Multiple Linear Regression

Subham Agrawal

23 April 2017

Question-1

```
library(UsingR)
```

```
## Loading required package: MASS
```

```
## Loading required package: HistData
```

```
## Loading required package: Hmisc
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
## Loading required package: ggplot2
```

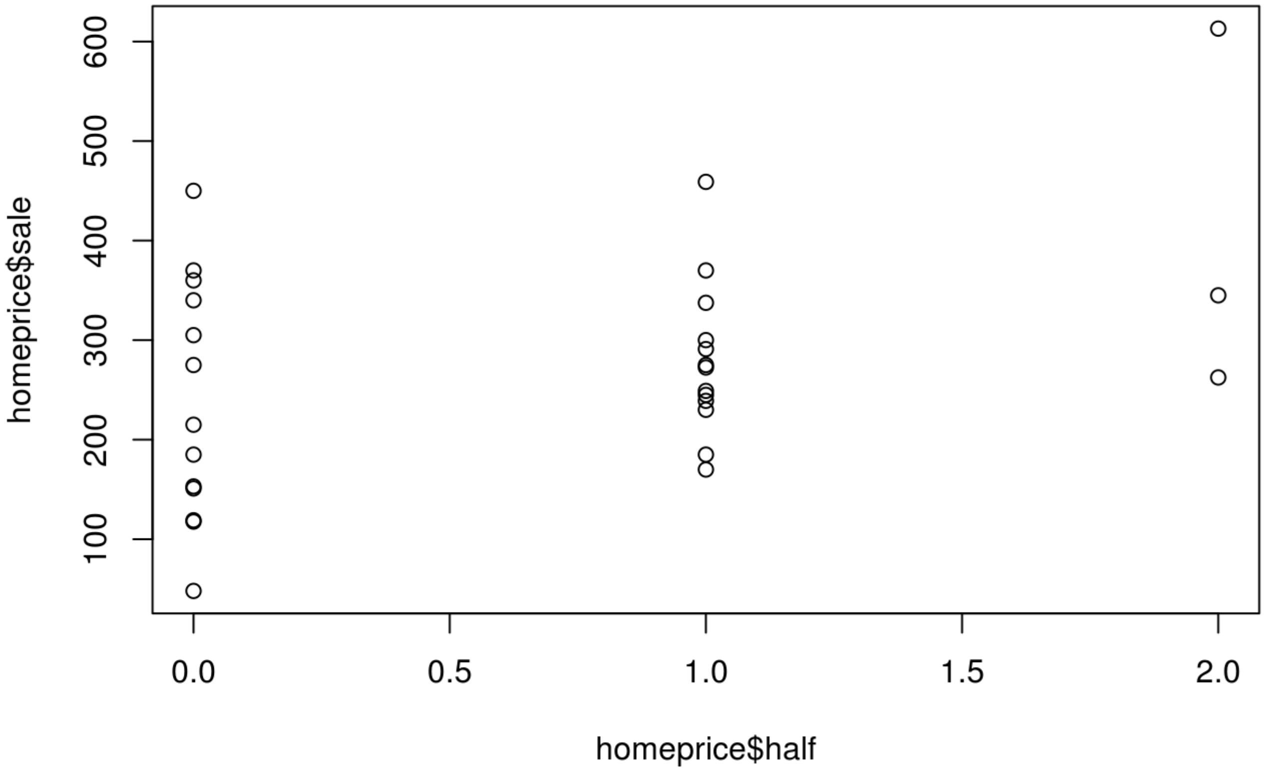
```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, round.POSIXt, trunc.POSIXt, units
```

```
##
## Attaching package: 'UsingR'
```

```
## The following object is masked from 'package:survival':
##
##     cancer
```

```
data(homeprice)
plot(homeprice$half, homeprice$sale)
```




```
summary(lm(sale~ half, data=homeprice))
```

```
##
## Call:
## lm(formula = sale ~ half, data = homeprice)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -180.27  -75.27  -22.34   72.66  246.58
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   228.27      28.78   7.932 1.59e-08 ***
## half           69.08      31.00   2.229   0.0344 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 109.8 on 27 degrees of freedom
## Multiple R-squared:  0.1554, Adjusted R-squared:  0.1241
## F-statistic: 4.966 on 1 and 27 DF,  p-value: 0.03436
```

From the above plots and regession model, we observe that sale price increases with number of half bathrooms.

```
summary(lm(sale~ full+half+bedrooms+rooms+neighborhood+list, data=homeprice))
```

```
##
## Call:
## lm(formula = sale ~ full + half + bedrooms + rooms + neighborhood +
##     list, data = homeprice)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.807  -6.626  -0.270   5.580  32.933
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.13359   17.15496   0.299    0.768
## full         -4.97759    5.48033  -0.908    0.374
## half         -1.00644    5.70418  -0.176    0.862
## bedrooms      2.49224    6.43616   0.387    0.702
## rooms        -0.43411    3.70424  -0.117    0.908
## neighborhood  2.03434    6.88609   0.295    0.770
## list          0.97131    0.07616  12.754 1.22e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.87 on 22 degrees of freedom
## Multiple R-squared:  0.989,  Adjusted R-squared:  0.986
## F-statistic: 330.5 on 6 and 22 DF,  p-value: < 2.2e-16
```

With every half bathroom in home, actual sale price goes up since coefficient is positive.

Question-2

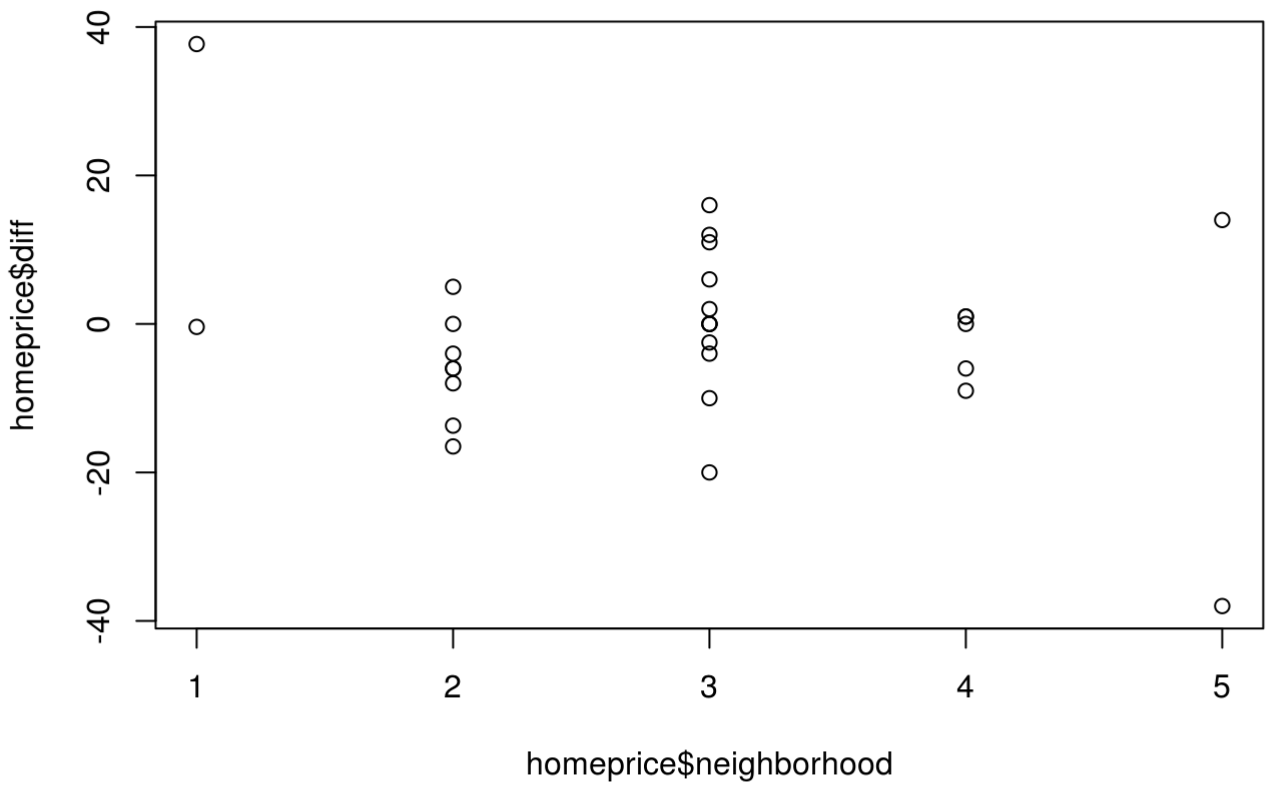
```
sale.lm<-lm(sale~ full+half+bedrooms+rooms+neighborhood-1, data=homeprice)
summary(sale.lm)
```

```
##
## Call:
## lm(formula = sale ~ full + half + bedrooms + rooms + neighborhood -
##     1, data = homeprice)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.510  -35.456    5.718   21.103   91.682
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## full           30.561     16.994   1.798  0.08471 .
## half           51.192     15.518   3.299  0.00302 **
## bedrooms       19.770     21.846   0.905  0.37447
## rooms          -9.911     11.474  -0.864  0.39625
## neighborhood   69.457     12.086   5.747 6.37e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 48.23 on 24 degrees of freedom
## Multiple R-squared:  0.9781, Adjusted R-squared:  0.9736
## F-statistic: 214.9 on 5 and 24 DF,  p-value: < 2.2e-16
```

There is not much change in coefficients if b0 is forced to be zero. Coefficient of full, half, bedrooms and neighborhood increase or decrease by 10 but are still positive i.e. with increase in this variables sale price increases. Whereas coefficient for rooms becomes negative. The adjusted R-squared value increases from 0.8879 to 0.9736. Thus it makes complete sense for this model to have no intercept term.

Question-3

```
homeprice$diff= homeprice$sale-homeprice$list
plot(homeprice$neighborhood, homeprice$diff)
```




```
tapply(homeprice$diff, homeprice$neighborhood, mean)
```

```
##       1       2       3       4       5
##  18.650  -6.150   0.875  -2.600 -12.000
```

As there is no pattern in means of difference between sale price and list price for different neighborhoods, we conclude there is no effect.

```
summary(lm(diff~neighborhood, data=homeprice))
```

```
##
## Call:
## lm(formula = diff ~ neighborhood, data = homeprice)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -30.05  -7.50  -0.85   5.80  33.05
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)     7.800      7.435   1.049    0.303
## neighborhood   -3.150      2.428  -1.298    0.205
##
## Residual standard error: 13 on 27 degrees of freedom
## Multiple R-squared:  0.0587, Adjusted R-squared:  0.02383
## F-statistic: 1.684 on 1 and 27 DF,  p-value: 0.2054
```

Same can be seen from the simple regression model above. As the p-value is greater than 0.05, we don't reject the null hypothesis. Hence there is no effect of neighborhood on the difference between sale price and list price.

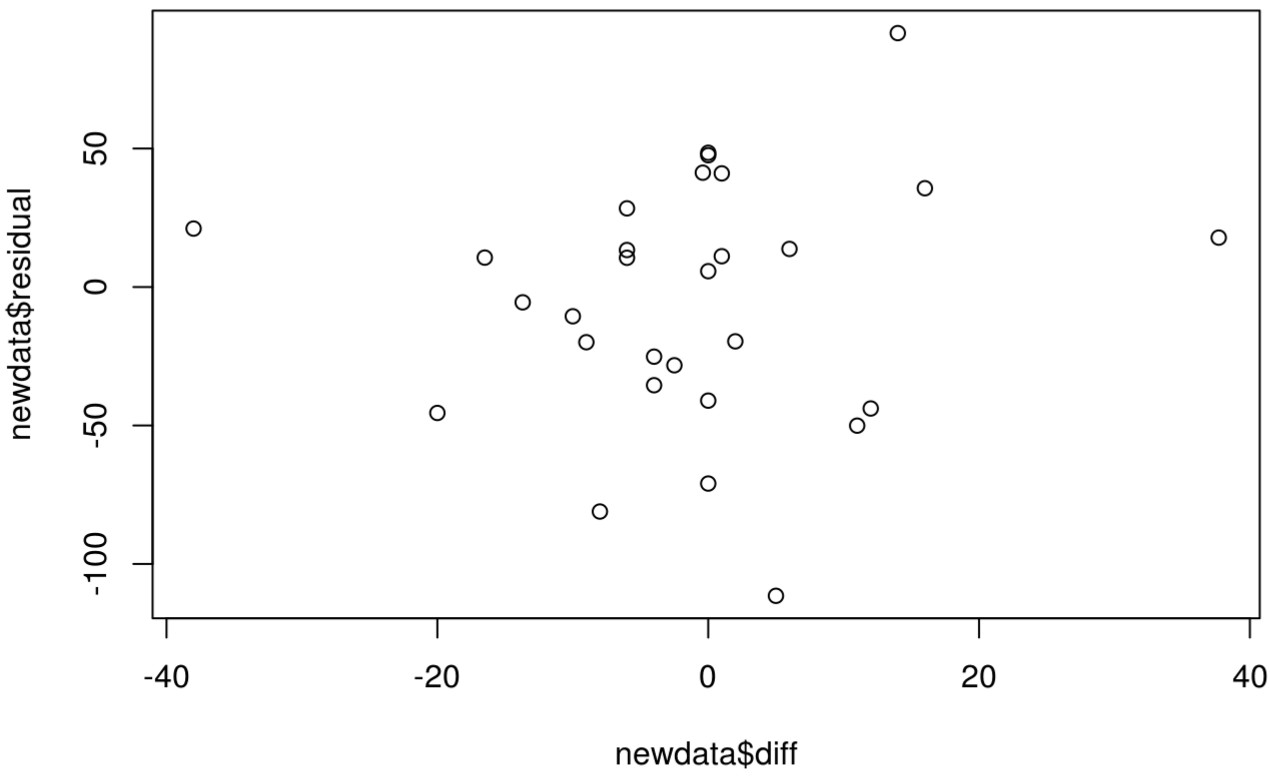
Question-4

(From question-3) No Nicer neighborhoods doesn't mean it is more likely to have a house go over the asking price.

Question-5

Lets see if there is a significant relationship between residual and difference between sale and list price if both are positive.

```
newdata=data.frame(homeprice, fitted.value=fitted(sale.lm), residual= resid(sale.lm))
plot(newdata$diff, newdata$residual)
```




```
newdata2= subset(newdata, residual>0 & diff>0)
summary(lm(residual~diff, data=newdata2))
```

```
##
## Call:
## lm(formula = residual ~ diff, data = newdata2)
##
## Residuals:
##        1        5        9       12       14       26
## -17.2937 -24.0845 -21.4412  56.5028   0.4899   5.8268
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 35.206790  19.499846   1.805    0.145
## diff        -0.001938   1.092556  -0.002    0.999
##
## Residual standard error: 33.78 on 4 degrees of freedom
## Multiple R-squared:  7.869e-07,  Adjusted R-squared:  -0.25
## F-statistic: 3.148e-06 on 1 and 4 DF,  p-value: 0.9987
```

As the p-value is greater than 0.05, we don’t reject the null hypothesis that β = 0. Hence there is no relationship between houses which sell for more than predicted (a positive residual) and houses which sell for more than asking.

Question-6

```
summary(lm(sale~list-1, data=homeprice))
```

```
##
## Call:
## lm(formula = sale ~ list - 1, data = homeprice)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.629  -4.576   1.066   4.589  38.417
##
## Coefficients:
##      Estimate Std. Error t value Pr(>|t|)
## list 0.991043   0.008033   123.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.95 on 28 degrees of freedom
## Multiple R-squared:  0.9982, Adjusted R-squared:  0.9981
## F-statistic: 1.522e+04 on 1 and 28 DF,  p-value: < 2.2e-16
```

The above simple linear regression model has adjusted R-squared value of 0.9981 when intercept is forced to be zero. The coefficient of list in this model is 0.991, can be approximated as 1. Thus, real estate agents are pricing the home correctly.

sale price= 0.991*list

Question-7

I’m not a facebook user. So will randomly generate data for this question.

```
data1=data.frame(sample(300:1000, 11))
colnames(data1)<-c("friends")

library(MASS)
xbar=mean(data1$friends)
stan_dev=sd(data1$friends) # Calculating sample standard deviation
n=length(data1$friends)

# Standard Error estimate
standard_Err=stan_dev/sqrt(n)
t_alphaby2=qt(0.975,df=n-1) # Quantile value
t_alphaby2
```

```
## [1] 2.228139
```

```
# Margin Of error
Err_Margin=t_alphaby2*standard_Err
Err_Margin
```

```
## [1] 150.5525
```

```
xbar+c(-Err_Margin,Err_Margin)
```

```
## [1] 544.8111 845.9161
```

OR

```
t.test(data1$friends)
```

```
##
##  One Sample t-test
##
## data:  data1$friends
## t = 10.291, df = 10, p-value = 1.221e-06
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  544.8111 845.9161
## sample estimates:
## mean of x
##  695.3636
```

confidence interval for 11 members is (601.686, 917.768)

```
data2=data.frame(sample(300:1000, 56))
colnames(data2)<-c("friends")
t.test(data2$friends)
```

```
##
##  One Sample t-test
##
## data:  data2$friends
## t = 22.012, df = 55, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  567.8861 681.6496
## sample estimates:
## mean of x
##  624.7679
```

confidence interval for 56 members is (566.63, 680.51)

As I have used randomly genearted data, cannot comment on average number of friends a profile can have.

Question-8

Null hypothesis: mu=8 Alternate hypothesis: mu!=8

Suppose we are testing at the 5 percent level of significance.

```
xbar=9.5  # sample mean
mu0=8     # true mean
sigma=2   # population standard deviation
n= 5      # sample size
z=(xbar-mu0)/(sigma/sqrt(n))
z         # test statistic
```

```
## [1] 1.677051
```

```
alpha=0.05
z.half_alpha=qnorm(1-alpha/2)
c(-z.half_alpha,z.half_alpha)
```

```
## [1] -1.959964  1.959964
```

The test statistic 1.68 lies between -1.96 to 1.96. Hence, at 0.05 significance level, we do not reject null hypothesis that mu=8

OR

```
pval=2*pnorm(z)
pval
```

```
## [1] 1.906467
```

Since it turns out to be greater than the 0.05 significance level, we do not reject the null hypothesis that μ=8.

For 10 percent significance level,

```
alpha=0.1
z.half_alpha=qnorm(1-alpha/2)
c(-z.half_alpha,z.half_alpha)
```

```
## [1] -1.644854  1.644854
```

The test statistic 1.68 is greater than critical value(upper bound) 1.645. Hence, at 0.1 significance level, we reject null hypothesis that mu=8

Question-9

```
pulse=c(54, 63, 58, 72, 49, 92, 70, 73, 69, 104, 48, 66, 80, 64, 77)
xbar=mean(pulse)
stan_dev=sd(pulse) # Calculating sample standard deviation
n=15

# Standard Error estimate
standard_Err=stan_dev/sqrt(n)
t_alphaby2=qt(0.975,df=n-1) # Quantile value
t_alphaby2
```

```
## [1] 2.144787
```

```
# Margin Of error
Err_Margin=t_alphaby2*standard_Err
Err_Margin
```

```
## [1] 8.39973
```

```
xbar+c(-Err_Margin,Err_Margin)
```

```
## [1] 60.86694 77.66640
```

OR

```
t.test(pulse)
```

```
##
##  One Sample t-test
##
## data:  pulse
## t = 17.687, df = 14, p-value = 5.652e-11
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  60.86694 77.66640
## sample estimates:
## mean of x
##  69.26667
```

95 percent confidence interval = (60.87, 77.67)

Lower confidence interval will have an upper bound. Calculation for upper bound is shown below.

```
t_alphaby2=qt(0.95,df=n-1) # Quantile value
t_alphaby2
```

```
## [1] 1.76131
```

```
# Margin Of error
Err_Margin=t_alphaby2*standard_Err
Err_Margin
```

```
## [1] 6.897903
```

```
xbar+c(-Inf,Err_Margin)
```

```
## [1]     -Inf 76.16457
```

95 percent lower confidence interval = (-inf, 76.1646)

Question-10

From central limit theorem, total yearly claim will have approximately a normal distribution with mean and standard deviation calculated below:

```
n=25000
mean=320
sd=540
new_mean= mean*n
new_sd= sd*sqrt(n)

1-pnorm(8300000, new_mean, new_sd)
```

```
## [1] 0.0002210042
```

probability=0.00022

Thus, there are only 2.2 chances out of 10,000 that the total yearly claim will exceed 8.3 million.