МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ІВАНА ФРАНКА

Звіт

до індивідуального завдання №6

з предмету Моделі статистичного навчання

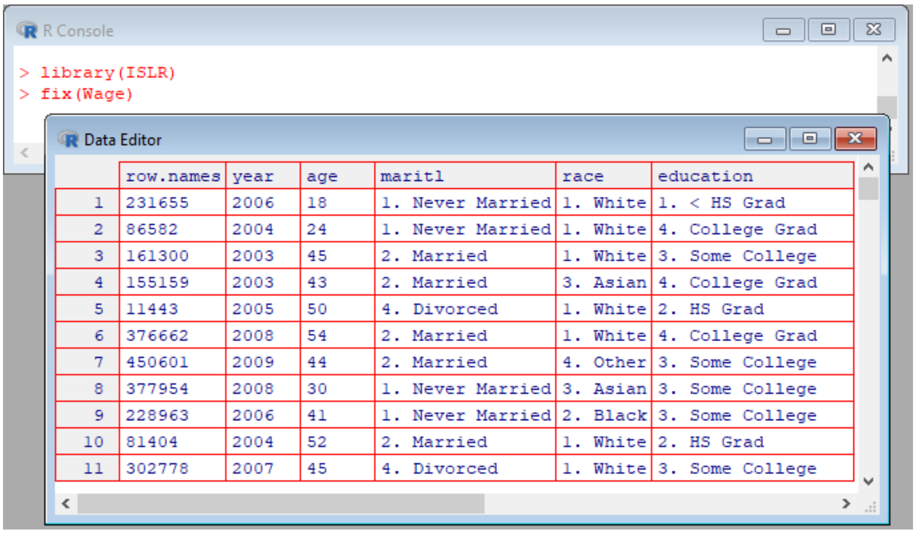
Роботу виконала:

Мерцало Ірина Ігорівна,

студентка групи ПМІМ-11

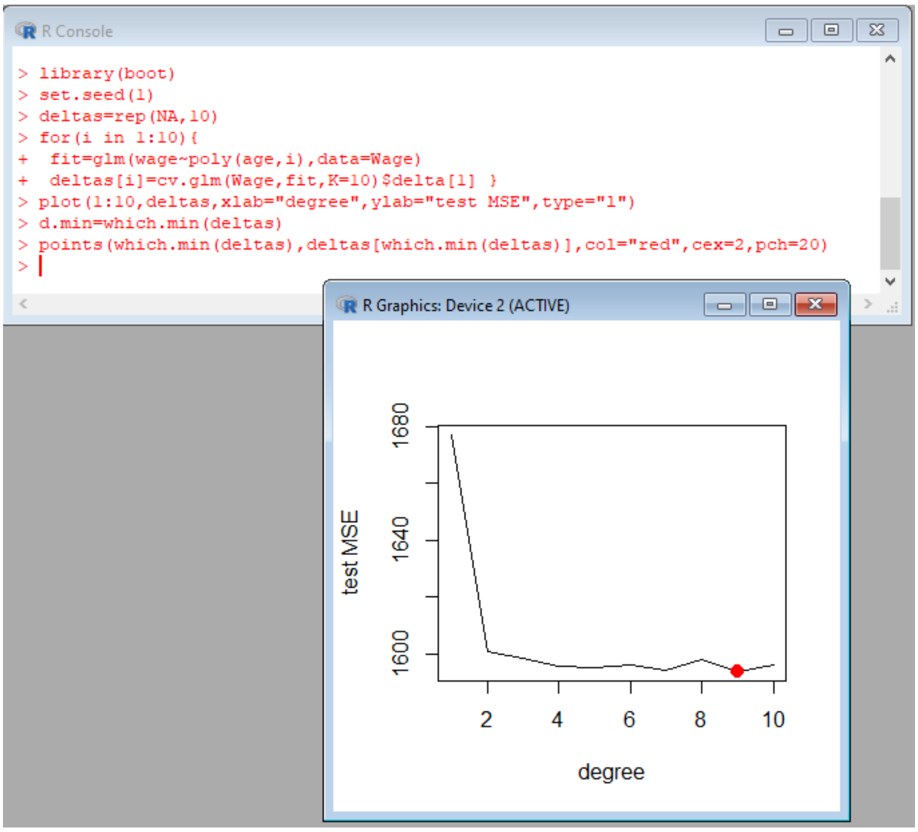
Львів – 2021

Завдання 1. Додатково проаналізувала набір даних Wage:

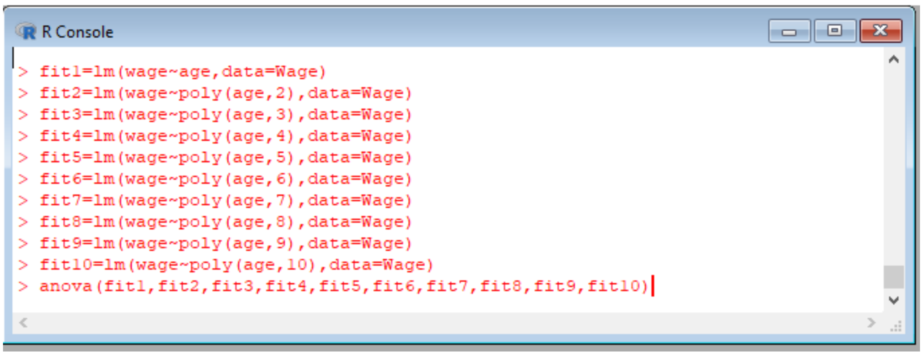


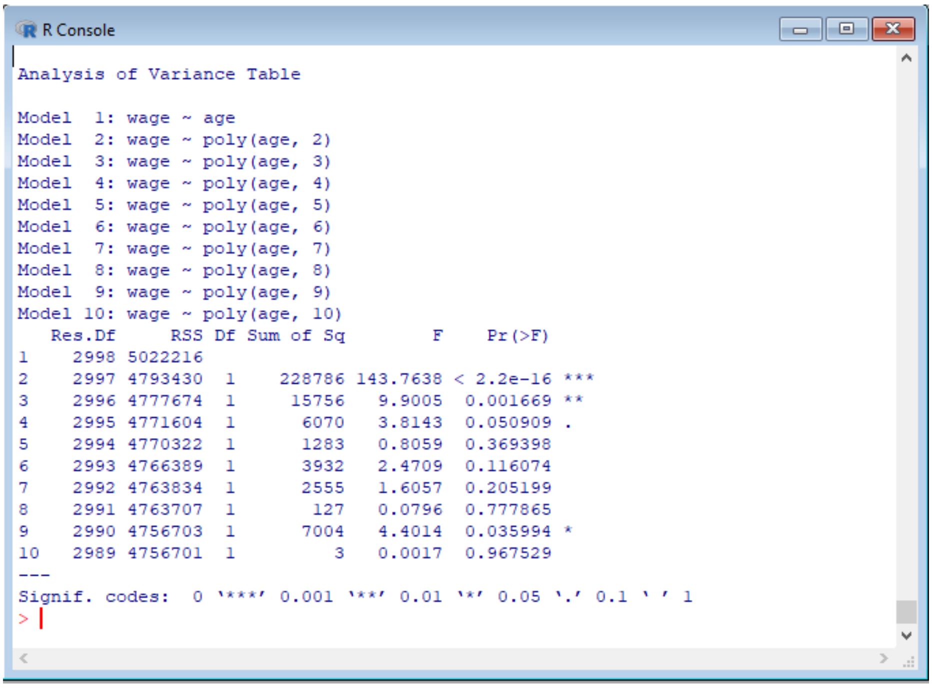
	row.names	year	age	maritl	race	education
1	231655	2006	18	1. Never Married	1. White	1. < HS Grad
2	86582	2004	24	1. Never Married	1. White	4. College Grad
3	161300	2003	45	2. Married	1. White	3. Some College
4	155159	2003	43	2. Married	3. Asian	4. College Grad
5	11443	2005	50	4. Divorced	1. White	2. HS Grad
6	376662	2008	54	2. Married	1. White	4. College Grad
7	450601	2009	44	2. Married	4. Other	3. Some College
8	377954	2008	30	1. Never Married	3. Asian	3. Some College
9	228963	2006	41	1. Never Married	2. Black	3. Some College
10	81404	2004	52	2. Married	1. White	2. HS Grad
11	302778	2007	45	4. Divorced	1. White	3. Some College

1.1 Використала поліноміальну регресію для прогнозування wage за age:



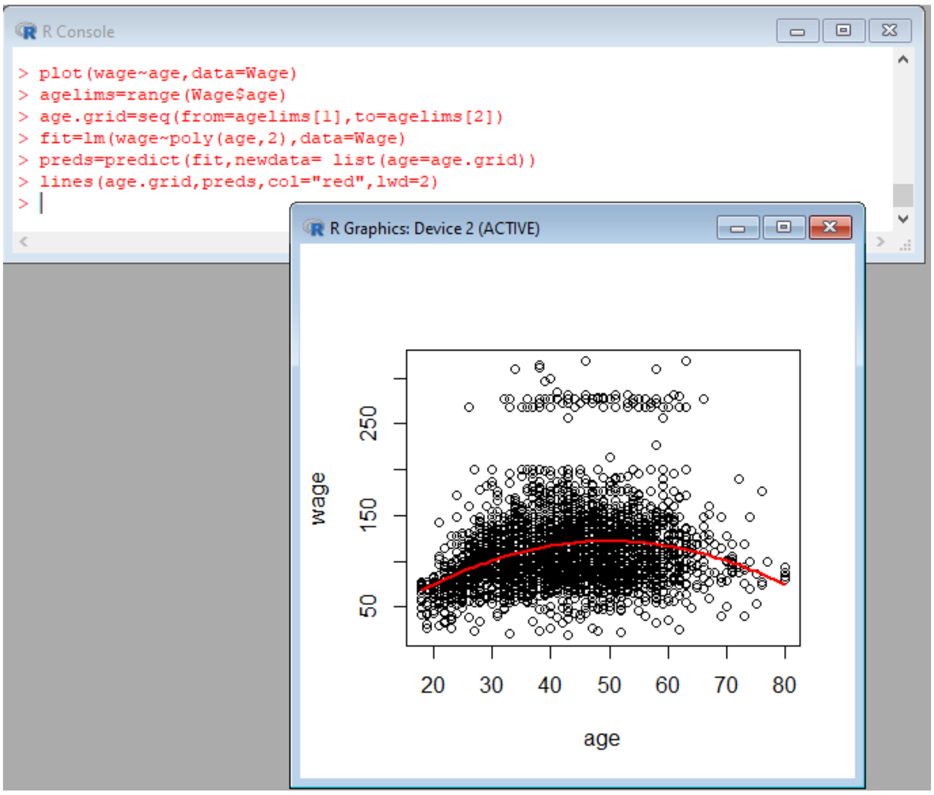
На графіку можна побачити, що оптимальний степінь полінома дорівнює 9.

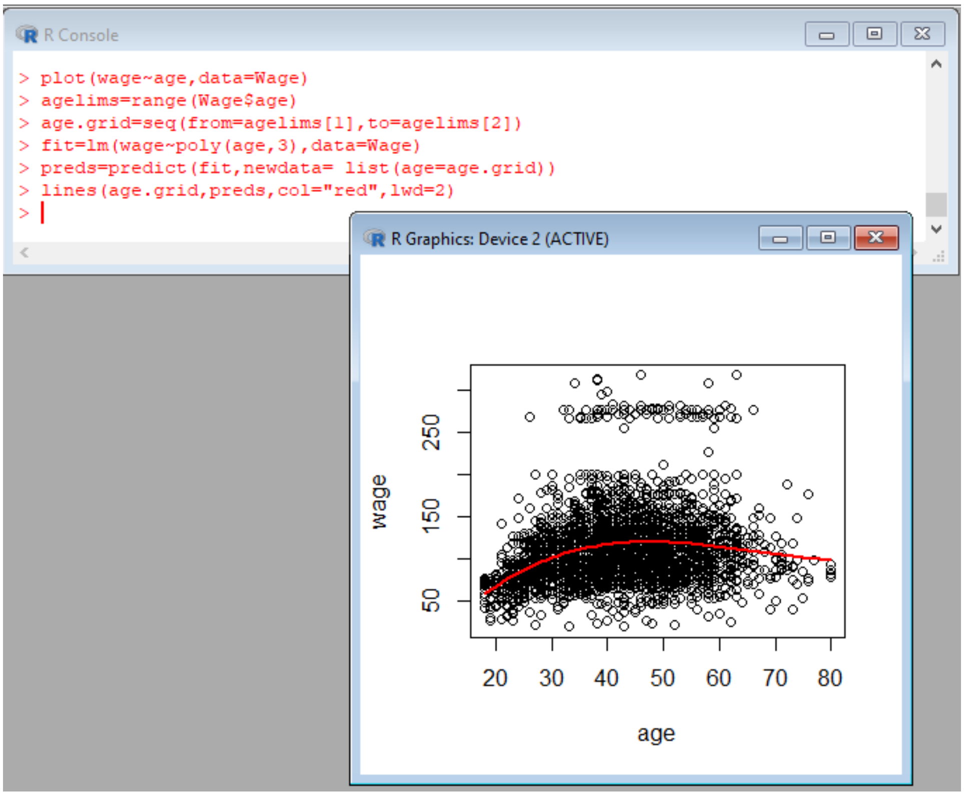




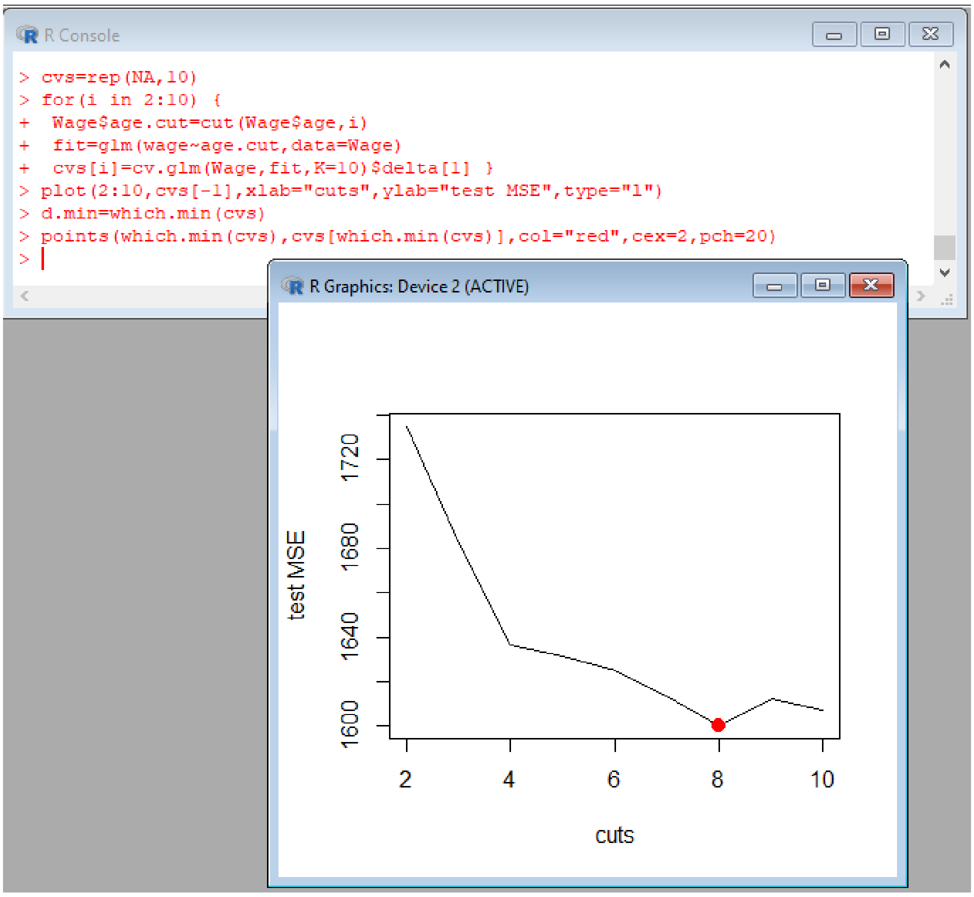
```
Analysis of Variance Table

Model  1: wage ~ age
Model  2: wage ~ poly(age, 2)
Model  3: wage ~ poly(age, 3)
Model  4: wage ~ poly(age, 4)
Model  5: wage ~ poly(age, 5)
Model  6: wage ~ poly(age, 6)
Model  7: wage ~ poly(age, 7)
Model  8: wage ~ poly(age, 8)
Model  9: wage ~ poly(age, 9)
Model 10: wage ~ poly(age, 10)
   Res.Df     RSS Df Sum of Sq        F    Pr(>F)
1    2998 5022216
2    2997 4793430  1    228786 143.7638 < 2.2e-16 ***
3    2996 4777674  1     15756   9.9005  0.001669 **
4    2995 4771604  1      6070   3.8143  0.050909 .
5    2994 4770322  1      1283   0.8059  0.369398
6    2993 4766389  1      3932   2.4709  0.116074
7    2992 4763834  1      2555   1.6057  0.205199
8    2991 4763707  1       127   0.0796  0.777865
9    2990 4756703  1      7004   4.4014  0.035994 *
10   2989 4756701  1         3   0.0017  0.967529
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
```

По p-значенню можна побачити, що найкращий результат отримуємо, коли степені поліному дорівнюють 2 і 3. Тому побудувала їхні графіки:



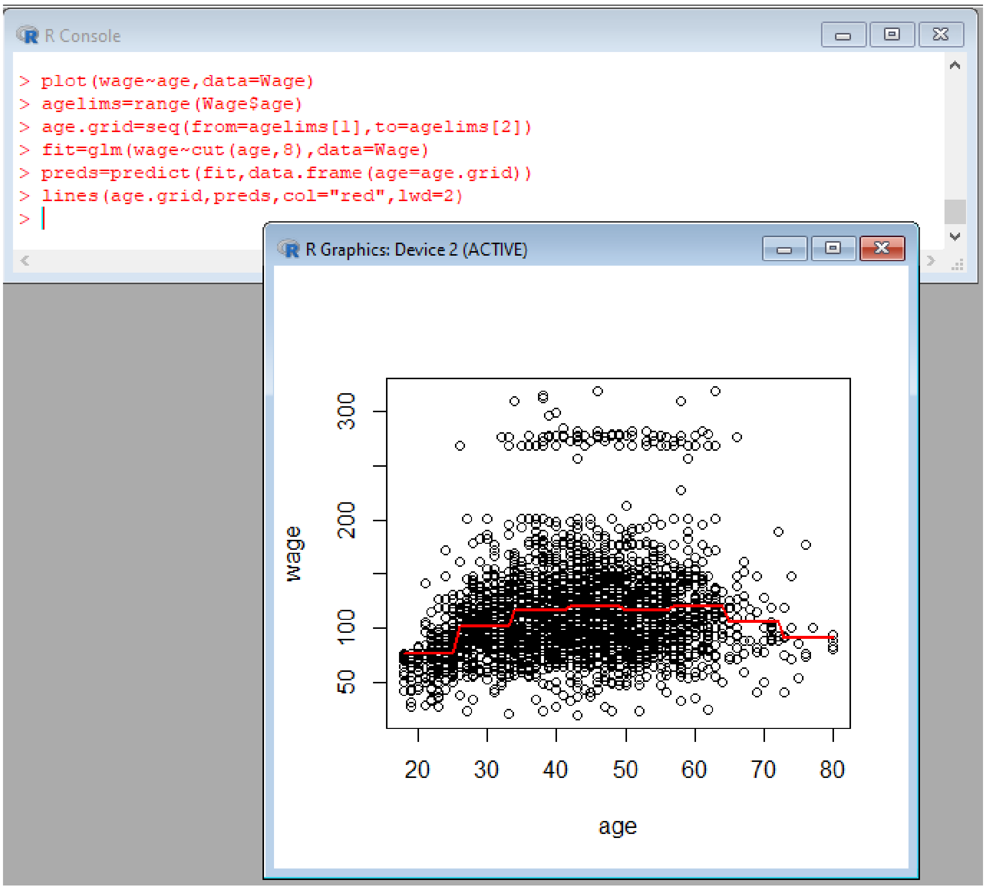


1.2 Використала східчасту функцію для прогнозування wage за age та провела перехресну перевірку для вибору оптимальної кількості розрізів:



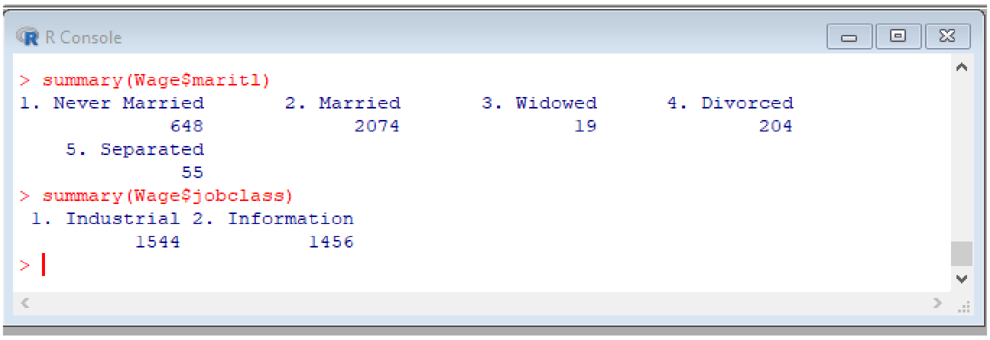
На графіку можна побачити, що мінімальна помилка буде тоді, коли кількість зрізів дорівнює 8.

Побудувала графік з отриманими результатами:



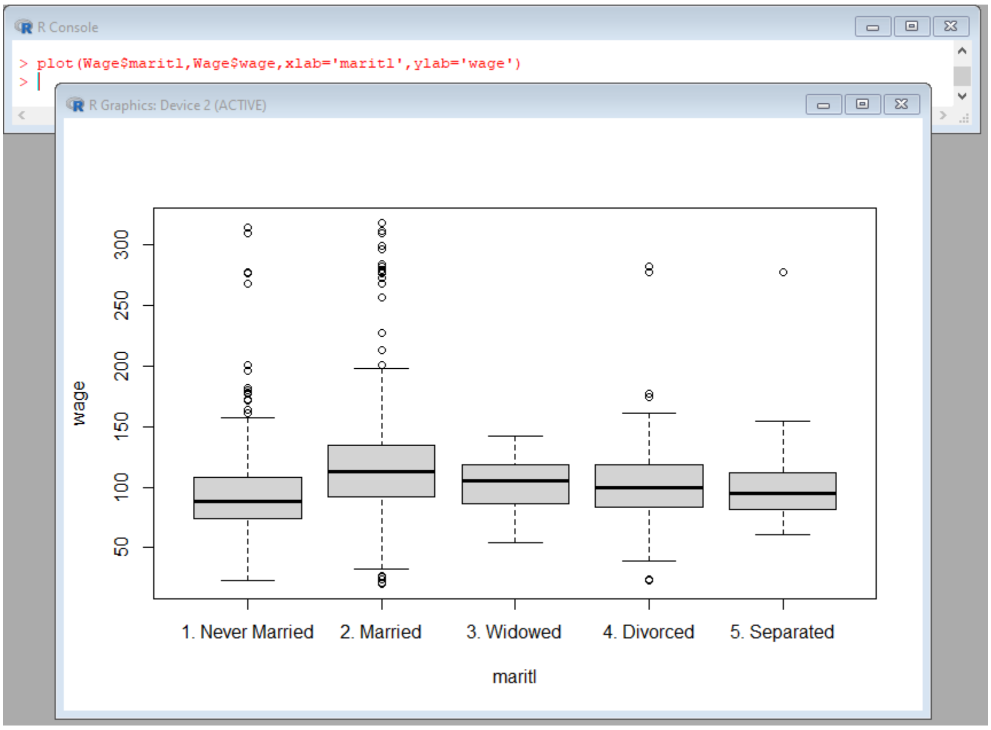
Завдання 2

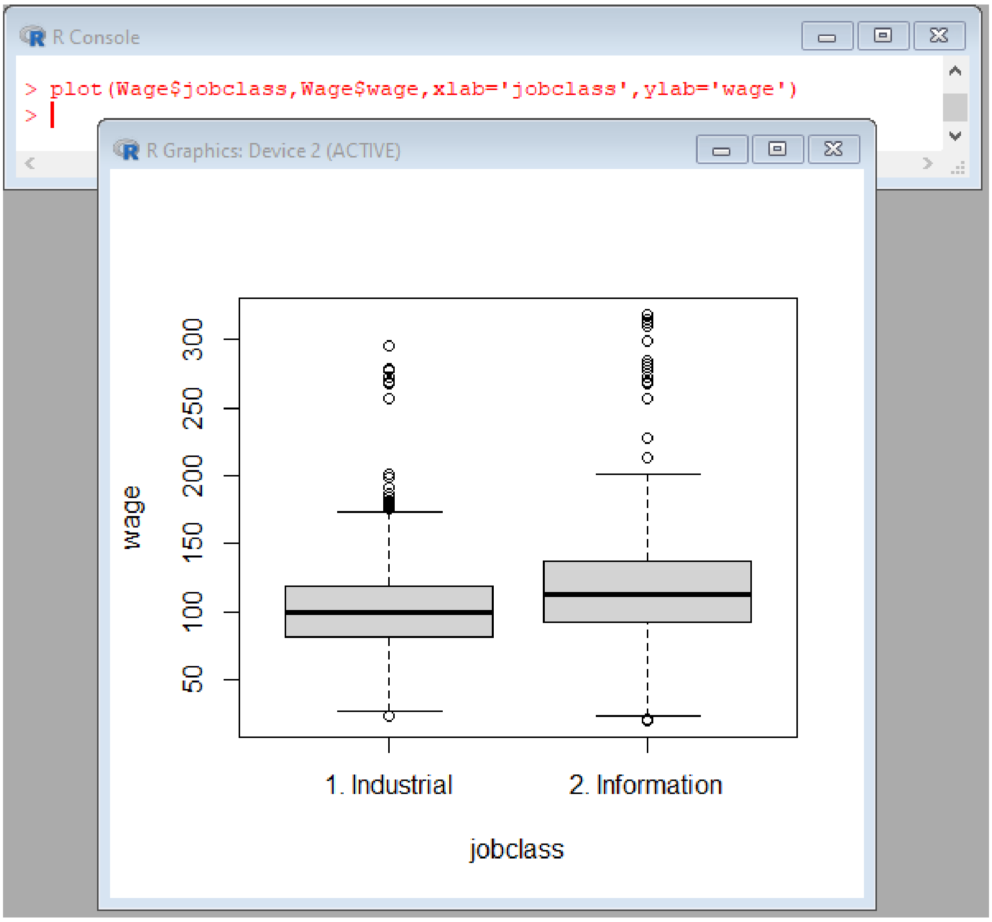
Набір даних Wage містить інші змінні наприклад, сімейний стан (maritl), робочий клас (jobclass):



```
> summary(Wage$maritl)
1. Never Married        2. Married       3. Widowed      4. Divorced
             648              2074               19              204
    5. Separated
              55
> summary(Wage$jobclass)
 1. Industrial 2. Information
          1544           1456
>
```

Дослідила зв’язки предикторів maritl і jobclass з wage:





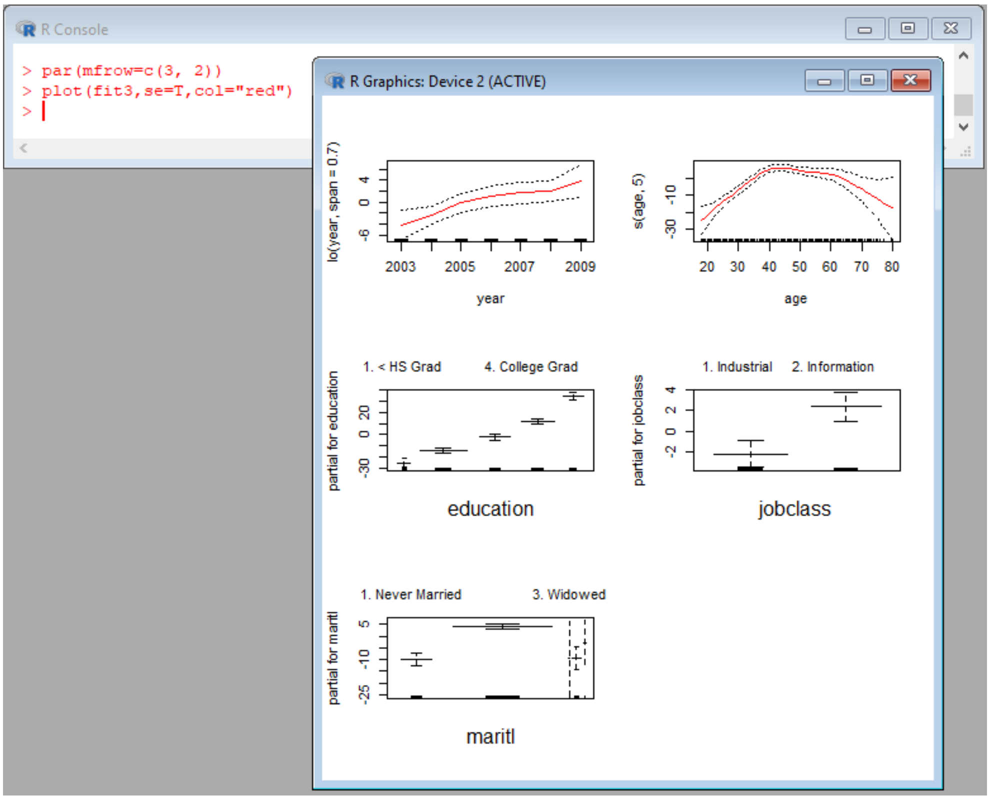
З першого графіку можна побачити, що максимальний заробіток є у сімейних пар, а мінімальний у тих, хто ніколи не одружувався.

А з другого, що особи з інформаційного класу діяльності заробляють більше, ніж особи з індустріального класу.

```
> fit0=gam(wage~lo(year,span=0.7)+s(age,5)+education,data=Wage)
> fit1=gam(wage~lo(year,span=0.7)+s(age,5)+education+jobclass,data=Wage)
> fit2=gam(wage~lo(year,span=0.7)+s(age,5)+education+maritl,data=Wage)
> fit3=gam(wage~lo(year,span=0.7)+s(age,5)+education+jobclass+maritl,data=Wage)
> anova(fit0,fit1,fit2,fit3)
Analysis of Deviance Table

Model 1: wage ~ lo(year, span = 0.7) + s(age, 5) + education
Model 2: wage ~ lo(year, span = 0.7) + s(age, 5) + education + jobclass
Model 3: wage ~ lo(year, span = 0.7) + s(age, 5) + education + maritl
Model 4: wage ~ lo(year, span = 0.7) + s(age, 5) + education + jobclass +
    maritl
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1    2987.1    3691855
2    2986.1    3679689  1    12166 0.0014637 **
3    2983.1    3597526  3    82163  9.53e-15 ***
4    2982.1    3583675  1    13852 0.0006862 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
```

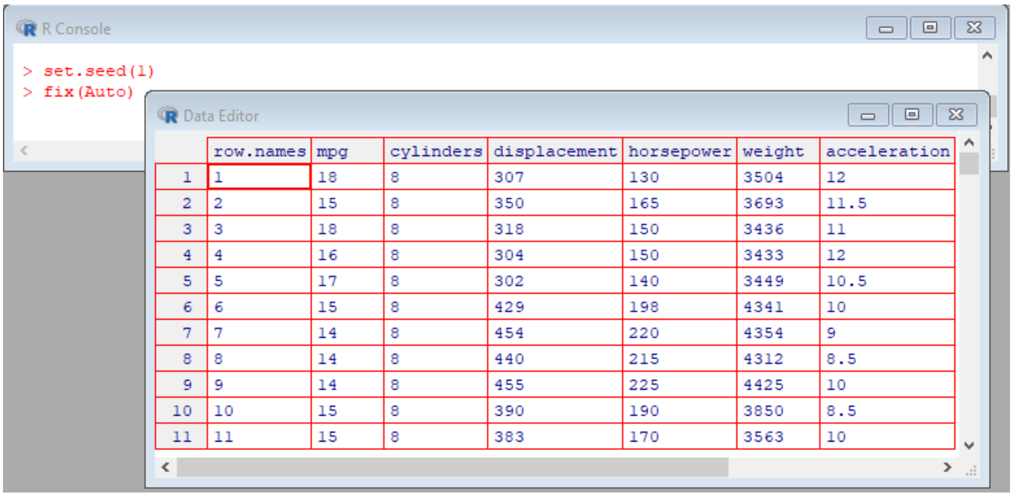
По р-значенню можна побачити, що найкраще підходить третя модель.

Побудувала графіки отриманих результатів:



Завдання 3

Проаналізувала набір даних Auto:

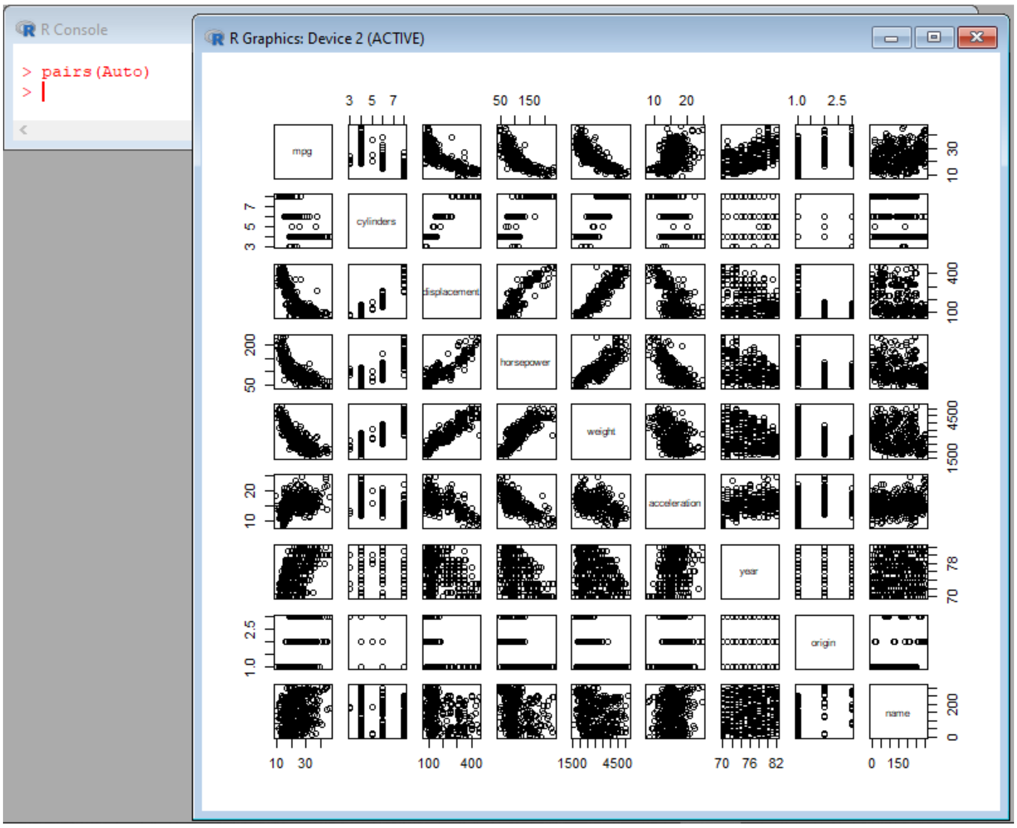


R Console

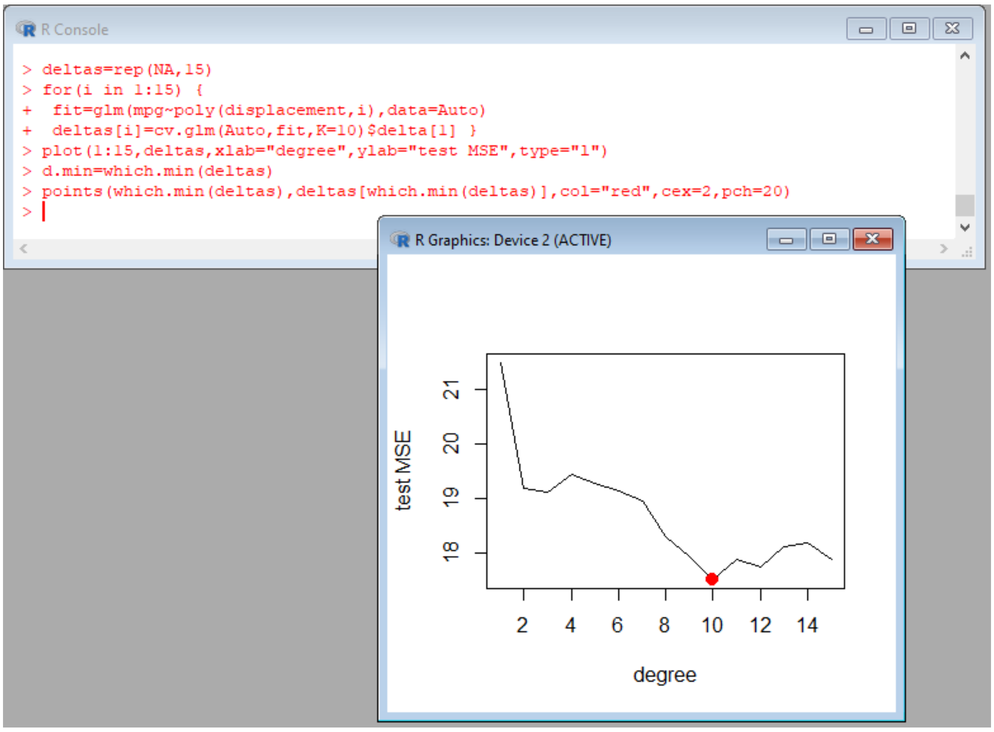
```
> set.seed(1)
> fix(Auto)
```

Data Editor

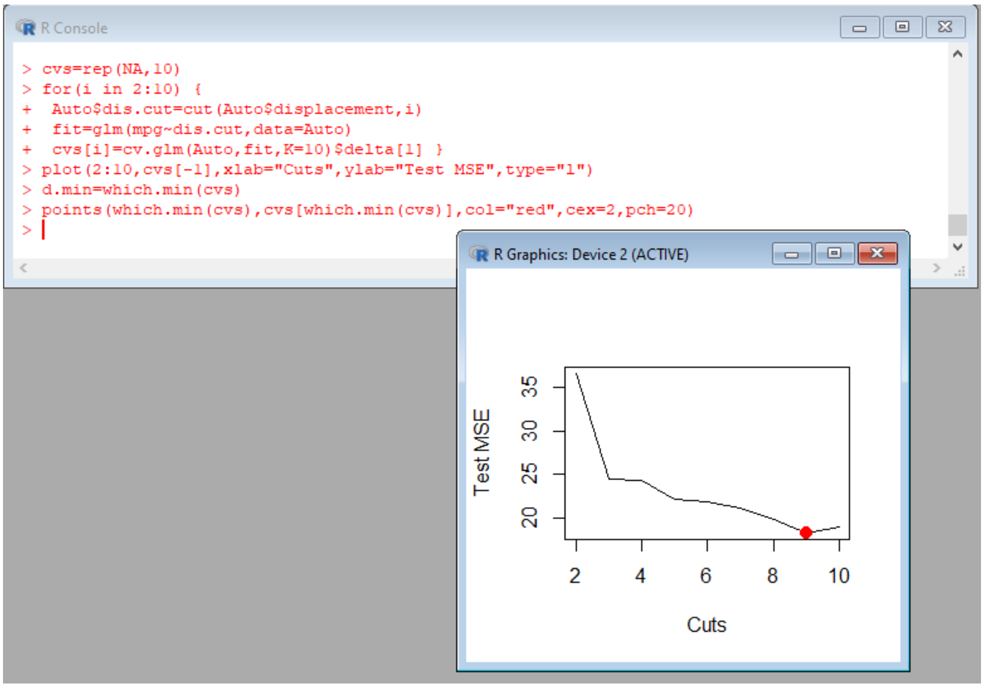
	row.names	mpg	cylinders	displacement	horsepower	weight	acceleration
1	1	18	8	307	130	3504	12
2	2	15	8	350	165	3693	11.5
3	3	18	8	318	150	3436	11
4	4	16	8	304	150	3433	12
5	5	17	8	302	140	3449	10.5
6	6	15	8	429	198	4341	10
7	7	14	8	454	220	4354	9
8	8	14	8	440	215	4312	8.5
9	9	14	8	455	225	4425	10
10	10	15	8	390	190	3850	8.5
11	11	15	8	383	170	3563	10



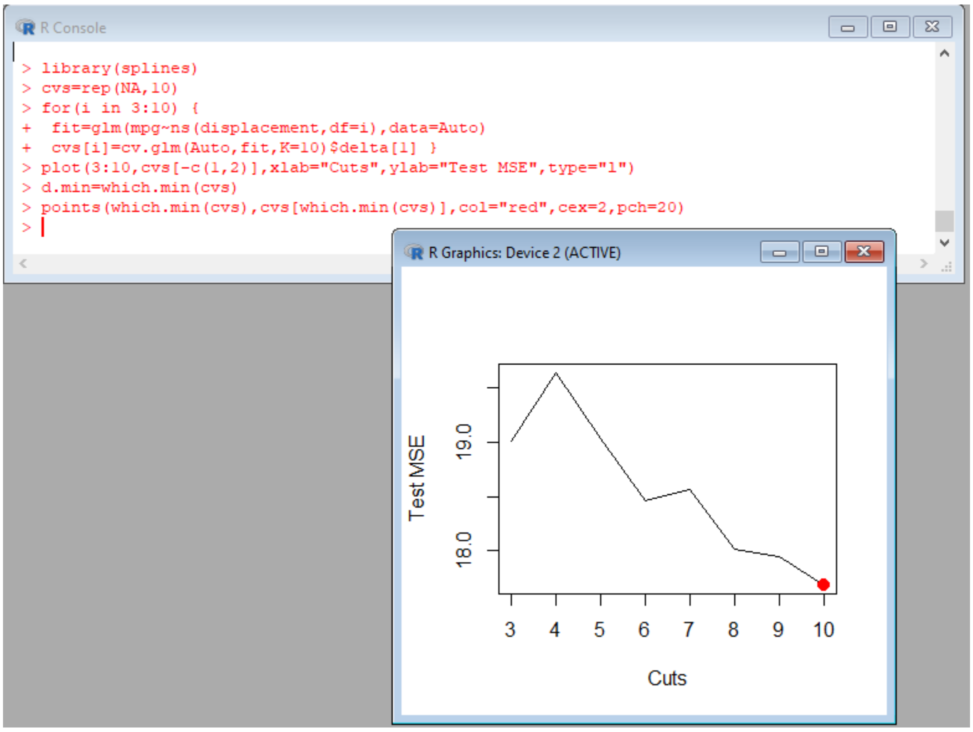
Можна побачити, що існує негативна кореляція mpg з cylinders, displacement, horsepower та weight.



На графіку можна побачити, що оптимальний степінь полінома дорівнює 10.



На графіку можна побачити, що мінімальна помилка буде тоді, коли кількість зрізів дорівнює 9.



На графіку можна побачити, що мінімальна помилка буде тоді, коли кількість ступенів свободи дорівнює 10.

```
> fit=gam(mpg~s(displacement,4)+s(horsepower,4),data=Auto)
> summary(fit)

Call: gam(formula = mpg ~ s(displacement, 4) + s(horsepower, 4), data = Auto)
Deviance Residuals:
     Min       1Q   Median       3Q      Max
-11.2982  -2.1592  -0.4394   2.1247  17.0946

(Dispersion Parameter for gaussian family taken to be 15.3543)

    Null Deviance: 23818.99 on 391 degrees of freedom
Residual Deviance: 5880.697 on 382.9999 degrees of freedom
AIC: 2194.05

Number of Local Scoring Iterations: NA

Anova for Parametric Effects
                    Df  Sum Sq Mean Sq F value   Pr(>F)
s(displacement, 4)   1 15254.9 15254.9 993.524 < 2e-16 ***
s(horsepower, 4)     1  1038.4  1038.4  67.632 3.1e-15 ***
Residuals          383  5880.7    15.4
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Anova for Nonparametric Effects
                   Npar Df Npar F     Pr(F)
(Intercept)
s(displacement, 4)       3 13.613 1.863e-08 ***
s(horsepower, 4)         3 15.606 1.349e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>
```

Завдання 4

Проаналізувала набір даних Boston:

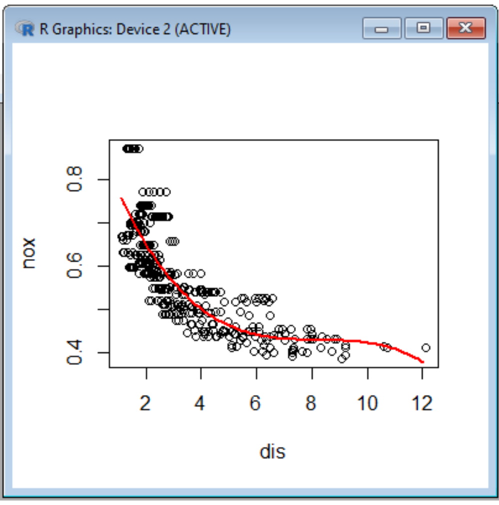
```
> library(MASS)
> fix(Boston)
```

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax
1	0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296
2	0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242
3	0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242
4	0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222
5	0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222
6	0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222
7	0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311
8	0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311
9	0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311
10	0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311
11	0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311

4.1 Використовуючи функцію poly(), встановила кубічну поліноміальну регресію для передбачення nox за допомогою dis:

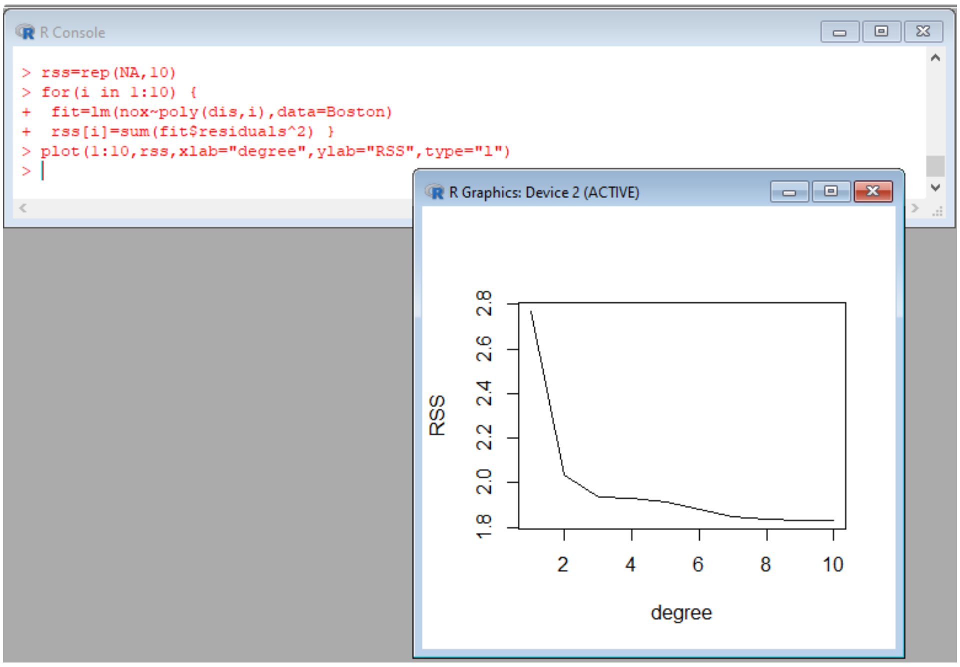
```
> set.seed(1)
> fit=lm(nox~poly(dis,3),data = Boston)
> summary(fit)

Call:
lm(formula = nox ~ poly(dis, 3), data = Boston)

Residuals:
      Min        1Q    Median        3Q       Max
-0.121130 -0.040619 -0.009738  0.023385  0.194904

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    0.554695   0.002759 201.021  < 2e-16 ***
poly(dis, 3)1 -2.003096   0.062071 -32.271  < 2e-16 ***
poly(dis, 3)2  0.856330   0.062071  13.796  < 2e-16 ***
poly(dis, 3)3 -0.318049   0.062071  -5.124 4.27e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.06207 on 502 degrees of freedom
Multiple R-squared:  0.7148,    Adjusted R-squared:  0.7131
F-statistic: 419.3 on 3 and 502 DF,  p-value: < 2.2e-16

>
```

```
> dislims=range(Boston$dis)
> dis.grid=seq(from=dislims[1],to=dislims[2],by= 0.1)
> preds=predict(fit,list(dis=dis.grid))
> plot(nox~dis,data=Boston)
> lines(dis.grid,preds,col="red",lwd=2)
>
```



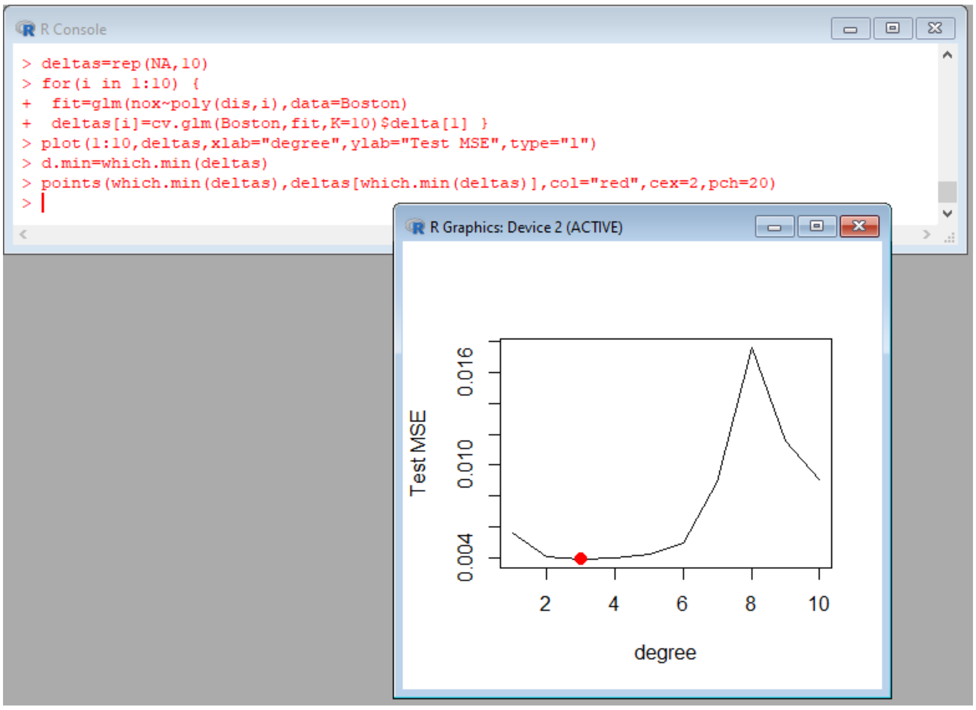
Можна побачити, що всі доданки в поліномі – значущі.

4.2 Побудувала поліноміальні моделі для різних степенів (скажімо, від 1 до 10), і навела їхні RSS:



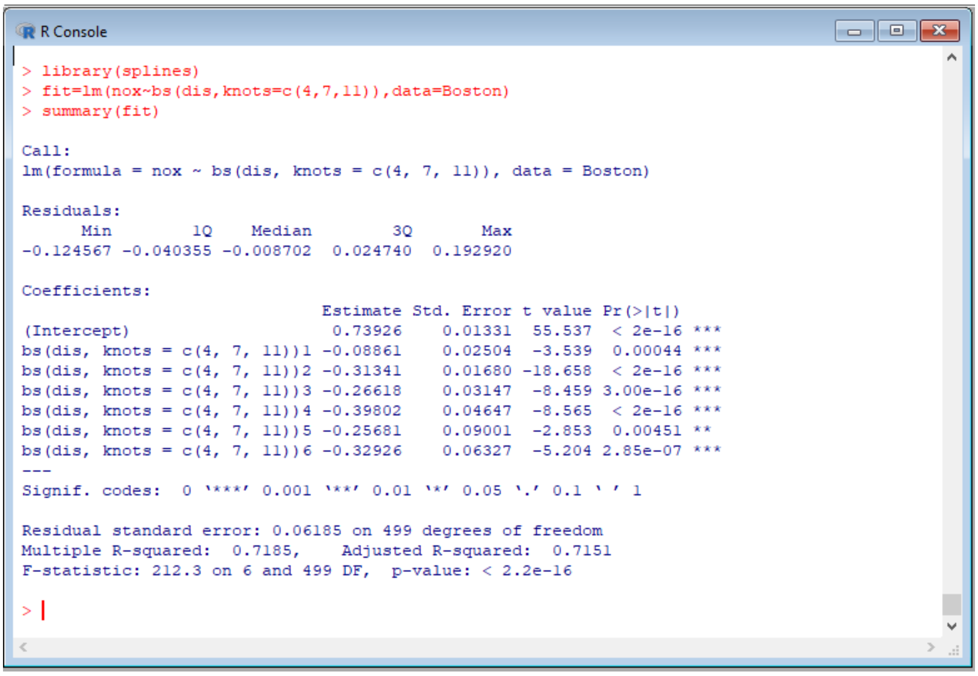
На графіку можна побачити, що чим більший степінь полінома, тим менше RSS.

4.3 Використала поліноміальну регресію для прогнозування оптимального степеня полінома.



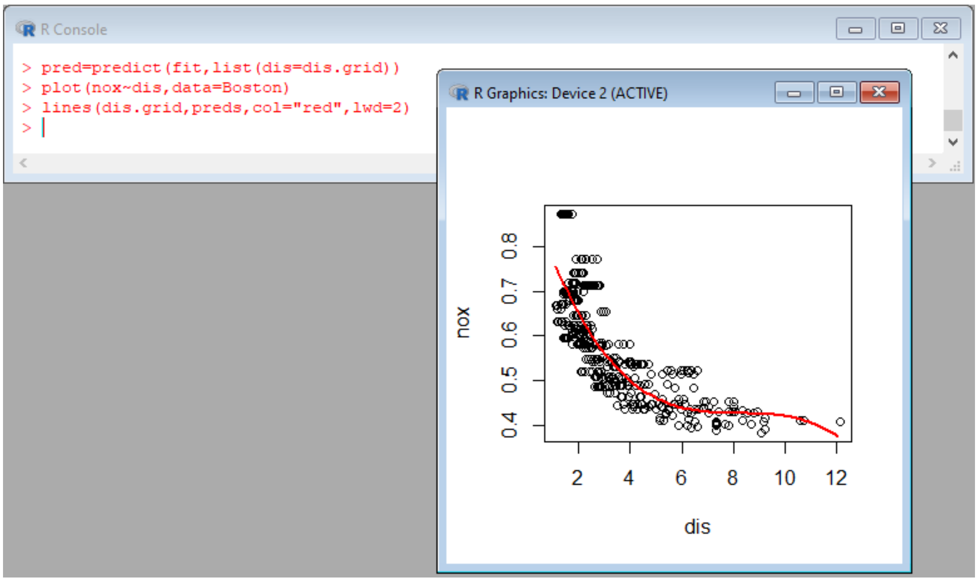
На графіку можна побачити, що оптимальний степінь полінома дорівнює 4.

4.4 Використовуючи функцію bs(), пристосувала сплайн регресію для прогнозування nox за допомогою dis:



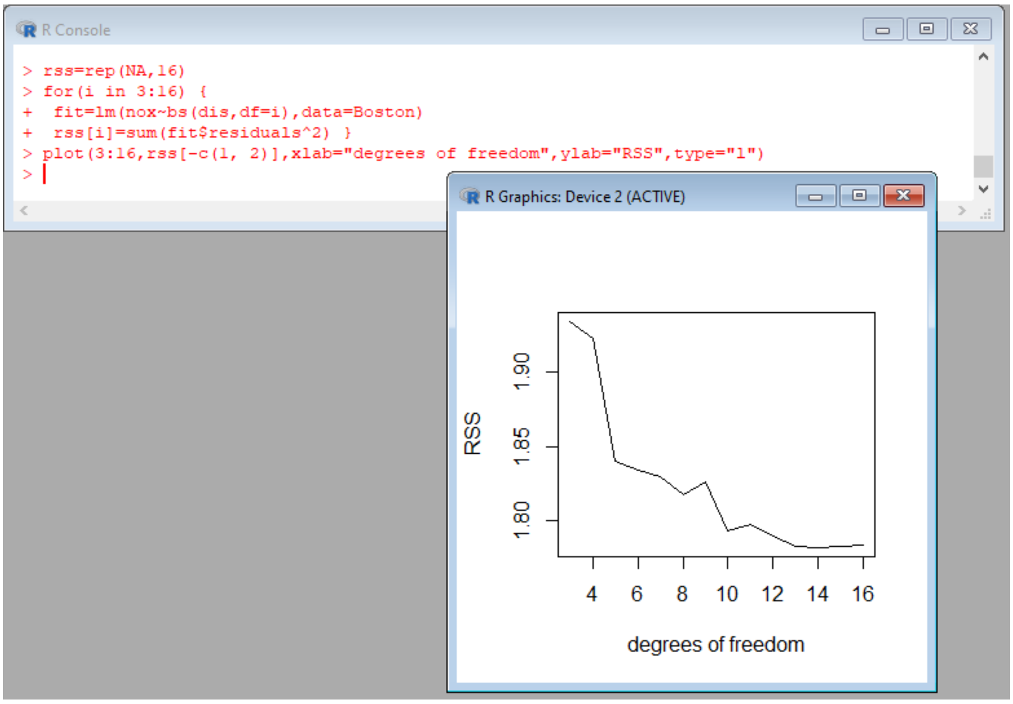
```
R Console
> library(splines)
> fit=lm(nox~bs(dis,knots=c(4,7,11)),data=Boston)
> summary(fit)

Call:
lm(formula = nox ~ bs(dis, knots = c(4, 7, 11)), data = Boston)

Residuals:
      Min        1Q    Median        3Q       Max
-0.124567 -0.040355 -0.008702  0.024740  0.192920

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                    0.73926    0.01331  55.537  < 2e-16 ***
bs(dis, knots = c(4, 7, 11))1 -0.08861    0.02504  -3.539  0.00044 ***
bs(dis, knots = c(4, 7, 11))2 -0.31341    0.01680 -18.658  < 2e-16 ***
bs(dis, knots = c(4, 7, 11))3 -0.26618    0.03147  -8.459 3.00e-16 ***
bs(dis, knots = c(4, 7, 11))4 -0.39802    0.04647  -8.565  < 2e-16 ***
bs(dis, knots = c(4, 7, 11))5 -0.25681    0.09001  -2.853  0.00451 **
bs(dis, knots = c(4, 7, 11))6 -0.32926    0.06327  -5.204 2.85e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.06185 on 499 degrees of freedom
Multiple R-squared:  0.7185,    Adjusted R-squared:  0.7151
F-statistic: 212.3 on 6 and 499 DF,  p-value: < 2.2e-16

>
```

Побудувала графік отриманої моделі:



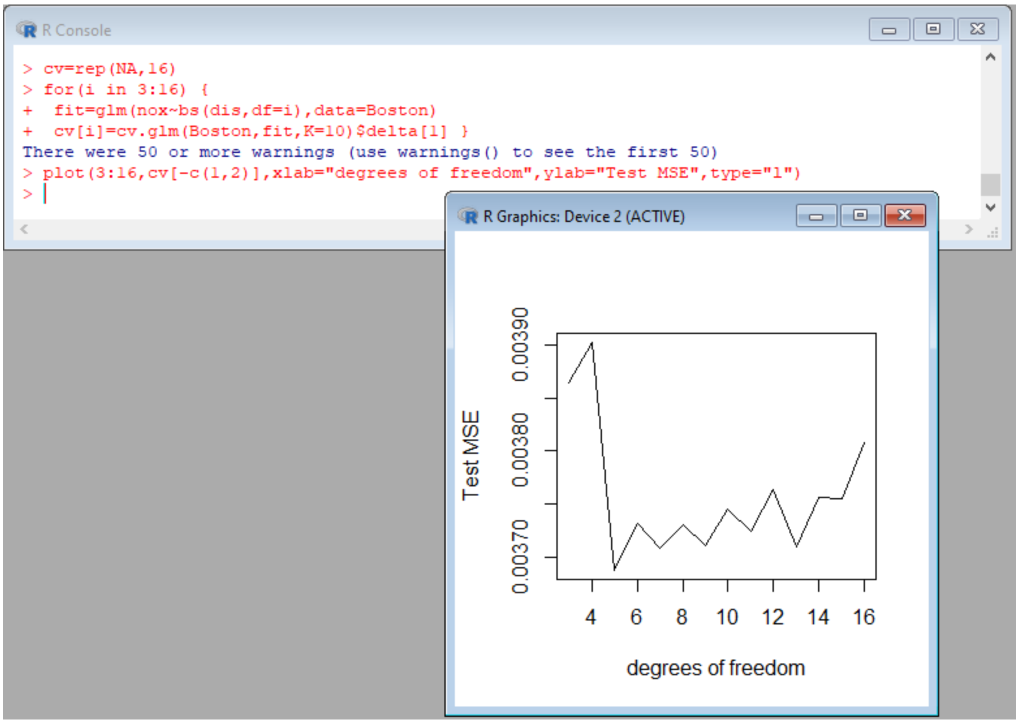
На графіку можна побачити, що у сплайні всі треми значущі.

4.5 Пристосувала сплайн регресію для діапазону ступенів свободи, і побудувала графік результатів:



На графіку можна побачити, що RSS спочатку спадає (до 14), а тоді трохи зростає.

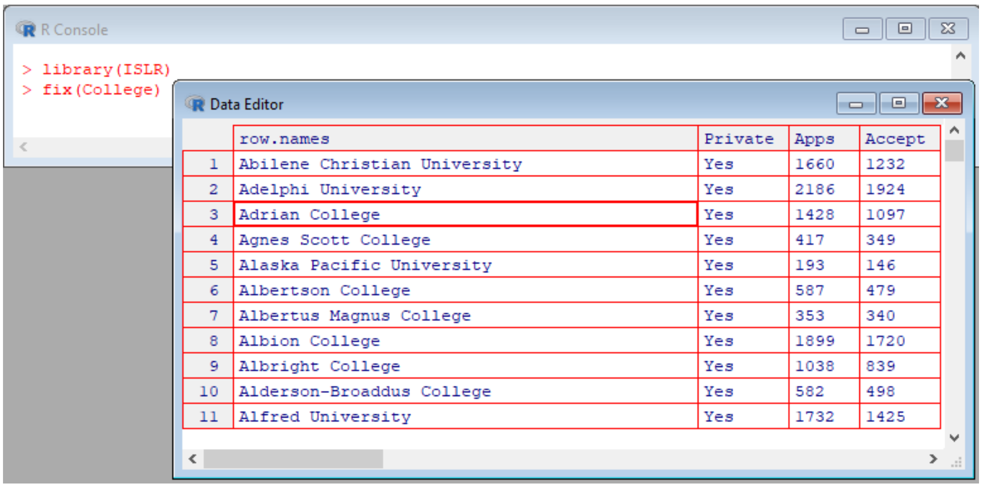
4.6 Використала перехресну перевірку, щоб вибрати найкращий ступінь свободи для сплайн регресії на цих даних:



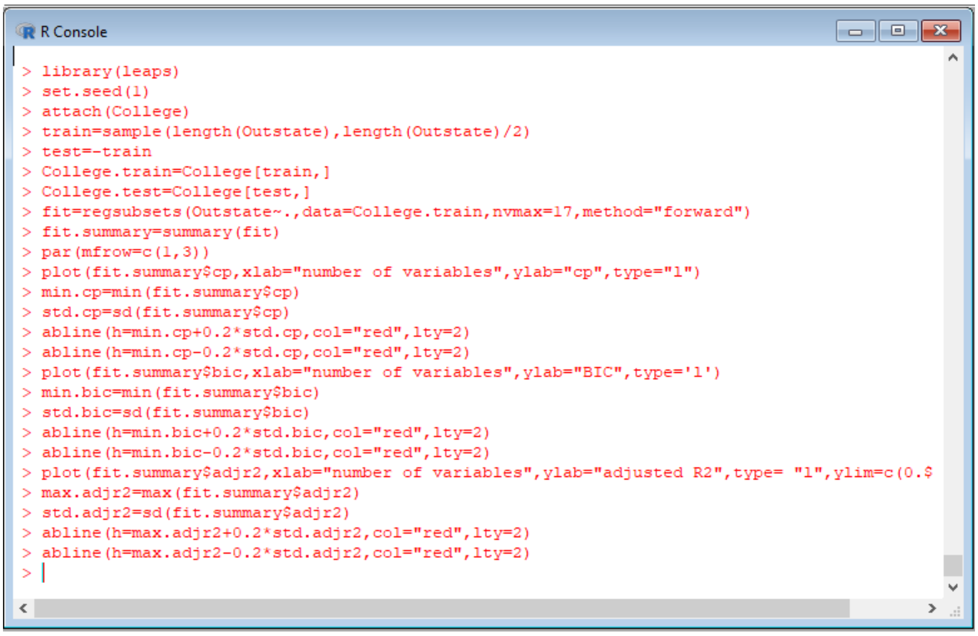
На графіку можна побачити, що мінімальна помилка буде тоді, коли кількість ступенів свободи дорівнює 5.

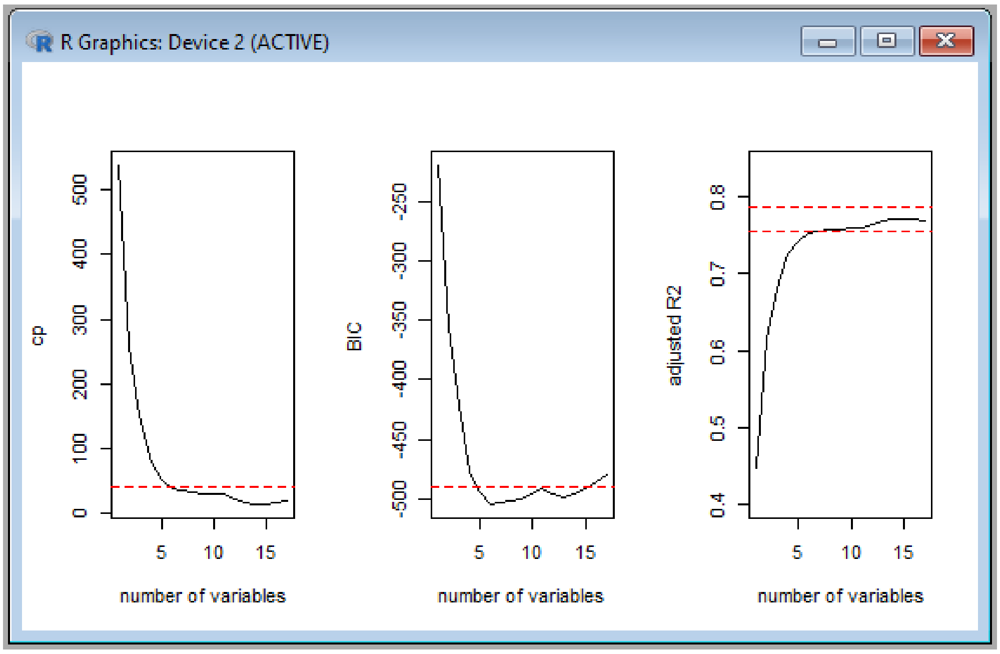
Завдання 5

Проаналізувала набір даних College:

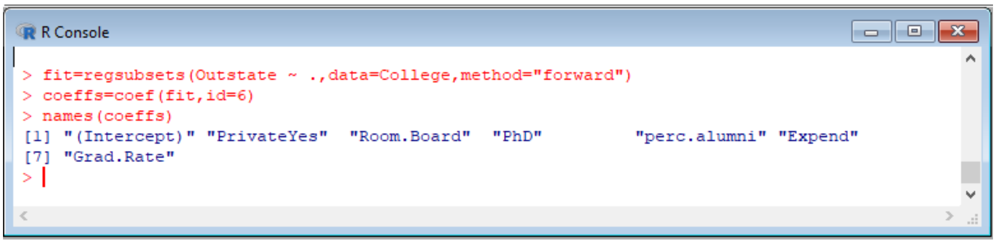


5.1 Розбила дані на навчальний та тестовий набори. Використала Outstate як залежну змінну, а інші змінні як предиктори. Виконала покроковий вибір вперед на навчальному наборі, щоб визначити задовільну модель, яка використовує лише підмножину предикторів:



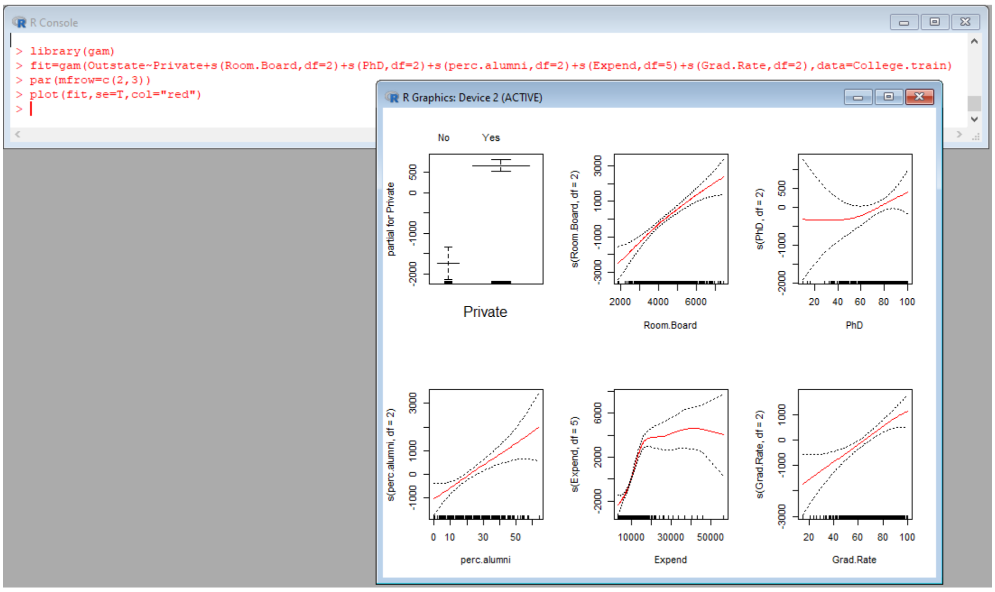


На графіках можна побачити, що мінімальний розмір підмножини дорівнює 6.



```
> fit=regsubsets(Outstate ~ .,data=College,method="forward")
> coeffs=coef(fit,id=6)
> names(coeffs)
[1] "(Intercept)" "PrivateYes"  "Room.Board"  "PhD"         "perc.alumni" "Expend"
[7] "Grad.Rate"
>
```

5.2 Оцінила УАМ модель на навчальних даних, використовуючи Outstate як залежну змінну та ознаки обрані на попередньому кроці як предиктори, побудувала графіки результатів:



5.3 Застосувала модель на тестовому наборі даних:

```
> preds=predict(fit,College.test)
> err=mean((College.test$Outstate-preds)^2)
> err
[1] 3349290
> tss=mean((College.test$Outstate-mean(College.test$Outstate))^2)
> rss=1-err/tss
> rss
[1] 0.7660016
>
```

Можна побачити, що R^2 дорівнює 0.766.

5.4 Для яких змінних, якщо такі є, є докази нелінійності взаємозв'язку з залежною змінною?

```
> summary(fit)

Call: gam(formula = Outstate ~ Private + s(Room.Board, df = 2) + s(PhD,
    df = 2) + s(perc.alumni, df = 2) + s(Expend, df = 5) + s(Grad.Rate,
    df = 2), data = College.train)
Deviance Residuals:
     Min       1Q   Median       3Q      Max
-7402.89 -1114.45   -12.67  1282.69  7470.60

(Dispersion Parameter for gaussian family taken to be 3711182)

    Null Deviance: 6989966760 on 387 degrees of freedom
Residual Deviance: 1384271126 on 373 degrees of freedom
AIC: 6987.021

Number of Local Scoring Iterations: NA

Anova for Parametric Effects
                        Df     Sum Sq    Mean Sq F value    Pr(>F)
Private                  1 1778718277 1778718277 479.286 < 2.2e-16 ***
s(Room.Board, df = 2)    1 1577115244 1577115244 424.963 < 2.2e-16 ***
s(PhD, df = 2)           1  322431195  322431195  86.881 < 2.2e-16 ***
s(perc.alumni, df = 2)   1  336869281  336869281  90.771 < 2.2e-16 ***
s(Expend, df = 5)        1  530538753  530538753 142.957 < 2.2e-16 ***
s(Grad.Rate, df = 2)     1   86504998   86504998  23.309 2.016e-06 ***
Residuals              373 1384271126    3711182
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Anova for Nonparametric Effects
                       Npar Df  Npar F     Pr(F)
(Intercept)
Private
s(Room.Board, df = 2)        1  1.9157    0.1672
s(PhD, df = 2)               1  0.9699    0.3253
s(perc.alumni, df = 2)       1  0.1859    0.6666
```

Завдяки ANOVA можна побачити, що є нелінійний з’язок між Outstate і Expend.

Завдпння 6

6.1 Згенерувала залежну змінну Y і два предиктори $X1$ і $X2$, з $n = 100$:

```
> set.seed(1)
> X1=rnorm(100)
> X2=rnorm(100)
> eps=rnorm(100,sd=0.1)
> Y=7+5*X1+2*X2+eps
>
```

$$Y = \beta0 + \beta1X1 + \beta2X2 + \epsilon$$

6.2 Ініціалізувала оцінку $\beta1$ довільним значенням на свій вибір:

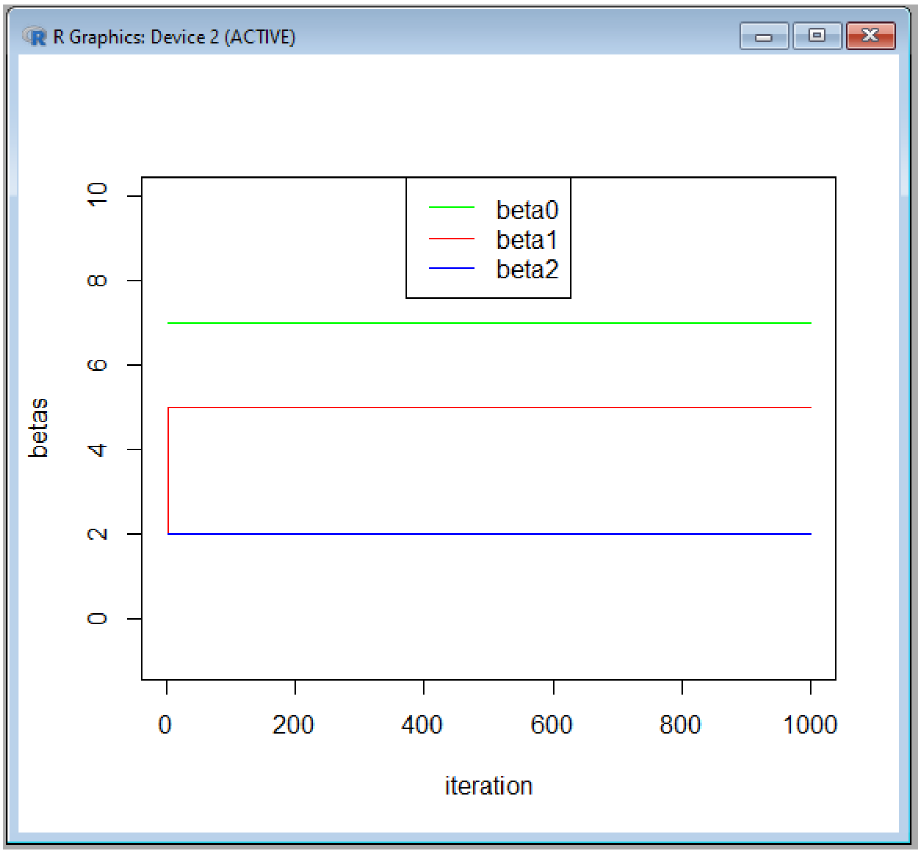
```
> beta0=rep(0,1000)
> beta1=rep(0,1000)
> beta2=rep(0,1000)
> beta1[1]=2
>
```

Нехай $\beta 1$=2.

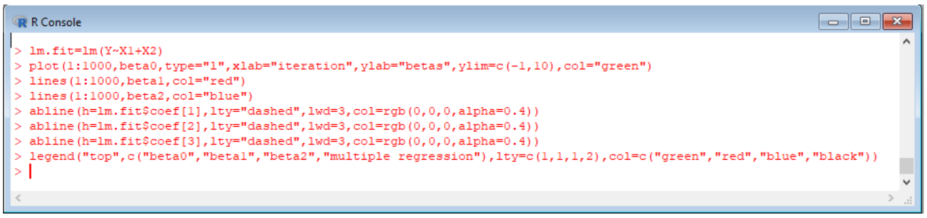
6.3-6.5 Не змінюючи $\beta 1$ оцінила модель $Y - \beta 1X1 = \beta 0 + \beta 2X2 + \varepsilon$. Зафіксувавши оцінку $\beta 2$, оцінила модель $Y - \beta 2X2 = \beta 0 + \beta 1X1 + \varepsilon$. Використала for для організації циклу з повторень кроків 6.3 та 6.4 1,000 разів.

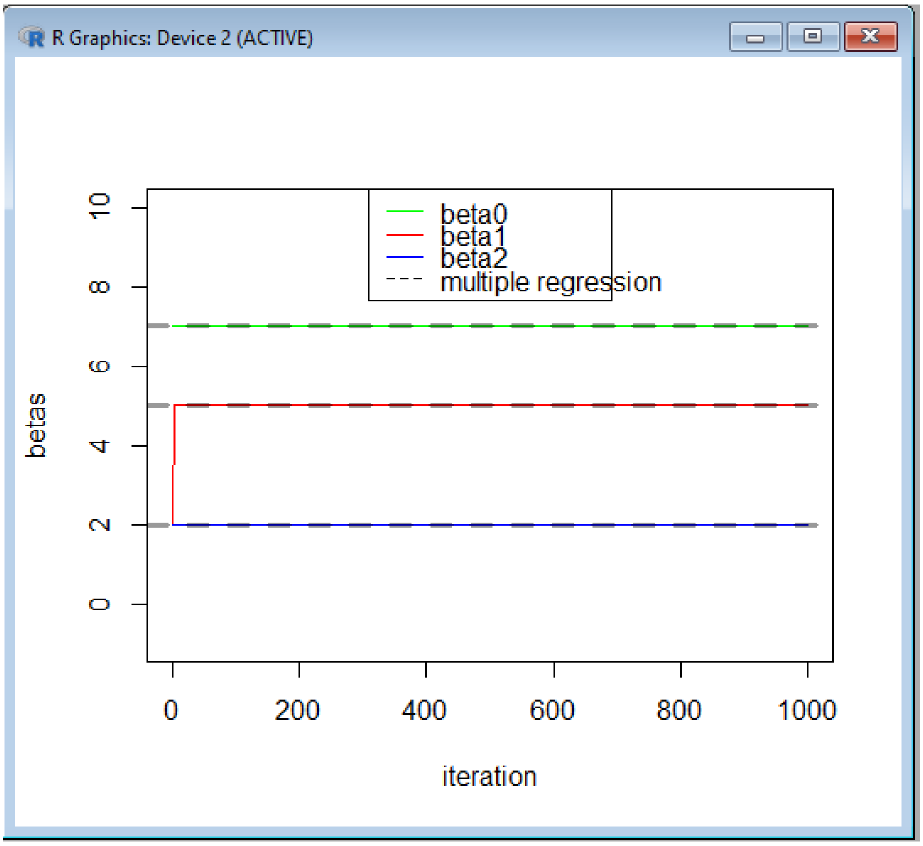
```
> for(i in 1:1000) {
+  a=Y-beta1[i]*X1
+  beta2[i]=lm(a~X2)$coef[2]
+  a=Y-beta2[i]*X2
+  lm.fit=lm(a~X1)
+  if(i<1000) {
+   beta1[i+1]=lm.fit$coef[2] }
+  beta0[i]=lm.fit$coef[1] }
> plot(1:1000,beta0,type="l",xlab="iteration",ylab="betas",ylim=c(-1,10),col="green")
> lines(1:1000,beta1,col="red")
> lines(1:1000,beta2,col="blue")
> legend("top",c("beta0","beta1","beta2"),lty=1,col=c("green","red","blue"))
>
```

Побудувала графіки, на яких відображено ці значення для $\beta 0$, $\beta 1$ і $\beta 2$ різними кольорами:



6.6 Використовуючи функцію abline(), наклала ці значення на графік отриманий в 6.5:



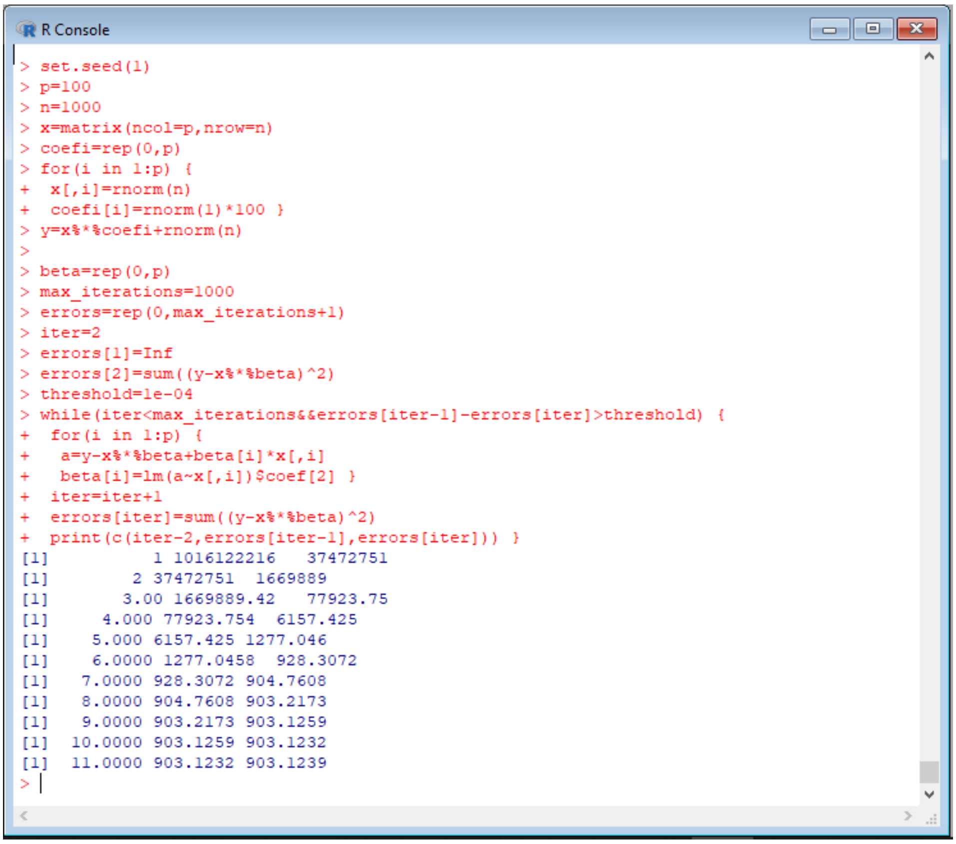


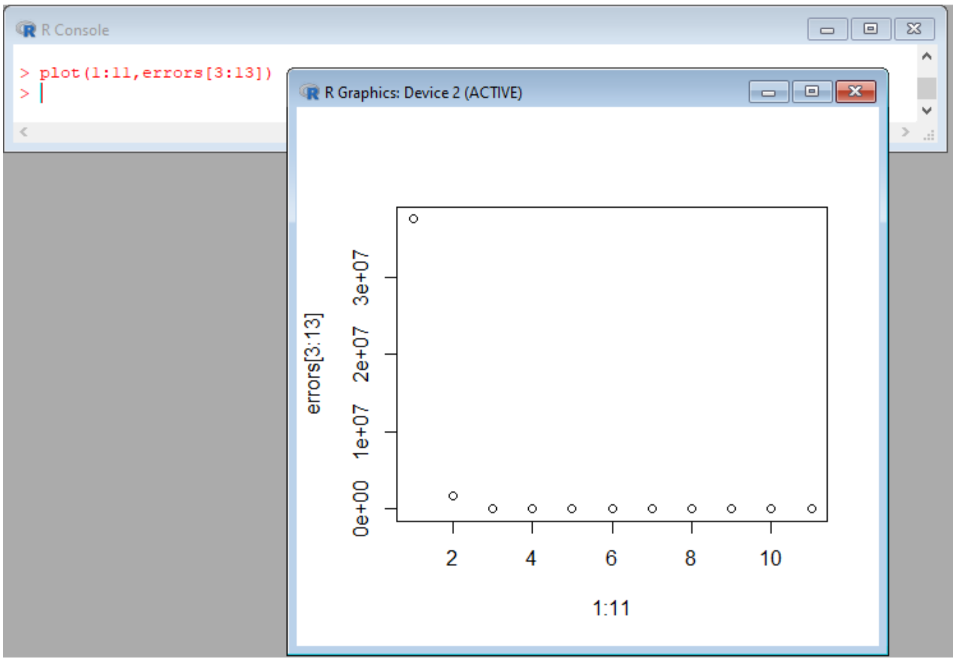
За допомогою пунктирних ліній можна побачити, що новий графік накладається на графік отриманий в 6.5, а це означає, що коефіцієнти однакові.

6.7 Достатньо було однієї ітерації підгонки для отримання «доброго» наближення до оцінок коефіцієнтів множинної регресії.

Завдання 7

Показала, що у випадку p = 100 можна отримати оцінки коефіцієнтів множинної регресії повторно застосовуючи метод підгонки:





Можна побачити, що отримати хорошу апроксимацію, можливо за десять ітерацій. А на 11-ій зростає значення похибки.