Проект по Приложна статистика

Силви-Мария Гюрова ПМ 3 курс , 31341, 2 група

Виктория Динкова ПМ 3 курс , 31343, 2 група

Анкетирахме 52 студенти от ФМИ с цел да разберем дали са доволни от образованието, което факултета им предоставя.

Въпросите:

Анкета за образованието.. или ?

1. "Колко % смятате, че висшето образование е необходимо?",
2. "Колко % смятате, че висшето образование в България е на ниво?",
3. "Колко % смятате, че са хората, които работят същата специалност, която са учили?",
4. "Колко % смятате, че е компетентността на преподавателите в българските университети?",
5. "Колко % смятате, че сте усвоили знанията, които са ви преподадени в университета?",
6. "Колко % смятате, че учите с желание избраната от Вас специалност?",
7. "Колко % смятате, че ще работите по специалността си в бъдеще?",
8. "Колко % смятате, че ще сте доволни от полученото образование след завършването Ви?"

С получените резултати решихме да направим линейна регресия , за да разберем кои компоненти са зависими. Като заначало ще сматаме ,че въпросите от 1 до 7 са отлици,а 8 е предиктор. В хода на анализа може нещо да се промени.

Код на R:

```
library (MASS)
library (ISLR)
library(UsingR)
install.packages("ISLR")
##install.packages("xlsx")
##install.packages("RODBC")
getwd()
results<-read.csv(file="anketa_1.csv",TRUE,sep=",")
results
results$X1 = as.numeric(sub("%","", results$X1))/100
results$X2 = as.numeric(sub("%","", results$X2))/100
results$X3 = as.numeric(sub("%","", results$X3))/100
results$X4 = as.numeric(sub("%","", results$X4))/100
results$X5 = as.numeric(sub("%","", results$X5))/100
results$X6 = as.numeric(sub("%","", results$X6))/100
results$X7 = as.numeric(sub("%","", results$X7))/100
results$X8 = as.numeric(sub("%","", results$X8))/100
results[,2:9]
```

Превръщаме % в десетични дроби и отпечатваме матричния вид на данните.

```
Console ~/
> results$X2 = as.numeric(sub("%","", results$X2))/100
> results$X3 = as.numeric(sub("%","", results$X3))/100
> results$X4 = as.numeric(sub("%","", results$X4))/100
> results$X5 = as.numeric(sub("%","", results$X5))/100
> results$X6 = as.numeric(sub("%","", results$X6))/100
> results$X7 = as.numeric(sub("%","", results$X7))/100
> results$X8 = as.numeric(sub("%","", results$X8))/100
> results[,2:9]
    X1  X2  X3  X4  X5  X6  X7  X8
1  1.0 0.8 0.6 0.8 0.8 1.0 0.8 1.0
2  0.6 0.4 0.6 0.4 0.4 0.8 1.0 0.6
3  0.8 0.6 0.2 0.4 0.6 1.0 1.0 0.8
4  0.8 0.2 0.2 0.2 0.4 0.6 1.0 0.6
5  0.6 0.6 0.4 0.4 0.8 1.0 1.0 0.6
6  0.8 0.4 0.6 0.6 0.4 0.4 0.6 0.6
7  0.6 0.4 0.6 0.4 0.6 0.6 0.8 0.4
8  0.8 0.4 0.4 0.6 0.8 1.0 1.0 0.8
9  1.0 0.2 0.2 0.4 0.4 1.0 0.6 0.4
10 0.8 0.6 0.6 0.6 0.4 0.6 1.0 0.4
11 0.6 0.4 0.6 0.6 0.4 0.4 1.0 0.8
12 0.8 0.6 0.4 0.6 0.8 1.0 0.8 0.6
13 0.8 0.8 0.4 0.8 0.6 0.8 1.0 0.6
14 0.6 0.6 0.6 0.8 0.6 0.8 0.8 0.8
15 0.8 0.6 0.2 0.0 1.0 1.0 1.0 0.8
16 0.8 0.8 0.4 0.8 0.6 0.6 0.4 0.6
```

Правим Shapiro.test() с цел да проверим H_0: данните да нормално разпределени

Shapiro.test()

	X1	X2	X3	X4	X5	X6	X7	X8
W=	0.86248	0.87901	0.8405	0.85029	0.90226	0.87223	0.75148	0.87805
p-value	2.49e-05	7.755e-05	6.115e-06	1.127e-05	0.0004394	4.824e-05	5.148e-08	7.243e-05

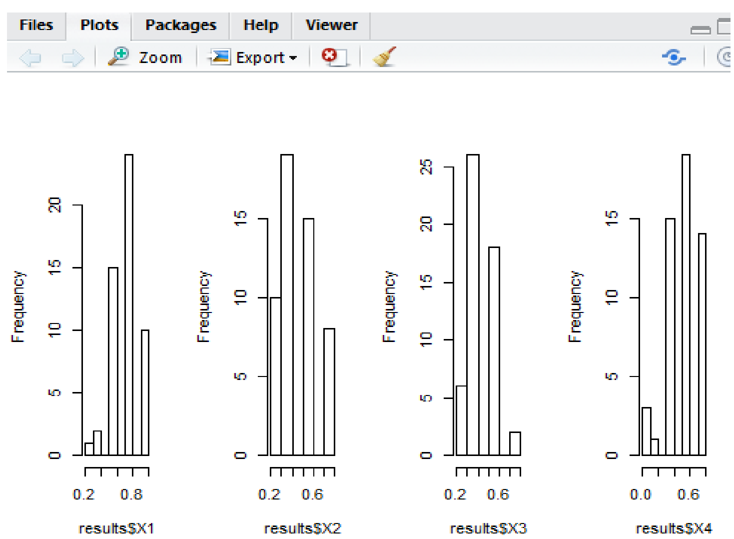
```
> shapiro.test(results$X8)

        Shapiro-Wilk normality test

data:  results$X8
W = 0.87805, p-value = 7.243e-05

> ## Shapiro-Wilk normality test
> ##data:  results$X8
> ##W = 0.87805, p-value = 7.243e-05
> shapiro.test(results$X1)

        Shapiro-Wilk normality test

data:  results$X1
W = 0.86248, p-value = 2.49e-05

> ##Shapiro-Wilk normality test
> ##data:  results$X1
> ##W = 0.86248, p-value = 2.49e-05
> shapiro.test(results$X2)

        Shapiro-Wilk normality test

data:  results$X2
W = 0.87901, p-value = 7.755e-05
```

```
> ##Shapiro-Wilk normality test
> ##data:  results$X4
> ##W = 0.85029, p-value = 1.127e-05
> shapiro.test(results$X5)

        Shapiro-Wilk normality test

data:  results$X5
W = 0.90226, p-value = 0.0004394

> ##Shapiro-Wilk normality test
> ##data:  results$X5
> ##W = 0.90226, p-value = 0.0004394
> shapiro.test(results$X6)

        Shapiro-Wilk normality test

data:  results$X6
W = 0.87223, p-value = 4.824e-05

> ##Shapiro-Wilk normality test
> ##data:  results$X6
> ##W = 0.87223, p-value = 4.824e-05
```

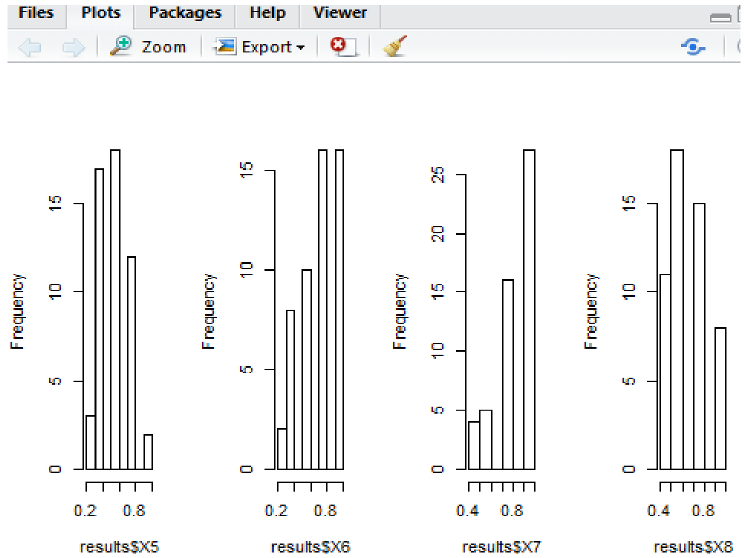
Забелязваме,че както за отлика , така и за предикторите p-value <0.05 ,т.е отхвърляме H_0.

Решихме да видим как нашите данни изглеждта в хистограма

```
par(mfrow=c(1,4))
hist(results$X1,main="")
hist(results$X2,main="")
hist(results$X3,main="")
hist(results$X4,main="")
```



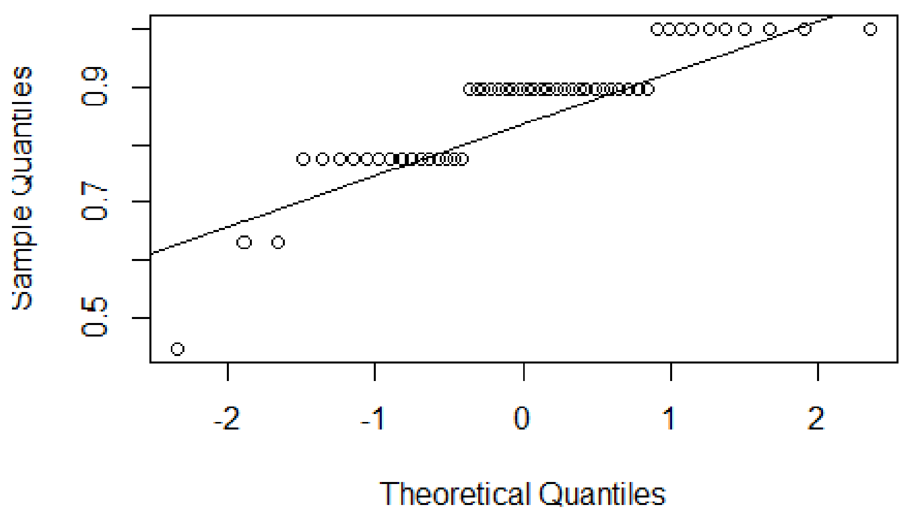
```
> par(mfrow=c(1,4))
> hist(results$X5,main="")
> hist(results$X6,main="")
> hist(results$X7,main="")
> hist(results$X8,main="")
>
```



От хистограмите се вижда,че не са нормално разпредлени. Решихме да направим няколко трансформации , но и те не дават желания резултат.

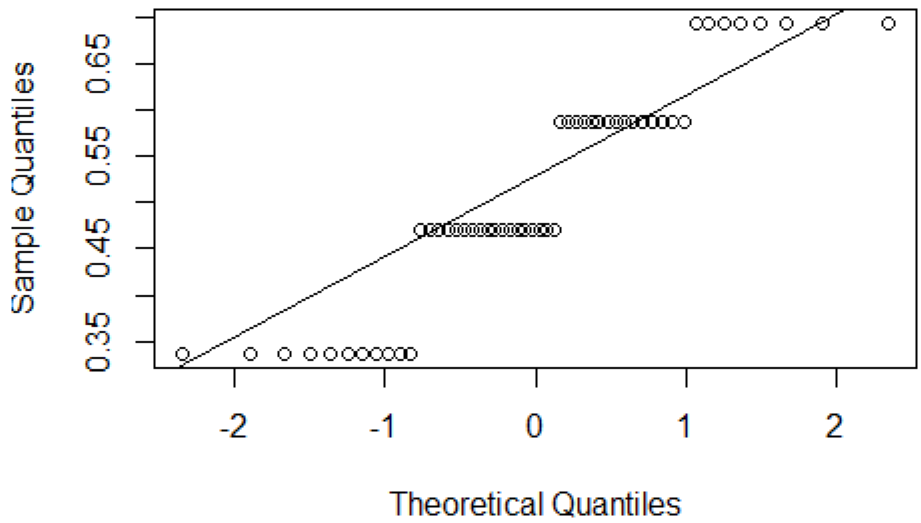
за предиктор X1

```
qqnorm(sqrt(results$X1),main="")
qqline(sqrt(results$X1),main="")
```



за отлик X8

```
qqnorm(log(results$X8+1),main="")
qqline(log(results$X8+1),main="")
```



Забелязваме, че трансформации не дават желания резултат. Получават се тежки опашки.

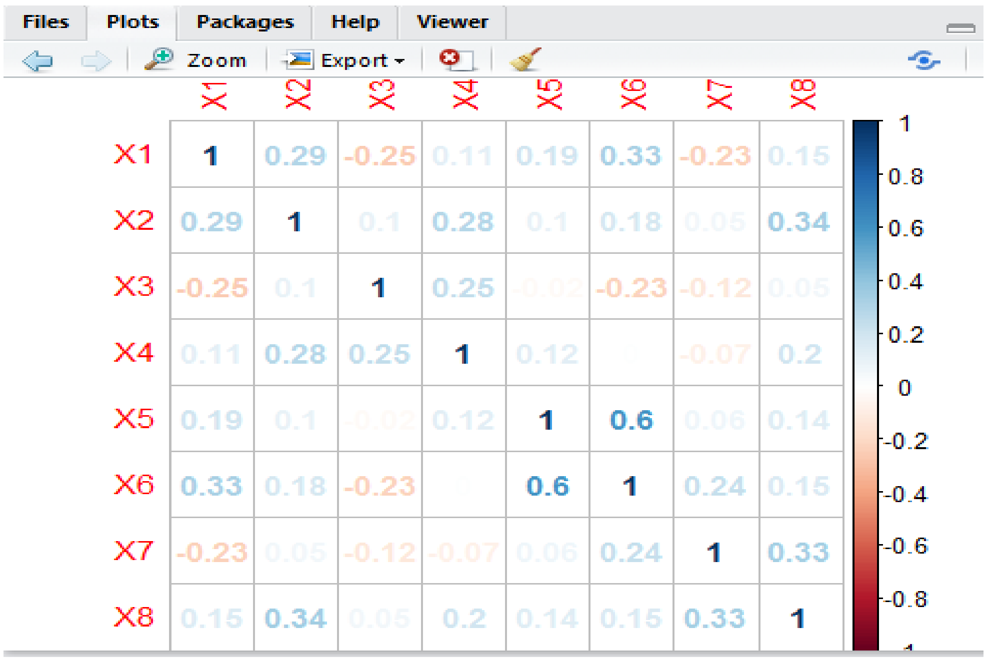
Правим корелационна матрица.

```
install.packages("Hmisc")
library(Hmisc)
library(corrplot)
rcorr(results,type="spearman")
rcorr(as.matrix(results[,2:9]))
```

```
> rcorr(as.matrix(results[,2:9]))
      X1   X2    X3    X4    X5    X6    X7   X8
X1  1.00 0.29 -0.25  0.11  0.19  0.33 -0.23 0.15
X2  0.29 1.00  0.10  0.28  0.10  0.18  0.05 0.34
X3 -0.25 0.10  1.00  0.25 -0.02 -0.23 -0.12 0.05
X4  0.11 0.28  0.25  1.00  0.12  0.00 -0.07 0.20
X5  0.19 0.10 -0.02  0.12  1.00  0.60  0.06 0.14
X6  0.33 0.18 -0.23  0.00  0.60  1.00  0.24 0.15
X7 -0.23 0.05 -0.12 -0.07  0.06  0.24  1.00 0.33
X8  0.15 0.34  0.05  0.20  0.14  0.15  0.33 1.00

n= 52


P
   X1     X2     X3     X4     X5     X6     X7     X8    
X1        0.0341 0.0682 0.4359 0.1702 0.0177 0.0942 0.2921
X2 0.0341        0.4911 0.0430 0.4808 0.2064 0.7214 0.0129
X3 0.0682 0.4911        0.0785 0.8689 0.1014 0.3793 0.7247
X4 0.4359 0.0430 0.0785        0.3899 0.9729 0.5976 0.1598
X5 0.1702 0.4808 0.8689 0.3899        0.0000 0.6595 0.3360
X6 0.0177 0.2064 0.1014 0.9729 0.0000        0.0902 0.2786
X7 0.0942 0.7214 0.3793 0.5976 0.6595 0.0902        0.0165
X8 0.2921 0.0129 0.7247 0.1598 0.3360 0.2786 0.0165       
```

Правим коварационна матрица с цел да видим дали имаме зависимости между отделните вектори данни. Ако $|cor(x,y)|$ >0.6 , можем да твърдим ,че имаме зависимост между x и y.

```
install.packages(corrplot)
M<-cor(results[,2:9])
corrplot(M, method="number")
```



От таблицата виждаме ,че има зависимост между въпрос 5 и 6. Правим

cor.test(*,* ,method=”spearman”), за да определим дали наистина са зависими .

```
> cor.test(results$X5, results$X6, method="spearman")

        Spearman's rank correlation rho

data:  results$X5 and results$X6
S = 9034, p-value = 1.269e-06
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.6143619
```

Виждаме ,че RHO е 0.6143619 > 0.6 => имаме зависимост. Ще разглеждаме линейна регресия относно тези два въпроса.

```
> lm.fit=lm(results$X6~results$X5)
> lm.fit

Call:
lm(formula = results$X6 ~ results$X5)

Coefficients:
(Intercept)   results$X5
     0.3229       0.7251

> summary(lm.fit)

Call:
lm(formula = results$X6 ~ results$X5)

Residuals:
     Min       1Q   Median       3Q      Max
-0.50300 -0.15798  0.04202  0.09700  0.38703

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.32293    0.08346   3.869 0.000317 ***
results$X5   0.72509    0.13808   5.251  3.1e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1914 on 50 degrees of freedom
Multiple R-squared:  0.3555,    Adjusted R-squared:  0.3426
F-statistic: 27.58 on 1 and 50 DF,  p-value: 3.102e-06
```

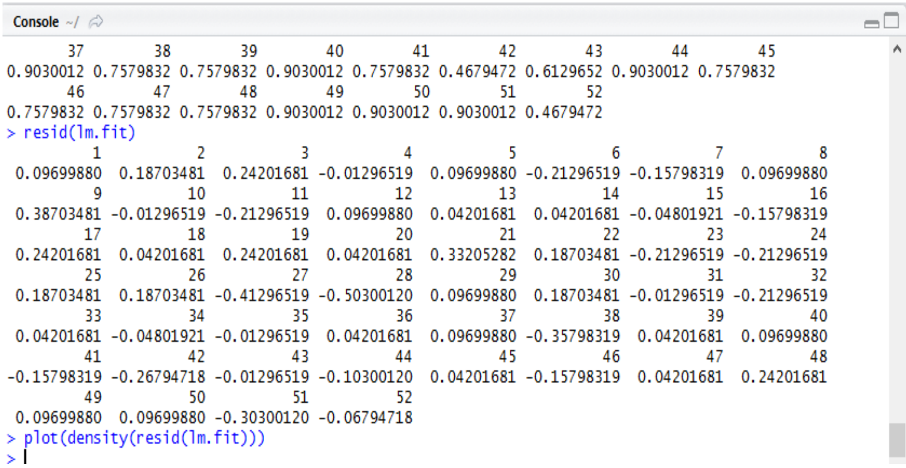
Виждаме ,че ни е пресметнат коефициента пред resultsX5.($\beta = 0.7251$).

$$\underline{results\$X6 = 0.7251 * results\$X5 + 0.3229}$$

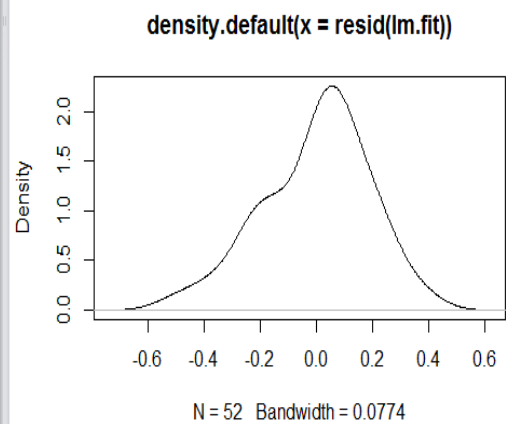
По-голяма информация получаваме от summary(). Резулатът включва по-голям набор от параметри като оценките на параметрите и стандартни грешки, както и остатъчната стандартна грешка и множествена R-квадрат. В една проста линейна регресия, R-квадрат е квадрата на съответствието между Y и X. **($0 \leq R^2 \leq 1$).**

Една основна разлика между R-squared и adjusted R-squared е това ,че R-squared предполага ,че всички независими променливи в модела обясняват изменението в зависимата променлива. Дава процентите на обяснението, ако всички независими променливи в модела засягат зависимата променллива, докато adjusted R-squared дава процента на изменение ,обяснявайки само за тези независими променливи ,които действителност засягат зависимата променлива.

Функцията resid(lm.fit) констроира лист от остатъци. Правим графика на гъстотата тия остатъци.

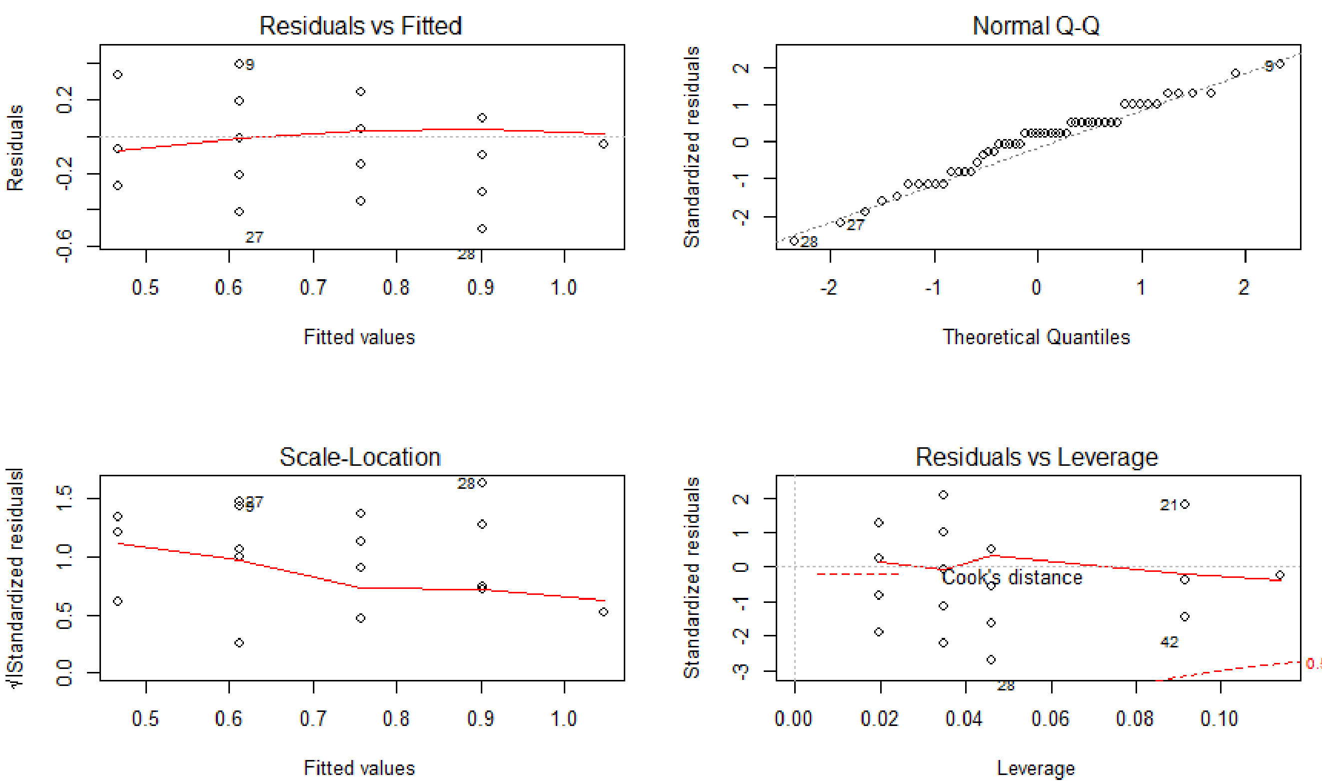


```
Console ~/
       37        38        39        40        41        42        43        44        45
0.9030012 0.7579832 0.7579832 0.9030012 0.7579832 0.4679472 0.6129652 0.9030012 0.7579832
       46        47        48        49        50        51        52
0.7579832 0.7579832 0.7579832 0.9030012 0.9030012 0.9030012 0.4679472
> resid(lm.fit)
          1           2           3           4           5           6           7           8
 0.09699880  0.18703481  0.24201681 -0.01296519  0.09699880 -0.21296519 -0.15798319  0.09699880
          9          10          11          12          13          14          15          16
 0.38703481 -0.01296519 -0.21296519  0.09699880  0.04201681  0.04201681 -0.04801921 -0.15798319
         17          18          19          20          21          22          23          24
 0.24201681  0.04201681  0.24201681  0.04201681  0.33205282  0.18703481 -0.21296519 -0.21296519
         25          26          27          28          29          30          31          32
 0.18703481  0.18703481 -0.41296519 -0.50300120  0.09699880  0.18703481 -0.01296519 -0.21296519
         33          34          35          36          37          38          39          40
 0.04201681 -0.04801921 -0.01296519  0.04201681  0.09699880 -0.35798319  0.04201681  0.09699880
         41          42          43          44          45          46          47          48
-0.15798319 -0.26794718 -0.01296519 -0.10300120  0.04201681 -0.15798319  0.04201681  0.24201681
         49          50          51          52
 0.09699880  0.09699880 -0.30300120 -0.06794718
> plot(density(resid(lm.fit)))
>
```



Построяваме графика за резидиумите (остатъци).

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



Residuals vs. fitted- Това е графика, която прави графика на fitted стойностите на ($\hat{y}$) срещу остатъците. Трябва да наблюдаваме линията y=0.Червената линия е загладена крива, която минава през действителните остатъци и тя е сравнително равна тук

,минаваща близо до сивата линия. Виждаме ,че в графиката имаме и номерирани точки (номер 9 и номер 27). Това не винаги е показател за наличие на проблем.

Normal qqplot- Едно от предположенията на метода на най-малките квадрати е , че грешките са нормлано разпределени.Това показва графиката.Ако грешките (остатъците) са точно нормално разпределени ,те ще лежат върху сивата линия.Може да очакваме някои отклонения ,но те трябва да бъдат малки.Остатъците са нормални, ако тази графика попада в близост до една права линия.

Scale-Location-Тоза графика показва корен квадратен от стандартизираните остатъци. X-оста показва fitted values, Y –оста – the square root от стандализираните остатъци.Всички стойности са положителни. Големите остатъци (положителни или отрицателни) се намират в горната чат на графиката, а малките в долната.Червената линия показва тенденцията. Регресията поема homoscedasticity, остатъците не се променят като функция на x. Ако това е вярно червената линия трябва да е сравнително равна. На графиката виждаме ,че това е така .

Cook's distance- .Тази графика идентифицира точките, които има голямо влияние в линията на регресия.Стандартизираните остатъци са центрирани около нулата и достига 1-2 стандартни отклонения от нулата, симетрично около нулата, тъй като се очаква нормално разпределение. Leverage е мярка за това колко много една данна (точка) влияе на регресията. Тъй като регресията трябва да минава през центъра, точките,които лежат далеч от центъра имат голямо влияние. В резултат, leverage се отразява както на разстояниеот ценъра и изолация на една точка. На графиката се вижда контурните стойности на Кук разстоянието, което измерва колко много регресията ще се промени, ако точката се премахне. Разстояние на Куксе увеличава от leverage и големите остатъци. На графиката, червената линия стои близо до хоризонталната сива линия и никакви точки имат разстояние на Кук (>0.5).

<u>Извод:</u>

1. От всички въпроси от анкетата разбрахме ,че усвояването на знания се дължи на мотивацията дали учиш в желаната от теб специалност.
2. Няма логическа свързаност в отговорите на въпросите от анкетираните.