Question 1

Based on Table 3.4, we can conclude that both TV and radio is has significant impact on sales performance. While newspaper does not prove to have a significant impact on sales.

Question 2

KNN regression averages the closest observations to estimate predict, KNN classifier assigns classification group based on majority of closest observations.

Question 3

(a)

i. inccorect, since there is an interaction term between gender and GPA, it is hard to determine whether X3 will be positively contributing to salary or negatively

ii. inccorect, same reason as above, if GPA is larger than 3.5, then male earns more than female, vice versa

iii. correct, if GPA is higher than .53, then male earns more than female

iv. inccorect, the GPA would need to be below 3.5 for female to earn more than male

(b)

50+20*4+0.07*110+35*1+0.01*4*110-10*4*1=137.1

(c)

False, small coefficient could be attributed to large variable values, unless statistical test is conducted and p-value is calculated, it is hard to prove whether a term is significantly/not significantly contributing to salary.

Question 4

(a)

Training RSS would be lower for cubic regression compared to the linear regression, although the true relation is linear, cubic regression provides more rooms for fitting the trainning set data, since the sole purpose of fitting is to reduce training RSS, with more terms added to the equation, smaller RSS would be expected.

(b)

Since the true relationship is linear, the test RSS would tend to reveal that cubic regression creates an overfit in the model with generally larger RSS than linear regression.

(c)

As we illustrated in question (a), with more terms added to the fitting model for cubic regression, the training RSS would be at least the same or lower compared to linear regression

(d)

Since cubic regression would help better fit the nonlinearity of the data set, it is likely that test RSS is better for cubic regression compared to linear regression

Question 5

$$\hat{y}i = xi \times \frac{\sum_{i'=1}^{n}(x_{i'}y_{i'})}{\sum_{j=1}^{n} x_j^2}$$

$$\hat{y}i = \sum_{i'=1}^{n} \frac{(x_{i'}y_{i'}) \times x_i}{\sum_{j=1}^{n} x_j^2}$$

$$\hat{y}i = \sum_{i'=1}^{n} \left(\frac{x_i x_{i'}}{\sum_{j=1}^{n} x_j^2} \times y_{i'} \right)$$

$$a_{i'} = \frac{x_i x_{i'}}{\sum_{j=1}^{n} x_j^2}$$

Question 6

According to the equation 3.4,

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

beta 1 would equal to 0 when xi equal to x average. beta 0 would equal to y average. Hence the model would prove to be valid regardless.

Question 7

An later exercise …

Question 8

(a)

```r
require(ISLR)
```

```
## Loading required package: ISLR
## Warning: package 'ISLR' was built under R version 3.3.3
```

```r
data(Auto)
lm.fit <- lm(mpg~horsepower, data=Auto)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.935861   0.717499   55.66   <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

i. Yes, according to p-value, there is a significant relationship between the predictor and the response.

ii. The relationship is really strong, since p-value is close to 0

iii. The coefficient estimate suggest a positive relationship

iv.

```
predict(lm.fit, data.frame(horsepower=c(98)))
```

```
##        1
## 24.46708
```

```
predict(lm.fit, data.frame(horsepower=c(98)), interval = "confidence")
```

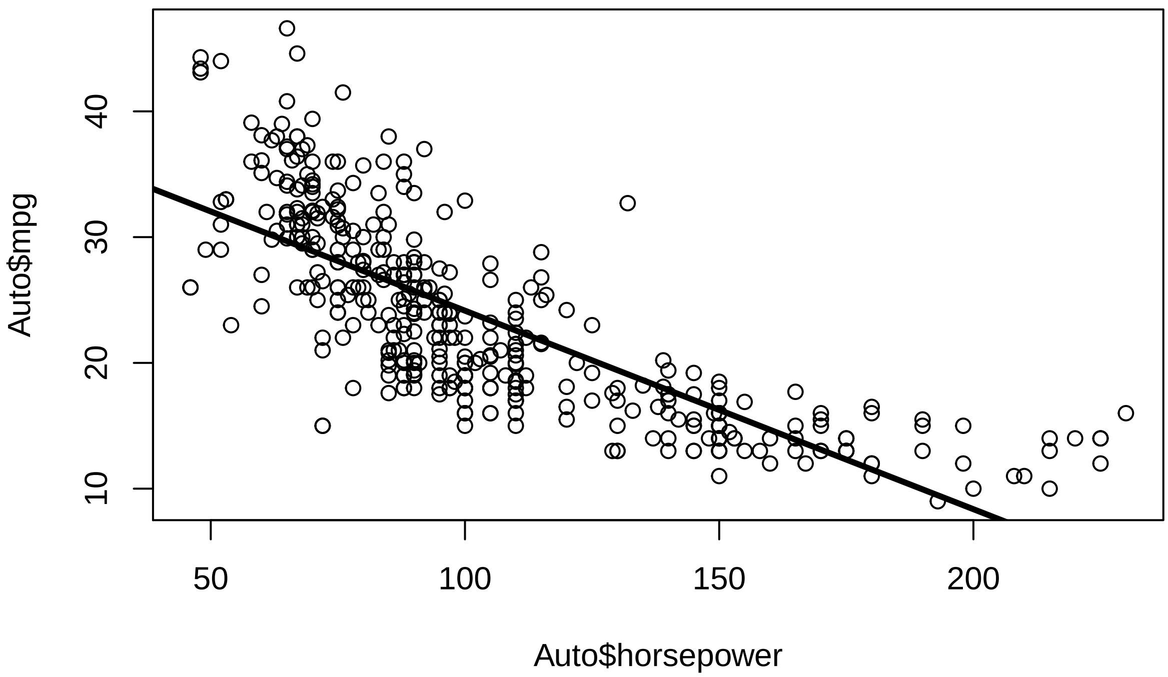
```
##        fit      lwr      upr
## 1 24.46708 23.97308 24.96108
```

```
predict(lm.fit, data.frame(horsepower=c(98)), interval = "prediction")
```

```
##        fit     lwr      upr
## 1 24.46708 14.8094 34.12476
```

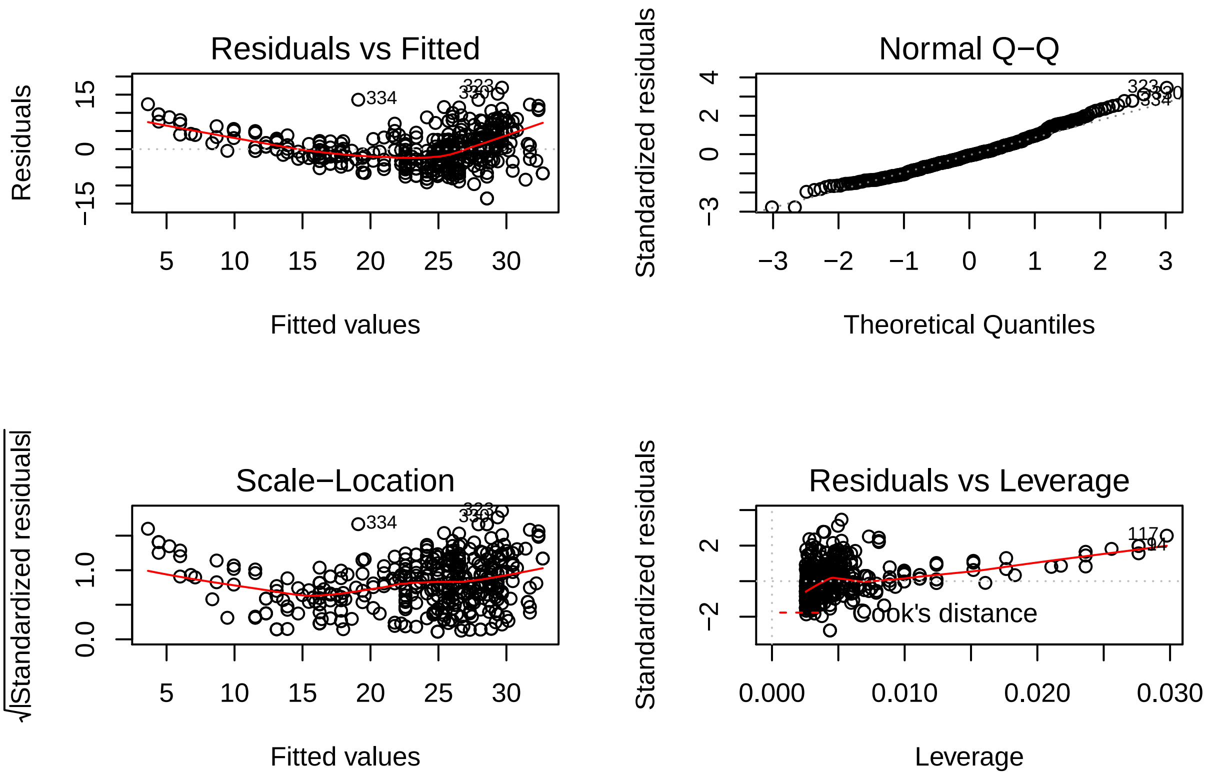
(b)

```
plot(Auto$horsepower,Auto$mpg)
abline(lm.fit, lwd=3)
```



(c)

```
par(mfrow=c(2,2))
plot(lm.fit)
```

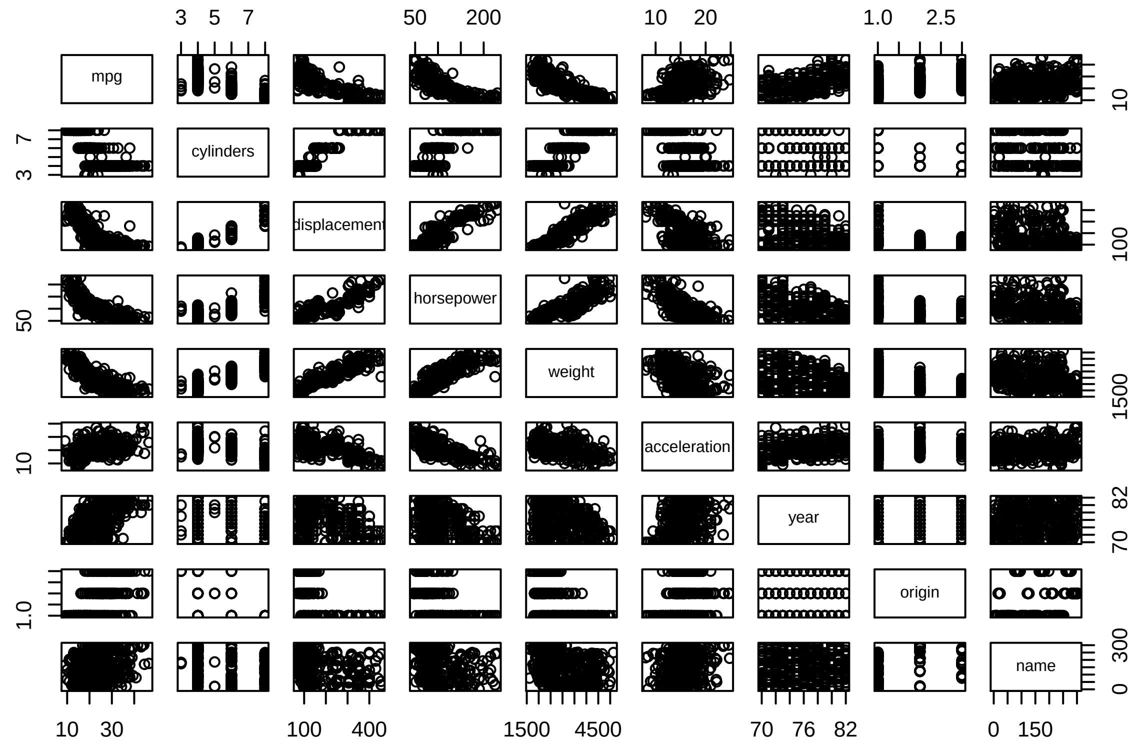


According to Residuals vs. Fitted plot, the residuals seem to be non-linear

Question 9

(a)

```
require(ISLR)
data(Auto)
pairs(Auto)
```



(b)

```r
cor(subset(Auto, select=-c(name)))
```

```
##                     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)

```r
lm.fit <- lm(mpg~.-name,data=Auto)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = mpg ~ . - name, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5903 -2.1565 -0.1169  1.8690 13.0604
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
## cylinders     -0.493376   0.323282  -1.526  0.12780
## displacement   0.019896   0.007515   2.647  0.00844 **
## horsepower    -0.016951   0.013787  -1.230  0.21963
## weight        -0.006474   0.000652  -9.929  < 2e-16 ***
## acceleration   0.080576   0.098845   0.815  0.41548
## year           0.750773   0.050973  14.729  < 2e-16 ***
## origin         1.426141   0.278136   5.127 4.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.328 on 384 degrees of freedom
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

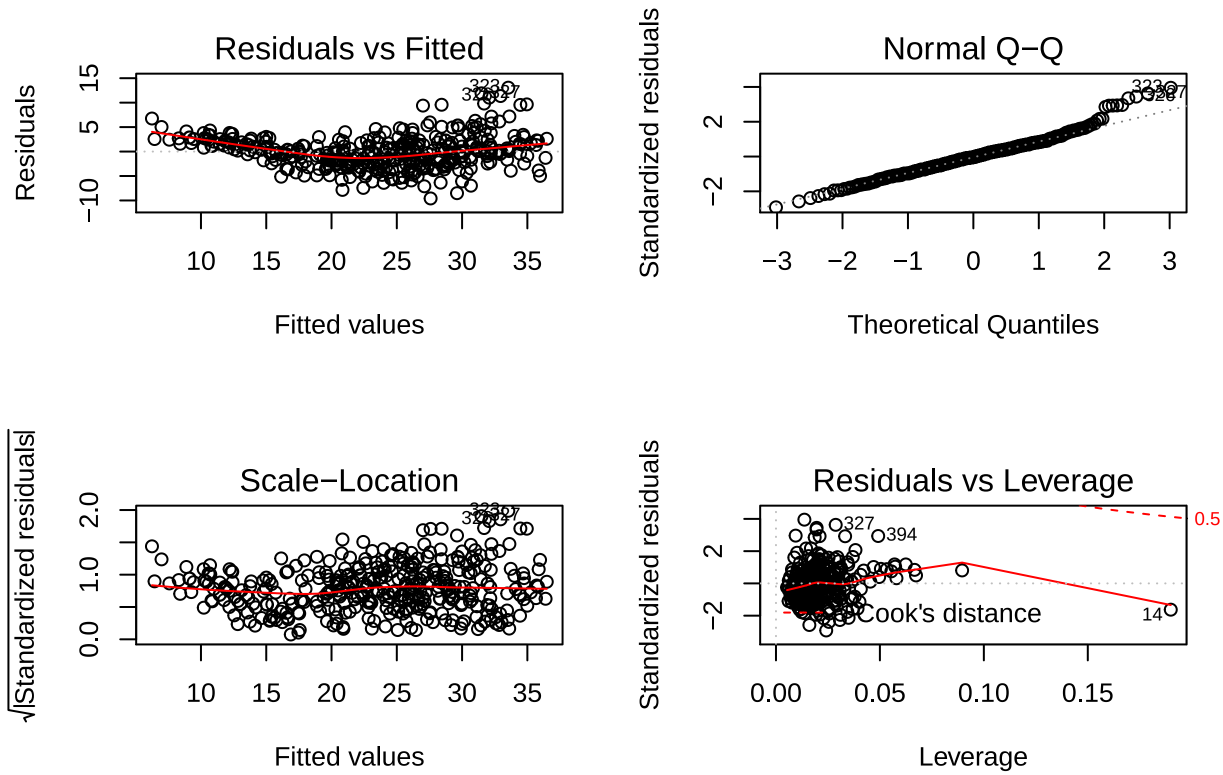
i. Yes, overall p-value is close to 0 indicating there is a strong relationship between the predictors and the response

ii. displacement, weight, year and origin were displaying a statistically significant relationship to the response.

iii. it suggest for every year of increment of the car which are produced, the mpg would increase for about 0.75.

(d)

```r
par(mfrow=c(2,2))
plot(lm.fit)
```



The residual plot suggested some outliers on top right corner of the chart, the leverage plot shows observation 14 has an outstanding leverage compared to rest of the data sets.

(e)

```
lm.fit <- lm(mpg~.-name-displacement-acceleration-cylinders+weight*year+horsepower*origin,data=Auto)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = mpg ~ . - name - displacement - acceleration - cylinders +
##     weight * year + horsepower * origin, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9589 -1.7259 -0.1997  1.4796 11.6792
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -9.916e+01  1.279e+01  -7.753 8.08e-14 ***
## horsepower         3.442e-02  1.491e-02   2.308   0.0215 *
## weight             2.307e-02  4.708e-03   4.900 1.42e-06 ***
## year               1.776e+00  1.741e-01  10.197  < 2e-16 ***
## origin             5.264e+00  8.234e-01   6.392 4.73e-10 ***
## weight:year       -3.783e-04  6.167e-05  -6.134 2.12e-09 ***
## horsepower:origin -5.081e-02  9.407e-03  -5.401 1.16e-07 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.99 on 385 degrees of freedom
## Multiple R-squared:  0.8555, Adjusted R-squared:  0.8533
## F-statistic:   380 on 6 and 385 DF,  p-value: < 2.2e-16
```

Tried the new model with a more spread and linear distribution of residuals

(f)

```
lm.fit <- lm(mpg~.-name-displacement-acceleration-cylinders+weight*year+log(horsepower)+I(year^2),data=A
summary(lm.fit)
```

```
##
## Call:
## lm(formula = mpg ~ . - name - displacement - acceleration - cylinders +
##     weight * year + log(horsepower) + I(year^2), data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5433 -1.6699 -0.0205  1.4391 11.4697
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.991e+02  8.105e+01   3.690 0.000256 ***
## horsepower       1.179e-01  2.193e-02   5.376 1.32e-07 ***
## weight           1.282e-02  4.794e-03   2.673 0.007829 **
## year            -6.415e+00  2.030e+00  -3.160 0.001701 **
## origin           9.791e-01  2.251e-01   4.350 1.75e-05 ***
## log(horsepower) -1.855e+01  2.431e+00  -7.632 1.85e-13 ***
## I(year^2)        5.114e-02  1.292e-02   3.958 9.00e-05 ***
## weight:year     -2.267e-04  6.384e-05  -3.551 0.000431 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.849 on 384 degrees of freedom
## Multiple R-squared:  0.8691, Adjusted R-squared:  0.8667
## F-statistic: 364.3 on 7 and 384 DF,  p-value: < 2.2e-16
```

log horsepower and year^2 does provide more explanatory power to the data set, resulting in higher adjusted R-Squared as well.

Question 10

(a)

```
data(Carseats)
lm.fit <- lm(Sales~Price+Urban+US,data=Carseats)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Sales ~ Price + Urban + US, data = Carseats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -6.9206 -1.6220 -0.0564  1.5786  7.0581
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.043469   0.651012  20.036  < 2e-16 ***
## Price       -0.054459   0.005242 -10.389  < 2e-16 ***
## UrbanYes    -0.021916   0.271650  -0.081    0.936
## USYes        1.200573   0.259042   4.635 4.86e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.472 on 396 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2335
## F-statistic: 41.52 on 3 and 396 DF,  p-value: < 2.2e-16
```

(b)

Price is negatively correlated with Sales, if the store is in urban areas, that would result in less sales, if the store is in US, it will result in more sales.

(c)

$$Sales_i = 13.043 - 0.054Price_i - 0.0219Urban_i + 1.201US_i$$

(d)

For Urban predictors, I can reject it since p-value does not suggest a significant relationship

(e)

```r
lm.fit <- lm(Sales~Price+US,data=Carseats)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Sales ~ Price + US, data = Carseats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.9269 -1.6286 -0.0574  1.5766  7.0515
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.03079    0.63098  20.652  < 2e-16 ***
## Price       -0.05448    0.00523 -10.416  < 2e-16 ***
## USYes        1.19964    0.25846   4.641 4.71e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.469 on 397 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2354
## F-statistic: 62.43 on 2 and 397 DF,  p-value: < 2.2e-16
```

(f)

The adjusted R-Square is slightly higher in later case, but both of them are pretty low

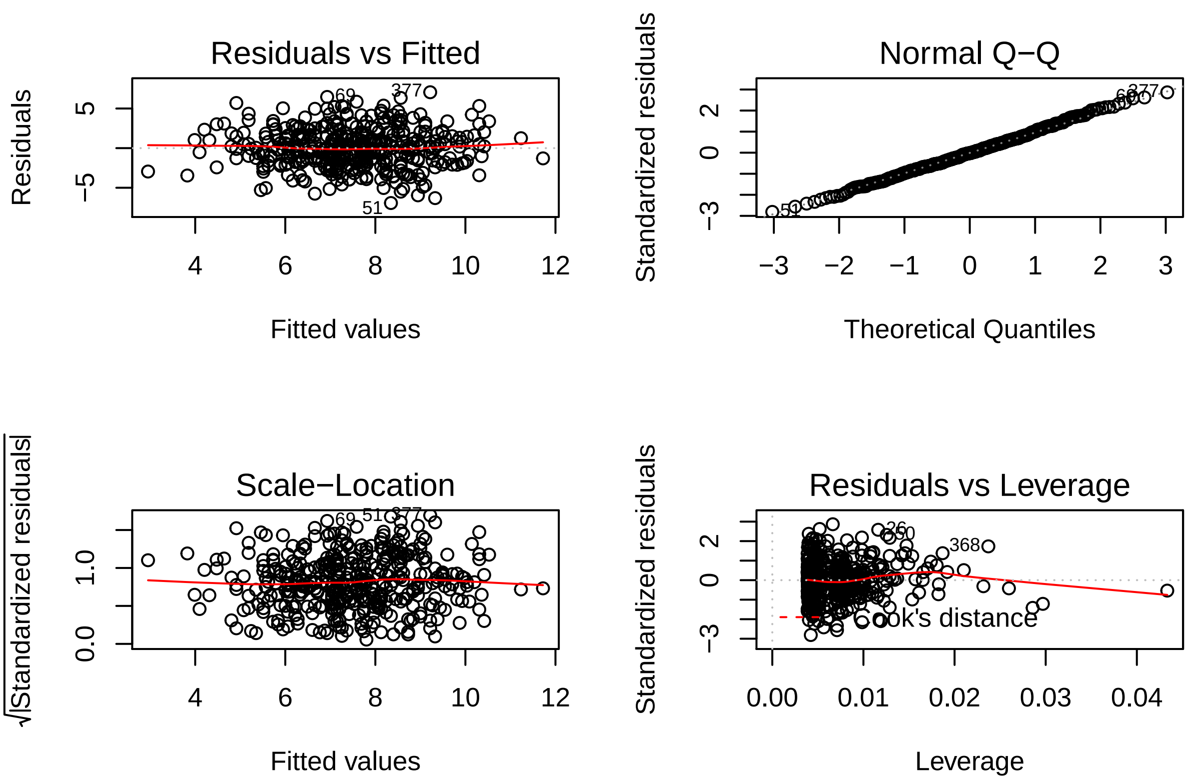
(g)

```
confint(lm.fit)
```

```
##                   2.5 %      97.5 %
## (Intercept) 11.79032020 14.27126531
## Price       -0.06475984 -0.04419543
## USYes        0.69151957  1.70776632
```

(h)

```
par(mfrow=c(2,2))
plot(lm.fit)
```



There is evidence of high leverage for one observation, no evidence of clear outliers

Question 11

```
set.seed(1)
x <- rnorm(100)
y <- 2*x+rnorm(100)
```

(a)

```
lm.fit <- lm(y~x+0)
summary(lm.fit)
```

```
##
## Call:
```

```
## lm(formula = y ~ x + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9154 -0.6472 -0.1771  0.5056  2.3109
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x   1.9939     0.1065   18.73   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9586 on 99 degrees of freedom
## Multiple R-squared:  0.7798, Adjusted R-squared:  0.7776
## F-statistic: 350.7 on 1 and 99 DF,  p-value: < 2.2e-16
```

The coefficient estimate

$$\hat{\beta}$$

is 1.9939. The standard error of the estimate is 0.1065. Given significant large t value coupled with close to 0 p-value we can reject null hypothesis and conclude that there is a significant relationship between y and x.

(b)

```
lm.fit <- lm(x~y+0)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8699 -0.2368  0.1030  0.2858  0.8938
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## y  0.39111    0.02089   18.73   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4246 on 99 degrees of freedom
## Multiple R-squared:  0.7798, Adjusted R-squared:  0.7776
## F-statistic: 350.7 on 1 and 99 DF,  p-value: < 2.2e-16
```

The coefficient estimate

$$\hat{\beta}$$

is 0.39111. The standard error of the estimate is 0.02089. Given significant large t value coupled with close to 0 p-value we can reject null hypothesis and conclude that there is a significant relationship between x and y.

(c)

inversely correlated

(d)

...

(e)

As the formula suggested, by exchanging x and y in the formulat does not really change the outcome of the result since it's all product terms, hence the t-statistic for both regression should be the same.

(f)

```
lm1.fit <- lm(y~x)
lm2.fit <- lm(x~y)
summary(lm1.fit)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8768 -0.6138 -0.1395  0.5394  2.3462
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.03769    0.09699  -0.389    0.698
## x            1.99894    0.10773  18.556   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9628 on 98 degrees of freedom
## Multiple R-squared:  0.7784, Adjusted R-squared:  0.7762
## F-statistic: 344.3 on 1 and 98 DF,  p-value: < 2.2e-16
```

```
summary(lm2.fit)
```

```
##
## Call:
## lm(formula = x ~ y)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.90848 -0.28101  0.06274  0.24570  0.85736
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.03880    0.04266    0.91    0.365
## y            0.38942    0.02099   18.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4249 on 98 degrees of freedom
## Multiple R-squared:  0.7784, Adjusted R-squared:  0.7762
## F-statistic: 344.3 on 1 and 98 DF,  p-value: < 2.2e-16
```

t-statistics for both are very close

Question 12

(a)

When xi equal to yi, then the estimator will be the same, and sd is small enough and not affecting prediction of estimator

(b)

```
set.seed(1)
x <- rnorm(100)
y <- 2*x+rnorm(100)
```

In this case coefficient would be different between X onto Y and Y onto X.

```
lm1.fit <- lm(y~x+0)
lm2.fit <- lm(x~y+0)
summary(lm1.fit)
```

```
##
## Call:
## lm(formula = y ~ x + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9154 -0.6472 -0.1771  0.5056  2.3109
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x   1.9939     0.1065   18.73   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9586 on 99 degrees of freedom
## Multiple R-squared:  0.7798, Adjusted R-squared:  0.7776
## F-statistic: 350.7 on 1 and 99 DF,  p-value: < 2.2e-16
```

```
summary(lm2.fit)
```

```
##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8699 -0.2368  0.1030  0.2858  0.8938
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## y  0.39111    0.02089   18.73   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4246 on 99 degrees of freedom
## Multiple R-squared:  0.7798, Adjusted R-squared:  0.7776
## F-statistic: 350.7 on 1 and 99 DF,  p-value: < 2.2e-16
```

(c)

```
set.seed(1)
x <- rnorm(100)
y <- x + rnorm(100, mean = 0, sd = 0.001)
```

In this case, they will be the same

```
lm1.fit <- lm(y~x+0)
lm2.fit <- lm(x~y+0)
summary(lm1.fit)
```

```
##
## Call:
## lm(formula = y ~ x + 0)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0019154 -0.0006472 -0.0001771  0.0005056  0.0023109
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x 0.9999939  0.0001065    9392   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0009586 on 99 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 8.82e+07 on 1 and 99 DF,  p-value: < 2.2e-16
```

```
summary(lm2.fit)
```

```
##
## Call:
## lm(formula = x ~ y + 0)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0023104 -0.0005047  0.0001771  0.0006482  0.0019152
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## y 1.0000050  0.0001065    9392   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0009586 on 99 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 8.82e+07 on 1 and 99 DF,  p-value: < 2.2e-16
```

Question 13

(a)

```
set.seed(1)
x <- rnorm(100, mean = 0, sd = 1)
```

(b)

```r
eps <- rnorm(100, mean = 0, sd = 0.25)
```

(c)

```r
y <- -1 + 0.5*x + eps
length(y)
```

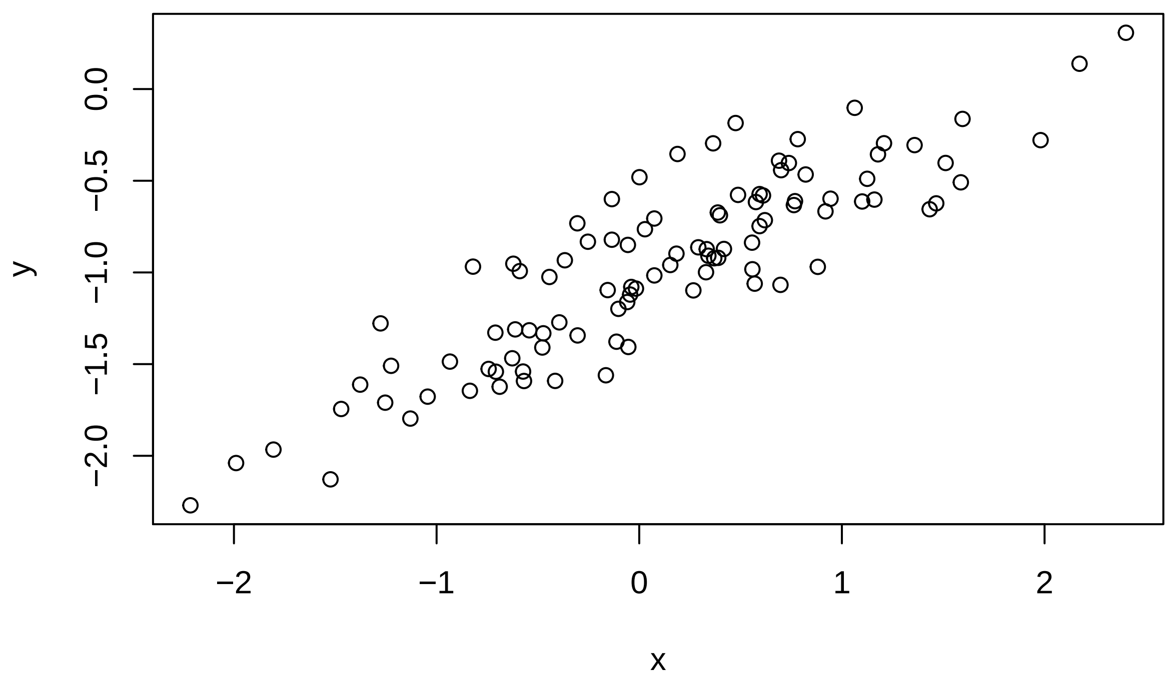
```
## [1] 100
```

vector length is 100.

intercept term is around -1 and coefficient for x is around 0.5.

(d)

```r
plot(x,y)
```



I can observe a relatively strong linear relationship

(e)

```r
lm.fit <- lm(y~x)
summary(lm.fit)
```

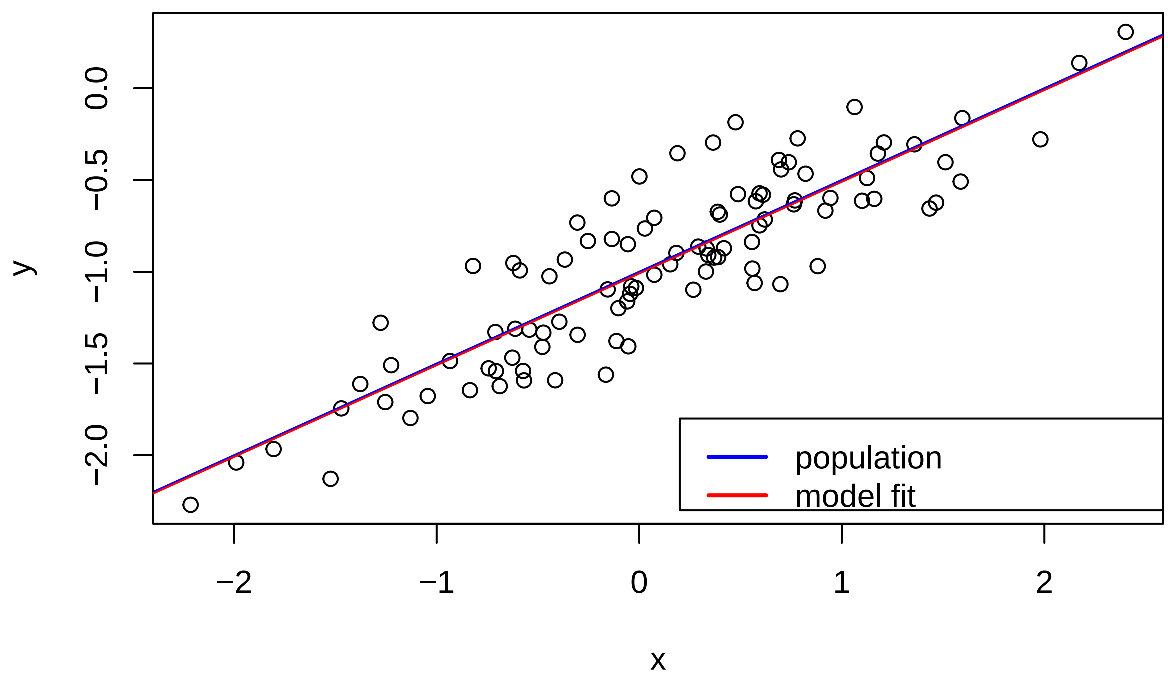
```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -0.46921 -0.15344 -0.03487  0.13485  0.58654
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.00942    0.02425  -41.63   <2e-16 ***
## x            0.49973    0.02693   18.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2407 on 98 degrees of freedom
## Multiple R-squared:  0.7784, Adjusted R-squared:  0.7762
## F-statistic: 344.3 on 1 and 98 DF,  p-value: < 2.2e-16
```

modelled intercept term is -1.00942, modelled coefficient term is 0.49973.

(f)

```r
plot(x,y)
abline(-1,0.5, col = "blue")
abline(lm.fit, col = "red")
legend(x = c(0.2,7),
       y = c(-1.8,-2.3),
       legend = c("population", "model fit"),
       col = c("blue","red"), lwd = 2)
```



(g)

```r
lm.fit.poly <- lm(y~x+I(x^2))
summary(lm.fit.poly)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4913 -0.1563 -0.0322  0.1451  0.5675
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.98582    0.02941 -33.516   <2e-16 ***
## x            0.50429    0.02700  18.680   <2e-16 ***
## I(x^2)      -0.02973    0.02119  -1.403    0.164
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2395 on 97 degrees of freedom
## Multiple R-squared:  0.7828, Adjusted R-squared:  0.7784
## F-statistic: 174.8 on 2 and 97 DF,  p-value: < 2.2e-16
```

No, the quadratic term does not prove to be significantly correlated with result term, hence it does not increase the model fit.

(h)

```r
eps <- rnorm(100, mean = 0, sd = 0.5)
y <- -1 + 0.5*x + eps
lm.fit.more.noisy <- lm(y~x)
summary(lm.fit.more.noisy)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.45706 -0.24115 -0.02266  0.32462  1.32079
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.98632    0.05235 -18.840  < 2e-16 ***
## x            0.51058    0.05815   8.781 5.34e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5197 on 98 degrees of freedom
## Multiple R-squared:  0.4403, Adjusted R-squared:  0.4346
## F-statistic:  77.1 on 1 and 98 DF,  p-value: 5.336e-14
```

(i)

```r
eps <- rnorm(100, mean = 0, sd = 0.1)
y <- -1 + 0.5*x + eps
```

```
lm.fit.less.noisy <- lm(y~x)
summary(lm.fit.less.noisy)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.251626 -0.054525 -0.003776  0.067289  0.187887
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.99423    0.01003  -99.14   <2e-16 ***
## x            0.49443    0.01114   44.39   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09955 on 98 degrees of freedom
## Multiple R-squared:  0.9526, Adjusted R-squared:  0.9521
## F-statistic:  1970 on 1 and 98 DF,  p-value: < 2.2e-16
```

(j)

```
confint(lm.fit)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0575402 -0.9613061
## x            0.4462897  0.5531801
```

```
confint(lm.fit.more.noisy)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0902064 -0.8824249
## x            0.3951885  0.6259784
```

```
confint(lm.fit.less.noisy)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0141340 -0.9743329
## x            0.4723272  0.5165356
```

The confidence interval is larger when the data set is noisier and vice versa

Question 14

(a)

```
set.seed(1)
x1 <- runif(100)
x2 <- 0.5*x1 + rnorm(100)/10
y <- 2 + 2*x1 + 0.3*x2 + rnorm(100)
```

population regression is

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$$

The coefficient are

$$\beta_0 = 2$$

,

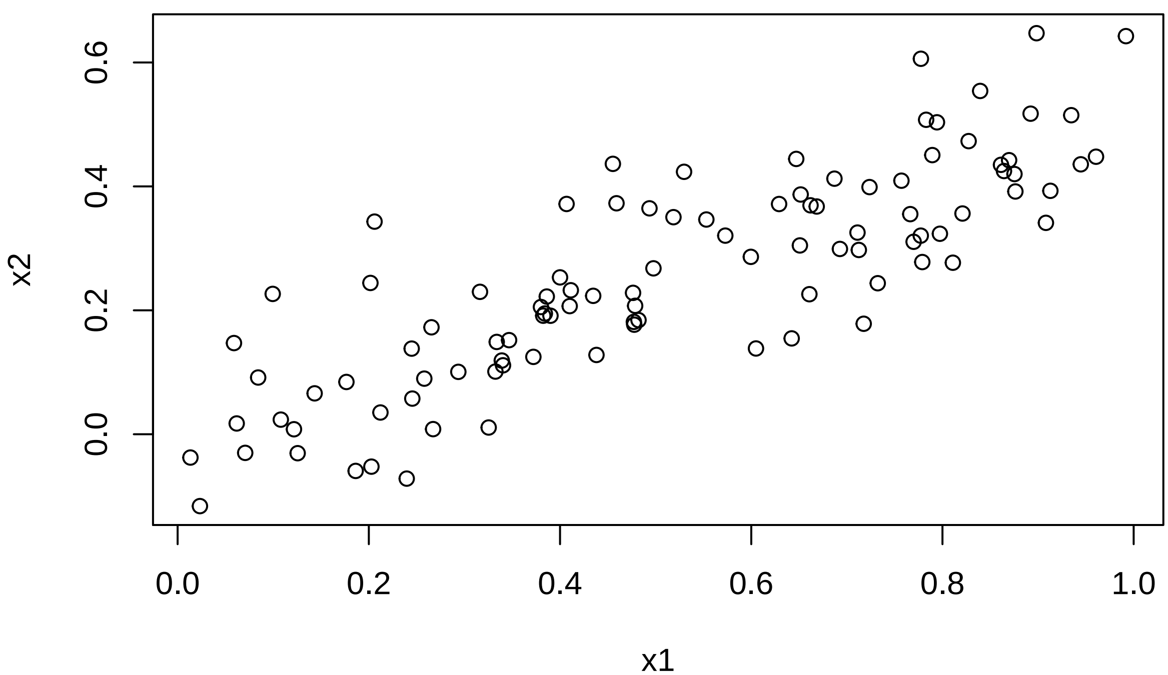
$$\beta_1 = 2$$

and

$$\beta_2 = 0.3$$

(b)

```
plot(x1, x2)
```



There is a relatively strong linear relationship between the two variables.

(c)

```
lm.fit <- lm(y~x1+x2)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.8311 -0.7273 -0.0537  0.6338  2.3359
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   2.1305     0.2319   9.188 7.61e-15 ***
## x1            1.4396     0.7212   1.996   0.0487 *
## x2            1.0097     1.1337   0.891   0.3754
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.056 on 97 degrees of freedom
## Multiple R-squared:  0.2088, Adjusted R-squared:  0.1925
## F-statistic:  12.8 on 2 and 97 DF,  p-value: 1.164e-05
```

Intercept is mostly close to population intercept, but both x1 and x2 are slightly far away from true x1 and x2.

Under 95% confidence interval we will reject hypothesis, but we won't reject hypothesis on beta2

(d)

```r
lm.fit <- lm(y~x1)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.89495 -0.66874 -0.07785  0.59221  2.45560
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.1124     0.2307   9.155 8.27e-15 ***
## x1            1.9759     0.3963   4.986 2.66e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.055 on 98 degrees of freedom
## Multiple R-squared:  0.2024, Adjusted R-squared:  0.1942
## F-statistic: 24.86 on 1 and 98 DF,  p-value: 2.661e-06
```

Yes, we can reject the null hypothesis, since the relationship is significant

(e)

```r
lm.fit <- lm(y~x2)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.62687 -0.75156 -0.03598  0.72383  2.44890
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3899     0.1949   12.26  < 2e-16 ***
## x2            2.8996     0.6330    4.58 1.37e-05 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.072 on 98 degrees of freedom
## Multiple R-squared:  0.1763, Adjusted R-squared:  0.1679
## F-statistic: 20.98 on 1 and 98 DF,  p-value: 1.366e-05
```

Yes, we can reject the null hypothesis, since the relationship is signficant

(f)

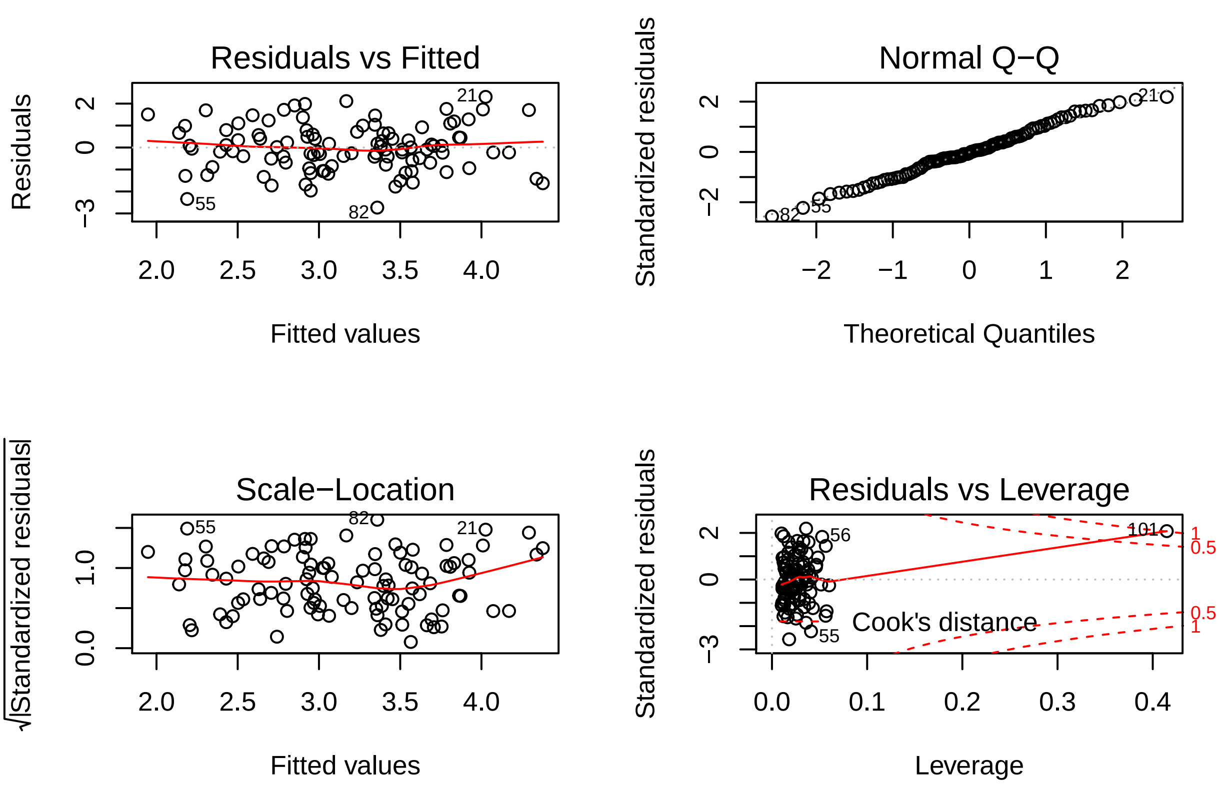
No, they don't. As far as we can tell based on the real relationship, x1 can mostly explained the movements in x2. Hence when both x1 and x2 are presented as coefficient, due to collinearility, only one variable is actually needed for the model resulting us rejecting x2.

In other scenarios, only one variable is used, hence we cannot reject any of them.

(g)

```r
x1 <- c(x1, 0.1)
x2 <- c(x2, 0.8)
y <- c(y, 6)
```

```r
lm.fit <- lm(y~x1+x2)
par(mfrow=c(2,2))
plot(lm.fit)
```



Clearly the newly added item 101 has a high leverage in the data set compared with other data

Question 15

(a)

```r
library(MASS)
names(Boston)[-1]
```

```
##  [1] "zn"      "indus"   "chas"    "nox"     "rm"      "age"     "dis"
##  [8] "rad"     "tax"     "ptratio" "black"   "lstat"   "medv"
```

```r
lmp <- function (modelobject) {
    if (class(modelobject) != "lm")
      stop("Not an object of class 'lm' ")
    f <- summary(modelobject)$fstatistic
    p <- pf(f[1],f[2],f[3],lower.tail=F)
    attributes(p) <- NULL
    return(p)
}

results <- combn(names(Boston), 2,
                 function(x) { lmp(lm(Boston[, x])) },
                 simplify = FALSE)

vars <- combn(names(Boston), 2)
names(results) <- paste(vars[1,],vars[2,],sep="~")
results[1:13]
```

```
## $`crim~zn`
## [1] 5.506472e-06
##
## $`crim~indus`
## [1] 1.450349e-21
##
## $`crim~chas`
## [1] 0.2094345
##
## $`crim~nox`
## [1] 3.751739e-23
##
## $`crim~rm`
## [1] 6.346703e-07
##
## $`crim~age`
## [1] 2.854869e-16
##
## $`crim~dis`
## [1] 8.519949e-19
##
## $`crim~rad`
## [1] 2.693844e-56
##
## $`crim~tax`
## [1] 2.357127e-47
##
## $`crim~ptratio`
## [1] 2.942922e-11
```

```
##
## $`crim~black`
## [1] 2.487274e-19
##
## $`crim~lstat`
## [1] 2.654277e-27
##
## $`crim~medv`
## [1] 1.173987e-19
```

The rad variable has a very significant correlation with per capita crime rate by town. At the same time, it produced higher R square as well.

(b)

```
lm.fit.multi <- lm(crim~., data=Boston)
summary(lm.fit.multi)
```

```
##
## Call:
## lm(formula = crim ~ ., data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.924 -2.120 -0.353  1.019 75.051
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.033228   7.234903   2.354 0.018949 *
## zn            0.044855   0.018734   2.394 0.017025 *
## indus        -0.063855   0.083407  -0.766 0.444294
## chas         -0.749134   1.180147  -0.635 0.525867
## nox         -10.313535   5.275536  -1.955 0.051152 .
## rm            0.430131   0.612830   0.702 0.483089
## age           0.001452   0.017925   0.081 0.935488
## dis          -0.987176   0.281817  -3.503 0.000502 ***
## rad           0.588209   0.088049   6.680 6.46e-11 ***
## tax          -0.003780   0.005156  -0.733 0.463793
## ptratio      -0.271081   0.186450  -1.454 0.146611
## black        -0.007538   0.003673  -2.052 0.040702 *
## lstat         0.126211   0.075725   1.667 0.096208 .
## medv         -0.198887   0.060516  -3.287 0.001087 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.439 on 492 degrees of freedom
## Multiple R-squared:  0.454,  Adjusted R-squared:  0.4396
## F-statistic: 31.47 on 13 and 492 DF,  p-value: < 2.2e-16
```

The above model is developed using backwardation method, and end up with only zn, dis, rad and medv as the variables. Under this scenario, we can reject null hypothesis.

(c)

```
results <- combn(names(Boston), 2,
                 function(x) { coefficients(lm(Boston[, x])) },
                 simplify = FALSE)
```

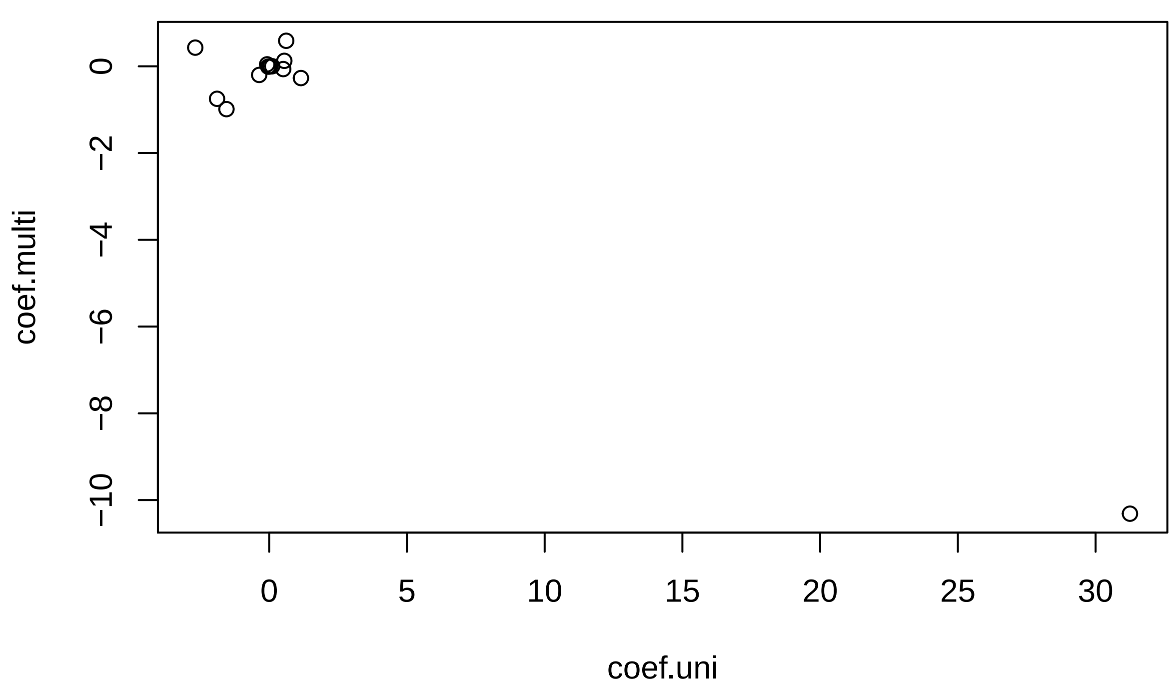
```r
(coef.uni <- unlist(results)[seq(2,26,2)])
```

```
##          zn       indus        chas         nox          rm         age
## -0.07393498  0.50977633 -1.89277655 31.24853120 -2.68405122  0.10778623
##         dis         rad         tax     ptratio       black       lstat
## -1.55090168  0.61791093  0.02974225  1.15198279 -0.03627964  0.54880478
##        medv
## -0.36315992
```

```r
(coef.multi <- coefficients(lm.fit.multi)[-1])
```

```
##            zn         indus          chas           nox            rm
##   0.044855215  -0.063854824  -0.749133611 -10.313534912   0.430130506
##           age           dis           rad           tax       ptratio
##   0.001451643  -0.987175726   0.588208591  -0.003780016  -0.271080558
##         black         lstat          medv
##  -0.007537505   0.126211376  -0.198886821
```

```r
plot(coef.uni, coef.multi)
```



beta coefficient tend to be very different between multivariate regression and single variable regression.

(d)

```r
lm.fit.poly <- lm(crim~poly(zn,3), data = Boston)
summary(lm.fit.poly)
```

```
##
## Call:
```

```
## lm(formula = crim ~ poly(zn, 3), data = Boston)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.821 -4.614 -1.294  0.473 84.130
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.6135     0.3722   9.709  < 2e-16 ***
## poly(zn, 3)1 -38.7498     8.3722  -4.628  4.7e-06 ***
## poly(zn, 3)2  23.9398     8.3722   2.859  0.00442 **
## poly(zn, 3)3 -10.0719     8.3722  -1.203  0.22954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.372 on 502 degrees of freedom
## Multiple R-squared:  0.05824,    Adjusted R-squared:  0.05261
## F-statistic: 10.35 on 3 and 502 DF,  p-value: 1.281e-06
```

The significance goes all the way up to square term then there is no evidence to prove further polynomial term to be significant for zn term