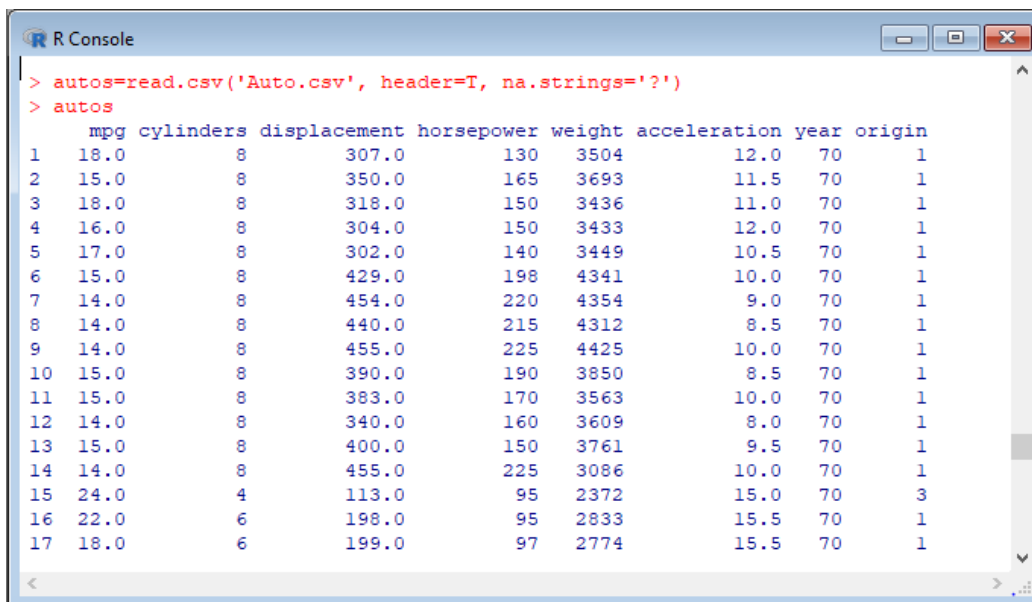


Звіт  
до індивідуального завдання №2  
з предмету Моделі статистичного навчання

Роботу виконала:  
**Мерцало Ірина Ігорівна,**  
студентка групи ПМІМ-11

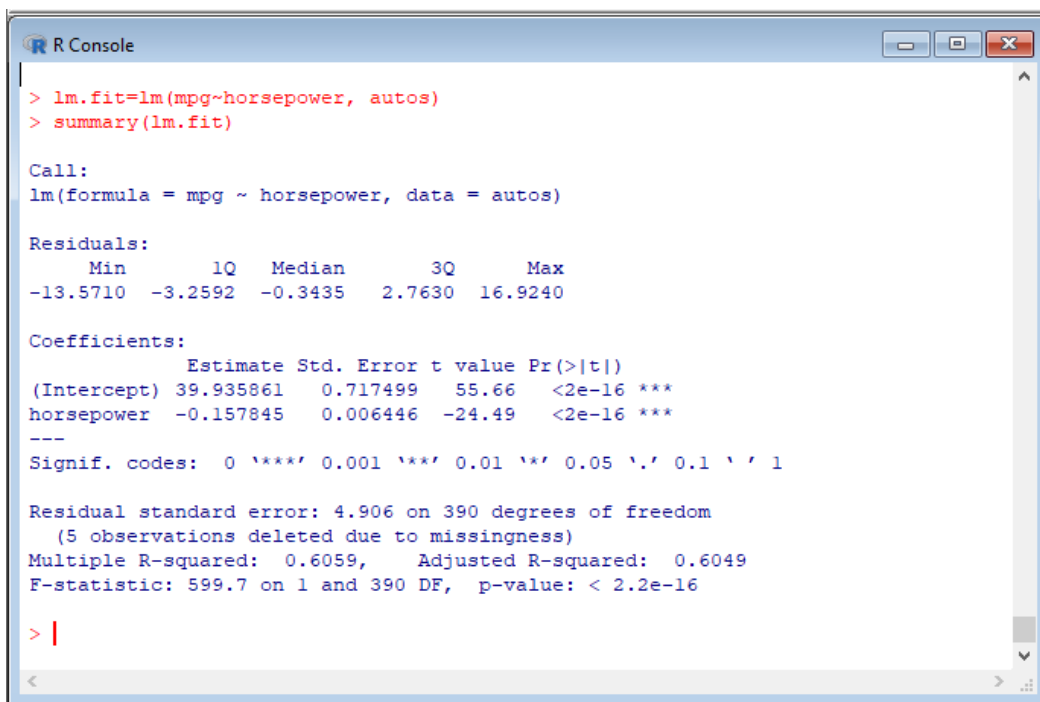
## Завдання 1. Проста лінійна регресія на основі даних.

1.1 Для опрацювання даних зчитала файл 'Auto.csv' функцією `read.csv()` та викликала їх:



```
R Console
> autos=read.csv('Auto.csv', header=T, na.strings='?')
> autos
  mpg cylinders displacement horsepower weight acceleration year origin
1  18.0         8      307.0         130   3504          12.0    70     1
2  15.0         8      350.0         165   3693          11.5    70     1
3  18.0         8      318.0         150   3436          11.0    70     1
4  16.0         8      304.0         150   3433          12.0    70     1
5  17.0         8      302.0         140   3449          10.5    70     1
6  15.0         8      429.0         198   4341          10.0    70     1
7  14.0         8      454.0         220   4354           9.0    70     1
8  14.0         8      440.0         215   4312           8.5    70     1
9  14.0         8      455.0         225   4425          10.0    70     1
10 15.0         8      390.0         190   3850           8.5    70     1
11 15.0         8      383.0         170   3563          10.0    70     1
12 14.0         8      340.0         160   3609           8.0    70     1
13 15.0         8      400.0         150   3761           9.5    70     1
14 14.0         8      455.0         225   3086          10.0    70     1
15 24.0         4      113.0          95   2372          15.0    70     3
16 22.0         6      198.0          95   2833          15.5    70     1
17 18.0         6      199.0          97   2774          15.5    70     1
```

Побудувала функцією `lm()` просту лінійну регресію з залежною змінною `mpg` і незалежною – `horsepower`. Для більш детального опису використала `summary()`.



```
R Console
> lm.fit=lm(mpg~horsepower, autos)
> summary(lm.fit)

Call:
lm(formula = mpg ~ horsepower, data = autos)

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
(5 observations deleted due to missingness)
Multiple R-squared:  0.6059,    Adjusted R-squared:  0.6049
F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16

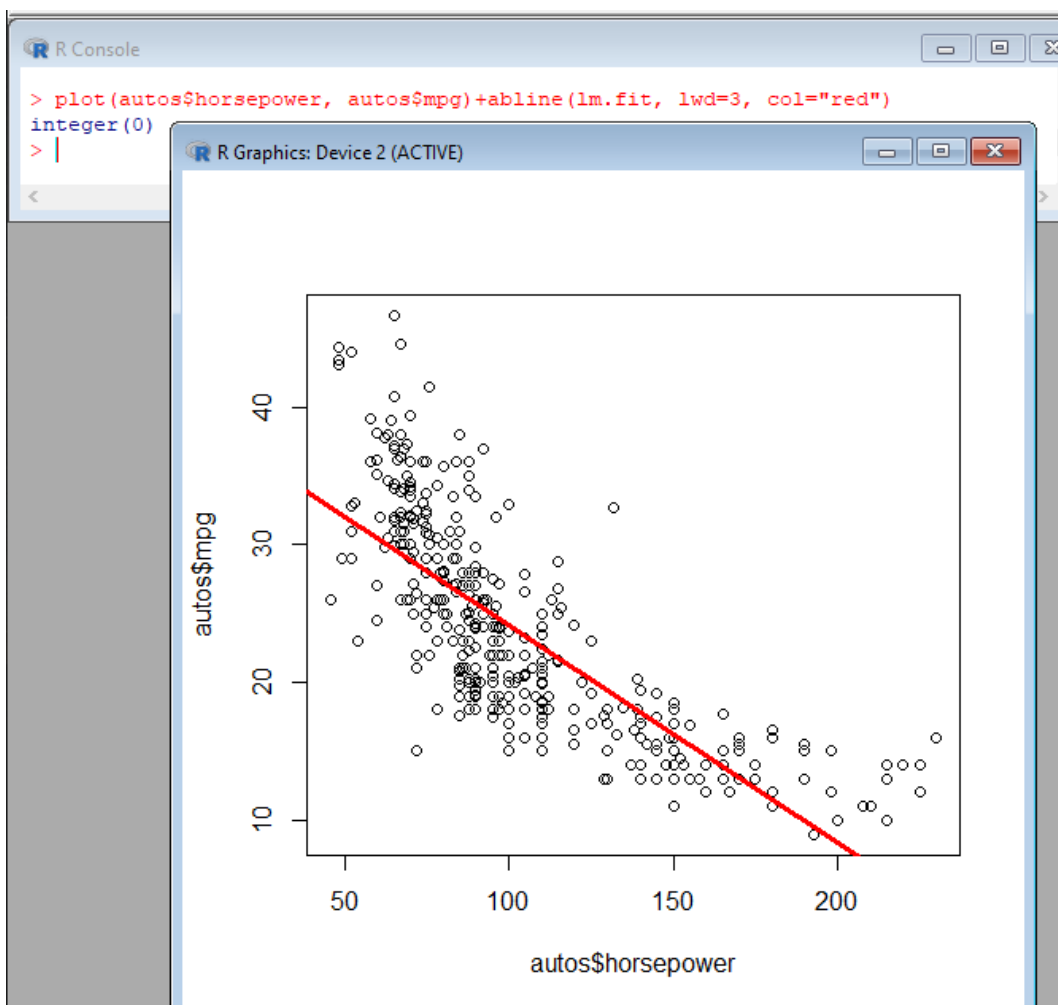
> |
```

По низькому значенню  $p$  можна побачити, що існує взаємозв'язок між цими двома змінними (альтернативна гіпотеза). Він сильний, це можна побачити по значенню  $R$ , вони пов'язані на 60%. Взаємозв'язок негативний, бо при збільшенні значення `horsepower`, зменшується значення `mpg`.

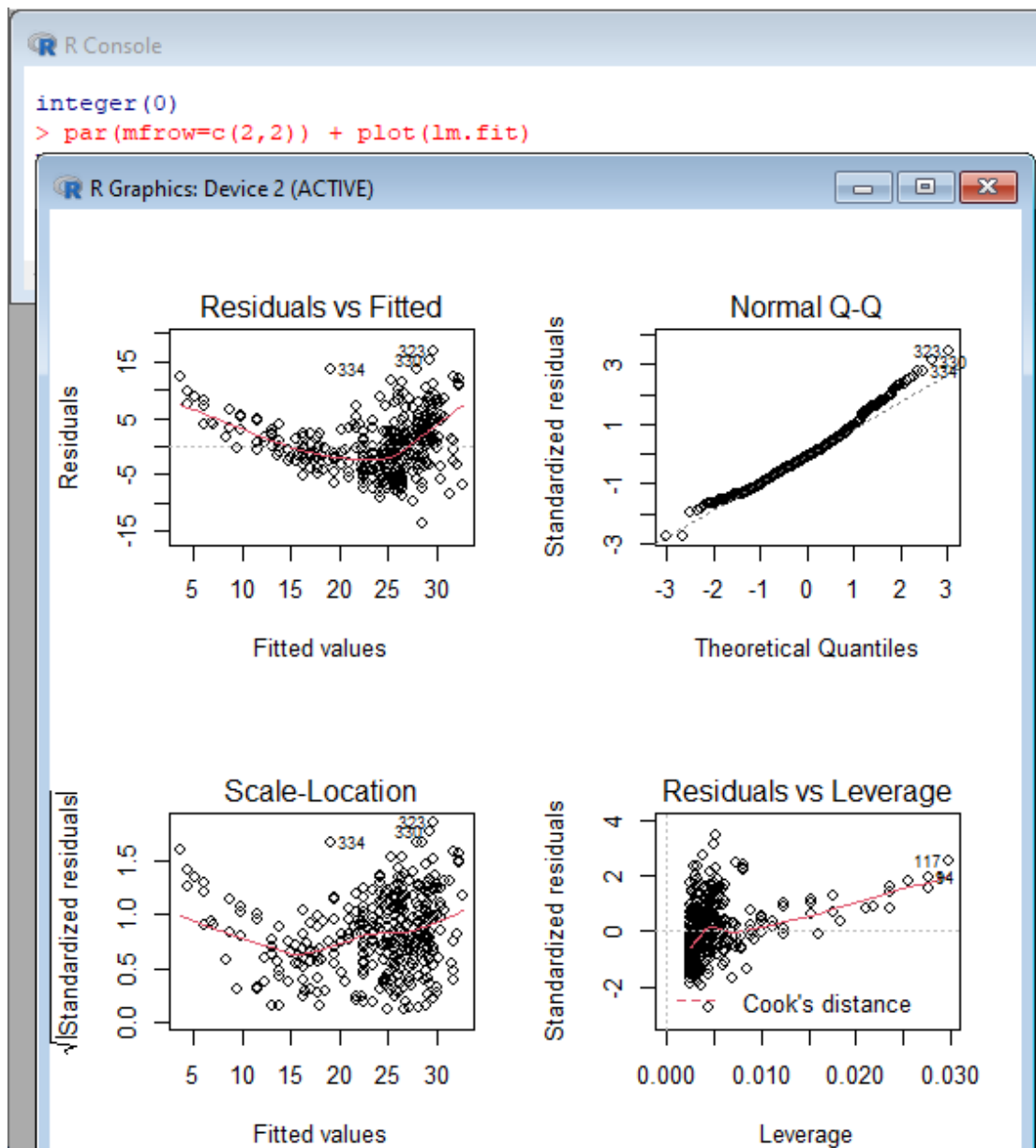
```
R Console
> predict(lm.fit, data.frame(horsepower=c(98)),interval="confidence")
      fit      lwr      upr
1 24.46708 23.97308 24.96108
> predict(lm.fit, data.frame(horsepower=c(98)),interval="prediction")
      fit      lwr      upr
1 24.46708 14.8094 34.12476
> |
```

Прогнозне значення залежної змінної при значенні предиктора 98 буде 24.46708, пов'язані 95% інтервали довіри та прогнозування відобразила за допомогою predict().

1.2 Використовуючи функцію plot(), зобразила графічно предиктор та залежну змінну, а за допомогою abline() - оцінену пряму:



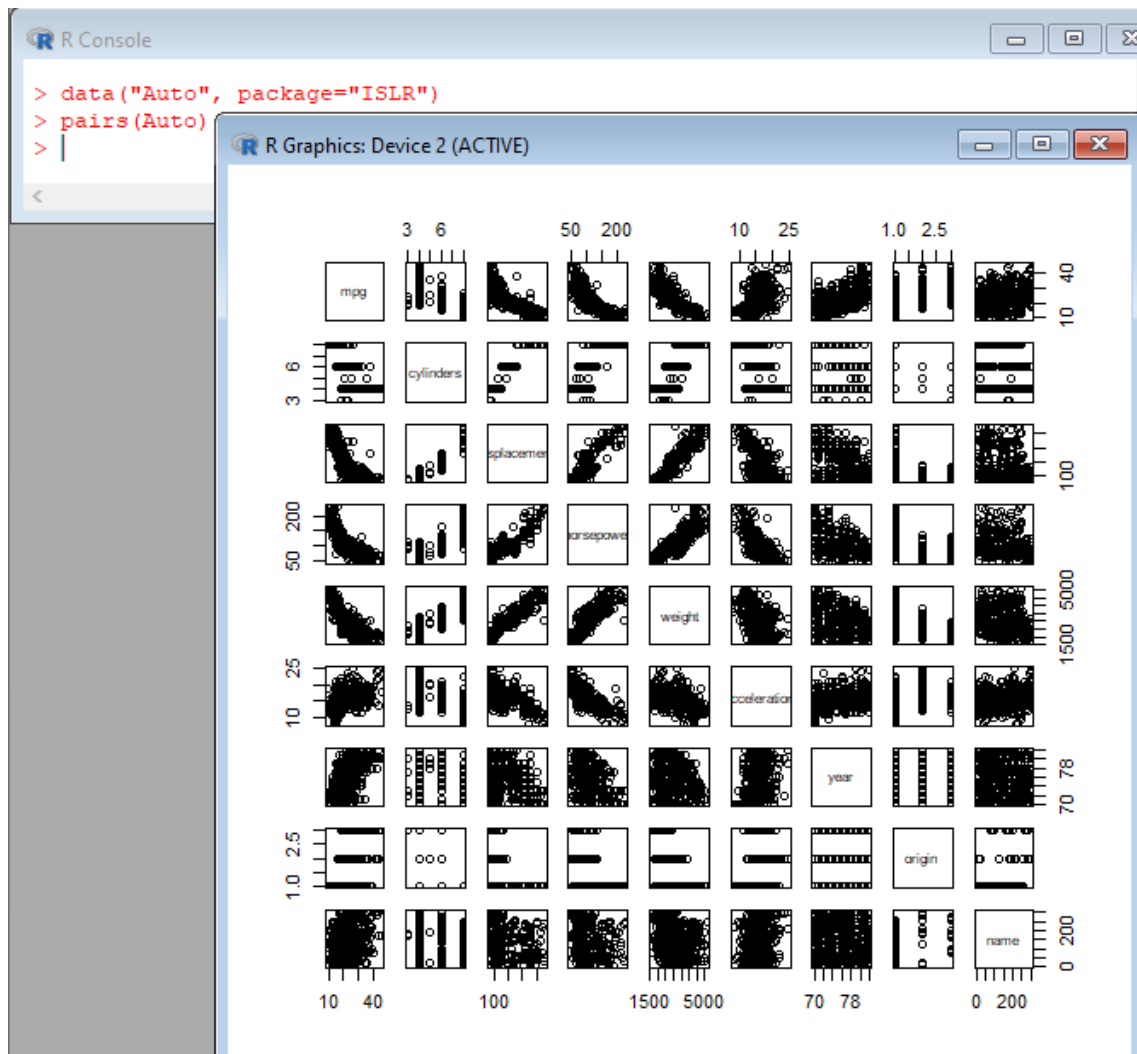
1.3 Використовуючи функцію plot(), зобразила діагностичні графіки:



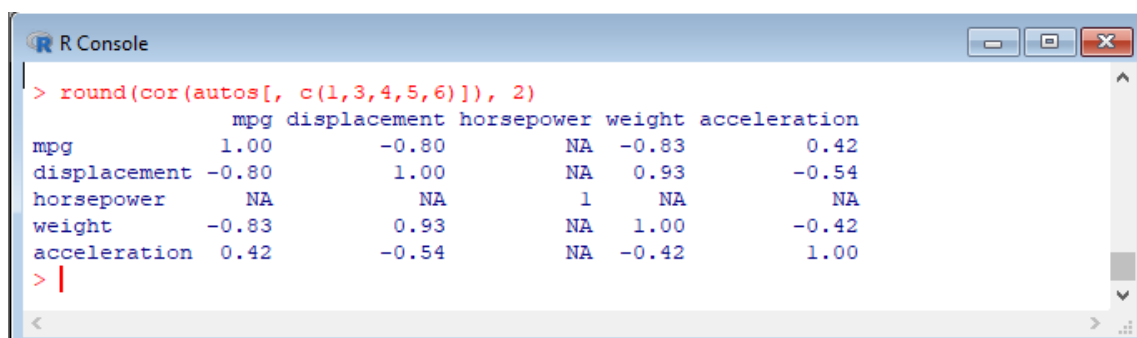
По них можна побачити, що проблема полягає у відсутності лінійної залежності між показниками.

**Завдання 2.** Множинна лінійна регресія на основі даних Auto.

2.1 Побудувала діаграми розкиду усіх змінних:



2.2 Використовуючи функцію `cor()`, обчислила матрицю кореляцій між змінними:



2.3 Використовуючи функцію `lm()` побудувала множинну регресію для залежної змінної `mpg` і всіх решту змінних окрім `names` як предикторів:

```
R Console
> lm.fit = lm(mpg~.-name, autos)
> summary(lm.fit)

Call:
lm(formula = mpg ~ . - name, data = autos)

Residuals:
    Min       1Q   Median       3Q      Max
-9.5903 -2.1565 -0.1169  1.8690 13.0604

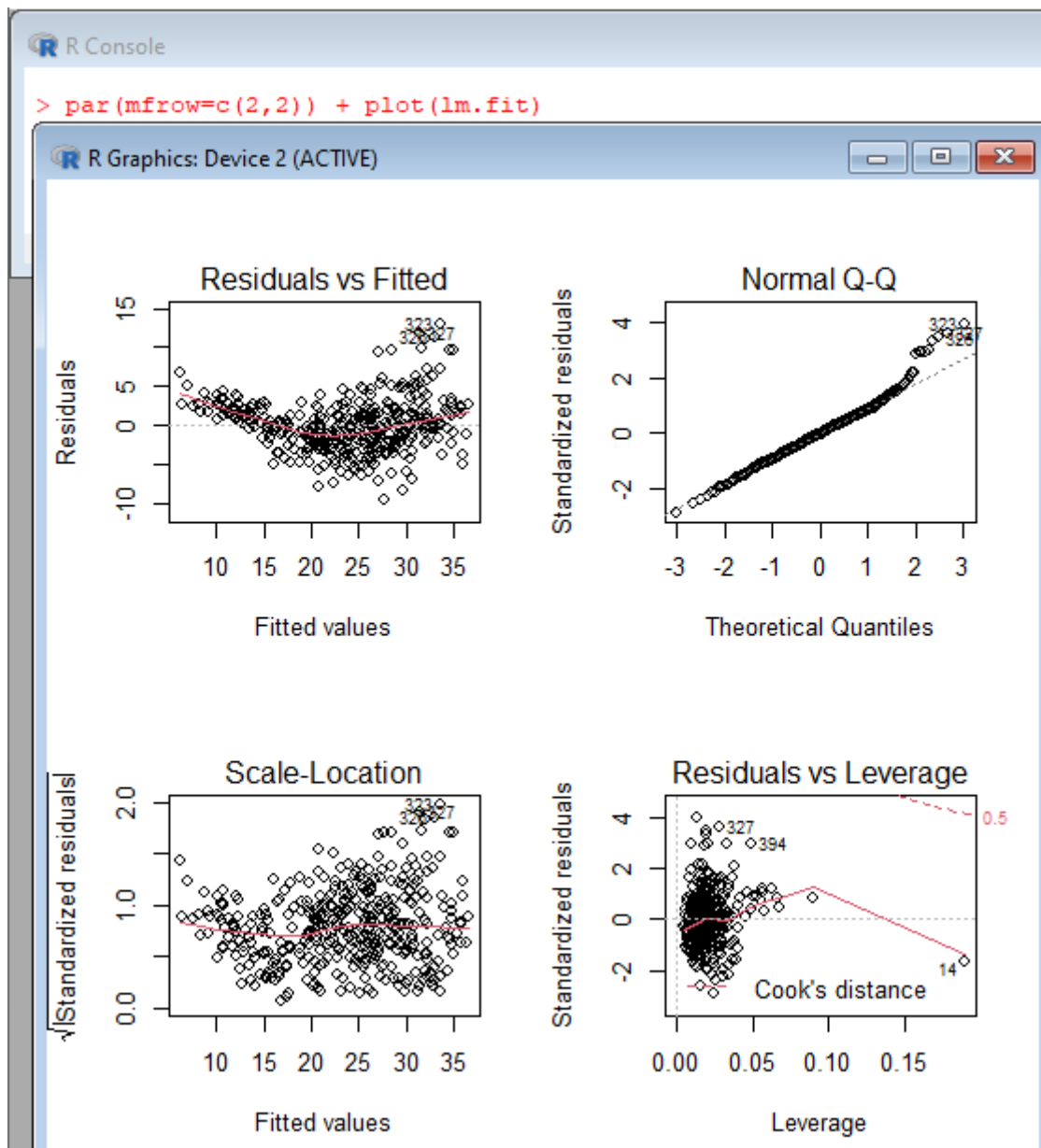
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.218435   4.644294  -3.707  0.00024 ***
cylinders    -0.493376   0.323282  -1.526  0.12780
displacement  0.019896   0.007515   2.647  0.00844 **
horsepower   -0.016951   0.013787  -1.230  0.21963
weight       -0.006474   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
(5 observations deleted due to missingness)
Multiple R-squared:  0.8215,    Adjusted R-squared:  0.8182
F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16

> |
```

По низькому значенню  $p$  можна побачити, що існує взаємозв'язок між залежною змінною та предиктором (альтернативна гіпотеза). Displacement, weight, year, origin – мають статистично значущий зв'язок з mpg. Коефіцієнт для year означає щорічне зростання mpg на 75%.

2.4. Використовуючи функцію plot () створила діагностичні графіки:



На першому графіку можна побачити, що проблема полягає у відсутності лінійної залежності між показниками (крива лінія). Великий викид можна спостерігати на четвертому графіку (327, 394,...). Спостереження з високим левереджем є на четвертому графіку (14).

2.5. Використовуючи символи \* та:, включила в модель лінійної регресії ефекти взаємодії.

```
R Console
> lm.fit = lm(mpg~acceleration*origin + horsepower*weight + displacement*horsepower, autos)
> summary(lm.fit)

Call:
lm(formula = mpg ~ acceleration * origin + horsepower * weight +
    displacement * horsepower, data = autos)

Residuals:
    Min       1Q   Median       3Q      Max
-11.0227  -2.3255  -0.3178   1.7502  16.6491

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.509e+01  4.503e+00  14.452  < 2e-16 ***
acceleration  -7.055e-01  2.333e-01  -3.024  0.002664 **
origin        -3.176e+00  2.064e+00  -1.539  0.124653
horsepower    -2.383e-01  3.147e-02  -7.573  2.75e-13 ***
weight        -2.573e-03  1.926e-03  -1.336  0.182339
displacement  -6.049e-02  1.661e-02  -3.642  0.000308 ***
acceleration:origin  2.450e-01  1.260e-01   1.944  0.052597 .
horsepower:weight   9.008e-06  1.290e-05   0.698  0.485315
horsepower:displacement 3.964e-04  1.155e-04   3.432  0.000665 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.804 on 383 degrees of freedom
(5 observations deleted due to missingness)
Multiple R-squared:  0.7674,    Adjusted R-squared:  0.7625
F-statistic: 157.9 on 8 and 383 DF,  p-value: < 2.2e-16

> |
```

По значенню р можна побачити, що статистично значимими змінними є acceleration:horsepower.

2.6. За допомогою функції anova() спробувала порівняти кілька різних перетворень змінних, таких як  $\log(X)$ ,  $X^2$ ,  $\sqrt{X}$  із нашим попереднім:



```
R Console

> lm.fit_new = lm(mpg~horsepower, autos)
> lm.fit_log = lm(mpg~horsepower + I(log(horsepower)), autos)
> anova(lm.fit_new, lm.fit_log)
Analysis of Variance Table

Model 1: mpg ~ horsepower
Model 2: mpg ~ horsepower + I(log(horsepower))
  Res.Df  RSS Df Sum of Sq    F    Pr(>F)
  1     390 9385.9
  2     389 7581.2  1    1804.7 92.601 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> lm.fit_to_2 = lm(mpg~horsepower + I(horsepower^2), autos)
> anova(lm.fit_new, lm.fit_to_2)
Analysis of Variance Table

Model 1: mpg ~ horsepower
Model 2: mpg ~ horsepower + I(horsepower^2)
  Res.Df  RSS Df Sum of Sq    F    Pr(>F)
  1     390 9385.9
  2     389 7442.0  1    1943.9 101.61 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> lm.fit_sqrt = lm(mpg~horsepower + I(sqrt(horsepower)), autos)
> anova(lm.fit_new, lm.fit_sqrt)
Analysis of Variance Table

Model 1: mpg ~ horsepower
Model 2: mpg ~ horsepower + I(sqrt(horsepower))
  Res.Df  RSS Df Sum of Sq    F    Pr(>F)
  1     390 9385.9
  2     389 7502.2  1    1883.7 97.672 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Моделі не описують дані однаково добре, тому приймаємо альтернативну гіпотезу.

**Завдання 3.** Розглянемо дані Carseats.

3.1. Побудувала модель множинної регресії для прогнозування Sales використовуючи Price, Urban, та US:

```
R Console

> library(ISLR)
> lm.fit = lm(Sales~Price+Urban+US, 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

> |
```

### 3.2. Інтерпретація кожного коефіцієнта в моделі

По низькому значенню p можна побачити, що існує взаємозв'язок між змінними Price та Sales і він негативний, бо при збільшенні значення Price, зменшується значення Sales.

По високому значенню p можна побачити, що взаємозв'язок з Urban не є статистично значущим.

Взаємозв'язок з US позитивний, при розташуванні магазину в US, кількість продажів збільшується на 1200.

### 3.3 Модель у формі рівняння

$$\text{Sales} = 13.04 + -0.05 * \text{Price} + -0.02 * \text{Urban} + 1.20 * \text{US}$$

3.4 Зважаючи на низькі значення p, нульову гіпотезу можна відхилити для предикторів Price та US.

3.5 Модель з меншою кількістю незалежних змінних, яка використовує лише ті предиктори, для яких зв'язок з залежною змінною є значимим:

```
R Console

> lm.fit_new_2=lm(Sales~Price+US, Carseats)
> summary(lm.fit_new_2)

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

> |
```

3.6 Скориставшись функцією `anova()`, можемо бачити, що друга модель краща.

```
R Console

> anova(lm.fit, lm.fit_new_2)
Analysis of Variance Table

Model 1: Sales ~ Price + Urban + US
Model 2: Sales ~ Price + US
  Res.Df    RSS Df Sum of Sq    F Pr(>F)
1     396 2420.8
2     397 2420.9 -1   -0.03979 0.0065 0.9357

> |
```

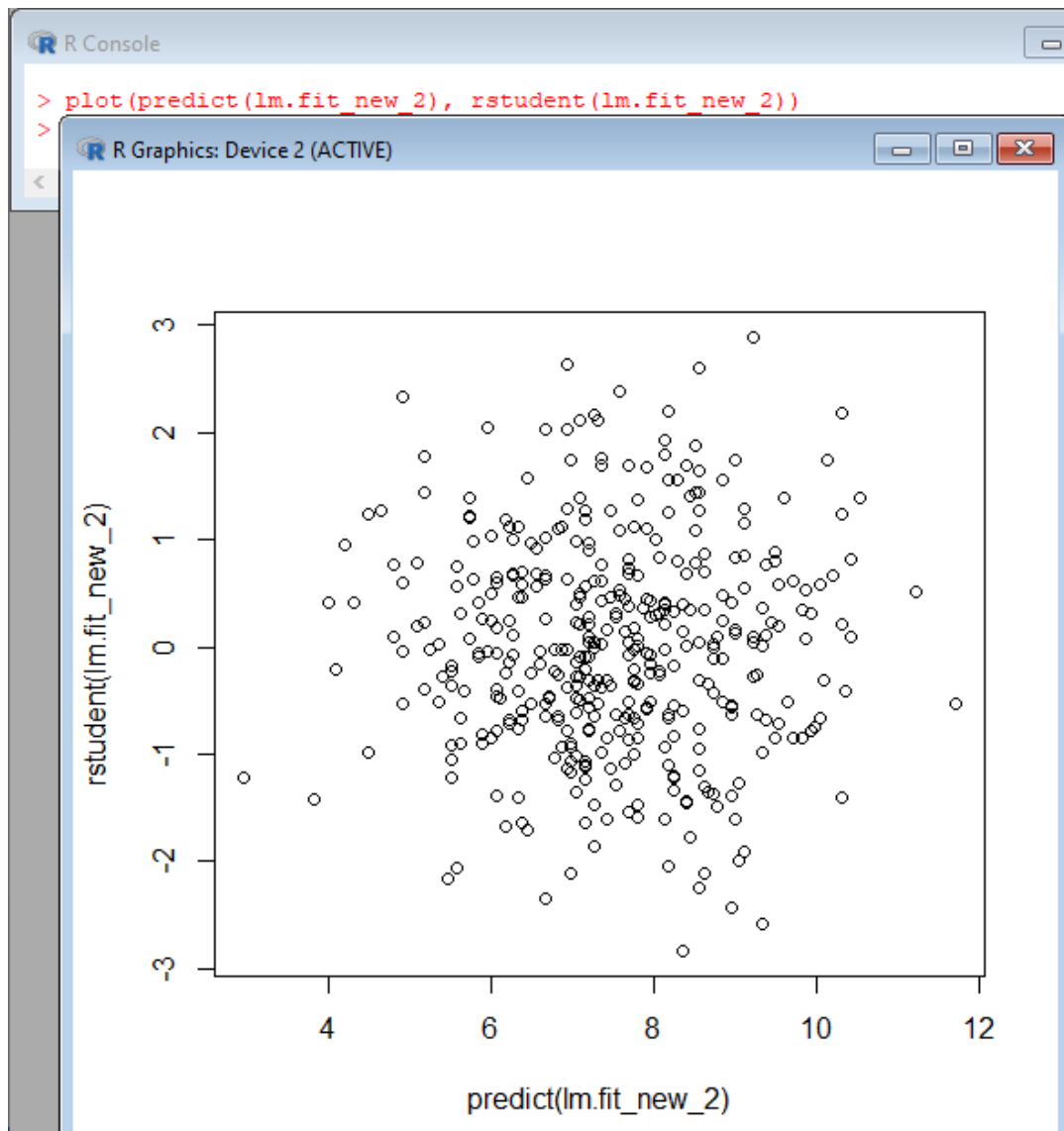
3.7 Використовуючи модель з 3.5, побудувала 95% інтервали довіри для коефіцієнтів:

```
R Console

> confint(lm.fit_new_2)
                2.5 %      97.5 %
(Intercept) 11.79032020 14.27126531
Price       -0.06475984 -0.04419543
USYes        0.69151957  1.70776632

> |
```

3.8 За допомогою функції `plot()` визначила, що потенційних викидів для моделі з 3.5 не видно, бо всі обмежені від -3 до 3. Спостереження з високим рівнем левериджу існують, бо є точки які значно перевищують  $(p+1)/n$  тобто значення більші за 0.0076.



**Завдання 4.** Згенерувала предиктор  $x$  та залежну змінну  $y$  :

The figure shows an 'R Console' window with the following code:

```
> set.seed(1)  
> x=rnorm(100)  
> y=2*x+rnorm(100)  
> |
```

4.1 Побудувала просту лінійну регресію  $y$  на  $x$  без  $\beta_0$  :

```
R Console

> lm.fit4=lm(y~x+0)
> summary(lm.fit4)

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

> |
```

По низькому значенню  $p$  можна побачити, що приймаємо альтернативну гіпотезу.

4.2 Побудувала просту лінійну регресію  $x$  на  $y$  без  $\beta_0$ :

```
R Console

> lm.fit4_1=lm(x~y+0)
> summary(lm.fit4_1)

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

> |
```

Так само, як в попередньому випадку, по низькому значенню  $p$  можна побачити, що приймаємо альтернативну гіпотезу.

4.3 За допомогою функції `cor()`, можна побачити кореляцію між результатами, отриманими в 4.1 та 4.2:

```
R Console
> cor(x,y)
[1] 0.8822902
> |
```

#### 4.4 Чисельна перевірка:

```
R Console
> (sqrt(length(x)-1)*sum(x*y))/(sqrt(sum(x*x)*sum(y*y)-(sum(x*y))^2))
[1] 18.72593
> |
```

Можна побачити, що значення таке саме, як в 4.1 та 4.2

4.5 З допомогою попереднього пункту можемо поабчити, що міняючи місцями аргументи в добутках значення не змюється, отже  $t$ -статистика для регресії  $y$  на  $x$  є те саме, що  $t$ -статистика для регресії  $x$  на  $y$ .

4.6 Показала, що коли будується регресія з коефіцієнтом  $\beta_0$ , то  $t$ -статистика для  $H_0: \beta_1 = 0$  однакова для регресії  $y$  на  $x$ , та для регресії  $x$  на  $y$ .

```
R Console
> print(coefficients(summary(lm.fit4)))
Estimate Std. Error t value Pr(>|t|)
x 1.993876 0.1064767 18.72593 2.642197e-34
> print(coefficients(summary(lm.fit4_1)))
Estimate Std. Error t value Pr(>|t|)
y 0.3911145 0.02088625 18.72593 2.642197e-34
> |
```

**Завдання 5.** Знову розгляну просту лінійну регресію без коефіцієнта  $\beta_0$ .

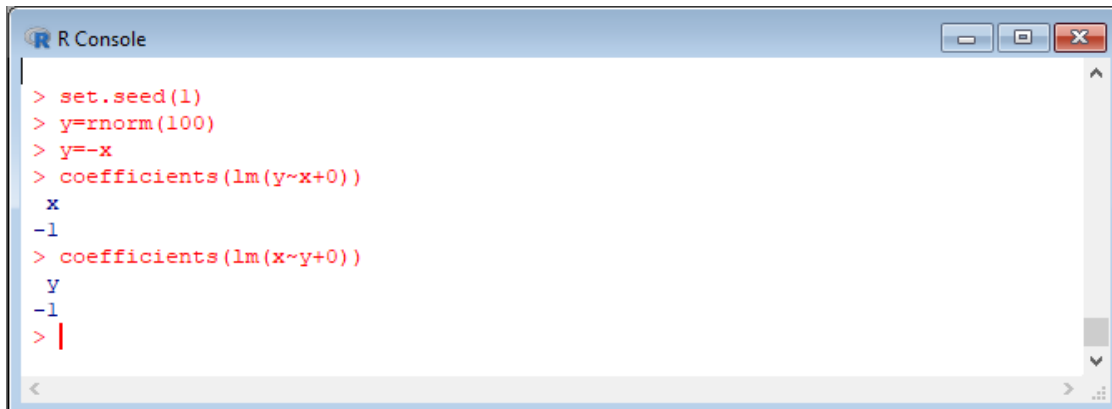
5.1 Оцінка коефіцієнта регресії  $X$  на  $Y$  дорівнює оцінці коефіцієнта регресії  $Y$  на  $X$ , коли  $\sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i^2$ .

5.2. Побудувала приклад у R з  $n = 100$  спостережень, в якому оцінка коефіцієнта для регресії  $X$  на  $Y$  не дорівнює оцінці коефіцієнта регресії  $Y$  на  $X$ :

```
R Console
> set.seed(1)
> y=2*x+rnorm(100)
> coefficients(lm(y~x+0))
x
3
> coefficients(lm(x~y+0))
y
0.3333333
> |
```

Можна побачити, що коефіцієнти різні.

5.3. Побудувала приклад у R з  $n = 100$  спостережень, в якому оцінка коефіцієнта для регресії  $X$  на  $Y$  дорівнює оцінці коефіцієнта регресії  $Y$  на  $X$ :



```
> set.seed(1)
> y=rnorm(100)
> y=-x
> coefficients(lm(y~x+0))
x
-1
> coefficients(lm(x~y+0))
y
-1
> |
```

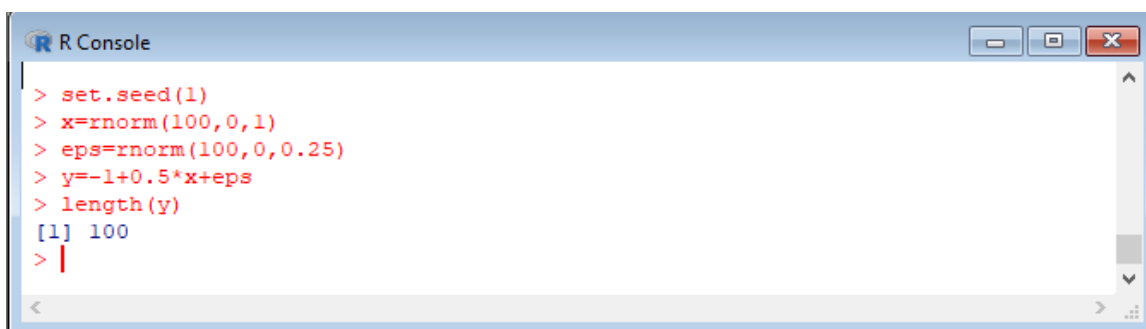
**Завдання 6.** Згенерувати набір даних та оцінити кілька простих лінійних моделей. Використайте `set.seed(1)` перед початком частини 6.1 для забезпечення однакових результатів.

6.1 - 6.3 За допомогою функції `rnorm()` створила вектор  $x$ , що містить 100 спостережень, отриманих з розподілу  $N(0, 1)$ .

За допомогою функції `rnorm()` створила вектор  $eps$ , що містить 100 спостереження, отриманих з розподілу  $N(0, 0,25)$ .

За допомогою  $x$  та  $eps$  згенерувала вектор  $y$  відповідно до моделі

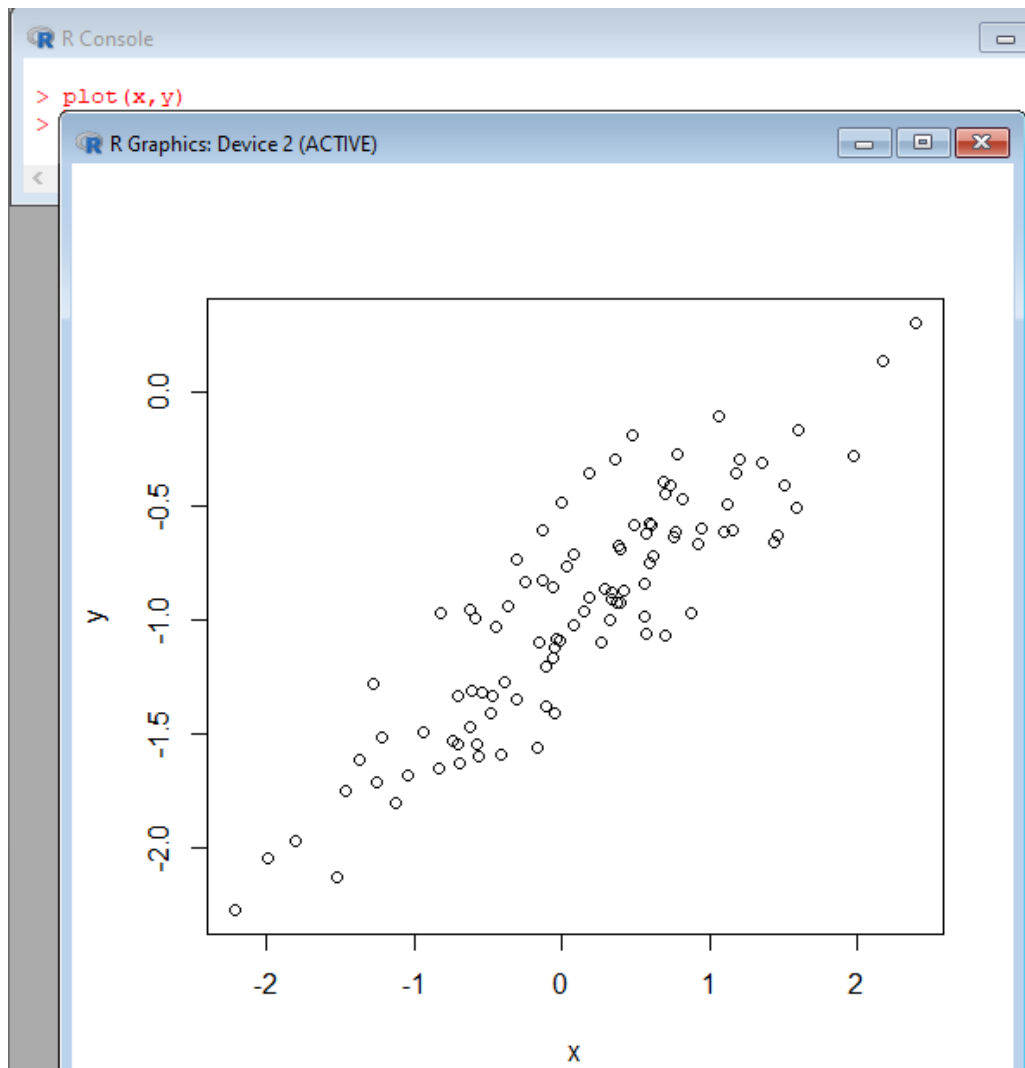
$$Y = -1 + 0,5X + \varepsilon$$



```
> set.seed(1)
> x=rnorm(100,0,1)
> eps=rnorm(100,0,0.25)
> y=-1+0.5*x+eps
> length(y)
[1] 100
> |
```

Довжина вектора  $y$  така сама, як  $x$  та  $eps$ . Значення  $\beta_0$  і  $\beta_1$  у цій лінійній моделі -1 та 0.5 відповідно.

6.4 Побудувала діаграму розсіювання, що відображає взаємозв'язок між  $x$  та  $y$ :



В результаті можна побачити лінійну залежність між x та y.

6.5. Побудуйте лінійну модель для прогнозування y на основі x:

```

R Console

> lm.fit4_2=lm(y~x)
> summary(lm.fit4_2)

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

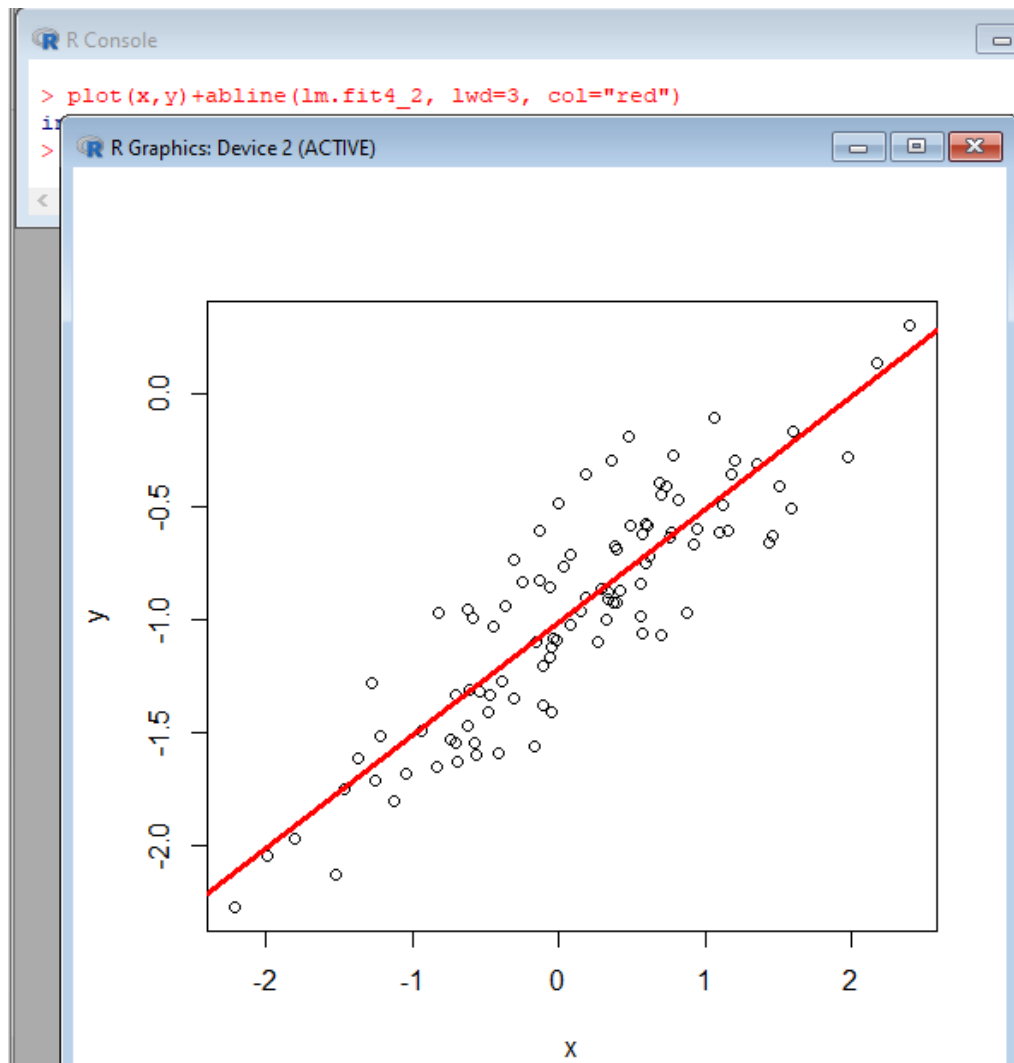
>

```

Оцінки параметрів  $\beta_0$  та  $\beta_1$  досить точні.

6.6 Відобразила оцінену лінію на діаграмі розсіювання, отриманій в 6.4:





6.7. Побудувала модель поліноміальної регресії, яка передбачає  $y$  на основі  $x$  і  $x^2$ :

```
R Console
> lm.fit4_2.poly=lm(y~poly(x,2))
> summary(lm.fit4_2.poly)

Call:
lm(formula = y ~ poly(x, 2))

Residuals:
    Min       1Q   Median       3Q      Max
-0.4913 -0.1563 -0.0322  0.1451  0.5675

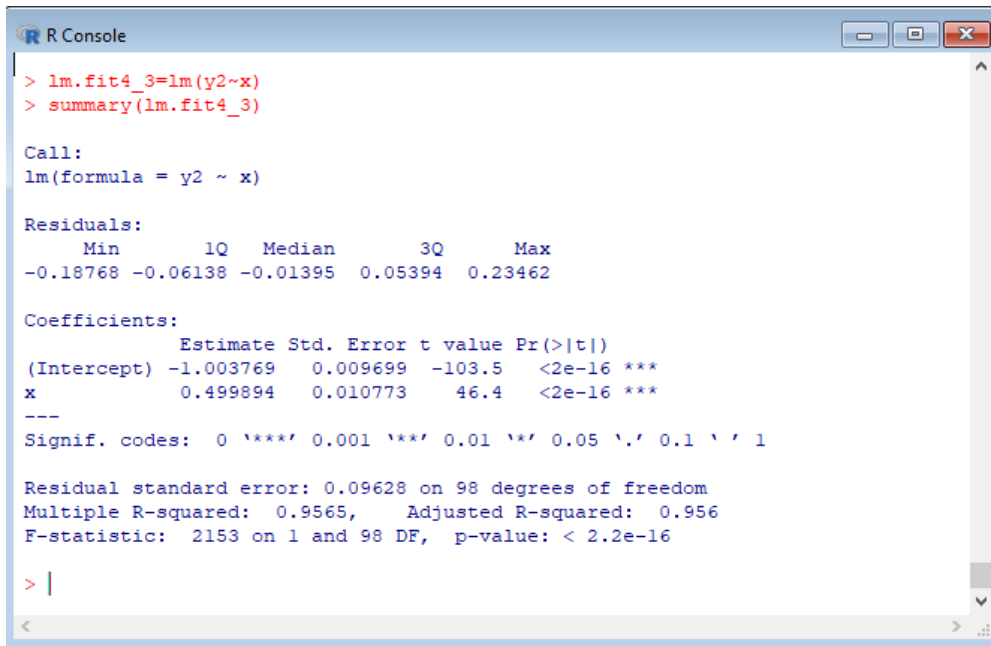
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.95501    0.02395  -39.874  <2e-16 ***
poly(x, 2)1   4.46612    0.23951   18.647  <2e-16 ***
poly(x, 2)2  -0.33602    0.23951   -1.403    0.164
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

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

> |
```

По високому значенню  $p$  можна прийняти нульову гіпотезу, отже квадратичний доданок не покращує модель.

6.8 Повторила 6.1. – 6.6. після модифікації процесу генерації даних у таким чином, щоб було менше шуму в даних. Для цього зменшила дисперсію нормального розподілу до  $N(0, 0.1)$ , що використовується для генерування залишків.



```
R Console
> lm.fit4_3=lm(y2~x)
> summary(lm.fit4_3)

Call:
lm(formula = y2 ~ x)

Residuals:
    Min       1Q   Median       3Q      Max
-0.18768 -0.06138 -0.01395  0.05394  0.23462

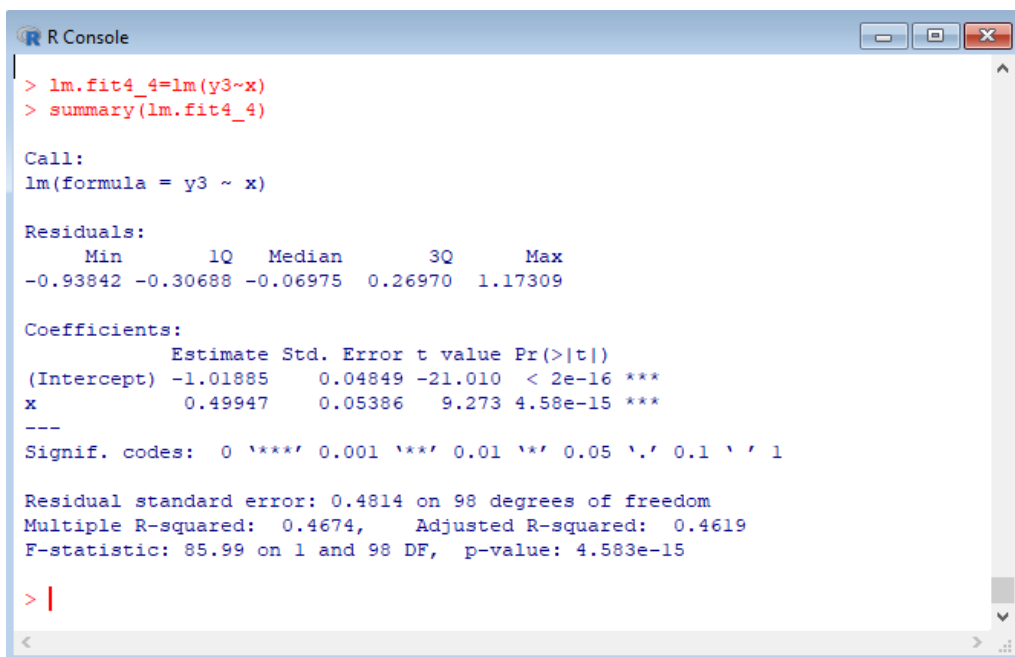
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.003769   0.009699  -103.5  <2e-16 ***
x             0.499894   0.010773   46.4  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.09628 on 98 degrees of freedom
Multiple R-squared:  0.9565,    Adjusted R-squared:  0.956
F-statistic: 2153 on 1 and 98 DF,  p-value: < 2.2e-16

> |
```

В результаті можна побачити, що значення R близькі до 1 (0,9565 та 0,956), отже з 98% відповідністю близькі до реальної регресії.

6.9 Повторила 6.1. – 6.6. після модифікації процесу генерації даних у таким чином, щоб було більше шуму в даних. Для цього збільшила дисперсію нормального розподілу до  $N(0, 0.5)$ , що використовується для генерування залишків.



```
R Console
> lm.fit4_4=lm(y3~x)
> summary(lm.fit4_4)

Call:
lm(formula = y3 ~ x)

Residuals:
    Min       1Q   Median       3Q      Max
-0.93842 -0.30688 -0.06975  0.26970  1.17309

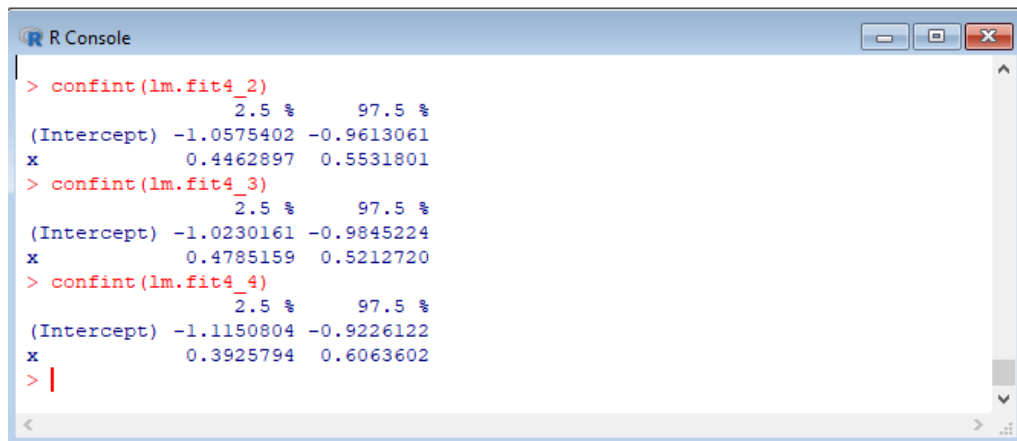
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.01885   0.04849  -21.010  < 2e-16 ***
x             0.49947   0.05386   9.273  4.58e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4814 on 98 degrees of freedom
Multiple R-squared:  0.4674,    Adjusted R-squared:  0.4619
F-statistic: 85.99 on 1 and 98 DF,  p-value: 4.583e-15

> |
```

В результаті можна побачити, що похибка збільшилась.

6.10 За допомогою функції `confint()` знайшла довірчі інтервали для реальної моделі, моделі з меншим шумом та моделі з більшим шумом.



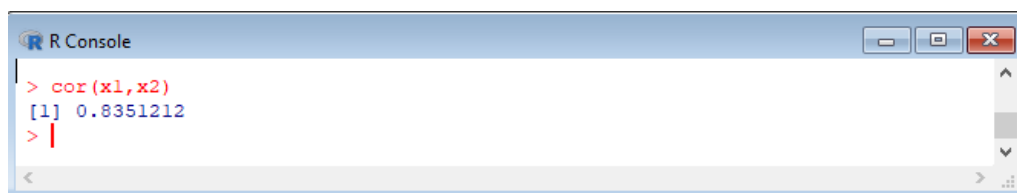
```
> confint(lm.fit4_2)
                2.5 %      97.5 %
(Intercept) -1.0575402 -0.9613061
x             0.4462897  0.5531801
> confint(lm.fit4_3)
                2.5 %      97.5 %
(Intercept) -1.0230161 -0.9845224
x             0.4785159  0.5212720
> confint(lm.fit4_4)
                2.5 %      97.5 %
(Intercept) -1.1150804 -0.9226122
x             0.3925794  0.6063602
> |
```

В результаті бачимо, що зі збільшенням шуму довірчі інтервали збільшуються.

### Завдання 7.

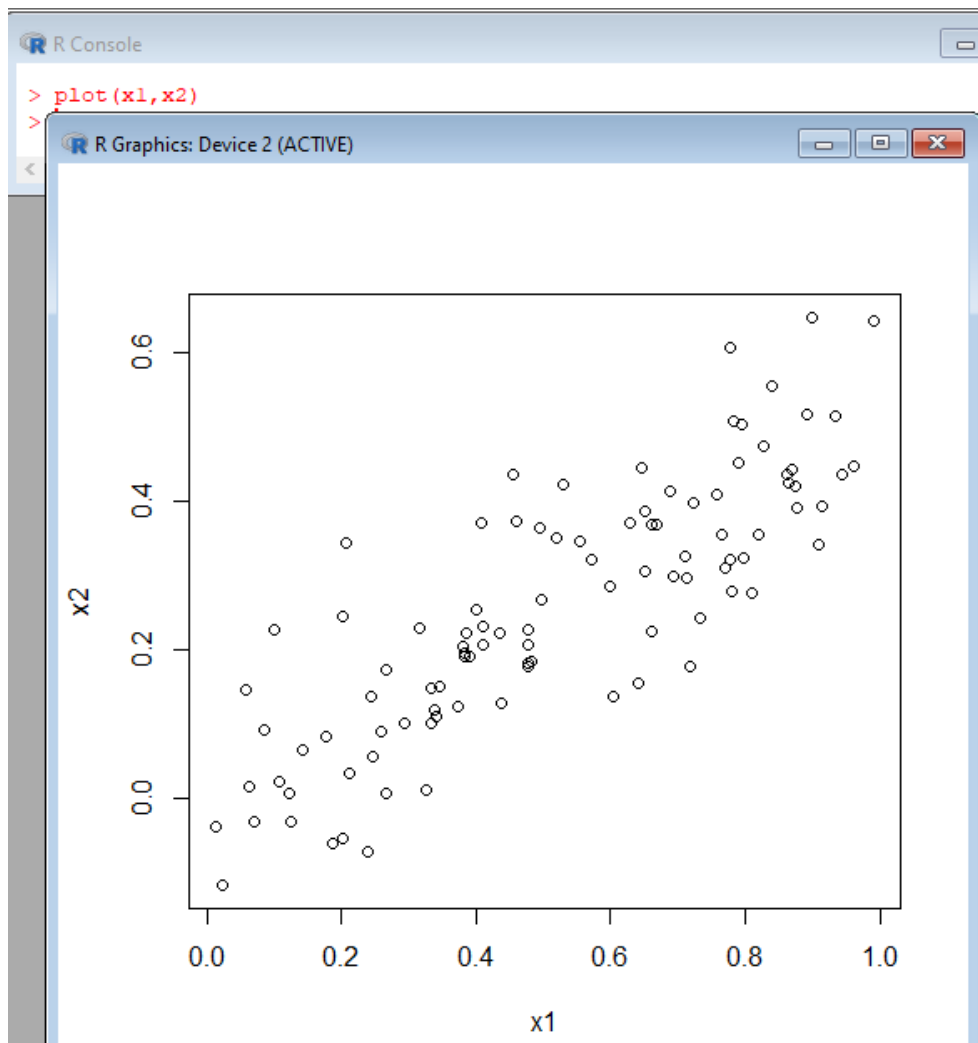
7.1 Створила лінійну модель, в якій  $y$  є функція від  $x_1$  і  $x_2$ . Форма лінійної моделі  $y = 2 + 2 \cdot x_1 + 0.3 \cdot x_2 + \text{eps}$ , коефіцієнти регресії 2, 2, 0.3 відповідно.

7.2 За допомогою функції `cor()`, можна побачити кореляцію між  $x_1$  та  $x_2$ :

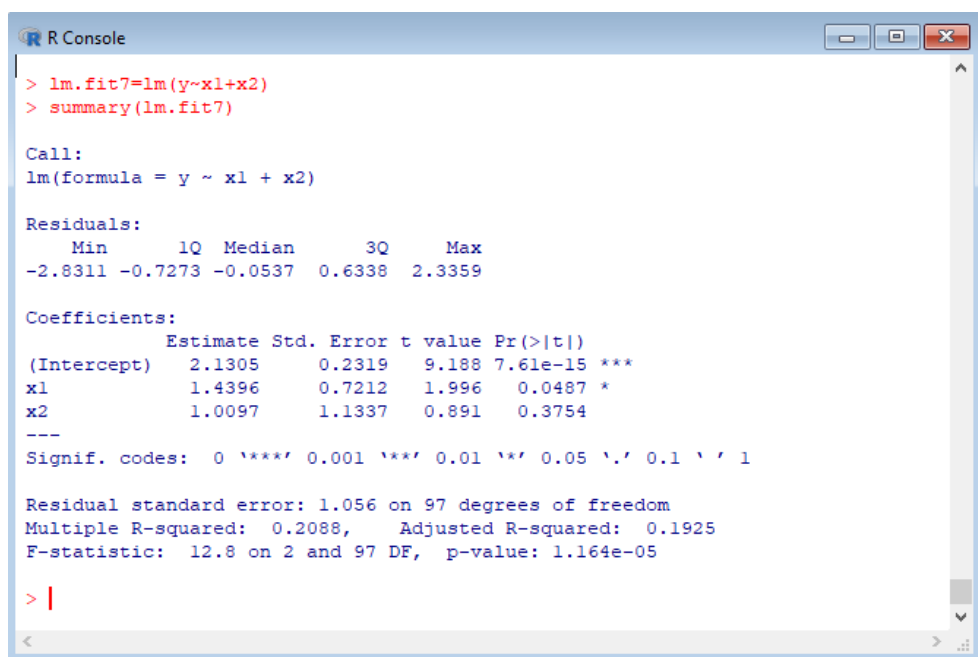


```
> cor(x1, x2)
[1] 0.8351212
> |
```

Побудувала діаграму розсіювання для відображення зв'язку між змінними:



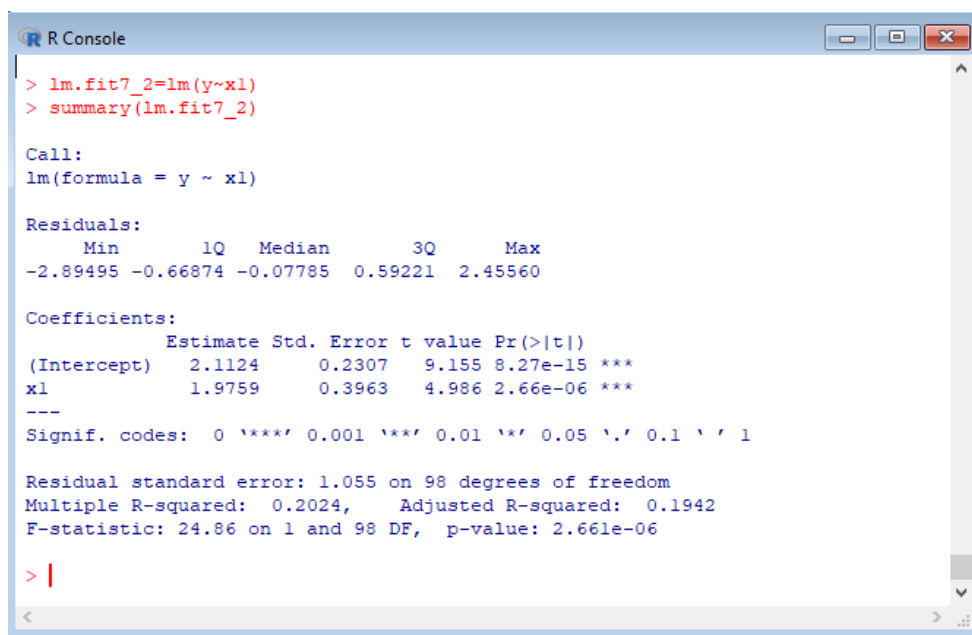
7.3 Використовуючи ці дані, оцінила регресію, щоб передбачити  $y$ , використовуючи  $x_1$  та  $x_2$ :



Параметр  $\beta_0$  близький до реального значення. Параметр  $\beta_1$  також, але можна побачити низьке значення  $p$ , тому приймаємо альтернативну гіпотезу. Параметр

$\beta_2$  найбільш відмінний від реального значення, можна побачити високе значення  $p$ , тому приймаємо нуль-гіпотезу.

#### 7.4 Побудувала регресію $y$ на $x_1$ :



```
R Console
> lm.fit7_2=lm(y~x1)
> summary(lm.fit7_2)

Call:
lm(formula = y ~ x1)

Residuals:
    Min       1Q   Median       3Q      Max
-2.89495 -0.66874 -0.07785  0.59221  2.45560

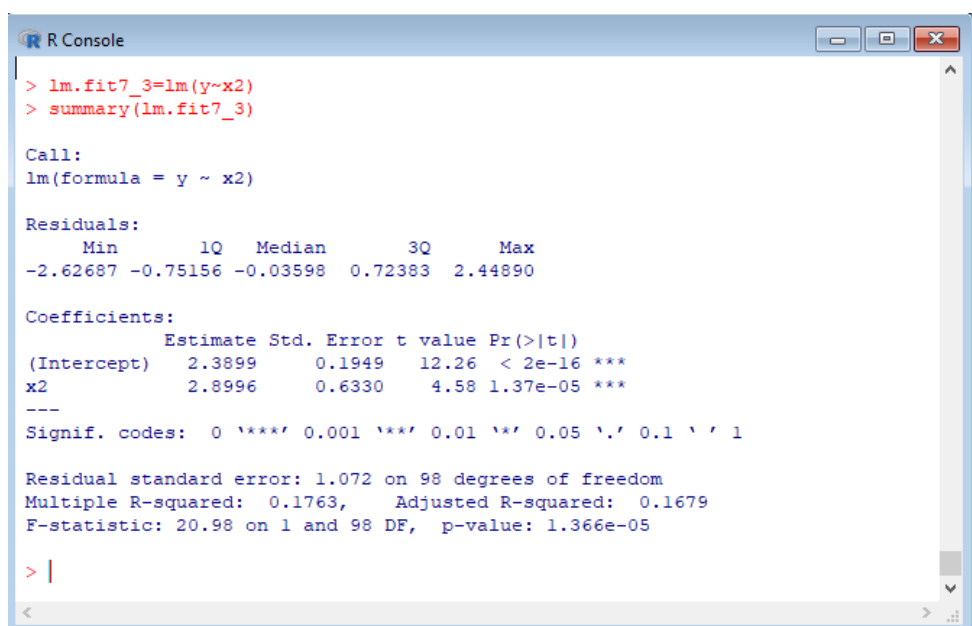
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.1124    0.2307   9.155 8.27e-15 ***
x1             1.9759    0.3963   4.986 2.66e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

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

> |
```

Можна побачити, що значення  $p$  низьке, тому відхиляємо нуль-гіпотезу.

#### 7.5 Побудувала регресію $y$ на $x_2$ :



```
R Console
> lm.fit7_3=lm(y~x2)
> summary(lm.fit7_3)

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

> |
```

Можна побачити, що значення  $p$  низьке, тому відхиляємо нуль-гіпотезу.

#### 7.6 Результати 7.3-7.5 не суперечать одні одним, оскільки $x_1$ та $x_2$ колінеарні.

Але коли ми розглядаємо зв'язок з кожним окремих пердиктором, то тоді з'являється лінійна залежність.

#### 7.7 Переоцінила попередні лінійні моделі, використовуючи нові дані про те, що отримано одне додаткове неправильне спостереження:

```
R Console
> x1=c(x1, 0.1)
> x2=c(x2, 0.8)
> y=c(y, 6)
> lm.fit7_ll=lm(y~x1+x2)
> summary(lm.fit7_ll)

Call:
lm(formula = y ~ x1 + x2)

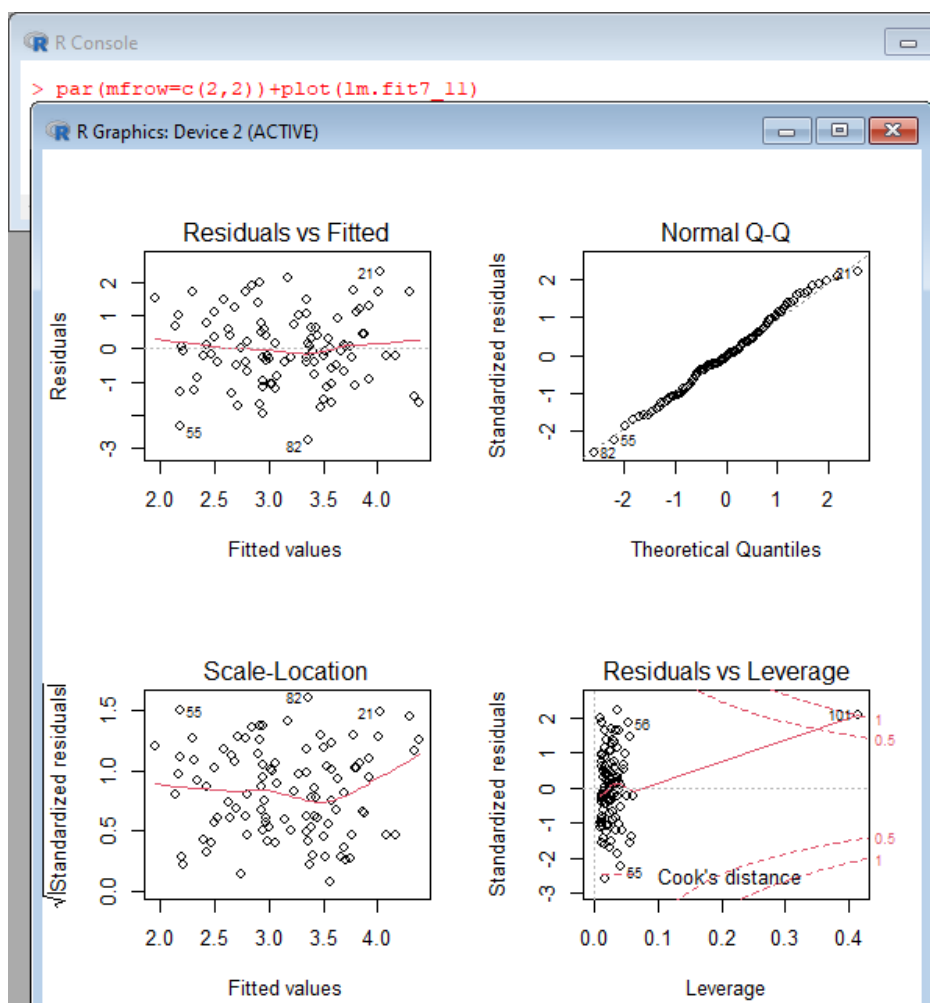
Residuals:
    Min       1Q   Median       3Q      Max
-2.73348 -0.69318 -0.05263  0.66385  2.30619

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.2267     0.2314   9.624 7.91e-16 ***
x1             0.5394     0.5922   0.911 0.36458
x2             2.5146     0.8977   2.801 0.00614 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.075 on 98 degrees of freedom
Multiple R-squared:  0.2188,    Adjusted R-squared:  0.2029
F-statistic: 13.72 on 2 and 98 DF,  p-value: 5.564e-06

> |
```

Нове спостереження впливає на цю модель: можна прийняти альтернативну гіпотезу для  $\beta_2$ , значення параметрів стали більш відмінні від реальних.



Можна побачити, що нове спостереження (101) є потенційним викидом з високим левереджем (більшим за  $(p+1)/n=0.03$ ).

```
R Console
> lm.fit7_l2=lm(y~x1)
> summary(lm.fit7_l2)

Call:
lm(formula = y ~ x1)

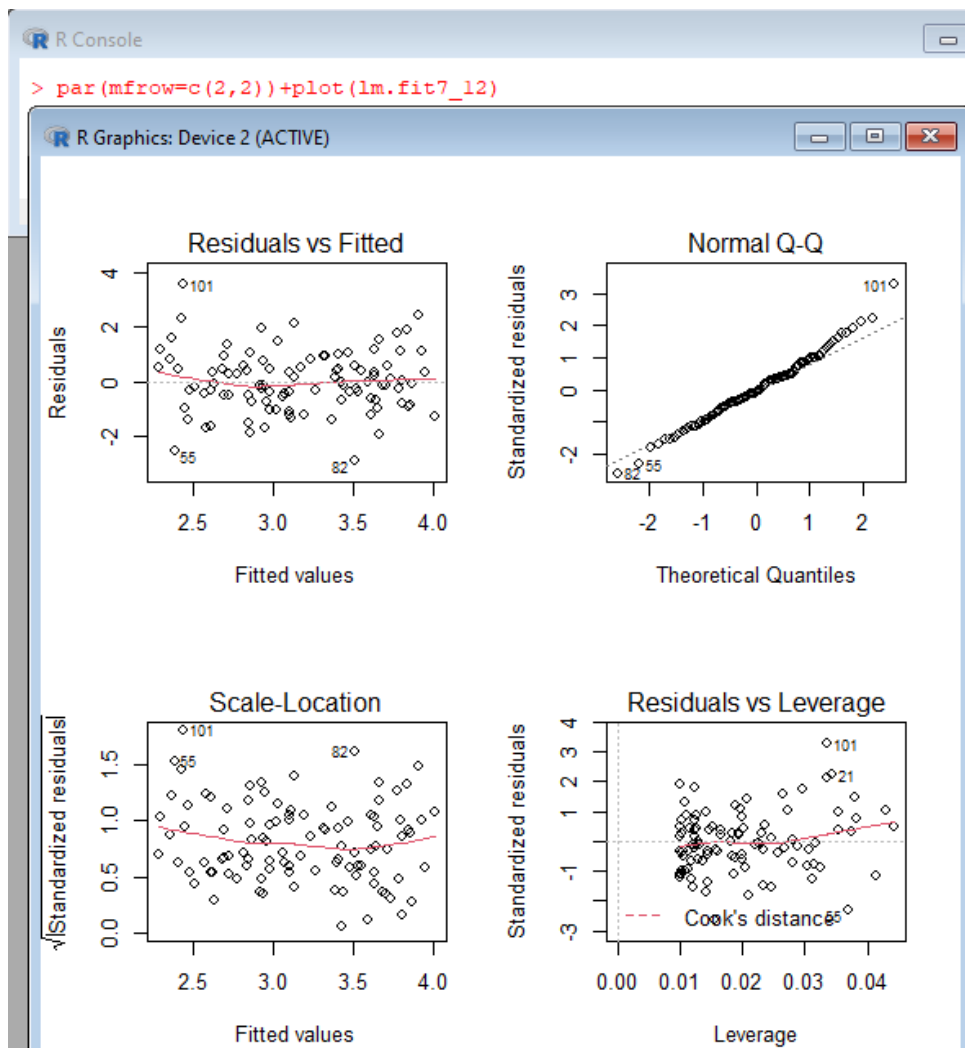
Residuals:
    Min       1Q   Median       3Q      Max
-2.8897 -0.6556 -0.0909  0.5682  3.5665

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.2569    0.2390   9.445 1.78e-15 ***
x1           1.7657    0.4124   4.282 4.29e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.111 on 99 degrees of freedom
Multiple R-squared:  0.1562,    Adjusted R-squared:  0.1477
F-statistic: 18.33 on 1 and 99 DF,  p-value: 4.295e-05

> |
```

На цю модель нове спостереження впливає не сильно, значення параметру  $\beta_0$  трохи збільшилось, а значення параметру  $\beta_1$  трохи зменшилось.



Можна побачити, що нове спостереження (101) є потенційним викидом, але низький левередж.

```
R Console
> lm.fit7_13=lm(y~x2)
> summary(lm.fit7_13)

Call:
lm(formula = y ~ x2)

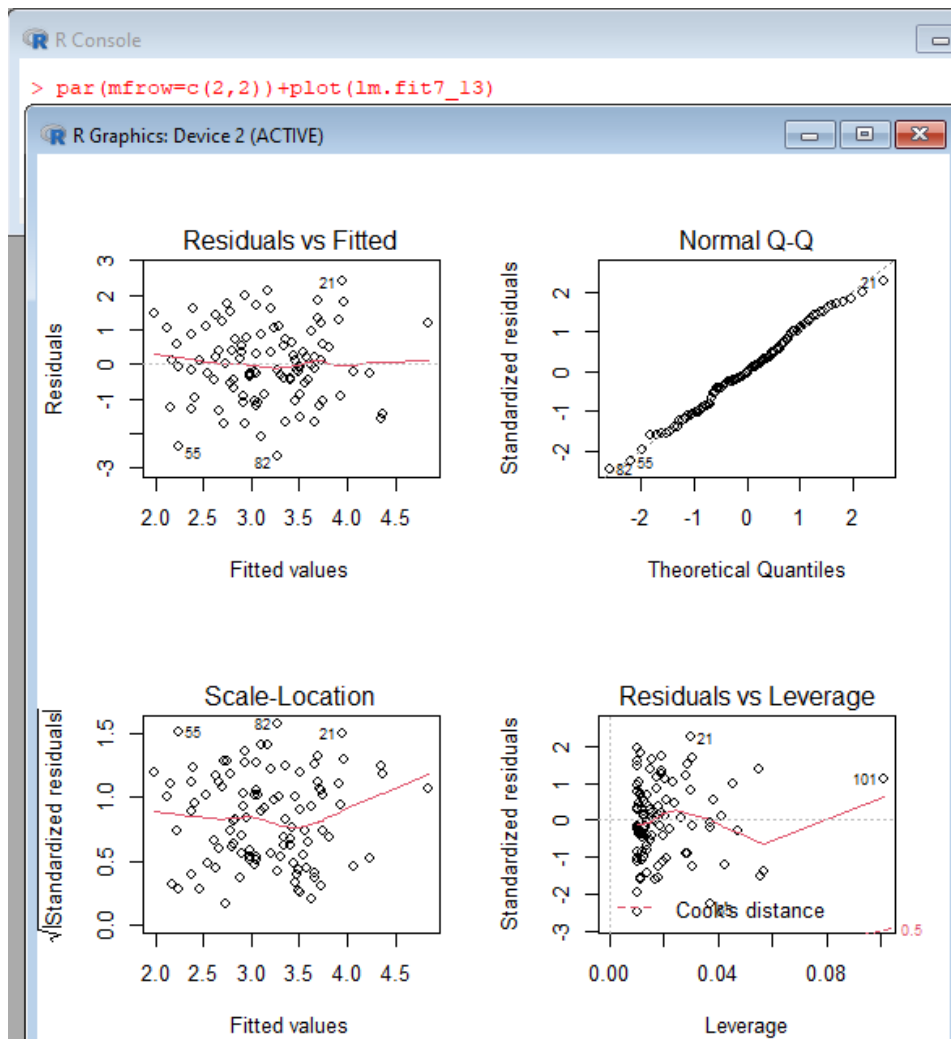
Residuals:
    Min       1Q   Median       3Q      Max
-2.64729 -0.71021 -0.06899  0.72699  2.38074

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.3451     0.1912  12.264 < 2e-16 ***
x2             3.1190     0.6040   5.164 1.25e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.074 on 99 degrees of freedom
Multiple R-squared:  0.2122,    Adjusted R-squared:  0.2042
F-statistic: 26.66 on 1 and 99 DF,  p-value: 1.253e-06

> |
```

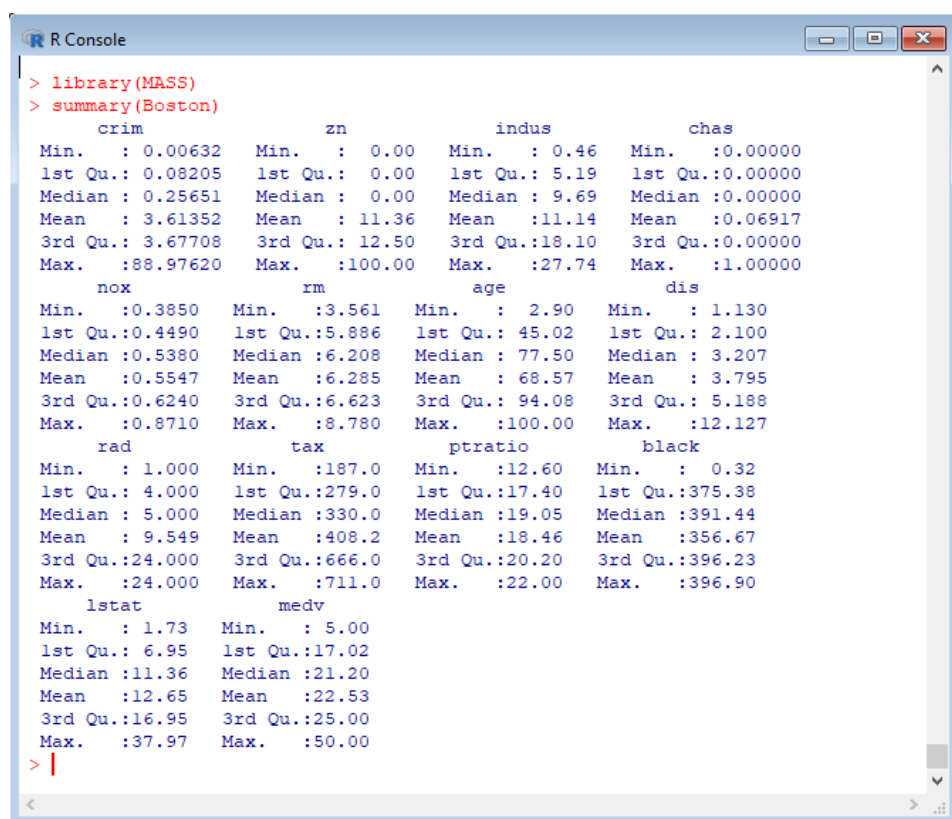
На цю модель нове спостереження також впливає не сильно, значення параметру  $\beta_2$  трохи збільшилось.



Можна побачити, що нове спостереження (101) має високий левередж, але не є потенційним викидом.



**Завдання 8.** Розглянула набір даних Boston. Спробувала спрогнозувати рівень злочинності на душу населення використовуючи інші змінні в цьому наборі даних. Рівень злочинності на душу населення – залежна змінна, а інші змінні - предиктори.



```
R Console
> library(MASS)
> summary(Boston)

      crim      zn      indus      chas
Min.   : 0.00632  Min.   : 0.00  Min.   : 0.46  Min.   :0.00000
1st Qu.: 0.08205  1st Qu.: 0.00  1st Qu.: 5.19  1st Qu.:0.00000
Median : 0.25651  Median : 0.00  Median : 9.69  Median :0.00000
Mean   : 3.61352  Mean   : 11.36  Mean   :11.14  Mean   :0.06917
3rd Qu.: 3.67708  3rd Qu.: 12.50  3rd Qu.:18.10  3rd Qu.:0.00000
Max.   :88.97620  Max.   :100.00  Max.   :27.74  Max.   :1.00000

      nox      rm      age      dis
Min.   :0.3850  Min.   :3.561  Min.   : 2.90  Min.   : 1.130
1st Qu.:0.4490  1st Qu.:5.886  1st Qu.: 45.02  1st Qu.: 2.100
Median :0.5380  Median :6.208  Median : 77.50  Median : 3.207
Mean   :0.5547  Mean   :6.285  Mean   : 68.57  Mean   : 3.795
3rd Qu.:0.6240  3rd Qu.:6.623  3rd Qu.: 94.08  3rd Qu.: 5.188
Max.   :0.8710  Max.   :8.780  Max.   :100.00  Max.   :12.127

      rad      tax      ptratio      black
Min.   : 1.000  Min.   :187.0  Min.   :12.60  Min.   : 0.32
1st Qu.: 4.000  1st Qu.:279.0  1st Qu.:17.40  1st Qu.:375.38
Median : 5.000  Median :330.0  Median :19.05  Median :391.44
Mean   : 9.549  Mean   :408.2  Mean   :18.46  Mean   :356.67
3rd Qu.:24.000  3rd Qu.:666.0  3rd Qu.:20.20  3rd Qu.:396.23
Max.   :24.000  Max.   :711.0  Max.   :22.00  Max.   :396.90

      lstat      medv
Min.   : 1.73  Min.   : 5.00
1st Qu.: 6.95  1st Qu.:17.02
Median :11.36  Median :21.20
Mean   :12.65  Mean   :22.53
3rd Qu.:16.95  3rd Qu.:25.00
Max.   :37.97  Max.   :50.00

> |
```

8.1. Для кожного предиктора побудувала просту модель лінійної регресії для прогнозування рівня злочинності на душу населення. Побудувала кілька графіків для більшої наочності.

```
R Console

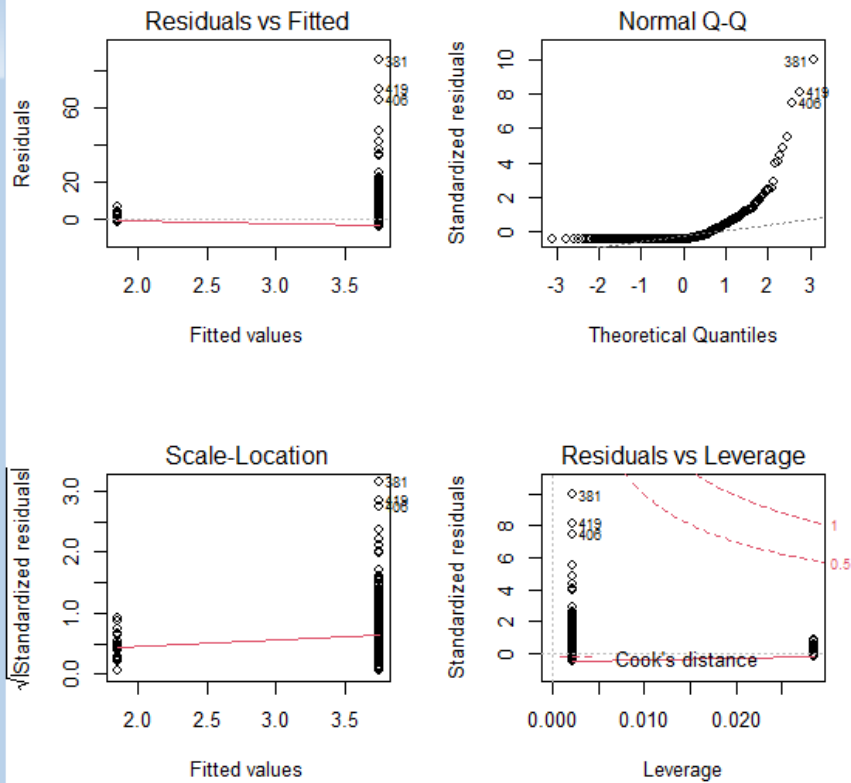
> lm.zn=lm(Boston$crim~Boston$zn)
> coefficients(summary(lm.zn))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)  4.45369376  0.4172178  10.674746 4.037668e-24
Boston$zn    -0.07393498  0.0160946  -4.593776 5.506472e-06
> lm.indus=lm(Boston$crim~Boston$indus)
> coefficients(summary(lm.indus))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) -2.0637426  0.66722830 -3.093008 2.091266e-03
Boston$indus  0.5097763  0.05102433  9.990848 1.450349e-21
> lm.chas=lm(Boston$crim~Boston$chas)
> coefficients(summary(lm.chas))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)  3.744447  0.3961111  9.453021 1.239505e-19
Boston$chas -1.892777  1.5061155 -1.256727 2.094345e-01
> lm.nox=lm(Boston$crim~Boston$nox)
> coefficients(summary(lm.nox))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) -13.71988  1.699479 -8.072992 5.076814e-15
Boston$nox   31.24853  2.999190 10.418989 3.751739e-23
> lm.rm=lm(Boston$crim~Boston$rm)
> coefficients(summary(lm.rm))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)  20.481804  3.3644742  6.087669 2.272000e-09
Boston$rm    -2.684051  0.5320411 -5.044819 6.346703e-07
> lm.age=lm(Boston$crim~Boston$age)
> coefficients(summary(lm.age))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) -3.7779063  0.94398472 -4.002084 7.221718e-05
Boston$age   0.1077862  0.01273644  8.462825 2.854869e-16
> lm.dis=lm(Boston$crim~Boston$dis)
> coefficients(summary(lm.dis))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)  9.499262  0.7303972 13.005611 1.502748e-33
Boston$dis   -1.550902  0.1683300 -9.213458 8.519949e-19
```

```
R Console

> lm.rad=lm(Boston$crim~Boston$rad)
> coefficients(summary(lm.rad))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) -2.2871594  0.44347583 -5.157349 3.605846e-07
Boston$rad   0.6179109  0.03433182 17.998199 2.693844e-56
> lm.tax=lm(Boston$crim~Boston$tax)
> coefficients(summary(lm.tax))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) -8.52836909  0.815809392 -10.45387 2.773600e-23
Boston$tax   0.02974225  0.001847415 16.09939 2.357127e-47
> lm.pptratio=lm(Boston$crim~Boston$ppratio)
> coefficients(summary(lm.pptratio))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) -17.646933  3.1472718 -5.607057 3.395255e-08
Boston$ppratio 1.151983  0.1693736  6.801430 2.942922e-11
> lm.black=lm(Boston$crim~Boston$black)
> coefficients(summary(lm.black))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) 16.55352922  1.425902755 11.609157 8.922239e-28
Boston$black -0.03627964  0.003873154 -9.366951 2.487274e-19
> lm.lstat=lm(Boston$crim~Boston$lstat)
> coefficients(summary(lm.lstat))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) -3.3305381  0.69375829 -4.800718 2.087022e-06
Boston$lstat  0.5488048  0.04776097 11.490654 2.654277e-27
> lm.medv=lm(Boston$crim~Boston$medv)
> coefficients(summary(lm.medv))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept) 11.7965358  0.93418916 12.62757 5.934119e-32
Boston$medv -0.3631599  0.03839017 -9.45971 1.173987e-19
> |
```

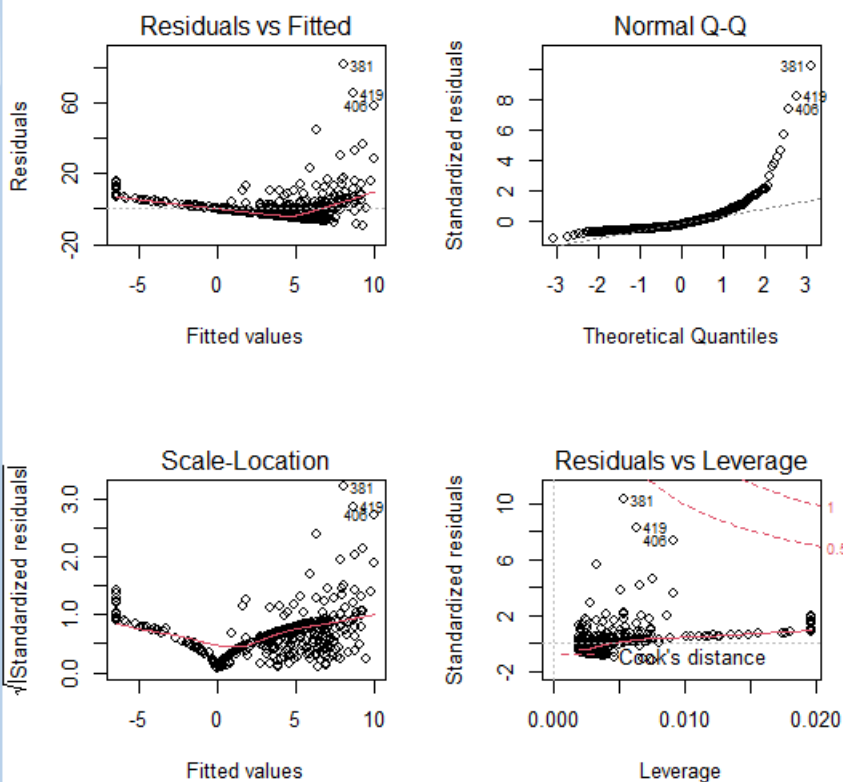
```
> par(mfrow=c(2,2))+plot(lm.chas)
```

R Graphics: Device 2 (ACTIVE)



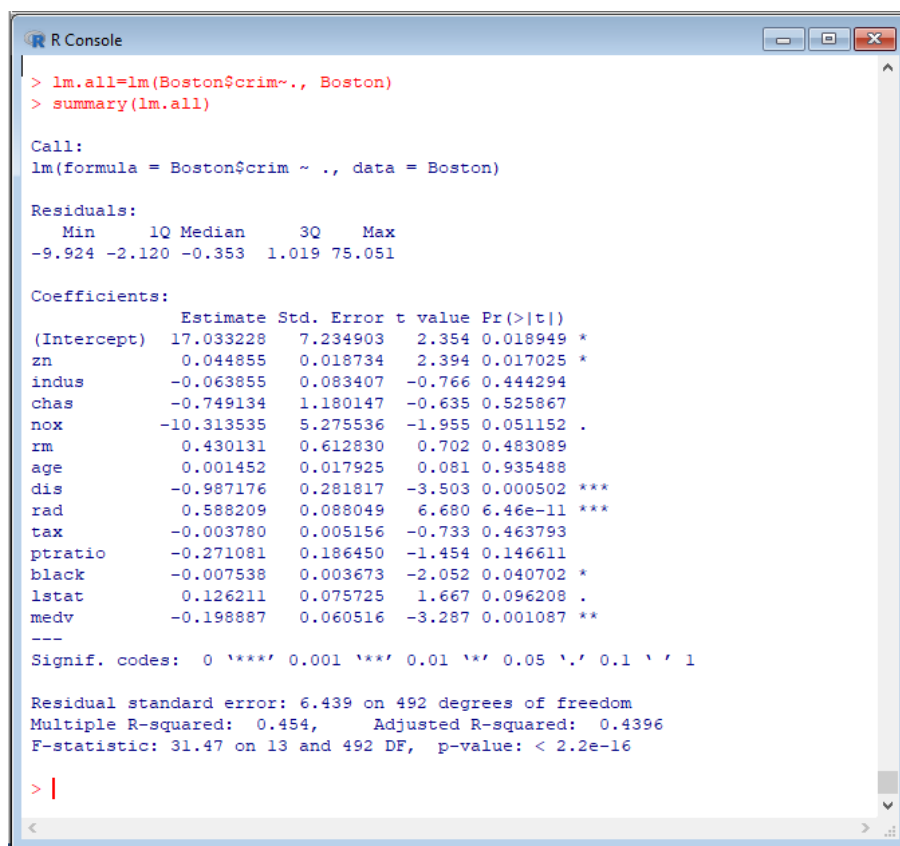
```
> par(mfrow=c(2,2))+plot(lm.medv)
```

R Graphics: Device 2 (ACTIVE)



Можна побачити, що значення p низьке для всіх предикторів, крім час. Тобто статистично значущий зв'язок залежної змінної існує з усіма ними, окрім chas.

8.2. Побудувала модель множинної регресії для прогнозування залежної змінної за допомогою всіх предикторів.



```
R Console
> lm.all=lm(Boston$crim~., Boston)
> summary(lm.all)

Call:
lm(formula = Boston$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

> |
```

Можна побачити, що альтернативну гіпотезу приймаємо для zn, dis, rad, black та medv. Тобто статистично значущий зв'язок залежної змінної існує з цими перерахованими.

8.3 Як результати з 8.1 співвідносяться з результатами з 8.2?

Результати з 8.1 та з 8.8 суттєво різні. Використовуючи окрему модель для окремого предиктора, то статистично значущий зв'язок був з більшістю з них. Більш точну інформацію стосовно цього ми отримуємо за допомогою множинної моделі. Також дуже відмінні коефіцієнти при користуванням моделями із 8.1 та 8.2.

8.4 Щоб визначити, чи є ознаки нелінійності зв'язку між будь-якими з предикторів та залежною змінною для кожного предиктора  $X$ , побудувала модель вигляду  $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \varepsilon$ .

```

R Console
> lm.zn=lm(Boston$crim~poly(Boston$zn, 3))
> coefficients(summary(lm.zn))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.372190   9.708814 1.547150e-20
poly(Boston$zn, 3)1 -38.749835    8.372207  -4.628389 4.697806e-06
poly(Boston$zn, 3)2  23.939832    8.372207  2.859441 4.420507e-03
poly(Boston$zn, 3)3 -10.071868    8.372207  -1.203012 2.295386e-01
> lm.indus=lm(Boston$crim~poly(Boston$indus, 3))
> coefficients(summary(lm.indus))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.329998  10.950138 3.606468e-25
poly(Boston$indus, 3)1  78.590819    7.423121  10.587301 8.854243e-24
poly(Boston$indus, 3)2 -24.394796    7.423121  -3.286326 1.086057e-03
poly(Boston$indus, 3)3 -54.129763    7.423121  -7.292049 1.196405e-12
> lm.nox=lm(Boston$crim~poly(Boston$nox, 3))
> coefficients(summary(lm.nox))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.321573  11.237025 2.742908e-26
poly(Boston$nox, 3)1  81.372015    7.233605  11.249165 2.457491e-26
poly(Boston$nox, 3)2 -28.828594    7.233605  -3.985370 7.736755e-05
poly(Boston$nox, 3)3 -60.361894    7.233605  -8.344649 6.961110e-16
> lm.rm=lm(Boston$crim~poly(Boston$rm, 3))
> coefficients(summary(lm.rm))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.3702993  9.7583873 1.026665e-20
poly(Boston$rm, 3)1 -42.379442    8.3296758  -5.0877661 5.128048e-07
poly(Boston$rm, 3)2  26.576770    8.3296758  3.1906128 1.508545e-03
poly(Boston$rm, 3)3  -5.510342    8.3296758  -0.6615314 5.085751e-01
> lm.age=lm(Boston$crim~poly(Boston$age, 3))
> coefficients(summary(lm.age))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.3485173  10.368276 5.918933e-23
poly(Boston$age, 3)1  68.182009    7.8397027  8.697015 4.878803e-17
poly(Boston$age, 3)2  37.484470    7.8397027  4.781364 2.291156e-06
poly(Boston$age, 3)3  21.353207    7.8397027  2.723727 6.679915e-03

```

```

R Console
> lm.dis=lm(Boston$crim~poly(Boston$dis, 3))
> coefficients(summary(lm.dis))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.325924  11.087013 1.060226e-25
poly(Boston$dis, 3)1 -73.388590    7.331479 -10.010066 1.253249e-21
poly(Boston$dis, 3)2  56.373036    7.331479  7.689176 7.869767e-14
poly(Boston$dis, 3)3 -42.621877    7.331479  -5.813544 1.088832e-08
> lm.rad=lm(Boston$crim~poly(Boston$rad, 3))
> coefficients(summary(lm.rad))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.297069  12.163920 5.149845e-30
poly(Boston$rad, 3)1 120.907446    6.682402  18.093412 1.053211e-56
poly(Boston$rad, 3)2  17.492299    6.682402  2.617666 9.120558e-03
poly(Boston$rad, 3)3  4.698457    6.682402  0.703109 4.823138e-01
> lm.pptratio=lm(Boston$crim~poly(Boston$ptratio, 3))
> coefficients(summary(lm.pptratio))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.3610484  10.008419 1.270767e-21
poly(Boston$ptratio, 3)1  56.045229    8.1215830  6.900777 1.565484e-11
poly(Boston$ptratio, 3)2  24.774824    8.1215830  3.050492 2.405468e-03
poly(Boston$ptratio, 3)3 -22.279737    8.1215830  -2.743275 6.300514e-03
> lm.black=lm(Boston$crim~poly(Boston$black, 3))
> coefficients(summary(lm.black))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524    0.353627  10.2184605 2.139710e-22
poly(Boston$black, 3)1 -74.431199    7.954643  -9.3569505 2.730082e-19
poly(Boston$black, 3)2  5.926419    7.954643  0.7450264 4.566044e-01
poly(Boston$black, 3)3 -4.834565    7.954643  -0.6077665 5.436172e-01

```

```
R Console
> lm.lstat=lm(Boston$crim~poly(Boston$lstat, 3))
> coefficients(summary(lm.lstat))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524   0.3391698  10.654025 4.939398e-24
poly(Boston$lstat, 3)1  88.069666   7.6294361  11.543404 1.678072e-27
poly(Boston$lstat, 3)2  15.888164   7.6294361   2.082482 3.780418e-02
poly(Boston$lstat, 3)3 -11.574022   7.6294361  -1.517022 1.298906e-01
> lm.medv=lm(Boston$crim~poly(Boston$medv, 3))
> coefficients(summary(lm.medv))
              Estimate Std. Error  t value    Pr(>|t|)
(Intercept)      3.613524   0.2920344  12.373622 7.024110e-31
poly(Boston$medv, 3)1 -75.057605   6.5691520 -11.425768 4.930818e-27
poly(Boston$medv, 3)2  88.086211   6.5691520  13.409069 2.928577e-35
poly(Boston$medv, 3)3 -48.033435   6.5691520  -7.311969 1.046510e-12
> |
```

Можна побачити, що квадратичний доданок є статистично значущим для zn, gm, rad, tax, lstat. Кубічний є сатичтично значущим для indus, nox, age, dis, patio, medv. Для black не є статистично значущим ні квадрайтчний, ні кубічний коефіцієнти.