0.00000)
Constant	−4,115.163	−9,710.394**	−23,003.860***
	(3,456.578)	(4,533.761)	(7,669.316)
Observations	546	546	546
R^2	0.663	0.666	0.669
Adjusted R^2	0.657	0.659	0.661
Residual Std. Error	15,636.530 (df = 535)	15,598.480 (df = 534)	15,546.140 (df = 533)
F Statistic	105.437*** (df = 10; 535)	96.648*** (df = 11; 534)	89.576*** (df = 12; 533)
Note:			*p<0.1; **p<0.05; ***p<0.01

- in lm10 estimate for lotsize = 3.535
- in quad estimate for lotsize = 5.807, estimate for lotsize2 = 0.000165 = 0

- in quad_cubic estimate for lotsize = 13.07, estimate for lotsize2 = -0.001244, estimate for lotsize3 = 0

We can see that-

$$y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \varepsilon$$

$$\frac{d(y)}{d(x)} = \beta_1 + 2\beta_2 X + 3\beta_3 X^2$$

$$\frac{\Delta(y)}{\Delta(x)} = \beta_1 + 2\beta_2 X + 3\beta_3 X^2$$

$$\Delta(y) = \Delta(x).(\beta_1 + 2\beta_2 X + 3\beta_3 X^2)$$

For the best model (lm10), the effect of lot size from mean to mean+sd is:

$$\Delta(y) = \Delta(x).(\beta_1)$$

Which is equal to 2168.159X3.535= 7664.44. Thus the price will increase by 7664.44 units for 1 std deviation change from mean in lot size.

For quad model, the effect of change in ot size by 1 std dev from mean is :

$$\Delta(y) = \Delta(x).(\beta_1 + 2\beta_2 X)$$

which is equal to 2168.159X(5.807+2X0)= 2168.159X5.807= 12590.499. Thus the price will increase by 12590.499 units for 1 std deviation change from mean in lot size.

For quad_cubic model, the effect of change in ot size by 1 std dev from mean is :

$$\Delta(y) = \Delta(x).(\beta_1 + 2\beta_2 X + 3\beta_3 X^2)$$

Which is equal to 2168.159X(13.07+2X0+3X0)= 28337.83. Thus the price will increase by 28337.83 units for 1 std deviation change from mean in lot size.

Thus, we can see that the estimate (beta1) of lotsize changes as other terms are introduced. I would conclude that the relation between lot size and price is not linear and thus would reject the null hypothesis.

Question 10

10. Using the best model as the nested model, test the hypothesis that the effect of lot size on price is moderated by prefarea.

Ans. One way to test this is to add an interaction between lotsize and prefarea. We will use best model (lm10) as reference:

```r
# Running regression
qi=lm(Housing$price~Housing$lotsize+Housing$garagepl+Housing$airco+
        Housing$bathrms+Housing$driveway+Housing$prefarea+
        Housing$bedrooms+Housing$recroom+Housing$stories+
        Housing$fullbase+Housing$lotsize*Housing$prefarea, data=Housing)

# Display regression with stargazer
stargazer(qi, type="latex", header = FALSE, title = "Output of qi",
          dep.var.labels = "price", float=FALSE)
```

	Dependent variable:
	price
lotsize	3.194***
	(0.417)
garagepl	4,418.705***
	(850.434)
airco	11,643.810***
	(1,556.569)
bathrms	14,732.770***
	(1,506.432)
driveway	7,137.239***
	(2,095.281)
prefarea	2,665.639
	(4,402.814)
bedrooms	1,977.337*
	(1,060.480)
recroom	4,662.496**
	(1,925.854)
stories	6,708.649***
	(936.517)
fullbase	5,384.786***
	(1,611.457)
prefarea	1.104
	(0.708)
Constant	−3,049.267
	(3,518.973)
Observations	546
R^2	0.665
Adjusted R^2	0.658
Residual Std. Error	15,615.640 (df = 534)
F Statistic	96.330*** (df = 11; 534)
Note:	*p<0.1; **p<0.05; ***p<0.01

The parameter for interaction term of lotsize and prefarea is statically significant at 1.1045 .

To test if the interaction should be part of the model, we can conduct an f-test. Note: lm10 is the nested version of this model.

```
anova(lm10,qi)
```

```
## Analysis of Variance Table
```

```
##
## Model 1: Housing$price ~ Housing$lotsize + Housing$garagepl + Housing$airco +
##     Housing$bathrms + Housing$driveway + Housing$prefarea + Housing$bedrooms +
##     Housing$recroom + Housing$stories + Housing$fullbase
## Model 2: Housing$price ~ Housing$lotsize + Housing$garagepl + Housing$airco +
##     Housing$bathrms + Housing$driveway + Housing$prefarea + Housing$bedrooms +
##     Housing$recroom + Housing$stories + Housing$fullbase + Housing$lotsize *
##     Housing$prefarea
##   Res.Df        RSS Df Sum of Sq      F Pr(>F)
## 1    535 1.3081e+11
## 2    534 1.3021e+11  1 593069912 2.4321 0.1195
```

p-value is greater than 0.05, thus we cant reject the hypothesis that the effect of lot size on price is moderated by prefarea. We dont need to include the interation term in the model.

4 (Unsupervised Machine Learning)

Question 11

11. Run a factor analysis or PCA on the Housing dataset, examine the loadings of the factors on the variables. Sort the variables by their loadings, and try to interpret what the first one mean.

Ans.

```r
#install.packages("GPArotation")
library(GPArotation)
library(psych)
```

```
## Warning: package 'psych' was built under R version 3.5.3

##
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
fact=fa(Housing,nfactors = 2)
fact1=fact$loading[,1]
fact1[order(fact1)]
```

```
##       gashw     stories    bedrooms     bathrms       airco    driveway
## 0.007361468 0.100426460 0.232825159 0.351343719 0.361199509 0.381280336
##    fullbase     recroom    garagepl    prefarea     lotsize       price
## 0.391277143 0.400448034 0.423111330 0.461010507 0.608631576 0.906643383
```

Looking at the variables of the first factor and after ordering them we can see that on the higher value side it is prefarea, lotsize and price. While on the lower value side it is gashw, stories and bedrooms. Thus, it can be said that while looking for housing at one side people look for facilies like gas connection in house, the number of stories and bedrooms in the house. While on the other side people also prefer looking for If the house is located in a preferred neighborhood, the size of the lot and also the price for the house

```r
#similar analysis can be done for the second factor

fact2=fact$loading[,2]
fact2[order(fact2)]
```

```
##    fullbase    prefarea     recroom     lotsize    driveway    garagepl
## -0.35485404 -0.23539311 -0.21106894 -0.13008623 -0.12774952 -0.05936481
##       gashw       price       airco     bathrms    bedrooms     stories
##  0.05304230  0.14348784  0.15386980  0.31140542  0.39043088  0.70766815
```

The second factor has higher value variables such as bathrooms, bedrooms and stories. The lower value variables are recroom, preferred area and fullbase. Thus it can be said that while looking for houses there are few people who look more for number of bedrooms, bathrooms and stories versus people who look for area preferance, fully furnished basement and presence of recreational rooms

Question 12

12. Use k-means algorithm and examine the centers of each cluster using only two centroids. How are they similar to and different from the factor loadings of the first factor?

Ans.

```r
set.seed(1)

kmout=kmeans(Housing,nstart=25, centers = 2)


centroids=kmout$centers
topvars_centroid1=centroids[1,order(centroids[1,])]
topvars_centroid2=centroids[2,order(centroids[2,])]

tail(topvars_centroid1)
```

```
##     garagepl      bathrms      stories     bedrooms      lotsize
## 1.036364e+00 1.624242e+00 2.369697e+00 3.333333e+00 6.650121e+03
##        price
## 1.007733e+05
```

```r
tail(topvars_centroid2)
```

```
##     driveway      bathrms      stories     bedrooms      lotsize
## 8.031496e-01 1.139108e+00 1.564304e+00 2.805774e+00 4.500722e+03
##        price
## 5.398110e+04
```

After taking out the values for the clustering centre 1 and 2. We can see centre 1 consists of variables like garagepl, bathrms, stories, bedrooms, lotsize and price, which means this center is mostly the cluster of things whose number matters. Like number of garageplaces, number of bedrooms, number of stories and number of bedrooms along with the area of the lot and price of it.

Very interestingly the center 2 also specifies the same things as center 1, while except for garagepl it has driveway as an addition. Where people try to find if there is a driveway in the house.

Similarity between center 1,2 and factor 1

1. Both the centers consists of cluster of things whose number matters like number of bathrooms, number of stories and number of bedroom, along with size of the lot, price of the lot and presence of a driveway.
2. The first factor has prefarea, lotsize and price towards the higher value side. The higher value side matches with both the centers.

Difference between center 1,2 and factor 1

1. The lower value side of the factor consists of gashw, stories and bedrooms. Though the centers have the variables stories and bedrooms, the variable gashw is missing in both center 1 and center 2.
2. Center 1 and 2 also do not contain variables like airco, fullbase and recroom which are there in factor 1

Thus, it can be said that factor 1 presents 2 groups of people according to preferences. The first group who focuses more on preferred area, the size of the lot and price of house versus the second group who focuses more on heating facilites, number of stories and number of bedrooms.

While the centers present cluster of people for whom the number of stories, bedrooms and bathrooms matter the most along with lotsize and price.

5 (Supervised Machine Learning)

Question 13

13. Divide the Housing data into two equally sized samples (one for training and one for testing). The dependent variable is price. Using the training sample, estimate a ridge model using the Housing dataset and find the optimal value of lambda.

Ans.

```
#divide the data into train and test set

suppressMessages(library(caTools))
```

```
## Warning: package 'caTools' was built under R version 3.5.3
```

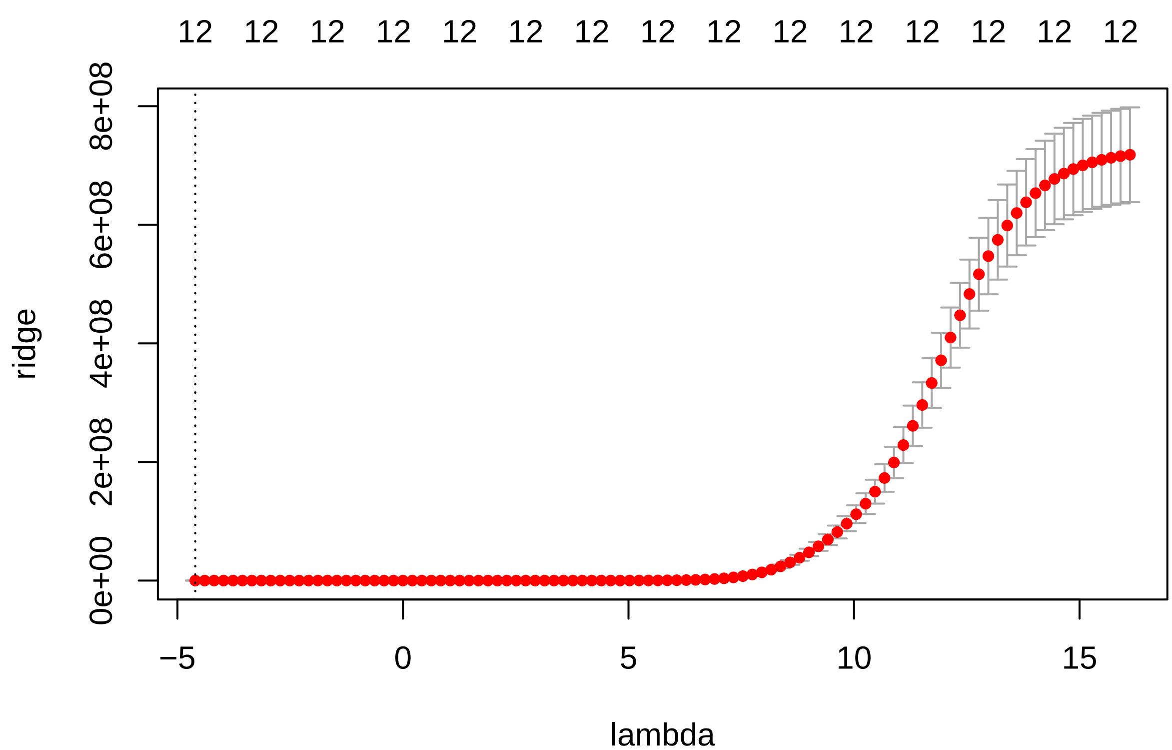
```
set.seed(123)
split=sample.split(Housing$lotsize, SplitRatio = 0.5)
in_sample=subset(Housing, split==TRUE)
out_sample=subset(Housing, split==FALSE)

is=as.matrix(in_sample)
os=as.matrix(out_sample)
y=in_sample$price
y_os=out_sample$price
sum(is.na(y))
```

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

```r
lambdalevels = 10^seq(7,-2,length=100)

#install.packages("glmnet")
suppressMessages(library(glmnet))
```

```
## Warning: package 'glmnet' was built under R version 3.5.3

## Warning: package 'foreach' was built under R version 3.5.3
```

```r
ridge = cv.glmnet(is, y, alpha = 0, lambda = lambdalevels)

plot(ridge, xlab= "lambda", ylab = "ridge")
```




```r
lambdaRidge = ridge$lambda.min # Getting optimal lambda
lambdaRidge
```

```
## [1] 0.01
```

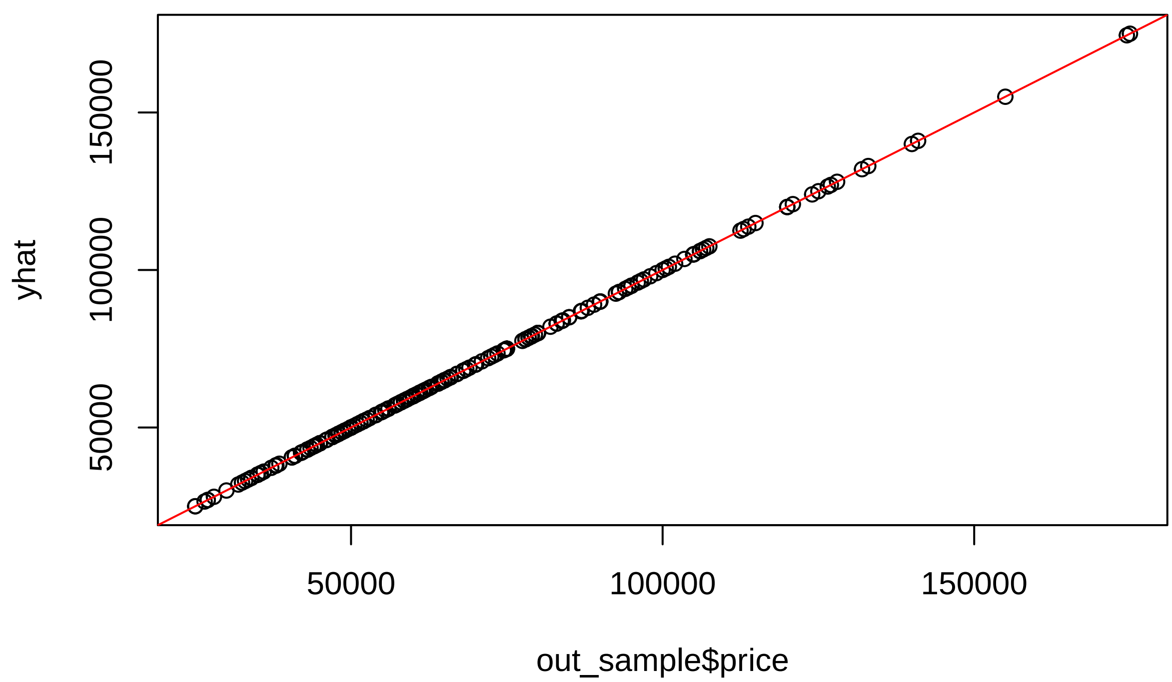
Thus, optimal value of lambda is 0.01

Question 14

14. How does the model performs in the testing sample? Compare the results of the ridge model with a linear regression. Which model performs best?

Ans.

```r
# Predicted values
yhat = predict(ridge$glmnet.fit, s = lambdaRidge, newx = os)

# Let's take a look at how the prediction looks like
plot(out_sample$price, yhat )+abline(0, 1, col = "red")
```




```
## integer(0)
```

```r
# Adding 45 degree line
```

The predictions look good, thus we can say that the model performs good in the testing sample.

```r
# Computing testing sample MSE
ridgeMSE = (1/length(out_sample$price))*sum((out_sample$price - yhat)^2)
ridgeMSE
```

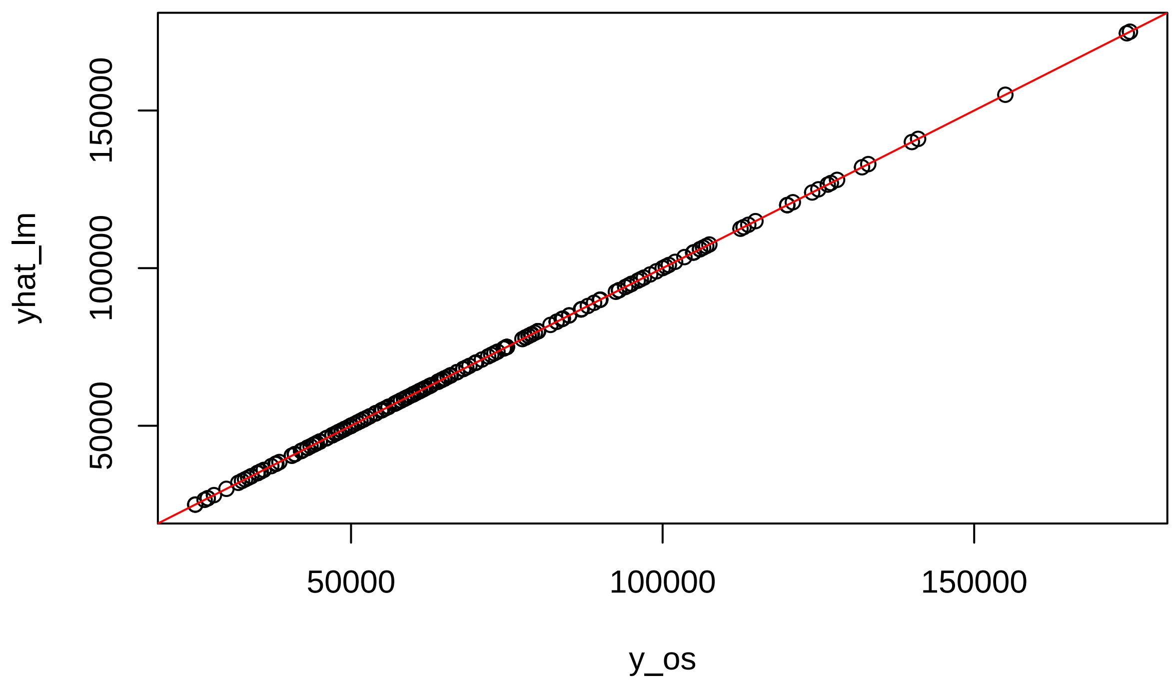
```
## [1] 0.0004642483
```

Such a low value of MSE of 0 strengthens our claim that the model performs good on test sample.

```r
# Estimating regression with insample data
lm_is = lm(y ~ is)
yhat_lm = cbind(1, os) %*% lm_is$coefficients
mse_lm = sum((y_os - yhat_lm)^2)/nrow(os)
mse_lm
```

```
## [1] 4.807222e-22
```

```r
# Let's take a look at how the LM prediction looks like
plot(y_os, yhat_lm )+abline(0, 1, col = "red") # Adding 45 degree line
```




```
## integer(0)
```

```r
lmMSE = (1/length(y_os))*sum(y_os - yhat_lm)^2
lmMSE
```

```
## [1] 7.756712e-25
```

So, we can see that both linear regression and ridge model have a negligible MSE and thus both perform well.

Question 15

15. Using the HealthInsurance dataset. Divide the data into two equally sized samples (one for training and one for testing). The dependent variable is health. Using the training sample; and a radial kernel and the following two values for cost C, estimate a support vector machine model and choose the optimal cost parameter using the function tune.

Ans.

```r
#install.packages("AER")
library(AER)
```

```
## Warning: package 'AER' was built under R version 3.5.3

## Loading required package: car

## Warning: package 'car' was built under R version 3.5.3

## Loading required package: carData

##
## Attaching package: 'carData'

## The following object is masked from 'package:Ecdat':
##
##     Mroz

##
## Attaching package: 'car'

## The following object is masked from 'package:psych':
##
##     logit

## Loading required package: lmtest

## Warning: package 'lmtest' was built under R version 3.5.3

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich

## Loading required package: survival
```

```r
#loading dataset
data(HealthInsurance)
#data exploration
summary(HealthInsurance)
```

```
##  health          age            limit        gender      insurance   married
##  no : 629   Min.   :18.00   no :7571   female:4169   no :1750   no :3369
##  yes:8173   1st Qu.:30.00   yes:1231   male  :4633   yes:7052   yes:5433
##             Median :39.00
##             Mean   :38.94
##             3rd Qu.:48.00
##             Max.   :62.00
##
##  selfemp        family              region      ethnicity
##  no :7731   Min.   : 1.000   northeast:1682   other: 365
##  yes:1071   1st Qu.: 2.000   midwest  :2023   afam :1083
##             Median : 3.000   south    :3075   cauc :7354
##             Mean   : 3.094   west     :2022
##             3rd Qu.: 4.000
##             Max.   :14.000
##
##      education
##  none      :1119
##  ged       : 374
##  highschool:4434
##  bachelor  :1549
##  master    : 524
##  phd       : 135
##  other     : 667
```

```r
#converting categorical to dummies where yes = 1 and no =0 in
#health,limit, insurance, married, selfemp.
#in gender column, male=1, female=0
#in ethinicity column, afam=1, cauc=2, other=3
# in region column, northeast=1, midwest=2,south=3,west=4
# education column to numeric factors
HealthInsurance$health=ifelse(HealthInsurance$health=="yes",1,0)
HealthInsurance$health=as.factor(HealthInsurance$health)

HealthInsurance$limit=ifelse(HealthInsurance$limit=="yes",1,0)

HealthInsurance$insurance=ifelse(HealthInsurance$insurance=="yes",1,0)

HealthInsurance$married=ifelse(HealthInsurance$married=="yes",1,0)

HealthInsurance$selfemp=ifelse(HealthInsurance$selfemp=="yes",1,0)

HealthInsurance$gender=ifelse(HealthInsurance$gender=="male",1,0)

HealthInsurance$ethnicity=ifelse(HealthInsurance$ethnicity=="afam",1,
                                 ifelse(HealthInsurance$ethnicity=="cauc",2,3))

HealthInsurance$region=ifelse(HealthInsurance$region=="northeast",1
```

```r
                              ,ifelse(HealthInsurance$region=="midwest", 2,
                                     ifelse(HealthInsurance$region=="south", 3, 4)))

HealthInsurance$education=as.numeric(as.factor(HealthInsurance$education))


set.seed(123)
split_HI=sample.split(HealthInsurance$health, SplitRatio = 0.5)
train_set=subset(HealthInsurance, split_HI==TRUE)
test_set=subset(HealthInsurance, split_HI==FALSE)
test_set=test_set[1:4400,] #making train and test similar in length

#Loading svm library
#install.packages("e1071")
library(e1071)
```

```
## Warning: package 'e1071' was built under R version 3.5.3
```

```r
# Setting cost values
costvalues = 10^seq(-5,1)

x_train=as.matrix(train_set[,2:11])
train=data.frame(train_set$health, x_train)

#estimate support vector machine model with radial kernel
classifier=svm(train_set$health~x_train, data=train,
               ranges=list(cost=costvalues), kernel = "radial")

#tuned svm
tuned_classifier=tune(svm, train_set$health~x_train,
                      data=train, ranges=list(cost=costvalues), kernel="radial")

summary(tuned_classifier)
```

```
##
## Parameter tuning of 'svm':
##
## - sampling method: 10-fold cross validation
##
## - best parameters:
##  cost
##    10
##
## - best performance: 0.061
##
## - Detailed performance results:
##    cost      error   dispersion
## 1 1e-05 0.07136364 0.0000000000
## 2 1e-04 0.07136364 0.0000000000
## 3 1e-03 0.07136364 0.0000000000
## 4 1e-02 0.07136364 0.0000000000
## 5 1e-01 0.07136364 0.0000000000
## 6 1e+00 0.07122727 0.0001916532
```

```
## 7 1e+01 0.06100000 0.0004815227
```

```
optimalCost = tuned_classifier$best.model$cost
optimalCost
```

```
## [1] 10
```

At cost =10, the error is minimized. Then we should proceed witha radial kernel and a cost parameter of 10.

Question 16

16. How does the svm model performs in the testing sample? How does the model compares to a logit in terms of accuracy?

Ans.

```
#tuning testing dataset
y_test=as.matrix(test_set[,2:11])
test=data.frame(test_set$health, y_test)

# Predicting y_svm using test data
y_svm = predict(classifier, newdata = y_test)

table(predicted=y_svm,truth=test_set$health)
```

```
##          truth
## predicted    0    1
##         0    0    0
##         1  315 4085
```

```
# Computing accuracy for svm
sum(y_svm == test_set$health)/length(test_set$health)
```

```
## [1] 0.9284091
```

```
# Estimating Logit
logit.test = glm(test_set$health~y_test,data=test,
                 family = "binomial")

#Predicting y_logit using test data
y_logit=predict(logit.test, newdata=as.data.frame(y_test),
                type="response")

y_logit=round(y_logit)

table(predicted=y_logit,truth=test_set$health)
```

```
##          truth
## predicted    0    1
##         1  315 4085
```

```r
# Computing accuracy for logit
sum(y_logit == test_set$health)/length(test_set$health)
```

```
## [1] 0.9284091
```

Both models, svm and logit give out predictions with similar accuracy of 92.84%