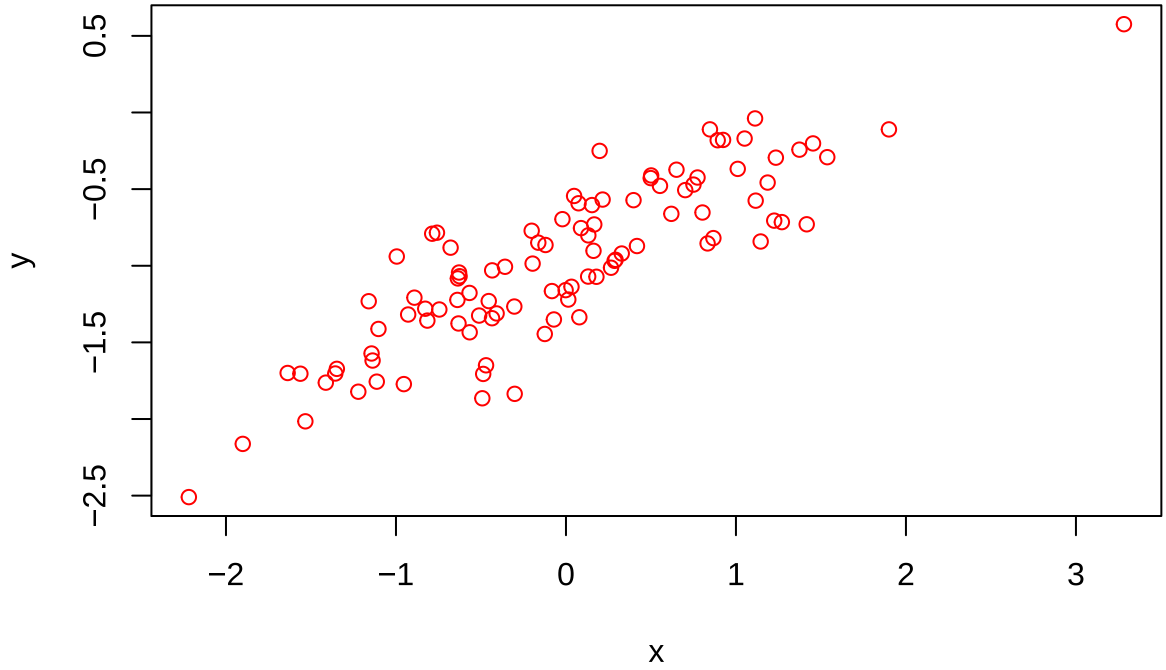
# HW1-Programming Problems

*Yihang Ding*

*9/20/2019*

## P3

```
x <- rnorm(100, 0, 1)
eps <- rnorm(100, 0, 0.25)
y <- -1 + 0.5*x + eps
plot(x,y,col="red")
```



(c) $\beta_0$ = -1, $\beta_1$ = 0.5 (d) There's a strong linear relationship between X and Y

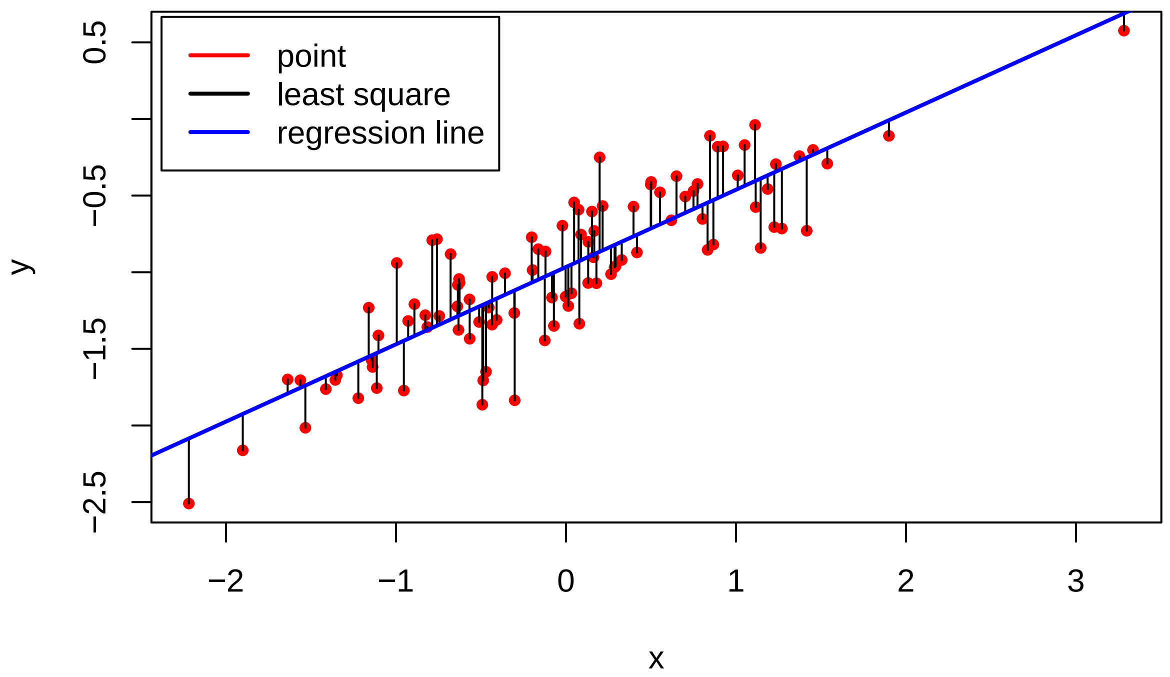
```
lm1 <- lm(y~x)
summary(lm1)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.71740 -0.17380  0.00607  0.20023  0.61592
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.96629    0.02753  -35.10   <2e-16 ***
## x            0.50423    0.02988   16.87   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2752 on 98 degrees of freedom
## Multiple R-squared:  0.7439, Adjusted R-squared:  0.7413
## F-statistic: 284.7 on 1 and 98 DF,  p-value: < 2.2e-16
```

(e) They are very close to $\beta_0$ and $\beta_1$, $\hat{\beta}_1$ is lower and $\hat{\beta}_0$ is higher than real value.

(f)

```
lm1_pred <- predict(lm1)
plot(x, y, col="red", pch=20)
segments(x, y, x, lm1_pred)
abline(lm1, col="blue", lwd=2)
legend("topleft", inset=.01, c("point", "least square", "regression line"), lwd=2, col=c("red", "black"
```



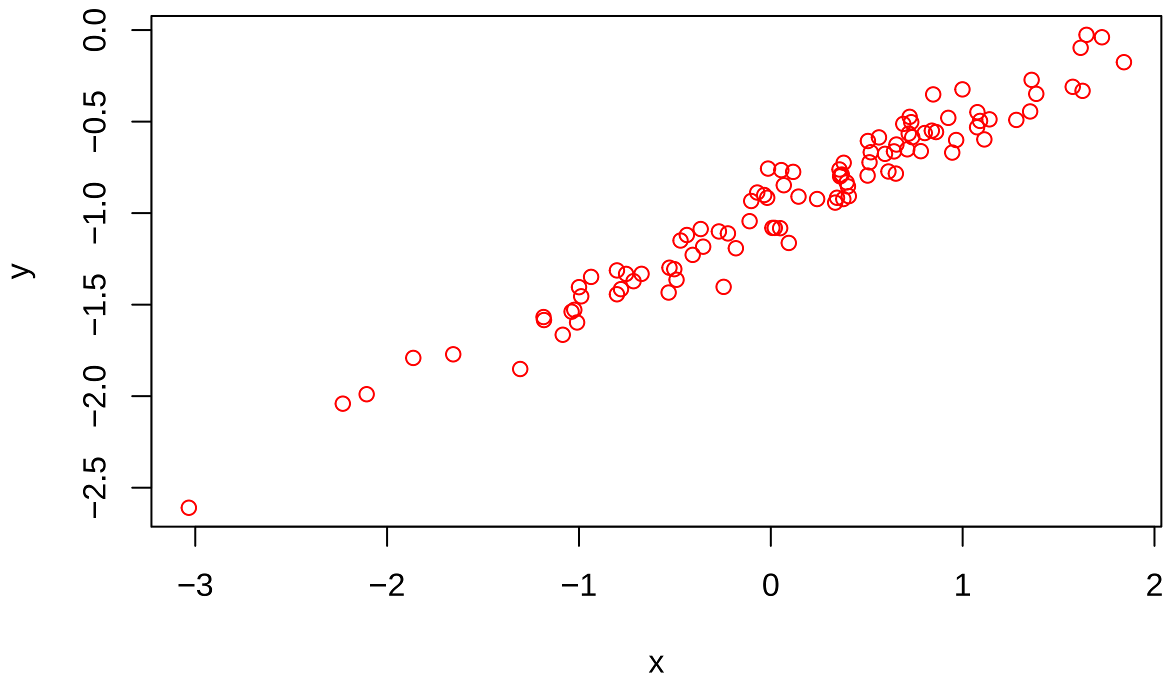
```
lm2 = lm(y~x+I(x^2))
summary(lm2)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73905 -0.18615  0.03637  0.17731  0.58857
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.93932    0.03260  -28.81   <2e-16 ***
## x            0.51239    0.03017   16.98   <2e-16 ***
## I(x^2)      -0.03150    0.02072   -1.52    0.132
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2733 on 97 degrees of freedom
## Multiple R-squared:  0.7499, Adjusted R-squared:  0.7447
## F-statistic: 145.4 on 2 and 97 DF,  p-value: < 2.2e-16
```

(g) There’s no evidence that the quadratic term improves the model fit, as the adjusted R-squared dropped from 0.767 to 0.7651

(h) After reducing the variance of data, the model fits more to the data, the residuals are smaller and the adjusted R-square increased from 0.767 to 0.9664.

```r
x <- rnorm(100, 0, 1)
eps <- rnorm(100, 0, 0.1)
y <- -1 + 0.5*x + eps
plot(x,y,col="red")
```

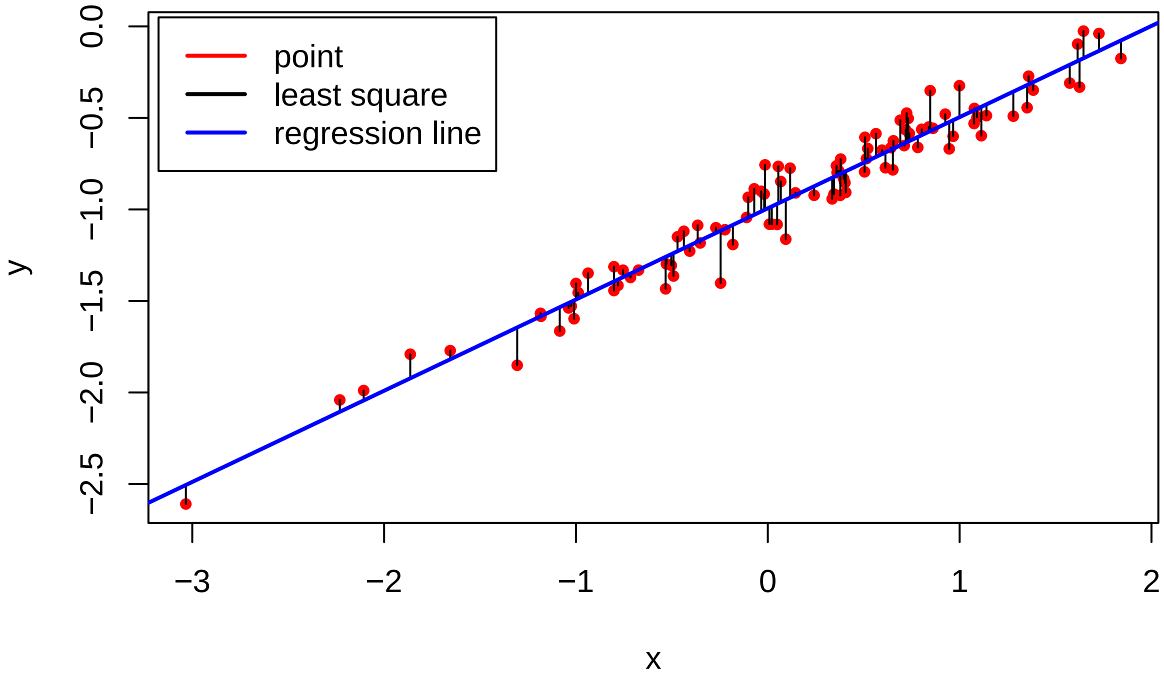


```r
lm3 <- lm(y~x)
summary(lm3)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.28636 -0.08495  0.00840  0.07913  0.24456
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.99395    0.01044  -95.23   <2e-16 ***
## x            0.49827    0.01117   44.62   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1035 on 98 degrees of freedom
## Multiple R-squared:  0.9531, Adjusted R-squared:  0.9526
## F-statistic:  1991 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
lm3_pred <- predict(lm3)
plot(x, y, col="red", pch=20)
segments(x, y, x, lm3_pred)
```

```r
abline(lm3, col="blue", lwd=2)
legend("topleft", inset=.01, c("point", "least square", "regression line"), lwd=2, col=c("red", "black"
```

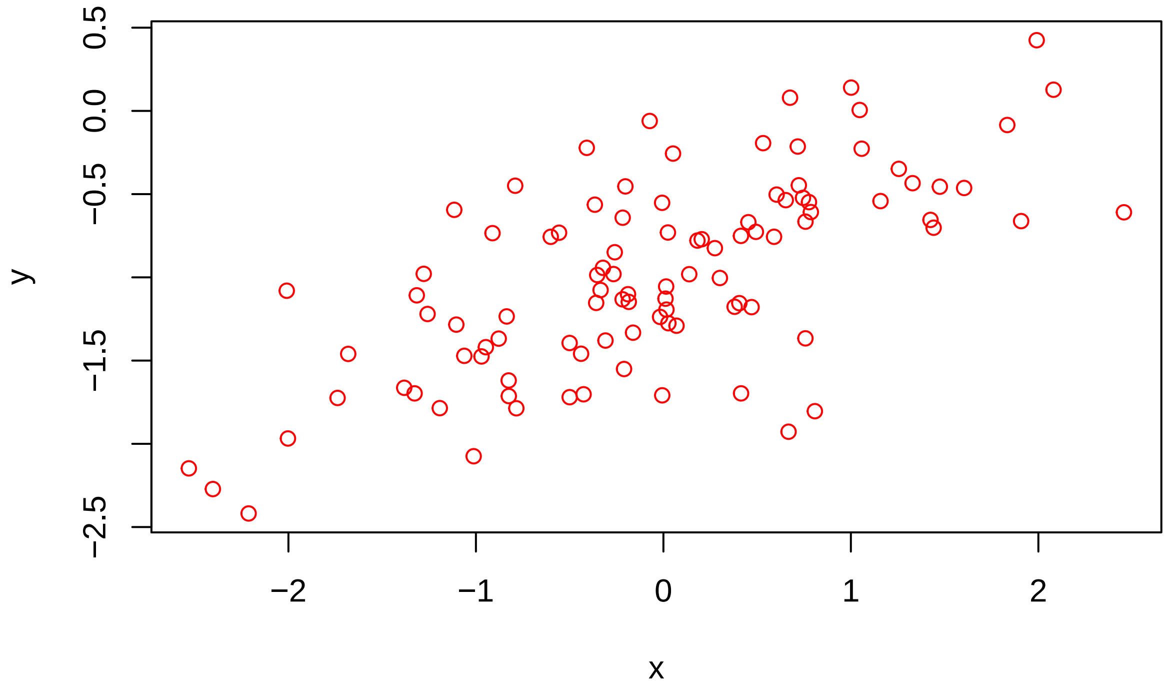


```r
lm4 = lm(y~x+I(x^2))
summary(lm4)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.291107 -0.072943  0.008329  0.076886  0.239923
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.989337   0.012971 -76.271   <2e-16 ***
## x            0.496468   0.011594  42.822   <2e-16 ***
## I(x^2)      -0.005024   0.008341  -0.602    0.548
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1038 on 97 degrees of freedom
## Multiple R-squared:  0.9533, Adjusted R-squared:  0.9523
## F-statistic: 989.2 on 2 and 97 DF,  p-value: < 2.2e-16
```

(i) After increasing the variance of data, the model fits less to the data, the residuals are larger and the adjusted R-square decreased from 0.767 to 0.473.

```
x <- rnorm(100, 0, 1)
eps <- rnorm(100, 0, 0.5)
y <- -1 + 0.5*x + eps
plot(x,y,col="red")
```

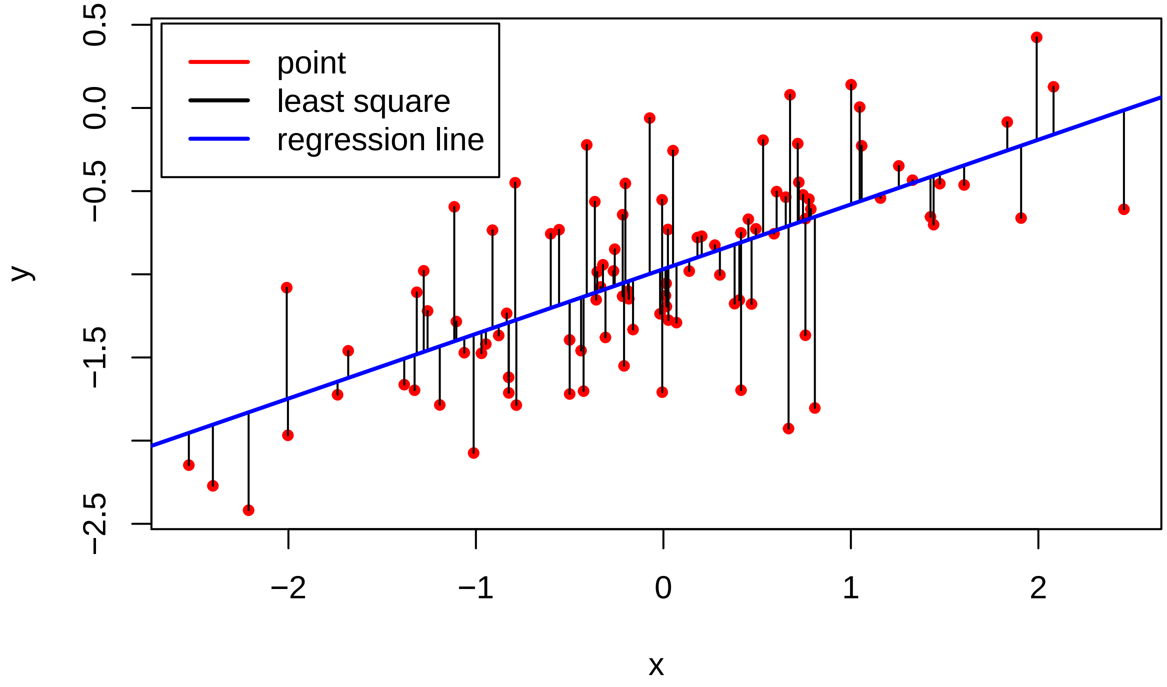


```
lm5 <- lm(y~x)
summary(lm5)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.21808 -0.29109 -0.02001  0.23298  0.93679
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.96901    0.04341 -22.322  < 2e-16 ***
## x            0.38902    0.04302   9.042 1.45e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.4336 on 98 degrees of freedom
## Multiple R-squared:  0.4548, Adjusted R-squared:  0.4493
## F-statistic: 81.77 on 1 and 98 DF,  p-value: 1.448e-14
```

```
lm5_pred <- predict(lm5)
plot(x, y, col="red", pch=20)
segments(x, y, x, lm5_pred)
abline(lm5, col="blue", lwd=2)
legend("topleft", inset=.01, c("point", "least square", "regression line"), lwd=2, col=c("red", "black"
```



```
lm6 = lm(y~x+I(x^2))
summary(lm6)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.22827 -0.27505 -0.02092  0.22876  0.91477
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.94710    0.05348 -17.708  < 2e-16 ***
## x             0.38593    0.04335   8.902 3.15e-14 ***
## I(x^2)       -0.02166    0.03073  -0.705    0.483
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4347 on 97 degrees of freedom
## Multiple R-squared:  0.4576, Adjusted R-squared:  0.4464
## F-statistic: 40.92 on 2 and 97 DF,  p-value: 1.299e-13
```

(j) As the variance of the data decreases, the confidence interval decreases as well.

```
confint(lm1)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0209235 -0.9116603
## x            0.4449257  0.5635342
```

```
confint(lm3)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0146611 -0.9732350
## x            0.4761058  0.5204252
```

```
confint(lm5)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0551505 -0.8828608
## x            0.3036443  0.4743937
```

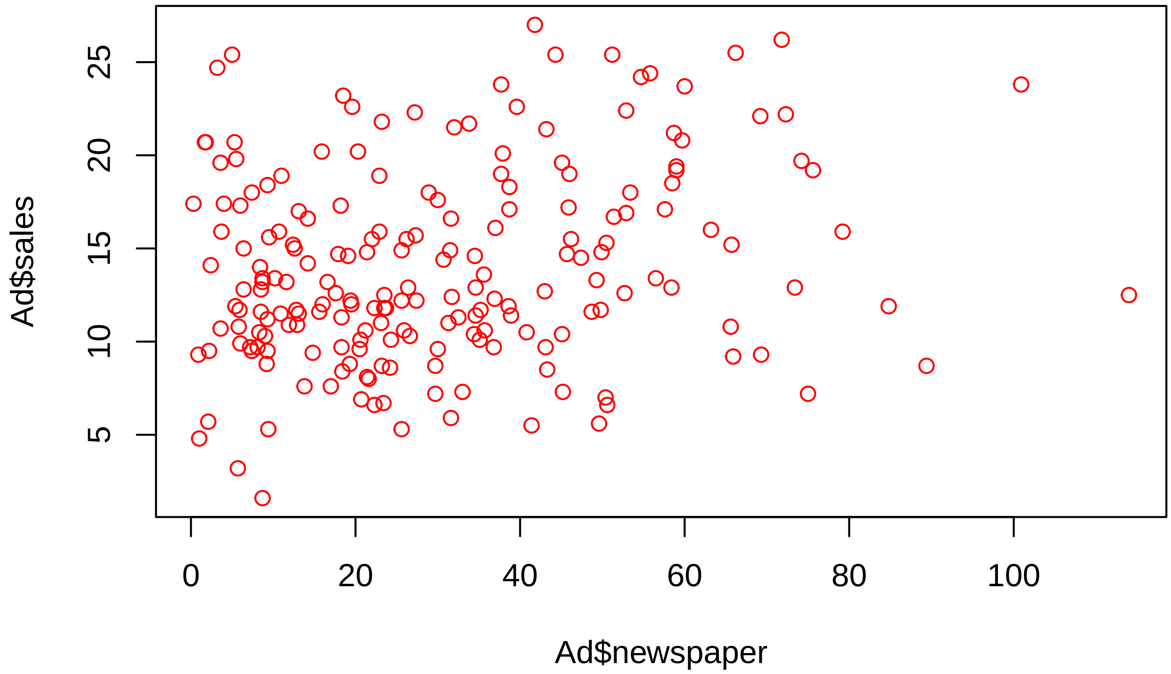
## P4

```
Ad = read.csv("Advertising.csv", header=T, na.strings="?")
dim(Ad)
```

```
## [1] 200   5
```

(1) newspaper

```
lm_ad_news = lm(Ad$sales~Ad$newspaper)
plot(Ad$newspaper,Ad$sales, col="red")
```



```
lm_ad_news = lm(Ad$sales~Ad$newspaper)
lm_ad_news_pred = predict(lm_ad_news, interval="confidence", level=0.92)
summary(lm_ad_news_pred)
```

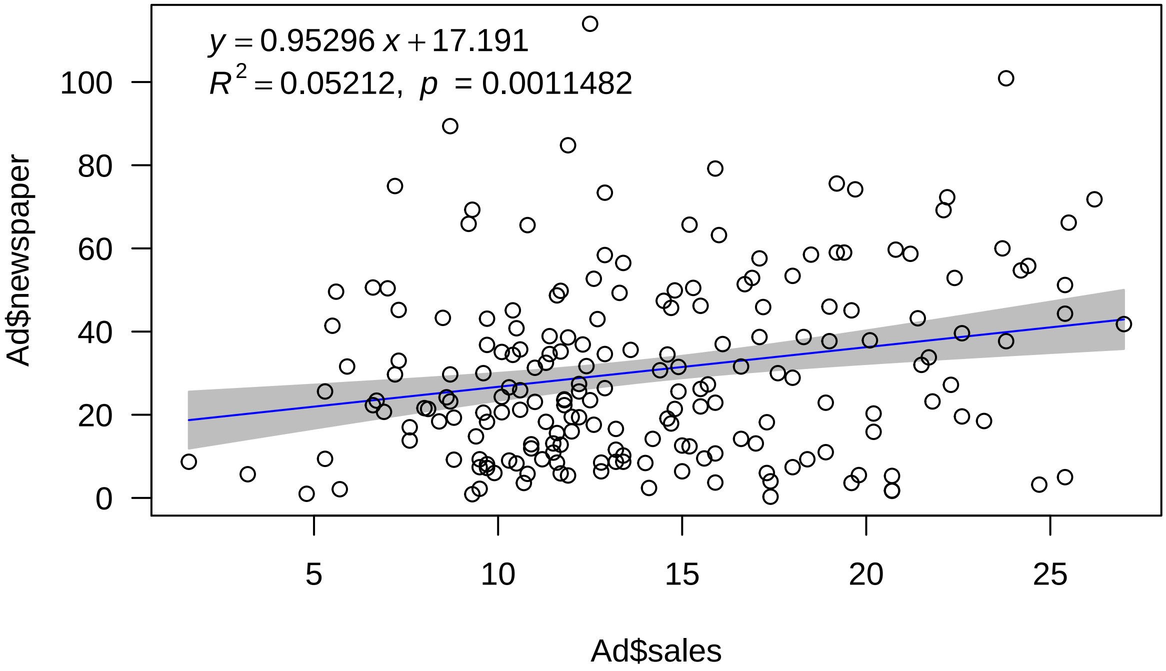
```
##       fit              lwr              upr
##  Min.   :12.37   Min.   :11.28   Min.   :13.45
##  1st Qu.:13.05   1st Qu.:12.23   1st Qu.:13.87
##  Median :13.76   Median :13.11   Median :14.41
##  Mean   :14.02   Mean   :13.16   Mean   :14.88
##  3rd Qu.:14.82   3rd Qu.:14.06   3rd Qu.:15.58
##  Max.   :18.59   Max.   :16.07   Max.   :21.10
```

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

```
trendline(Ad$sales, Ad$newspaper, CI.level=0.92)
```

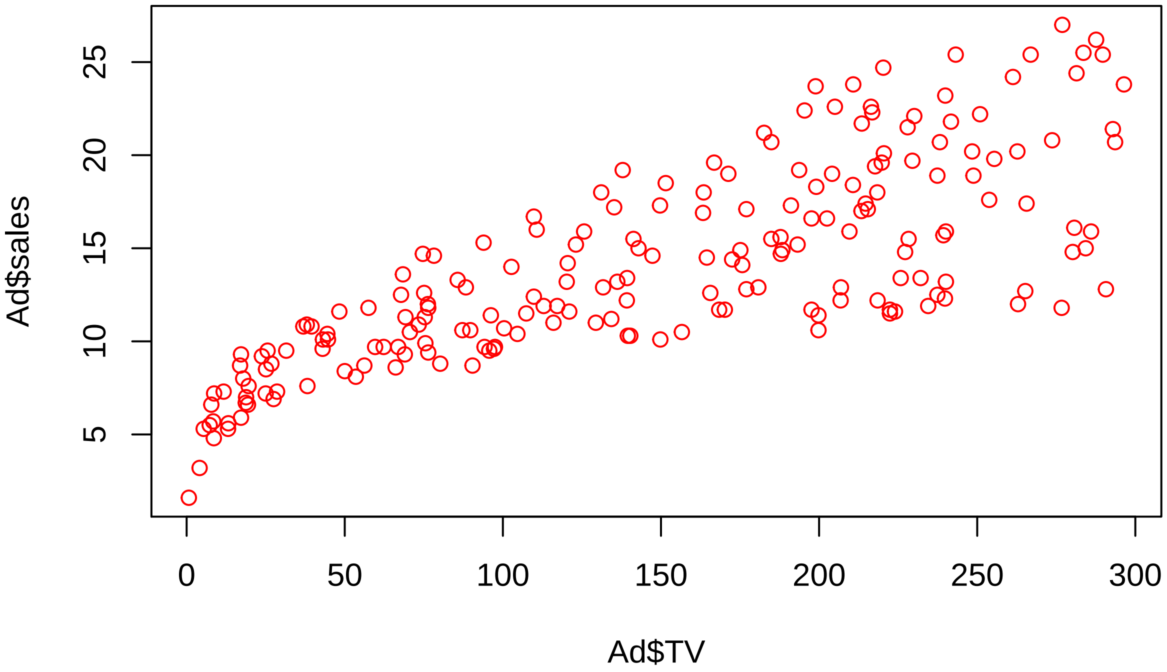
```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -37.5293 -16.6284  -3.1495  13.8392  84.8969
```

```
##
## Coefficients:
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 17.19109    4.31981  3.9796 9.683e-05 ***
## x            0.95296    0.28881  3.2996  0.001148 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.257 on 198 degrees of freedom
## Multiple R-squared:  0.05212,    Adjusted R-squared:  0.047333
## F-statistic: 10.887 on 1 and 198 DF,  p-value: 0.0011482
##
##
## N: 200 , AIC: 1794.2 , BIC:  1804.1
## Residual Sum of Squares:  89468
```



(2) TV

```
lm_ad_tv = lm(Ad$sales~Ad$TV)
plot(Ad$TV,Ad$sales, col="red")
```

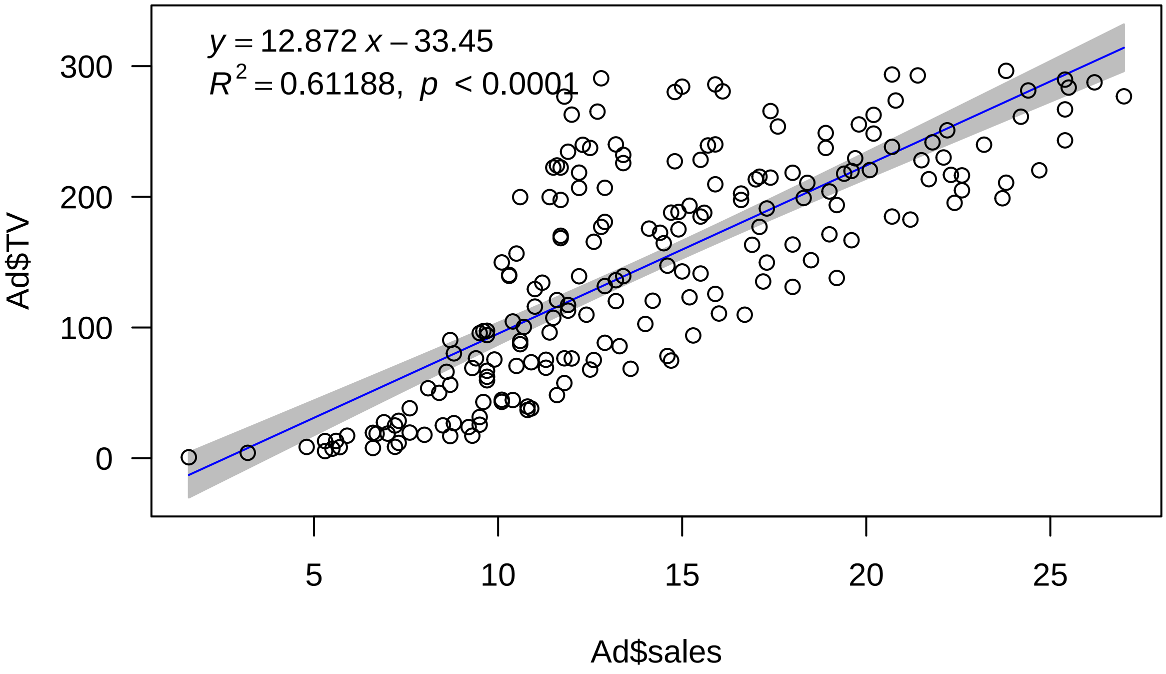


```r
lm_ad_tv = lm(Ad$sales~Ad$TV)
lm_ad_tv_pred = predict(lm_ad_tv, interval="confidence", level=0.92)
summary(lm_ad_tv_pred)
```

```
##       fit              lwr              upr        
##  Min.   : 7.066   Min.   : 6.263   Min.   : 7.869  
##  1st Qu.:10.568   1st Qu.:10.036   1st Qu.:11.100  
##  Median :14.151   Median :13.746   Median :14.557  
##  Mean   :14.023   Mean   :13.462   Mean   :14.583  
##  3rd Qu.:17.435   3rd Qu.:16.906   3rd Qu.:17.964  
##  Max.   :21.122   Max.   :20.307   Max.   :21.938  
```

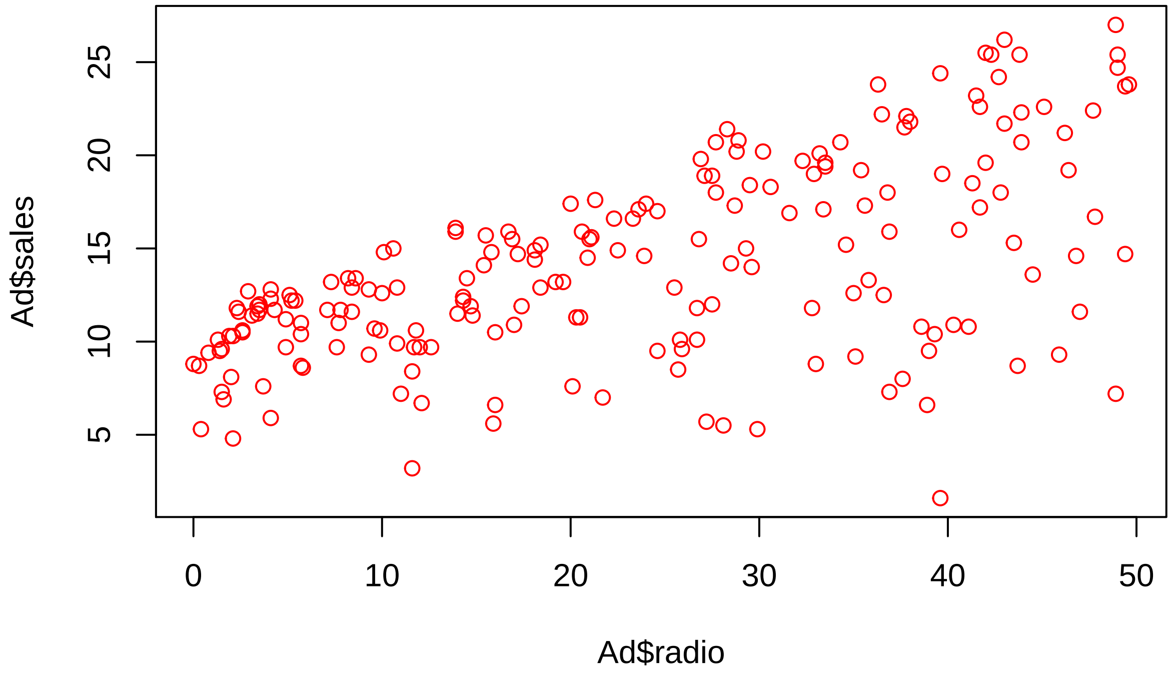
```r
trendline(Ad$sales, Ad$TV, CI.level=0.92)
```

```
## 
## Call:
## lm(formula = y ~ x)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -81.063 -40.121 -11.145  27.753 159.393 
## 
## Coefficients:
##              Estimate Std. Error t value  Pr(>|t|)   
## (Intercept) -33.45023   10.89695 -3.0697  0.002443 **
## x             12.87165    0.72854 17.6676 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 53.622 on 198 degrees of freedom
## Multiple R-squared:  0.61188,    Adjusted R-squared:  0.60991
## F-statistic: 312.14 on 1 and 198 DF,  p-value: < 2.22e-16
##
##
## N: 200 , AIC: 2164.3 , BIC:  2174.2
## Residual Sum of Squares:  569309
```



(3) Radio

```
lm_ad_radio = lm(Ad$sales~Ad$radio)
plot(Ad$radio,Ad$sales, col="red")
```

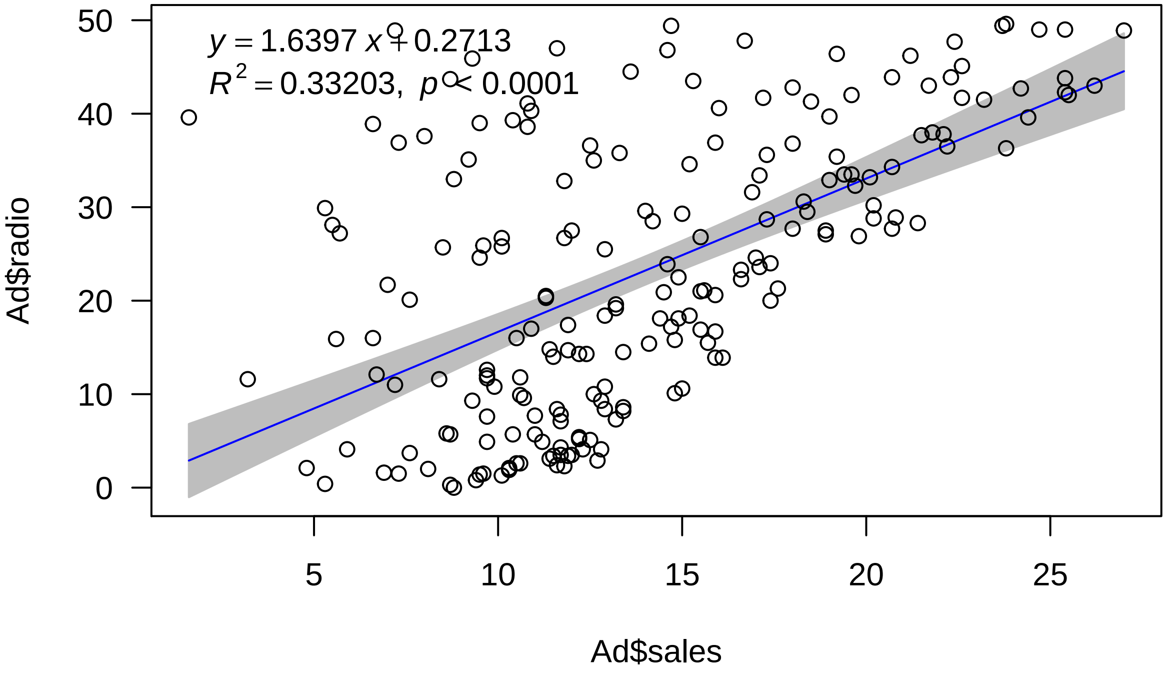


```
lm_ad_radio = lm(Ad$sales~Ad$radio)
lm_ad_radio_pred = predict(lm_ad_radio, interval="confidence", level=0.92)
summary(lm_ad_radio_pred)
```

```
##       fit               lwr               upr
##  Min.   : 9.312   Min.   : 8.321   Min.   :10.30
##  1st Qu.:11.332   1st Qu.:10.617   1st Qu.:12.05
##  Median :13.949   Median :13.417   Median :14.48
##  Mean   :14.023   Mean   :13.287   Mean   :14.76
##  3rd Qu.:16.708   3rd Qu.:15.994   3rd Qu.:17.42
##  Max.   :19.355   Max.   :18.270   Max.   :20.44
```

```
trendline(Ad$sales, Ad$radio, CI.level=0.92)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.1955  -8.8107  -2.3495   7.4133  36.8229
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.27130    2.47211  0.1097   0.9127
## x            1.63970    0.16528  9.9208   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.165 on 198 degrees of freedom
## Multiple R-squared:  0.33203,    Adjusted R-squared:  0.32866
## F-statistic: 98.422 on 1 and 198 DF,  p-value: < 2.22e-16
##
##
## N: 200 , AIC: 1571 , BIC:  1580.9
## Residual Sum of Squares:  29300
```



P5

```
Auto = read.csv("Auto.csv", header=T, na.strings="?")
dim(Auto)
```

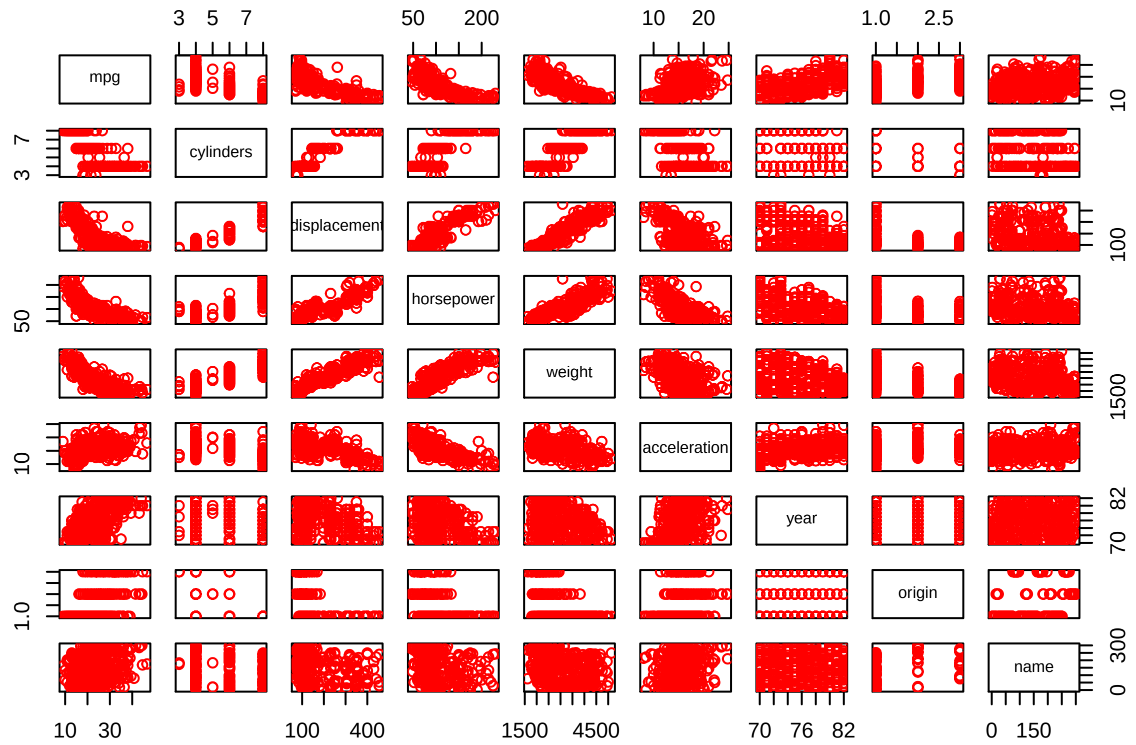
```
## [1] 392   9
```

```
Auto[1:4,]
```

```
##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
## 4  16         8          304        150   3433         12.0   70      1
##                          name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
```

(a)

```r
pairs(Auto, col="red")
```



(b)

```r
selected = Auto[,1:8]
dim(selected)
```

```
## [1] 392   8
```

```r
cor(selected)
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
##              acceleration       year     origin
## mpg             0.4233285  0.5805410  0.5652088
## cylinders      -0.5046834 -0.3456474 -0.5689316
## displacement   -0.5438005 -0.3698552 -0.6145351
## horsepower     -0.6891955 -0.4163615 -0.4551715
## weight         -0.4168392 -0.3091199 -0.5850054
## acceleration    1.0000000  0.2903161  0.2127458
## year            0.2903161  1.0000000  0.1815277
## origin          0.2127458  0.1815277  1.0000000
```

(c)

```
lm_mul = lm(Auto$mpg~Auto$cylinders+Auto$displacement+Auto$horsepower+Auto$weight+Auto$acceleration+Auto
summary(lm_mul)
```

```
##
## Call:
## lm(formula = Auto$mpg ~ Auto$cylinders + Auto$displacement +
##     Auto$horsepower + Auto$weight + Auto$acceleration + Auto$year)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6927 -2.3864 -0.0801  2.0291 14.3607
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -1.454e+01  4.764e+00  -3.051  0.00244 **
## Auto$cylinders    -3.299e-01  3.321e-01  -0.993  0.32122
## Auto$displacement  7.678e-03  7.358e-03   1.044  0.29733
## Auto$horsepower   -3.914e-04  1.384e-02  -0.028  0.97745
## Auto$weight       -6.795e-03  6.700e-04 -10.141  < 2e-16 ***
## Auto$acceleration  8.527e-02  1.020e-01   0.836  0.40383
## Auto$year          7.534e-01  5.262e-02  14.318  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.435 on 385 degrees of freedom
## Multiple R-squared:  0.8093, Adjusted R-squared:  0.8063
## F-statistic: 272.2 on 6 and 385 DF,  p-value: < 2.2e-16
```

i: Yes, there's a relationship between the predictors and the response, as the adjusted R-square score is 0.8063. ii: Weight and year appear to have a statistically significant relationship to the response. iii: It suggests that the year is very important to the mpg, as year increases(i.e. the car models are newer), the mpg increases significantly.

(d)

```r
lm_mul2 = lm(Auto$mpg~sqrt(Auto$cylinders)+sqrt(Auto$displacement)+sqrt(Auto$horsepower)+sqrt(Auto$weig
summary(lm_mul2)
```

```
##
## Call:
## lm(formula = Auto$mpg ~ sqrt(Auto$cylinders) + sqrt(Auto$displacement) +
##     sqrt(Auto$horsepower) + sqrt(Auto$weight) + sqrt(Auto$acceleration) +
##     sqrt(Auto$year))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0770 -1.9915 -0.2719  1.7993 13.9583
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -45.0956     9.3107  -4.843 1.85e-06 ***
## sqrt(Auto$cylinders)      1.0224     1.5417   0.663   0.5076
## sqrt(Auto$displacement)  -0.1794     0.2132  -0.841   0.4007
## sqrt(Auto$horsepower)    -0.5345     0.3090  -1.730   0.0845 .
## sqrt(Auto$weight)        -0.6222     0.0807  -7.709 1.09e-13 ***
## sqrt(Auto$acceleration)  -0.9155     0.8524  -1.074   0.2835
## sqrt(Auto$year)          12.7588     0.8777  14.537  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.281 on 385 degrees of freedom
## Multiple R-squared:  0.826,  Adjusted R-squared:  0.8233
## F-statistic: 304.7 on 6 and 385 DF,  p-value: < 2.2e-16
```

For $\sqrt{X}$, the model is very similar to X, and the adjusted R-square score increased. The most important features are still year and weight.

```r
lm_mul3 = lm(Auto$mpg~I(Auto$cylinders^2)+I(Auto$displacement^2)+I(Auto$horsepower^2)+I(Auto$weight^2)+
summary(lm_mul3)
```

```
##
## Call:
## lm(formula = Auto$mpg ~ I(Auto$cylinders^2) + I(Auto$displacement^2) +
##     I(Auto$horsepower^2) + I(Auto$weight^2) + I(Auto$acceleration^2) +
##     I(Auto$year^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9076 -2.6160 -0.0569  2.1774 14.7696
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             3.084e+00  2.437e+00   1.265  0.20654
## I(Auto$cylinders^2)    -9.796e-02  2.626e-02  -3.730  0.00022 ***
## I(Auto$displacement^2)  4.477e-05  1.428e-05   3.135  0.00185 **
## I(Auto$horsepower^2)    1.975e-05  5.101e-05   0.387  0.69886
## I(Auto$weight^2)       -1.014e-06  9.272e-08 -10.934  < 2e-16 ***
## I(Auto$acceleration^2)  5.966e-03  2.808e-03   2.124  0.03429 *
## I(Auto$year^2)          5.078e-03  3.683e-04  13.788  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.695 on 385 degrees of freedom
## Multiple R-squared:  0.7793, Adjusted R-squared:  0.7759
## F-statistic: 226.6 on 6 and 385 DF,  p-value: < 2.2e-16
```

For $X^2$, the adjusted R-square score decreases. The most important features include cylinders as well.

```
lm_mul4 = lm(Auto$mpg~log(Auto$cylinders)+log(Auto$displacement)+log(Auto$horsepower)+log(Auto$weight)+
summary(lm_mul4)
```

```
##
## Call:
## lm(formula = Auto$mpg ~ log(Auto$cylinders) + log(Auto$displacement) +
##     log(Auto$horsepower) + log(Auto$weight) + log(Auto$acceleration) +
##     log(Auto$year))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5641 -1.7873 -0.0611  1.5810 13.2714
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -62.413     17.650  -3.536 0.000456 ***
## log(Auto$cylinders)       2.750      1.626   1.691 0.091585 .
## log(Auto$displacement)   -3.406      1.355  -2.513 0.012371 *
## log(Auto$horsepower)     -6.386      1.563  -4.085 5.36e-05 ***
## log(Auto$weight)        -11.905      2.240  -5.316 1.80e-07 ***
## log(Auto$acceleration)   -5.326      1.622  -3.283 0.001119 **
## log(Auto$year)           54.825      3.595  15.250  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.103 on 385 degrees of freedom
## Multiple R-squared:  0.8444, Adjusted R-squared:  0.8419
## F-statistic: 348.1 on 6 and 385 DF,  p-value: < 2.2e-16
```

For $log(X)$, the model's adjusted R-square score increased significantly. The most important features also includes horsepower.