Dev Submission for Assignment 2

Installing necessary packages:

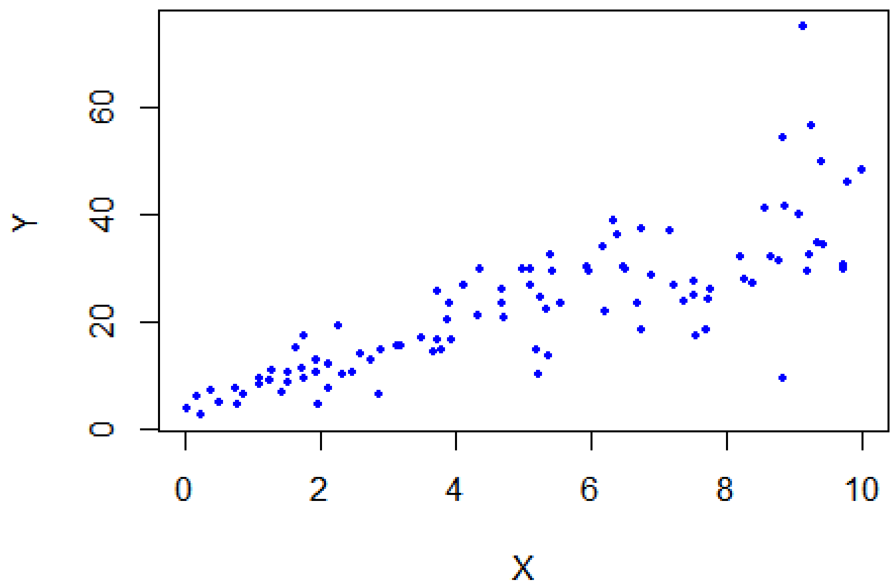
```
#install.packages('mlbench')
library(mlbench)
#install.packages("lmtest")
library(lmtest)

## Loading required package: zoo

##
## Attaching package: 'zoo'

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

#install.packages('mlbench')
library(mlbench)
```

****Question 1

```
set.seed(2017)
X=runif(100)*10
Y=X*4+3.45
Y=rnorm(100)*0.29*Y+Y
```

****a)

```
plot(X,Y,pch = 16, cex = 0.5, col = "blue")
```




```
#By looking at the plot we see that there is a positive linear relationship
between x & y. Therefore we can fit a linear model to explain Y based on X.
```

****b)

```
lm <- lm(Y ~ X)
summary(lm)

##
## Call:
## lm(formula = Y ~ X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -26.755  -3.846  -0.387   4.318  37.503
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.4655     1.5537   2.874  0.00497 **
## X             3.6108     0.2666  13.542  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.756 on 98 degrees of freedom
## Multiple R-squared:  0.6517, Adjusted R-squared:  0.6482
## F-statistic: 183.4 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
lm$coefficients

## (Intercept)           X
##    4.465490    3.610759

# Equation that explains Y based on X is  4.4655 = 3.6108 * X
# For every one unit change in X, Y increases by 3.6108 units.
# By R-squared value we know that 65% of the variance in Y was explained by
the variance in X.
```

****c)

```
cor(X,Y)

## [1] 0.807291

(cor(X,Y))^2

## [1] 0.6517187

#R-squared is simply the correlation squared for a simple linear regression.
```

****d) Reference taken: https://blog.minitab.com/blog/statistics-and-quality-data-analysis/violations-of-the-assumptions-for-linear-regression-the-trial-of-lionel-loosefit-day-1

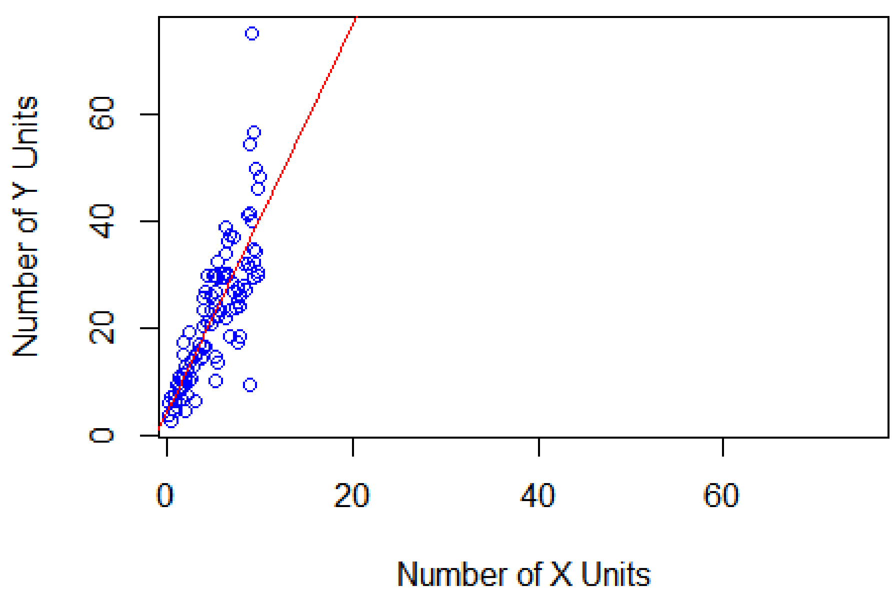
```
summary(X)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.02021 2.31519 5.14681 5.04920 7.53777 9.99147

summary(Y)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.735  11.999  22.820  22.697  29.834  74.995

summary(lm)

##
## Call:
## lm(formula = Y ~ X)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -26.755  -3.846  -0.387   4.318  37.503
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.4655     1.5537   2.874  0.00497 **
## X             3.6108     0.2666  13.542  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 7.756 on 98 degrees of freedom
## Multiple R-squared:  0.6517, Adjusted R-squared:  0.6482
## F-statistic: 183.4 on 1 and 98 DF,  p-value: < 2.2e-16
```

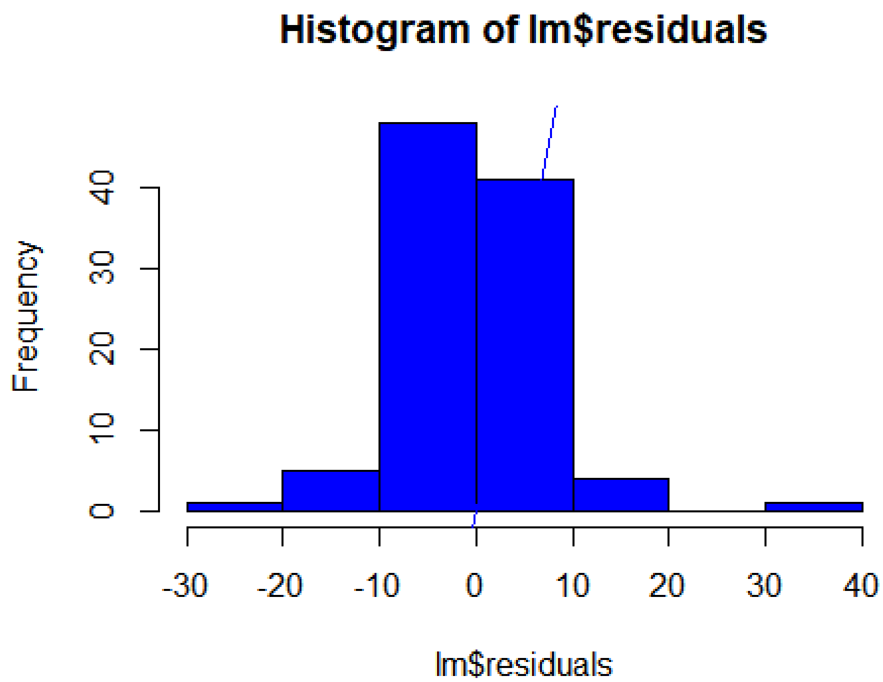
```
dwtest(lm)
```

```
##
##  Durbin-Watson test
##
## data:  lm
## DW = 2.0925, p-value = 0.68
## alternative hypothesis: true autocorrelation is greater than 0
```

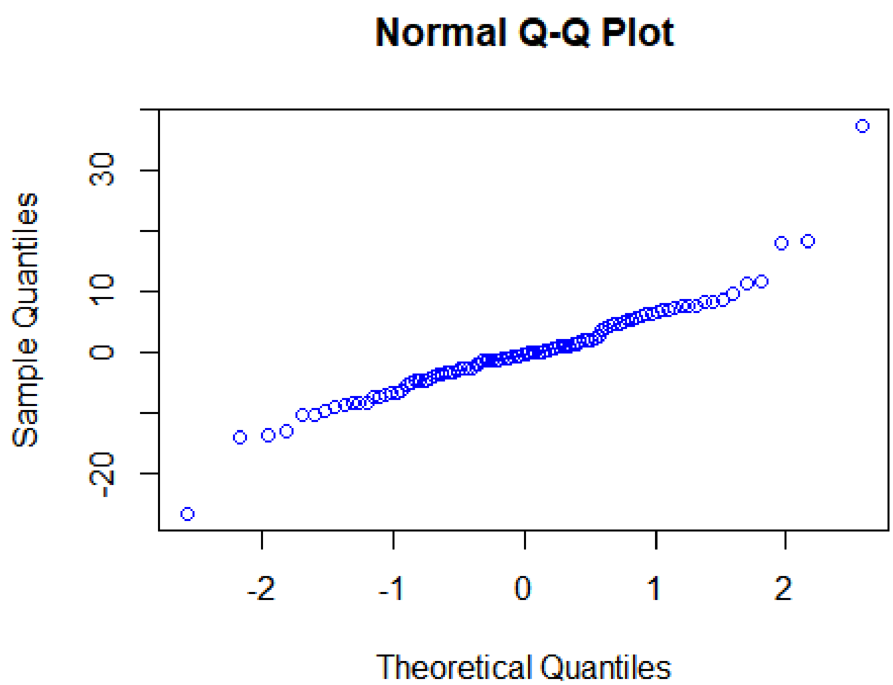
```
plot(X,Y,xlim=c(2,75),xlab="Number of X Units",ylab="Number of Y
Units",col="blue")
abline(lsfit(X,Y),col="red")
```



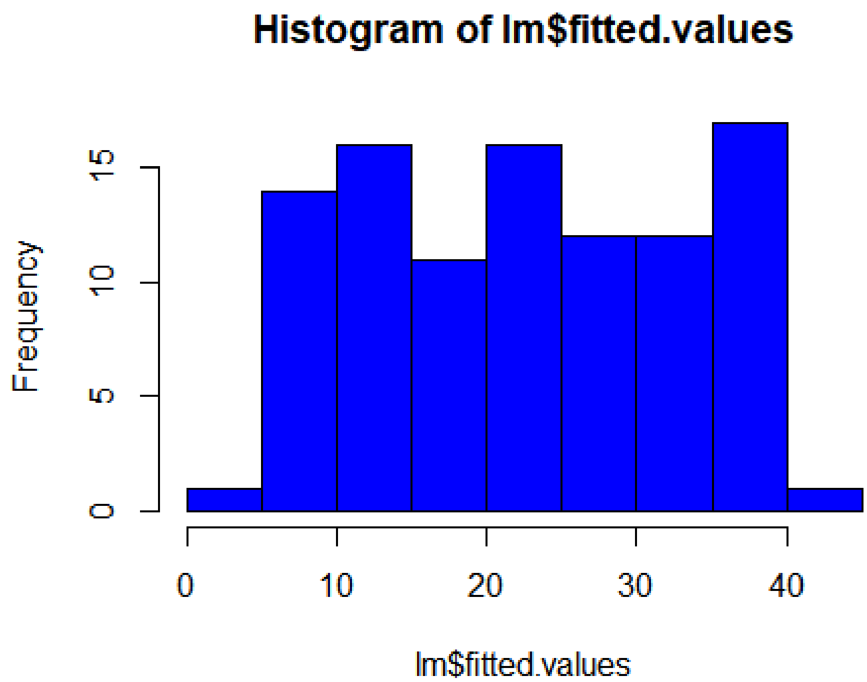

```
hist(lm$residuals, col="blue")
qqline(lm$residuals, col="blue")
```



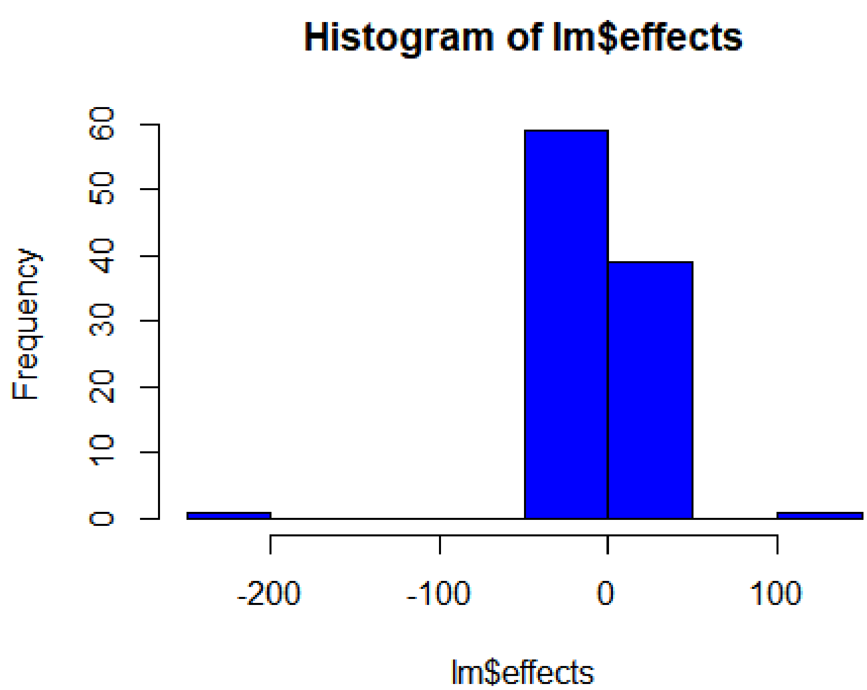

```r
qqnorm(lm$residuals, col="blue")
```




```r
hist(lm$fitted.values, col="blue")
```




```r
hist(lm$effects, col="blue")
```



```r
# 1. It is depicted in all the plots that there is a strong linear
relationship between X and Y.
# 2. The residual plots  show that there is a good fit of the dataset in the
simple linear model.
# 3. As illustrated in the residual-effects plot, the mean residuals centered
approximately on 0.
# 4. Most points fall on the theoretical 45-degree line.
# 5. Mean and median illustrate that the distribution is close to normal.

# From all the above points and looking at the graphs we can say that it is
appropriate to use linear regression for this case.
```

****Question 2

****a)

```r
summary(mtcars$hp)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    52.0    96.5   123.0   146.7   180.0   335.0

summary(mtcars$wt)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.513   2.581   3.325   3.217   3.610   5.424

summary(mtcars$mpg)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.40   15.43   19.20   20.09   22.80   33.90

# Model to estimate hp by wt
lmwt <- lm(hp ~ wt, data = mtcars)
lmwt$coefficients

## (Intercept)          wt
##   -1.820922   46.160050

summary(lmwt)

##
## Call:
## lm(formula = hp ~ wt, data = mtcars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -83.430 -33.596 -13.587   7.913 172.030
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.821     32.325  -0.056    0.955
## wt            46.160      9.625   4.796 4.15e-05 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 52.44 on 30 degrees of freedom
## Multiple R-squared:  0.4339, Adjusted R-squared:  0.4151
## F-statistic:    23 on 1 and 30 DF,  p-value: 4.146e-05
```

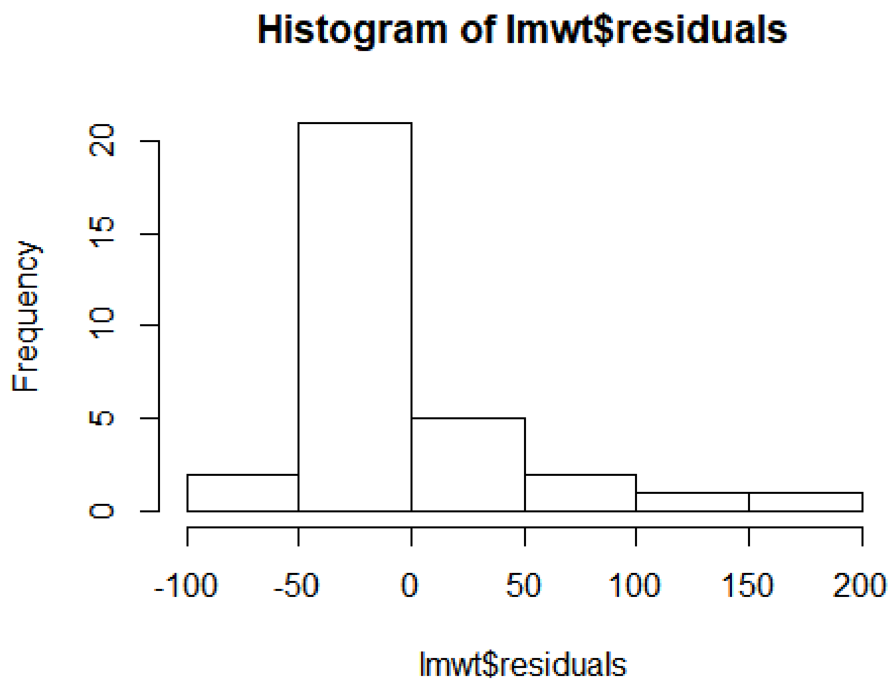
```
# Model to estimate hp by mpg
lmmpg <- lm(hp ~ mpg, data = mtcars)
lmmpg$coefficients
```

```
## (Intercept)         mpg
##  324.082314   -8.829731
```

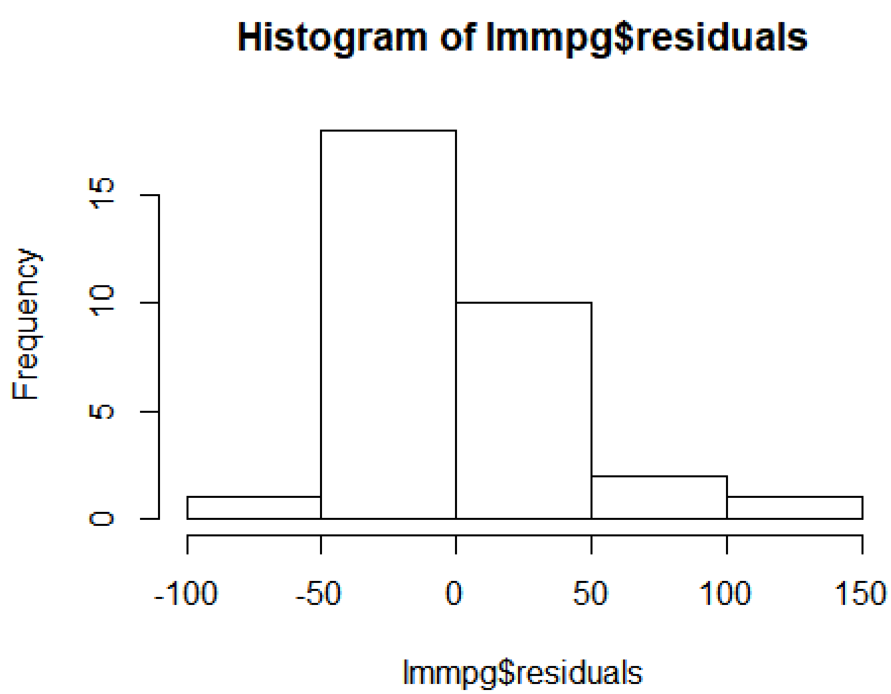
```
summary(lmmpg)
```

```
##
## Call:
## lm(formula = hp ~ mpg, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.26 -28.93 -13.45  25.65 143.36
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   324.08      27.43  11.813 8.25e-13 ***
## mpg            -8.83       1.31  -6.742 1.79e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 43.95 on 30 degrees of freedom
## Multiple R-squared:  0.6024, Adjusted R-squared:  0.5892
## F-statistic: 45.46 on 1 and 30 DF,  p-value: 1.788e-07
```

```
hist(lmwt$residuals)
```




```
hist(lmmpg$residuals)
```



```r
# R-squared for wt = 43.39%
# R-squared for mpg = 60.24%

# We can clearly see that mpg is more significant and explains 60% of the
data.

# Model to estimate hp by wt and mpg

lmboth <- lm(hp ~ mpg + wt, data = mtcars)
summary(lmboth)

##
## Call:
## lm(formula = hp ~ mpg + wt, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.42 -30.75 -12.07  24.82 141.84
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  349.287    103.509   3.374  0.00212 **
## mpg           -9.417      2.676  -3.519  0.00145 **
## wt            -4.168     16.485  -0.253  0.80217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 44.65 on 29 degrees of freedom
## Multiple R-squared:  0.6033, Adjusted R-squared:  0.576
## F-statistic: 22.05 on 2 and 29 DF,  p-value: 1.505e-06

anova(lmboth)

## Analysis of Variance Table
##
## Response: hp
##           Df Sum Sq Mean Sq F value    Pr(>F)
## mpg        1  87791   87791 44.0414 2.825e-07 ***
## wt         1    127     127  0.0639    0.8022
## Residuals 29  57808    1993
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# We can clearly see that mpg is the most significant variable and wt is not
statistically significant at all in estimating for hp.

# Therefore we can say that Chris is right in thinking that mpg is a better
estimator of the hp.
```

****b)

```r
# Model to estimate hp by cyl and mpg
lmnew <- lm(hp ~ mpg + cyl, data = mtcars)
summary(lmnew)
```

```
##
## Call:
## lm(formula = hp ~ mpg + cyl, data = mtcars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -53.72 -22.18 -10.13  14.47 130.73
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   54.067     86.093   0.628  0.53492
## mpg           -2.775      2.177  -1.275  0.21253
## cyl           23.979      7.346   3.264  0.00281 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 38.22 on 29 degrees of freedom
## Multiple R-squared:  0.7093, Adjusted R-squared:  0.6892
## F-statistic: 35.37 on 2 and 29 DF,  p-value: 1.663e-08
```

****1)

```r
# Prdicting HP with mpg = 22 & cyl = 4

predict(lmnew, data.frame(mpg = 22, cyl = 4))
```

```
##        1
## 88.93618
```

```r
# We could also use the quation : hp = 54.067 + -2.775* mpg + 23.979* cyl
```

****2)

```r
# Constructing a 85% confidence interval:

predict(lmnew, data.frame(mpg = 22, cyl = 4), interval = "prediction", level
= 0.85)
```

```
##        fit      lwr      upr
## 1 88.93618 28.53849 149.3339
```

****3)

****a)

```r
data(BostonHousing)
head(BostonHousing)
```

```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio      b
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90
## 6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7 394.12
##   lstat medv
## 1  4.98 24.0
## 2  9.14 21.6
## 3  4.03 34.7
## 4  2.94 33.4
## 5  5.33 36.2
## 6  5.21 28.7
```

```
# Model to estimate medv by crim, zn, ptratio & chas
lmboston <- lm(medv ~ crim + zn + ptratio + chas, data = BostonHousing)
summary(lmboston)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + ptratio + chas, data = BostonHousing)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.282  -4.505  -0.986   2.650  32.656
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 49.91868    3.23497  15.431  < 2e-16 ***
## crim        -0.26018    0.04015  -6.480 2.20e-10 ***
## zn           0.07073    0.01548   4.570 6.14e-06 ***
## ptratio     -1.49367    0.17144  -8.712  < 2e-16 ***
## chas1        4.58393    1.31108   3.496 0.000514 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.388 on 501 degrees of freedom
## Multiple R-squared:  0.3599, Adjusted R-squared:  0.3547
## F-statistic: 70.41 on 4 and 501 DF,  p-value: < 2.2e-16
```

```
lmboston$coefficients
```

```
## (Intercept)        crim          zn     ptratio       chas1
## 49.91868439 -0.26017612  0.07072809 -1.49367255  4.58392591
```

```
# R-squared is 35.99%, which means 64.01% of the data is not being explained
by the model. Hence, it is  not a very good model.

# All variables are statistically significant though.
```

****b)

****1)

```r
aggregate(medv ~ chas, data = BostonHousing, FUN= "mean" )

##   chas     medv
## 1    0 22.09384
## 2    1 28.44000

# Houses that do not bound river with chas = 0, avg median cost is $22,093.84

# Houses that bound river with chas = 1, avg median cost is $28,440.00

# Therefore the house which bounds Chas River is more expensive by:

28440.00 - 22093.84

## [1] 6346.16
```

****2)

```r
# Keeping all the aspects of house identical other than ptratio.

# Data frame with ptratio = 15
data1 <- data.frame(crim = 0.00632, zn = 2, ptratio = 15, chas = 1)

data1$chas = as.factor(data1$chas)

predict(lmboston, data1)

##        1
## 32.23733

# Data frame with ptratio = 18
data2 <- data.frame(crim = 0.00632, zn = 2,  ptratio = 18, chas = 1)

data2$chas = as.factor(data2$chas)


predict(lmboston, data2)

##        1
## 27.75632

diff <- predict(lmboston, data1) - predict(lmboston, data2)

# House with ptratio = 15 is more expensive by:
diff * 10000

##        1
## 44810.18
```

****c)

```r
# Model to estimate medv by using all the variables present:

lmbostonall <- lm(medv ~ crim + zn + indus + chas + nox + rm + age + dis +
rad + tax + ptratio + b + lstat, data = BostonHousing)
summary(lmbostonall)

##
## Call:
## lm(formula = medv ~ crim + zn + indus + chas + nox + rm + age +
##     dis + rad + tax + ptratio + b + lstat, data = BostonHousing)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.595  -2.730  -0.518   1.777  26.199
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
## crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
## zn           4.642e-02  1.373e-02   3.382 0.000778 ***
## indus        2.056e-02  6.150e-02   0.334 0.738288
## chas1        2.687e+00  8.616e-01   3.118 0.001925 **
## nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
## rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
## age          6.922e-04  1.321e-02   0.052 0.958229
## dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
## rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
## tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
## ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
## b            9.312e-03  2.686e-03   3.467 0.000573 ***
## lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.745 on 492 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7338
## F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16

anova(lmbostonall)

## Analysis of Variance Table
##
## Response: medv
##            Df  Sum Sq Mean Sq  F value    Pr(>F)
## crim        1  6440.8  6440.8 286.0300 < 2.2e-16 ***
## zn          1  3554.3  3554.3 157.8452 < 2.2e-16 ***
## indus       1  2551.2  2551.2 113.2984 < 2.2e-16 ***
## chas        1  1529.8  1529.8  67.9393 1.543e-15 ***
## nox         1    76.2    76.2   3.3861 0.0663505 .
```

```
## rm          1 10938.1 10938.1 485.7530 < 2.2e-16 ***
## age         1    90.3    90.3   4.0087 0.0458137 *
## dis         1  1779.5  1779.5  79.0262 < 2.2e-16 ***
## rad         1    34.1    34.1   1.5159 0.2188325
## tax         1   329.6   329.6  14.6352 0.0001472 ***
## ptratio     1  1309.3  1309.3  58.1454 1.266e-13 ***
## b           1   593.3   593.3  26.3496 4.109e-07 ***
## lstat       1  2410.8  2410.8 107.0634 < 2.2e-16 ***
## Residuals 492 11078.8    22.5
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# We see that from "Pr(>|t|)" column, other than indus & age all the other
variables are statistically important, those having "***" are the most
important.

# From "Pr(>F)" column by using anova we see that other than nox & rad all
the other variables are statistically important, those having "***" are the
most important.
```

****d)

```
anova(lmboston)

## Analysis of Variance Table
##
## Response: medv
##            Df  Sum Sq Mean Sq F value    Pr(>F)
## crim        1  6440.8  6440.8 118.007 < 2.2e-16 ***
## zn          1  3554.3  3554.3  65.122 5.253e-15 ***
## ptratio     1  4709.5  4709.5  86.287 < 2.2e-16 ***
## chas        1   667.2   667.2  12.224 0.0005137 ***
## Residuals 501 27344.5    54.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Order of importance of the four variables we used to create the "lmboston"
model is as follows by looking at the F values and other attributes if the
table below:

# 1. crim
# 2. ptratio
# 3. zn
# 4. chas
```