Универзитет у Београду

Математички факултет

Статистички софтвер 1

Семинарски рад

Илустрација истраживања о наследности висине

децембар 2016.
Београд

Аутор

Марија Костић 286/14

Ментор

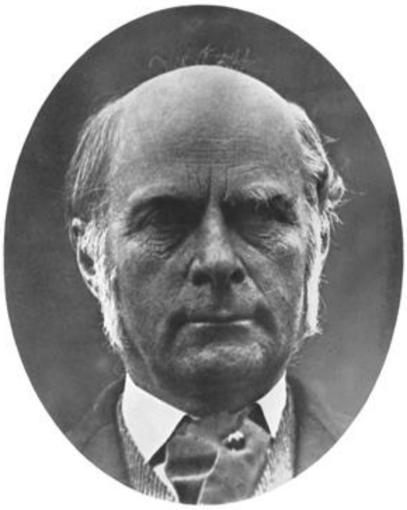
Бојана Тодић

Садржај

УВОД **3**
Опис 3
Формат 4
Приказ уопштених података о бази 4
ГРАФИЧКО ПРЕДСТАВЉАЊЕ ПОДАТАКА **6**
Барплотови 6
Питице 10
Хистограми 12
Kernal density график 14
Боксплотови 16
Scatterplots графици 18
ОБРАДА ПОДАТАКА **21**
Испитивање расподела скупа података 21
Статистички тестови 26
Тестови зависности 27
Још неки тестови зависности 28
ЗАКЉУЧАК **29**

Увод

Тема овог рада базира се на подацима пакета ***mosaicData*** базе ***Galton***. База података добила је име по њеном творцу Сер Френсис Галтону (енглески статистичар , социолог , психолог , анторополог , еугеничар , географ , проналазач, метеоролог и про-генетичар,16. фебруар 1822 - 17. јануар 1911). Галтон је током 19ог века проучавао везу између висине родитеља и деце. Приметио је да висина родитеља није прешла у потпуности на потомство. Мерењем висине стотина људи, проценио је да је регресија на средини.

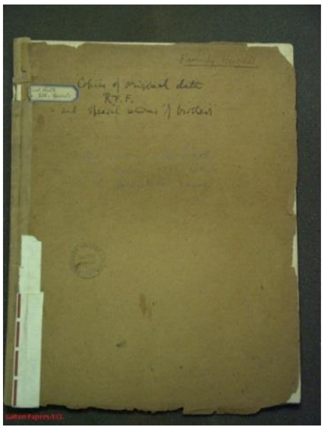
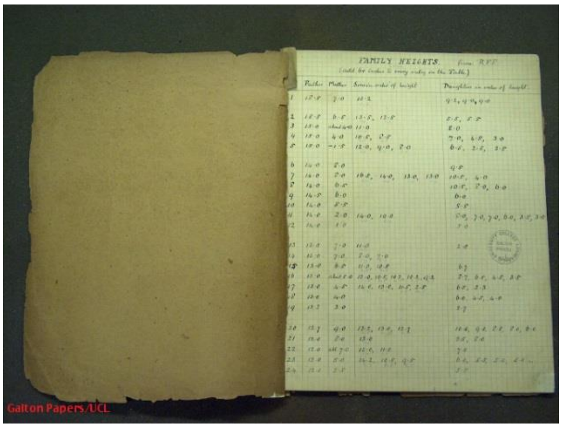



Да би радили са поменутом базом, потребно је прво инсталирати пакет и укључити пакет ***mosaicData*** , а затим и базу ***Galton***.

```
> install.packages("mosaicData")
> library(mosaicData)

> data(Galton)
> attach(Galton)
> help("Galton")
> str(Galton)
```

Опис

База података садржи појединачна запажања за 205 породица (898 деце узраста од 1-15 година) која је Галтон забележо 1880.године.

Формат

База података садржи 898 посматрања у 6 променљивих.
Радови су избрисани за ону децу чија висина није евидентирана нумерички, Галтон је понекад користио уносе као што су "висок", "кратак", "идиотски", "деформисан" и тако даље.
Променљиве су :

1. ***family*** - фактор са нивоима 1-135, 136A, 136-204.
2. ***father*** - висина оца (у инчима)
3. ***mother*** - висина мајке (у инчима)
4. ***sex*** - пол детета (фактор са нивоима *F* и *M*)
5. ***height*** – висина детета као одрасле особе (у инчима)
6. ***nkids*** - број одрасле деце у породици (или барем број одрасле деце чије је висине Галтон снимио)

Приказ уопштених података о бази

За приказ уопштених података о бази користимо функцију ***summary*** :

```
> summary(Galton)
     family        father          mother       sex        height          nkids       
 185    : 15   Min.   :62.00   Min.   :58.00   F:433   Min.   :56.00   Min.   : 1.000  
 166    : 11   1st Qu.:68.00   1st Qu.:63.00   M:465   1st Qu.:64.00   1st Qu.: 4.000  
 66     : 11   Median :69.00   Median :64.00           Median :66.50   Median : 6.000  
 130    : 10   Mean   :69.23   Mean   :64.08           Mean   :66.76   Mean   : 6.136  
 136    : 10   3rd Qu.:71.00   3rd Qu.:65.50           3rd Qu.:69.70   3rd Qu.: 8.000  
 140    : 10   Max.   :78.50   Max.   :70.50           Max.   :79.00   Max.   :15.000  
 (Other):831                                                                          
```

За ***family*** смо добили да породица са фактором 185 има 15оро деце, породица са фактором 166 има 11оро деце итд.
За ***father*** смо добили да је најмања забележена висина оца 62.00, највећа забележена висина је 78.50, а узорачка средина је 69.23. Медијана је 69.00, а одговарајући квартили су q_1 =68.00 и q_3 =71.00.
За ***mother*** смо добили да је најмања забележена висина мајке 58.00, највећа забележена висина је 70.50, а узорачка средина је 64.08. Медијана је 64.00, а одговарајући квартили су q_1 =63.00 и q_3 =65.50.
За ***sex*** смо добили да број мушке деце 465, а женске 433.
За ***height*** смо добили да је најмања забележена висина детета 56.00, највећа забележена висина је 79.00, а узорачка средина је 66.76. Медијана је 66.50, а одговарајући квартили су q_1 =64.00 и q_3 =69.70.
За ***nkids*** смо добили да најмањи број деце у породици 1, а највећи 15. Медијана је 6,а одговарајући квартили су q_1 =4 и q_3 =8.

Помоћу функције ***tapply*** ћемо израчунати највећу,најмању и средњу вредност висине за сваки пол. То постижемо на следећи начин :

```
> HSmax<-tapply(height,sex,max)
> HSmin<-tapply(height,sex,min)
> HSmeans<-tapply(height,sex,mean)
```

Резултати су следећи :

```
> HSmax
   F    M
70.5 79.0
> HSmin
 F  M
56 60
> HSmeans
       F        M
64.11016 69.22882
```

Ове резултате можемо чувати у листи. Листу правимо помоћу функције ***list*** на следећи начин :

```
> ListaVisina<-list(Najveca_visina_za_svaki_pol_deteta=HSmax,
+                   Najmanja_visina_za_svaki_pol_deteta=HSmin,
+                   Srednja_vrednost_visine_za_svaki_pol_deteta=HSmeans)
```

Позивањем листе добијамо следећи резултат :

```
> ListaVisina
$Najveca_visina_za_svaki_pol_deteta
   F    M
70.5 79.0

$Najmanja_visina_za_svaki_pol_deteta
 F  M
56 60

$Srednja_vrednost_visine_za_svaki_pol_deteta
       F        M
64.11016 69.22882
```

Графичко представљање података

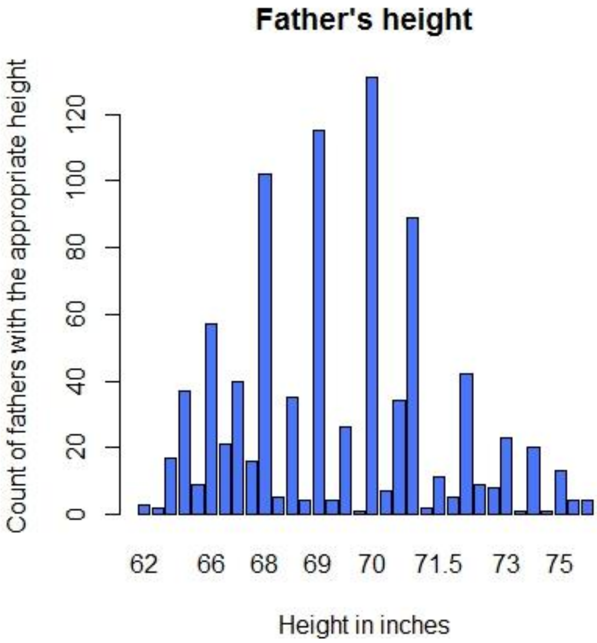
Да бисмо радили са неким подацима, потребно је за почетак стећи представу како ти подаци изгледају.

Барплотови

Представљање података барплотовима постижемо коришћењем функције ***barplot***.

- Барплот висина очева

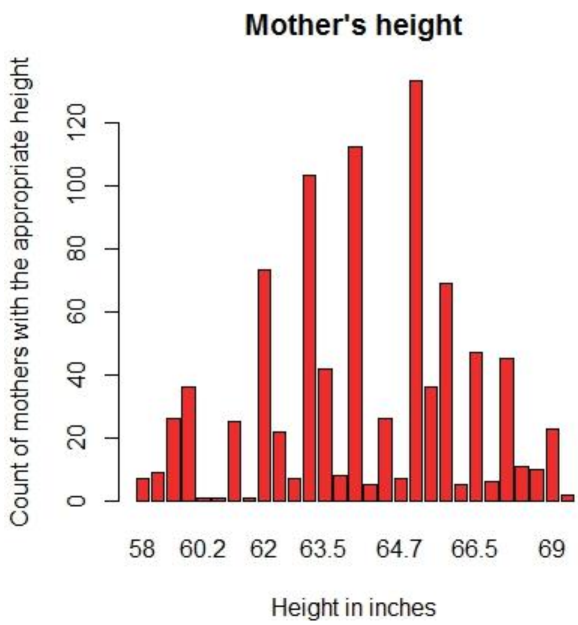
```
> barplot(table(father),main="Father's height",xlab = "Height in inches",
+         ylab = "Count of fathers with the appropriate height",col="royalblue1")
```



Из добијеног барплота види се да највећи број очева има висину око 70 инча.

- Барплот висина мајки

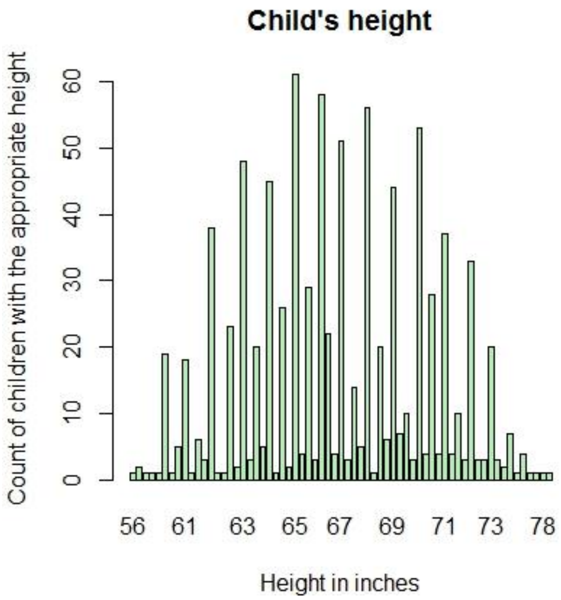
```
> barplot(table(mother),main="Mother's height",xlab = "Height in inches",
 +         ylab = "Count of mothers with the appropriate height",col="firebrick2")
```



Из добијеног барплота види се да највећи број мајки има висину око 65 инча.

- <u>Барплот висина деце</u>

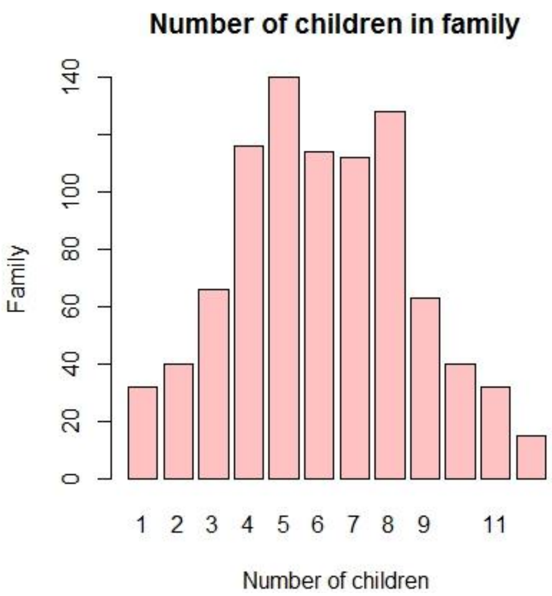
```
> barplot(table(height),main="Child's height",xlab = "Height in inches",
+       ylab = "Count of children with the appropriate height",col="darkseagreen2")
```



Из добијеног барплота види се да највећи број деце има висину око 66 инча.

- Барплот броја деце у породици

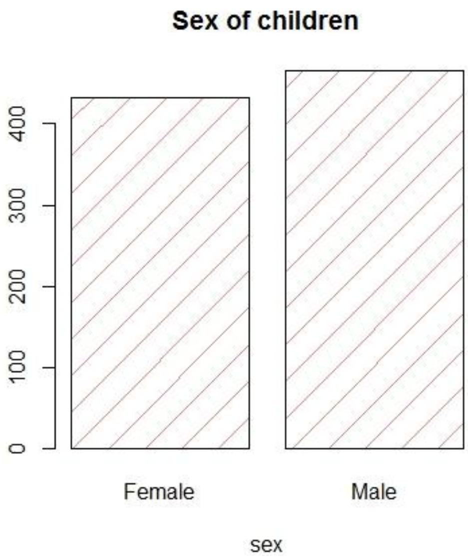
```
> barplot(table(nkids),main="Number of children in family",xlab = "Number of children",
+         ylab = "Family",col="rosybrown1")
```



Из добијеног барплота види се да највећи број породица има петоро деце.

- <u>Барплот броја деце сваког пола</u>

```
> barplot(table(sex),main="Sex of children",xlab = "sex",
+         names.arg = c("Female","Male"),col = "salmon",density=5)
```

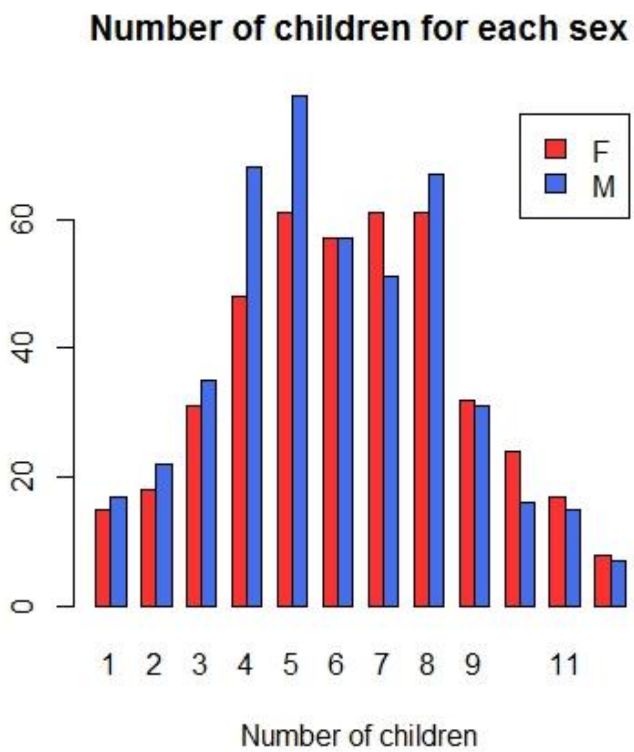


Из добијеног барплота се види да је већи број дечака него девојчица.

Такође, могуће је формирање барплота за груписане променљиве. Најпре ћемо направити табелу која ће садржати податке које желимо да упоредимо.

- <u>Број деце у породици подељен према полу</u>

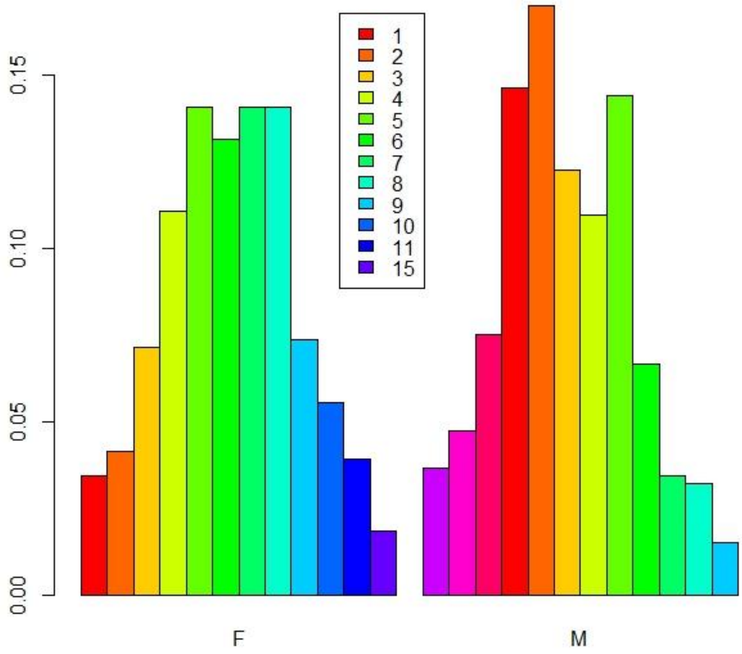
```
> barplotHS <- table(Galton$sex,Galton$nkids)
> barplot((barplotHS), main="Number of children for each sex",xlab="Number of children",
+         col=c("firebrick1","royalblue2"),legend = rownames(barplotHS),beside = TRUE)
```



Из добијеног барплота закључујемо да највише дечака има у породици са петоро деце, а највише девојчица у породицама са петоро,седморо и осморо деце.

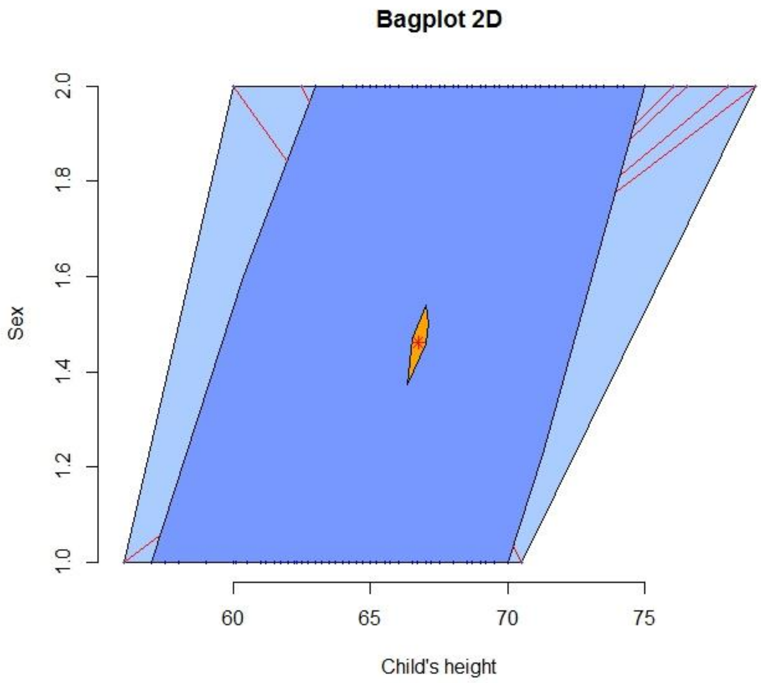
Такође до истих резултата долазимо цртањем барплота на следећи начин :

```
> barplot(prop.table(table(nkids,sex),2),beside = T,col=rainbow(15))
> legend(locator(n=1),legend = rownames(table(nkids,sex)),fill = rainbow(15))
```



Могуће је представљање података и у 2D. За то ће нам бити потребан пакет ***aplpack.***

```
> library(aplpack)
> bagplot(height,sex, xlab="Child's height", ylab="Sex",main="Bagplot 2D")
```



Питице

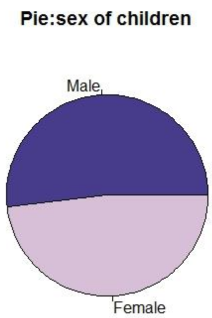
Представљање података питицама постижемо коришћењем функције ***pie***.

- Број деце подељен према полу
  Потребрно је прво да израчунамо број дечака и број девојчица из узорка на следећи начин :

```
> NumberM<-length(subset(Galton,sex=="M")$sex)
> NumberF<-length(subset(Galton,sex=="F")$sex)
```

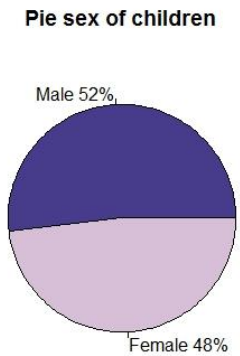
  Сада можемо нацртати питицу :

```
> slices<-c(NumberM,NumberF)
> lbls<-c("Male","Female")
> pie(slices,labels=lbls,main="Pie:sex of children",col = c("slateblue4","thistle"))
```



Оваквим приказом података не знамо колико тачно има дечака, а колико девојчица.Из приказа можемо да закључимо да више има дечака. Да би смо то потврдили додаћемо проценат на постојећи график :

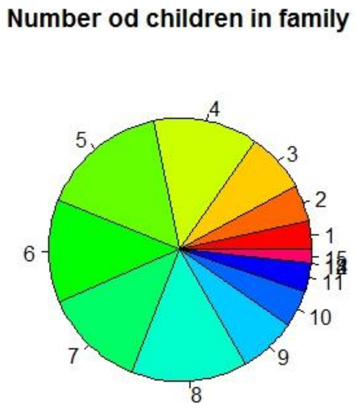
```
> pct <- round(slices/sum(slices)*100)
> lbls <- paste(lbls, pct) # dodavanje procenta
> lbls <- paste(lbls,"%",sep="") # dodavanje %
> pie(slices,labels = lbls, col = c("slateblue4","thistle"),main="Pie sex of children")
```



Сада имамо тачан однос између дечака и девојчица.

- Број деце у породици

```
> slices2<-c()
> for(i in 1:15)
+   slices2[i]<-sum(Galton$nkids==i)
> lbls<-c(1:15)
> pie(slices2,lbls,col=rainbow(15),main ="Number od children in family")
```

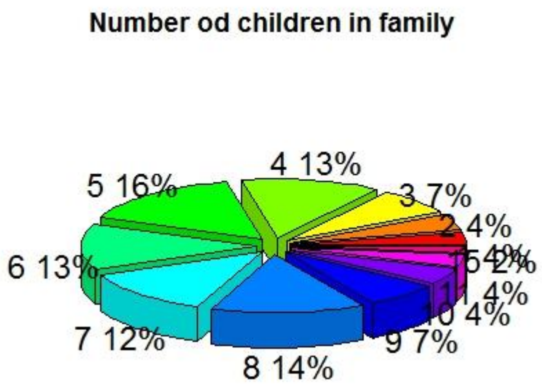


Из овог графика можемо видети да не постоје породице из узорка са дванаесторо, тринаесторо и четрнаесторо деце.
Да би нацртали 3D питицу морамо да инсталирамо и укључимо пакет ***plotrix*** .

```
> install.packages("plotrix")
> library(plotrix)
```

Цртање 3D питице:

```
> slices3<-slices2[slices2!=0]
> lbls3 <- c(1,2,3,4,5,6,7,8,9,10,11,15)
> pct2 <- round(slices3/sum(slices3)*100)
> lbls2 <- paste(lbls3, pct2)
> lbls2<- paste(lbls2, "%", sep = "")
> pie3D(slices3, labels = lbls2, explode = 0.1,main ="Number od children in family",
col = rainbow(12))
```



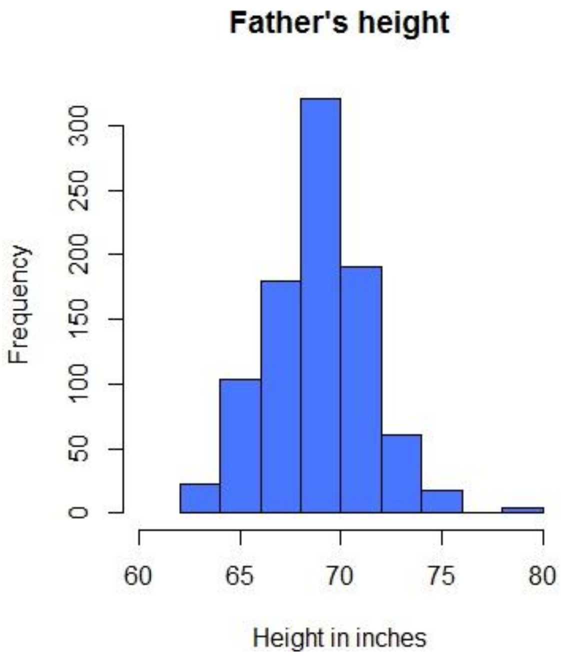
Хистограми

Представљање података хистограмима постижемо коришћењем функције *__hist__*.

На овај начин цртамо следеће хистограме :

- <u>Хистограм висина очева</u>

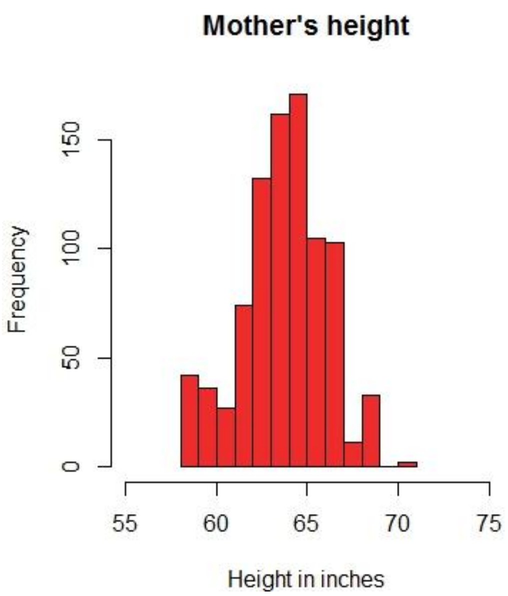
```
> hist(Galton$father, main="Father's height",xlab = "Height in inches",
xlim=c(60,80),col="royalblue1")
```



Из добијеног хистограма види се да највећи број очева има висину око 70 инча.

- <u>Хистограм висина мајки</u>

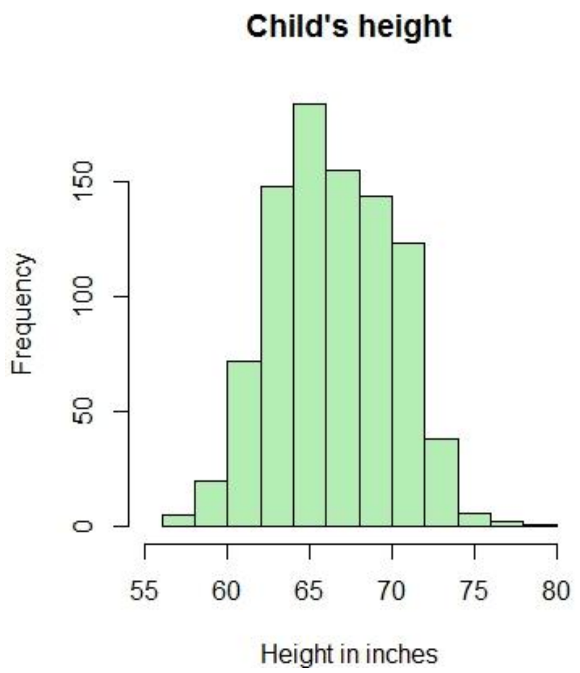
```
> hist(Galton$mother, main="Mother's height",xlab = "Height in inches",xlim=c(55,75),
col="firebrick2")
```



Из добијеног хистограма види се да највећи број мајки има висину око 65 инча.

- <u>Хистограм висина деце</u>

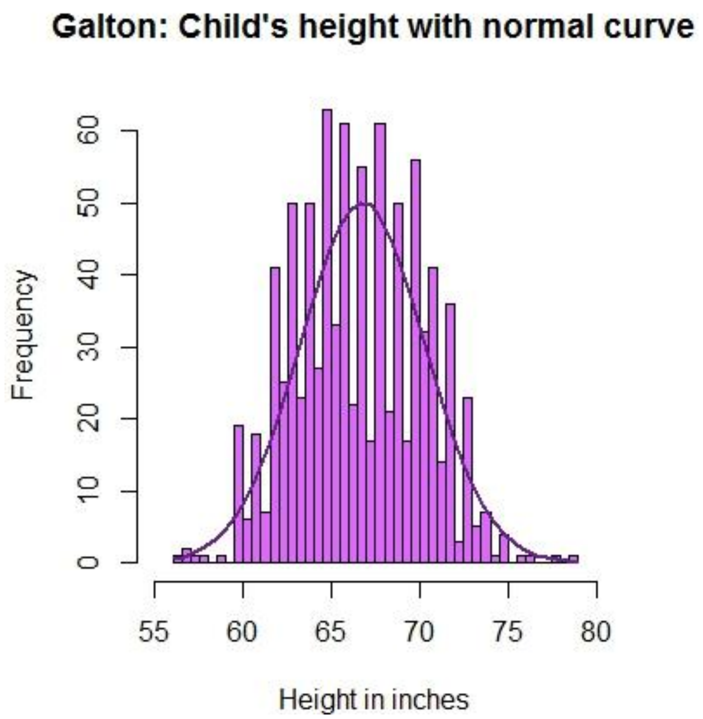
```
> hist(Galton$height, main="Child's height",xlab = "Height in inches",xlim=c(55,80),
col="darkseagreen2")
```



Из добијеног хистограма види се да највећи број деце има висину око 65 инча.

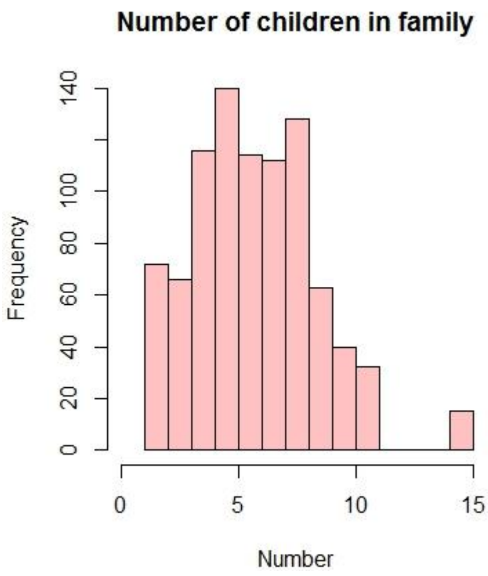
- <u>Хистограм висине деце са нормалном кривом</u>

```
> x <- Galton$height
> h <- hist(x,xlim=c(55,80),breaks = 40, col = "mediumorchid1",
+           xlab = "Height in inches",
+           main = "Galton: Child's height with normal curve")
> xfit <- seq(min(x), max(x), length = 40)
> yfit <- dnorm(xfit, mean = mean(x), sd = sd(x))
> yfit <- yfit * diff(h$mids[1:2]) * length(x)
> lines(xfit, yfit, col = "darkorchid4", lwd =2)
```



- <u>Хистограм броја деце у породици</u>

```
> hist(Galton$nkids, main="Number of children in family",xlab = "Number",xlim=c(0,15),
col="rosybrown1")
```

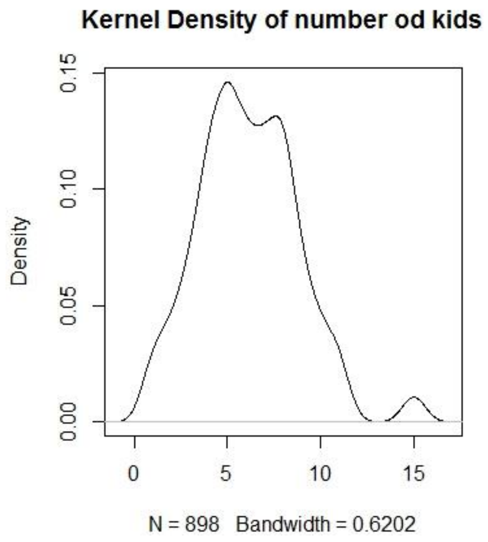


Из добијеног хистограма види се да највећи број породица има петоро деце.

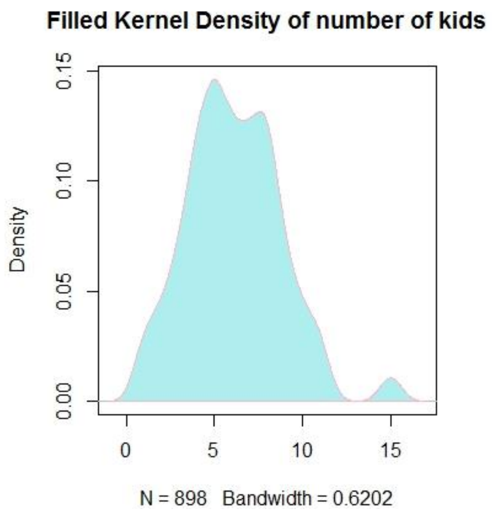
Kernal density график

Много ефикаснији начин за приказ расподеле података су ***Kernal density plots***. Нацртаћемо ***Kernal density plots*** график за променљиву број деце у породици :

```
> d1 <- density(Galton$nkids)
> plot(d1,main="Kernel Density of number od kids")
```




```
> d2 <- density(Galton$nkids)
> plot(d2, main="Filled Kernel Density of number of kids")
> polygon(d2, col="paleturquoise2", border="pink1")
```

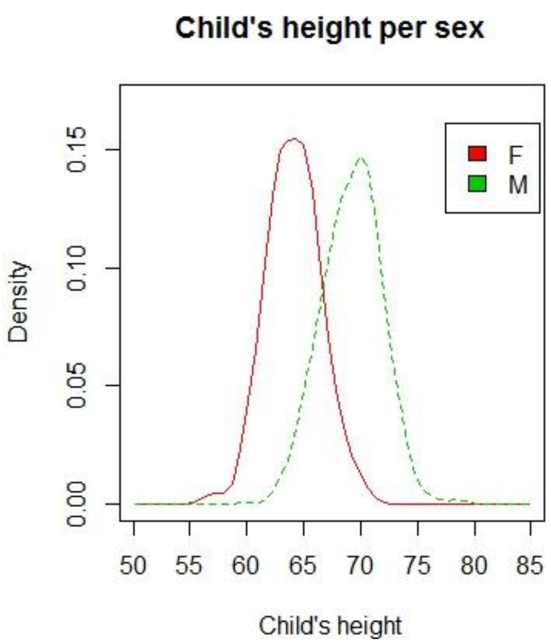


Помоћу функције ***sm.density.compare*** пакета ***sm*** можемо да прикажемо више *Kernal density* графика.
Прво инсталирамо и учитавамо пакет :

```
> install.packages("sm")
> library(sm)
```

Сада цртамо график :

```
> sex.f <- factor(sex, levels= c(0,1),labels = c("Male", "Female") )
> sm.density.compare(height,sex, xlab="Child's height")
> title(main="Child's height per sex")
> # dodavanje legende klikom misa na mesto gde zelimo da se pojavi legenda
> colfill<-c(2:(2+length(levels(sex))))
> legend(locator(1), levels(sex), fill=colfill)
```



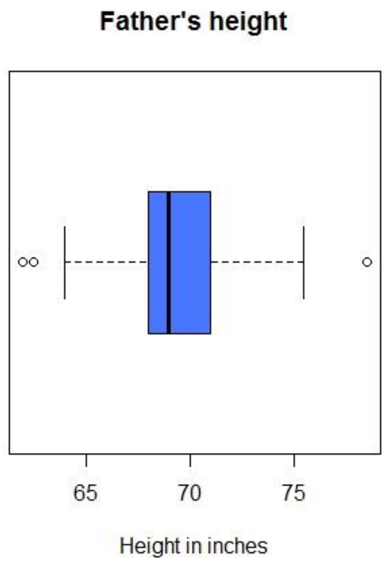
Боксплотови

Коришћењем овог дијаграма могу се пронаћи аутлајери (највероватније погрешни подаци, подаци које треба проверити). Кутија оксплота садржи [25%,75%] узорка,дебела линија на кутији представља медијану, цртице иду до најмање/највеће вредности која упада у дужину од 1.5 кутије од крајева кутије. Аутлајери су представњени кружићима

Представљање података боксплотовима постижемо коришћењем функције ***boxplot***.

- <u>Боксплот висина очева</u>

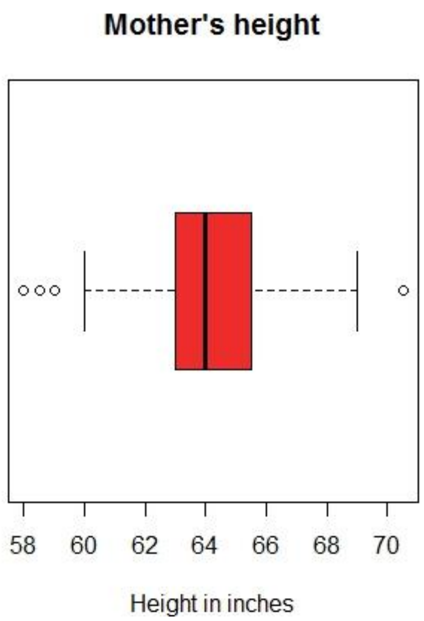
```
> boxplot(father,horizontal=TRUE,main="Father's height",xlab = "Height in inches",col="
royalblue1")
```



На основу графика можемо закључити да је расподела обележја померена удесно (на самом почетку смо добили да је медијана 69.00 а узорачка средња вредност 69.23) и да постоје аутлајери.

- <u>Боксплот висина мајки</u>

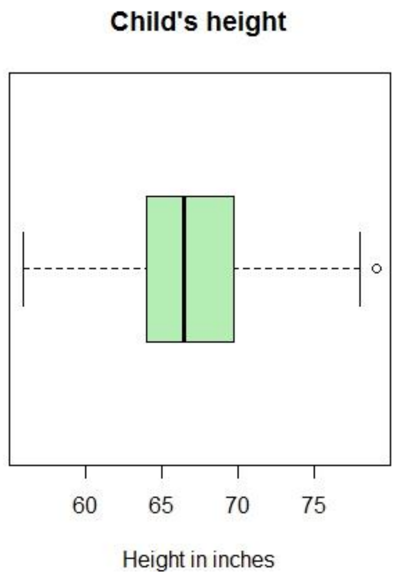
```
> boxplot(mother,horizontal=TRUE,main="Mother's height",xlab = "Height in inches",col="
firebrick2")
```



На основу графика можемо закључити да је расподела обележја померена удесно (на самом почетку смо добили да је медијана 64.00 а узорачка средња вредност 69.08) и да постоје аутлајери.

- <u>Боксплот висина деце</u>

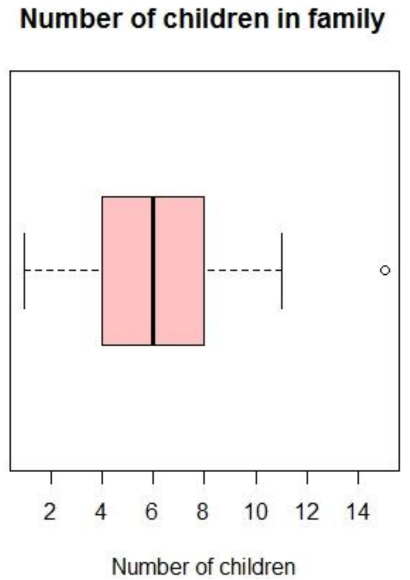
```
> boxplot(height,horizontal=TRUE,main="Child's height",xlab = "Height in inches",col="d
arkseagreen2")
```



- На основу графика можемо закључити да је расподела обележја померена удесно (на самом почетку смо добили да је медијана 66.50 а узорачка средња вредност 66.76) и да постоје аутлајери.

- <u>Боксплот броја деце у породици</u>

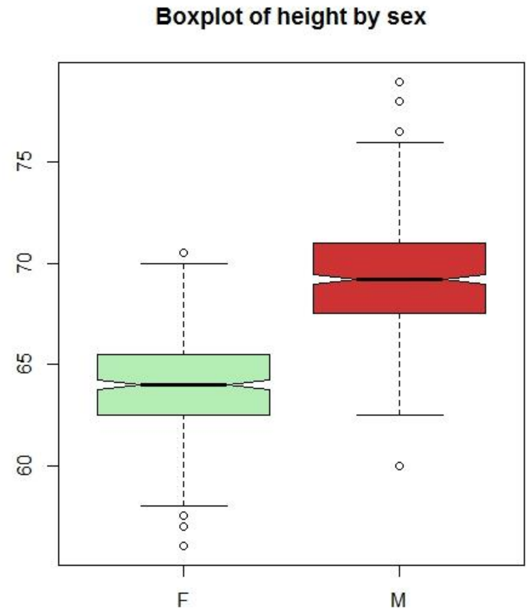
```
> boxplot(nkids,horizontal=TRUE,main="Number of children in family",xlab = "Number of c
hildren",col="rosybrown1")
```



На основу графика можемо претпоставити да је расподела обележја на средини, тј. да су медијана и узорачка средина једнаке.Међутим, на почетку смо видели да је медијана 6.00 а узорачка средња вредност 6.136 па одбацујемо претпоставку о једнакости. Дакле, закључујемо да је расподела обележја померена удесно и да постоје аутлајери.

- <u>Боксплот висина деце по полу</u>

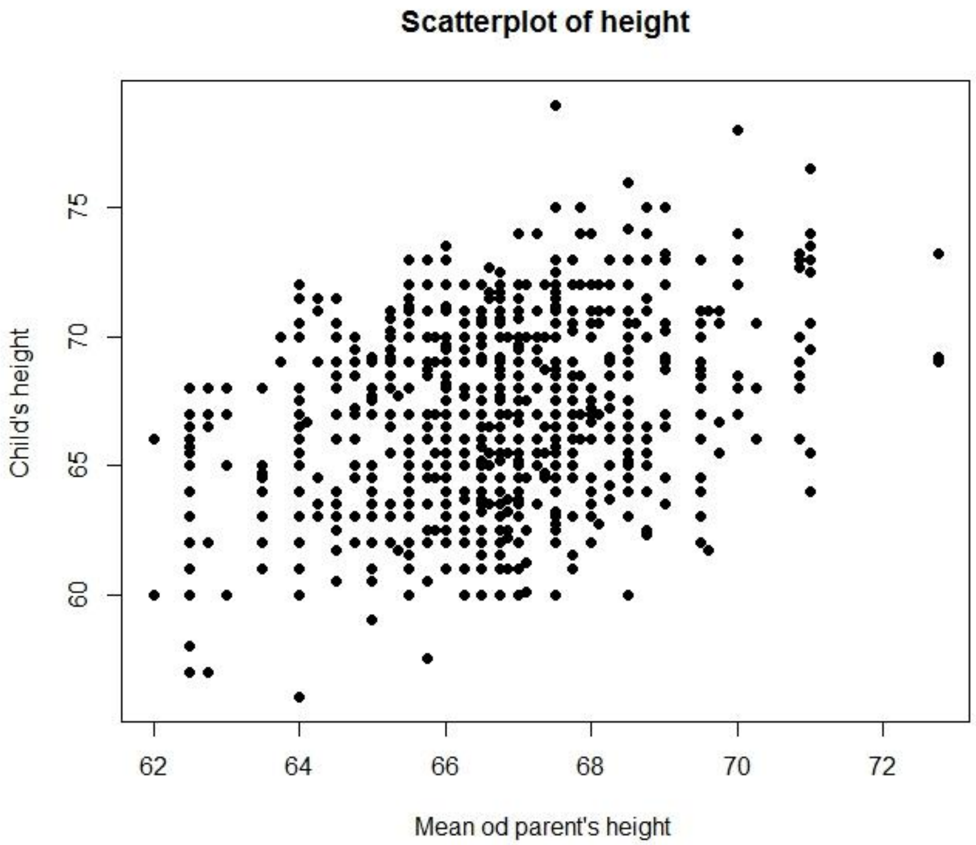
```
> boxplot(height~sex,col=c("darkseagreen2","brown3"),main="Boxplot of height by sex",
notch=TRUE)
```



Scatterplots графици

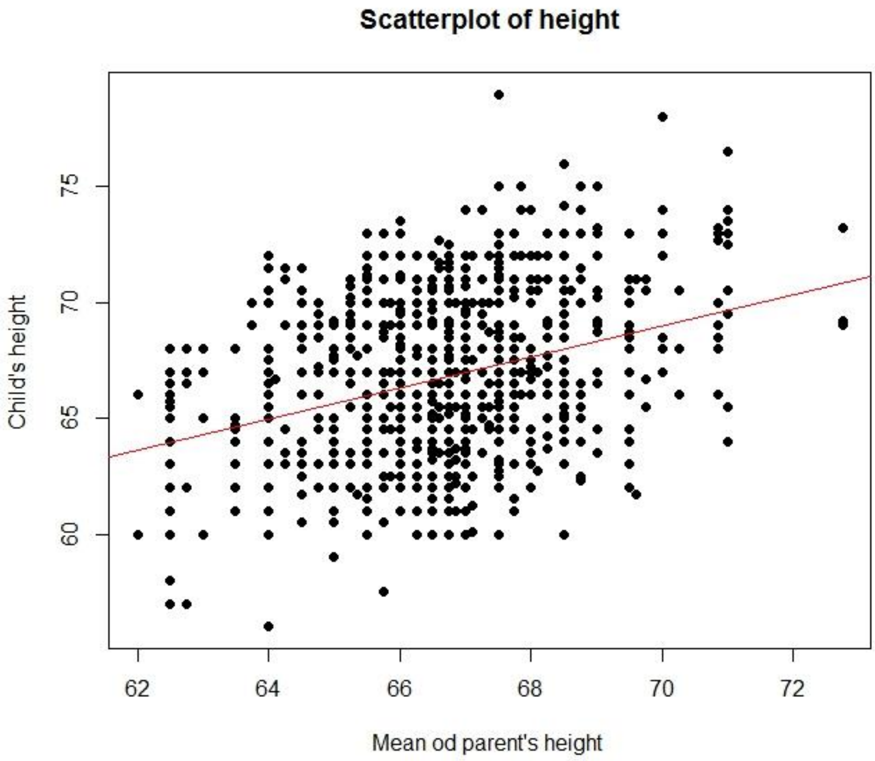
Заједнички график за средњу вредност висине родитеља и висине деце.

```
> Hmean<-((Galton$father+Galton$mother)/2)
> plot(Hmean, height, main="Scatterplot of height",
+      xlab="Mean od parent's height ", ylab="Child's height ", pch=19)
```



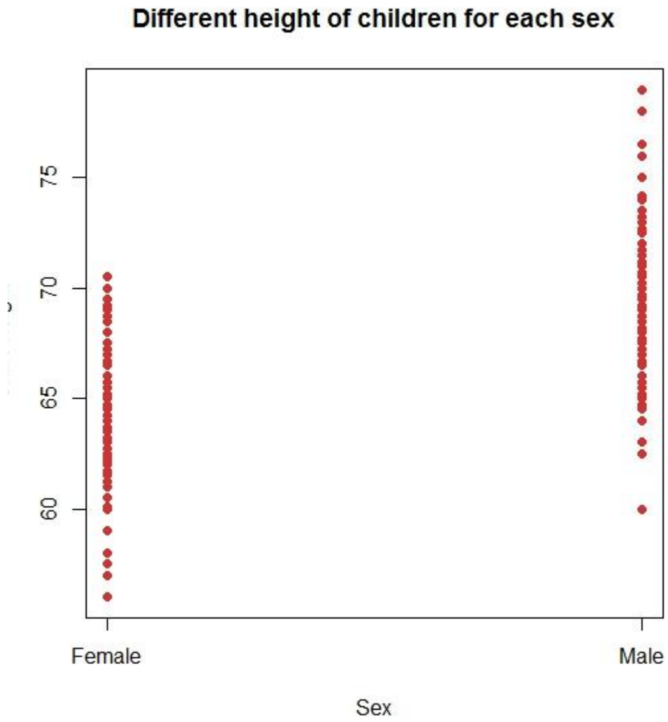
Додавање регресионе линије на график:

```
> abline(lm(height~Hmean), col="red")
```



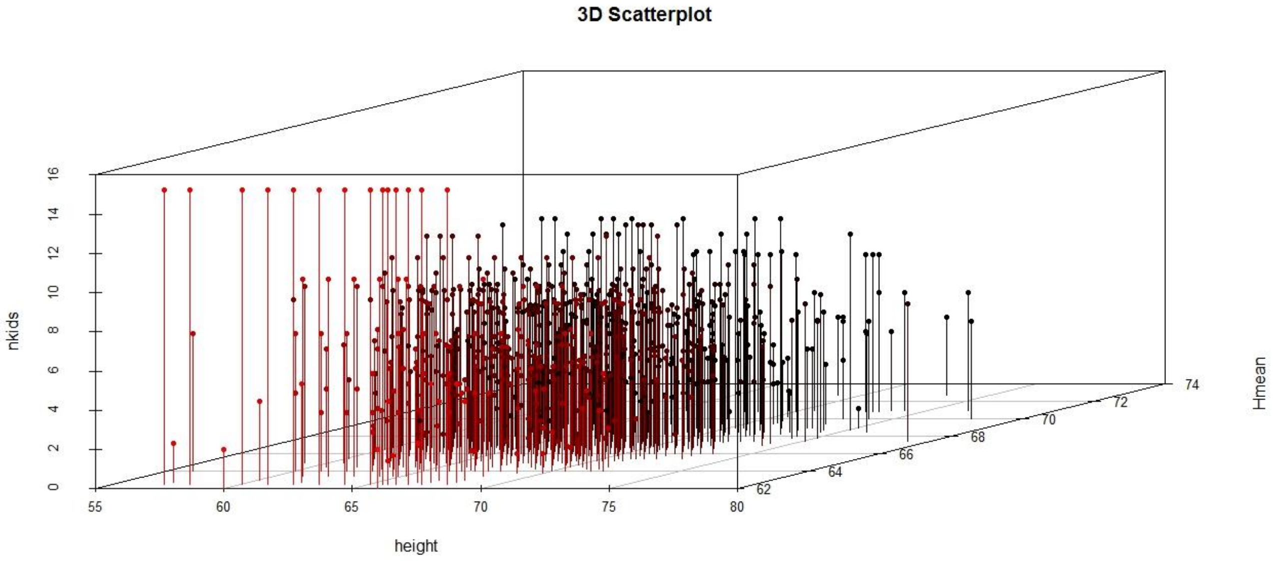
Разлику у висини деце за сваки пол можемо представити :

```
> stripchart(height~sex,
+            main="Different height of children for each sex",
+            xlab="Sex",
+            ylab="Chil's height",
+            col="brown3",
+            group.names=c("Female","Male"),
+            vertical=T,
+            pch=16
+ )
```



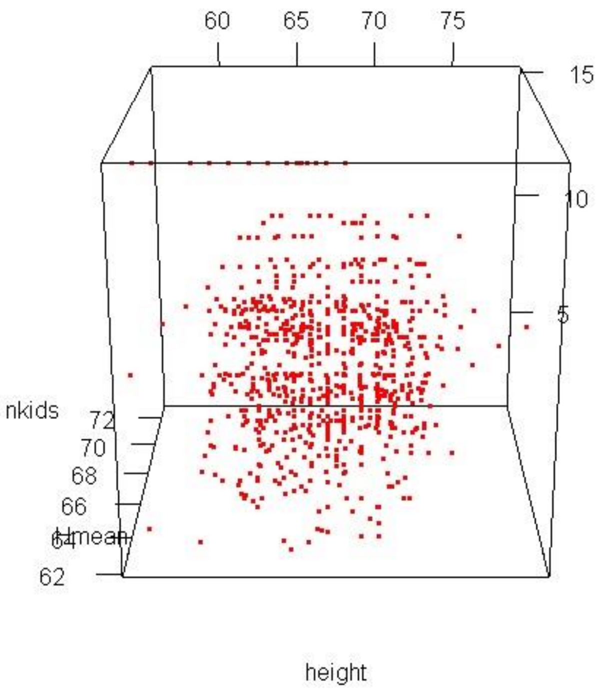
За цртање 3D Scatterplot-a потребно је инсталирати пакет ***scatterplot3d.***

```
> install.packages("scatterplot3d")
> library(scatterplot3d)
> scatterplot3d(height,Hmean,nkids,pch=20, highlight.3d=TRUE,
+               type="h", main="3D Scatterplot")
```



Још један начин за цртање 3D Scatterplot-a потребно је инсталирати пакет ***Rcmdr.***

```
> install.packages("rgl")
> library(rgl)
> plot3d(height,Hmean,nkids, col="red", size=3)
```



Обрада података

Испитивање расподела скупа података

Расподелу података можемо испитати на различите начине. Најједноставнији начин је испитивање бројева. На самом почетку смо податке представили функцијом ***summary*,** а још један начин сажетка података можемо добити употребом функције ***fivenum***. Ова функција нам даје редом минимум, доњи квартил, медијану, горњи квартил и максимум.

```
> fivenum(mother)
[1] 58.0 63.0 64.0 65.5 70.5
> fivenum(father)
[1] 62.0 68.0 69.0 71.0 78.5
> fivenum(height)
[1] 56.0 64.0 66.5 69.7 79.0
```

Податке графички можемо представити и помоћу стабло-лишће дијаграма употребом функције ***stem*** .

Стабло лишће дијаграм за висину очева добијамо на следећи начин :

```
> stem(father)

  The decimal point is at the |

  62 | 00055
  63 |
  64 | 00000000000000000
  65 | 00000000000000000000000000000000000000555555555
  66 | 0000000000000000000000000000000000000000000000000000000000055555555555
  67 | 00000000000000000000000000000000000000000555555555555555
  68 | 0000000000000000000000000000000000000000000000000000000000000000000000+66
  69 | 0000000000000000000000000000000000000000000000000000000000000000000000+66
  70 | 0000000000000000000000000000000000000000000000000000000000000000000000+92
  71 | 0000000000000000000000000000000000000000000000000000000000000000000000+27
  72 | 000000000000000000000000000000000000000000055555555577777777
  73 | 000000000000000000000002
  74 | 000000000000000000005
  75 | 00000000000005555
  76 |
  77 |
  78 | 5555
```

Дијаграм нас мало подсећа на нормалну расподелу па ћемо то проверити. Користићемо ***Shapiro-Wilk*** тест нормалности који нам говори да ли узорак има нормалну расподелу. Дакле, нулта хипотеза је H_0 : узорак има нормалну расподелу, а алтернативна H_1 : узорак нема нормалну расподелу.

```
> shapiro.test(father)

	Shapiro-Wilk normality test

data:  father
W = 0.98433, p-value = 3.211e-08
```

p- вредност теста је мала тако да одбацујемо нулту у корист алтернативне.Закључак овог тестирања је да узорак нема нормалну расподелу.

Стабло лишће дијаграм за висину мајки добијамо на следећи начин :

```
> stem(mother)

  The decimal point is at the |

  58 | 0000000555555555
  59 | 00000000000000000000000000
  60 | 00000000000000000000000000000000000025
  61 | 00000000000000000000000005
  62 | 00000000000000000000000000000000000000000000000000000000000000000000+22
  63 | 00000000000000000000000000000000000000000000000000000000000000000000+73
  64 | 00000000000000000000000000000000000000000000000000000000000000000000+70
  65 | 00000000000000000000000000000000000000000000000000000000000000000000+89
  66 | 00000000000000000000000000000000000000000000000000000000000000000000+47
  67 | 000000000000000000000000000000000000000000000
  68 | 000000000005555555555
  69 | 00000000000000000000000
  70 | 55
```

Дијаграм нас мало подсећа на нормалну расподелу па ћемо то проверити. Користићемо ***Shapiro-Wilk*** тест нормалности који нам говори да ли узорак има нормалну расподелу. Дакле, нулта хипотеза је H_0 : узорак има нормалну расподелу, а алтернативна H_1 : узорак нема нормалну расподелу.

```
> shapiro.test(mother)

  Shapiro-Wilk normality test

data:  mother
W = 0.9779, p-value = 1.992e-10
```

p- вредност теста је мала тако да одбацујемо нулту у корист алтернативне.Закључак овог тестирања је да узорак нема нормалну расподелу.

Стабло лишће дијаграм за висину деце добијамо на следећи начин :

```
> stem(height)

  The decimal point is at the |

  56 | 0
  57 | 005
  58 | 0
  59 | 0
  60 | 0000000000000000000155555
  61 | 0000000000000000002555555777
  62 | 00000000000000000000000000000000000000235555555555555555555555577
  63 | 00000000000000000000000000000000000000000000000022255555555555555555
  64 | 00000000000000000000000000000000000000000000025555555555555555555555
  65 | 00000000000000000000000000000000000000000000000000000000000002222555+17
  66 | 00000000000000000000000000000000000000000000000000000000005555555555+4
  67 | 00000000000000000000000000000000000000000000000000022255555555555555
  68 | 00000000000000000000000000000000000000000000000000000000255555555555+3
  69 | 0000000000000000000000000000000000000000000022222225555555555777
  70 | 00000000000000000000000000000000000000000000000000000222255555555555+9
  71 | 000000000000000000000000000000000000022225555555555777
  72 | 000000000000000000000000000000000555777
  73 | 0000000000000000000022255
  74 | 00000002
  75 | 0000
  76 | 05
  77 |
  78 | 0
  79 | 0
```

Дијаграм нас мало подсећа на нормалну расподелу па ћемо то проверити. Користићемо ***Shapiro-Wilk*** тест нормалности који нам говори да ли узорак има нормалну расподелу. Дакле, нулта хипотеза је H_0 : узорак има нормалну расподелу, а алтернативна H_1 : узорак нема нормалну расподелу.

```
> shapiro.test(height)

   Shapiro-Wilk normality test

data:  height
W = 0.98978, p-value = 6.713e-06
```

p- вредност теста је мала тако да одбацујемо нулту у корист алтернативне.Закључак овог тестирања је да узорак нема нормалну расподелу.

Стабло лишће дијаграм за број деце у породици добијамо на следећи начин :

```
> stem(nkids) #stablo lisce dijagram

  The decimal point is at the |

   1 | 00000000000000000000000000000000
   2 | 0000000000000000000000000000000000000000
   3 | 000000000000000000000000000000000000000000000000000000000000000000
   4 | 00000000000000000000000000000000000000000000000000000000000000000000+36
   5 | 00000000000000000000000000000000000000000000000000000000000000000000+60
   6 | 00000000000000000000000000000000000000000000000000000000000000000000+34
   7 | 00000000000000000000000000000000000000000000000000000000000000000000+32
   8 | 00000000000000000000000000000000000000000000000000000000000000000000+48
   9 | 000000000000000000000000000000000000000000000000000000000000000
  10 | 0000000000000000000000000000000000000000
  11 | 00000000000000000000000000000000
  12 |
  13 |
  14 |
  15 | 000000000000000
```

Дијаграм нас мало подсећа на нормалну расподелу па ћемо то проверити. Користићемо ***Shapiro-Wilk*** тест нормалности који нам говори да ли узорак има стандардну нормалну расподелу. Дакле, нулта хипотеза је H_0 : узорак има стандардну нормалну расподелу, а алтернативна H_1 : узорак нема стандардну нормалну расподелу.

```
> shapiro.test(nkids)

Shapiro-Wilk normality test

data:  nkids
W = 0.96587, p-value = 1.186e-13
```

p- вредност теста је мала тако да одбацујемо нулту у корист алтернативне.Закључак овог тестирања је да број деце у породици нема стандардну нормалну расподелу.

Сада ћемо испитати да ли ова променљива има нормалну расподелу.

```
> shapiro.test(rnorm(length(nkids),mean = mean(nkids), sd=sd(nkids)))

   Shapiro-Wilk normality test

data:  rnorm(length(nkids), mean = mean(nkids), sd = sd(nkids))
W = 0.99842, p-value = 0.599
```

p- вредност теста је велика тако да прихватамо H_0 да променљива има нормалну расподелу са параметрима m= 6.135857 и sd=2.685156.

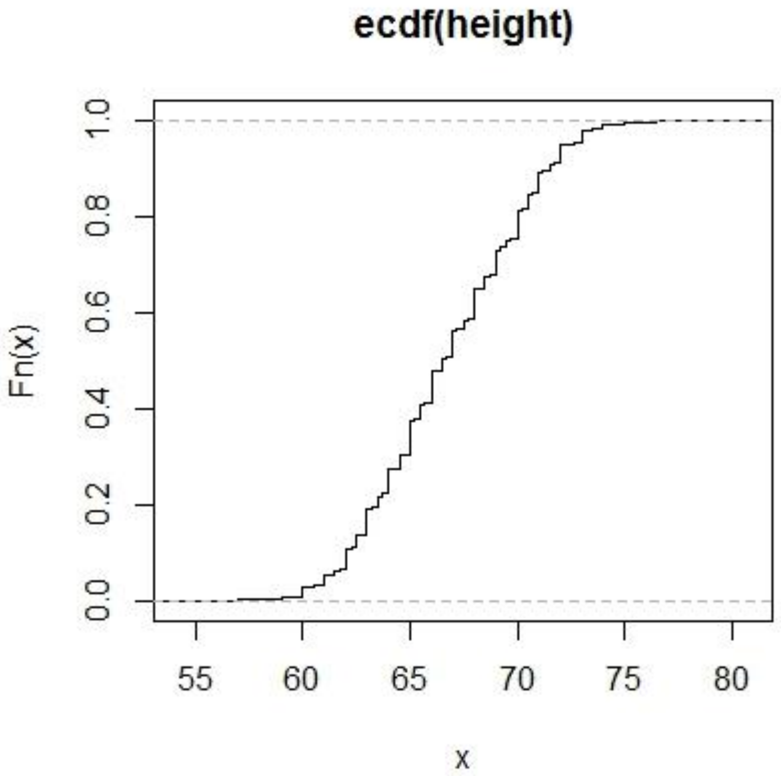
```
> mean(nkids)
[1] 6.135857
> sd(nkids)
[1] 2.685156
```

График емпиријске функције расподеле можемо нацртати употребом функције ***ecdf*** , стандардног пакета ***stepfun.***
Инсталирамо и учитавамо пакет ***stepfun*** за почетак.

```
> install.packages("stepfun")
> library(stepfun)
```

Емпиријска функција расподеле висине деце добија се на следећи начин :

```
> plot(ecdf(height), do.points=FALSE, verticals=TRUE)
```



Ова расподела је очигледно далеко од било које стандардне расподеле. Испитаћемо колико одступа од нормане расподеле :

```
> x <- seq(56, 80, 2)
> lines(x, pnorm(x, mean=mean(height), sd=sqrt(var(height))), lty=3,col="red")
```

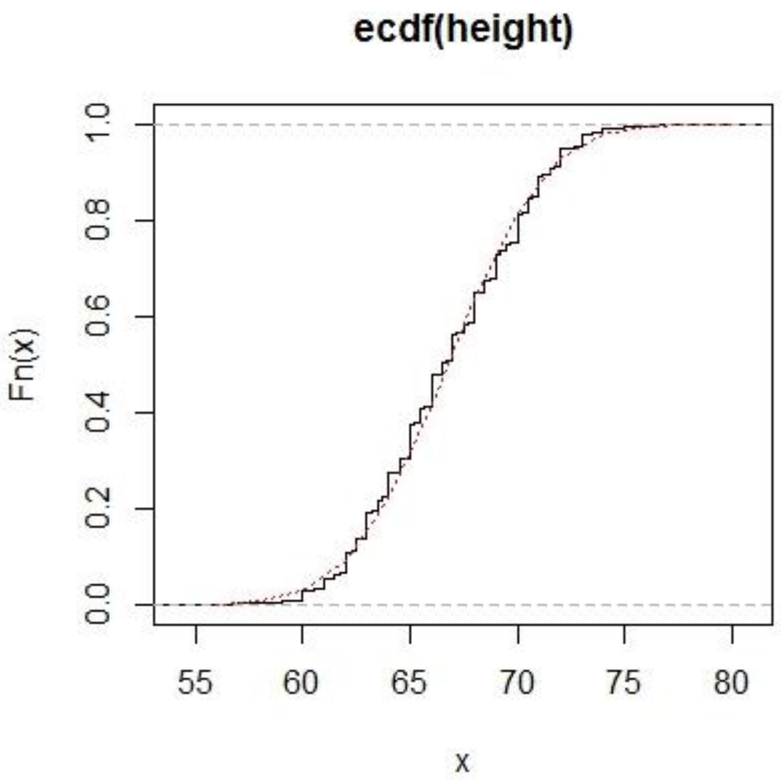
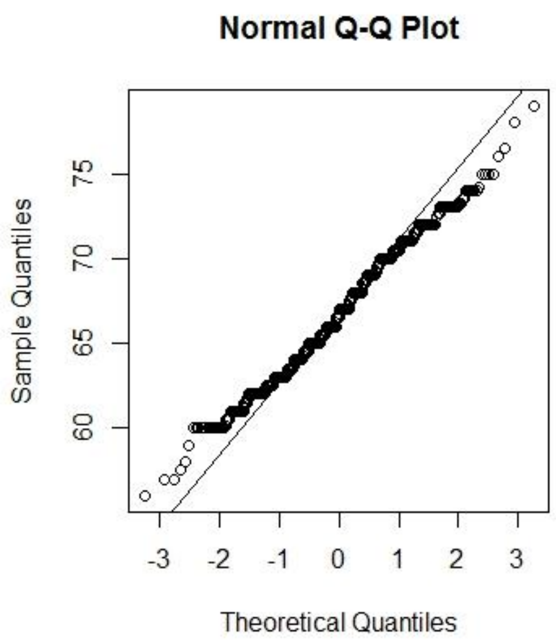


График квантил-квантил (Q-Q) нам може помоћи да то детаљније испитамо.

```
> qqnorm(height)
> qqline(height)
```



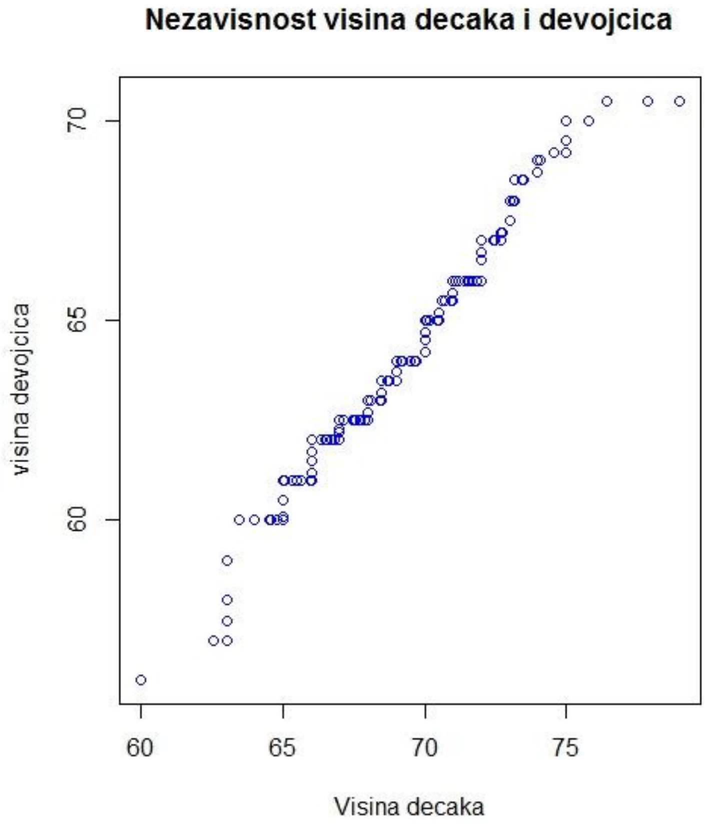
Са графика можемо закључити да узорак нема нормалну расподелу. Међутим, да би то формалније испитали искористићемо неки од статистичких тестова. Користићемо ***Kolmogorov -Smirnov*** тест нормалности.Нулта хипотеза је H_0:узорак има стандардну нормалну расподелу, а алтернативна H_1 : узорак нема стандардну нормалну расподелу.

```
> ks.test(height, "pnorm", mean=mean(height), sd=sqrt(var(height)))

    One-sample Kolmogorov-Smirnov test

data:  height
D = 0.065358, p-value = 0.0009315
alternative hypothesis: two-sided
```

p- вредност теста је мала тако да одбацујемо нулту у корист алтернативне.

Претпоставимо да између висина дечака и висина девојчица постоји зависност. То графички можемо проверити помоћу Q-Q теста:

```
> qqplot(height[sex=="M"],height[sex=="F"], col="blue", main="Nezavisnost visina deca
ka i devojcica",
+ xlab = "Visina decaka", ylab = "visina devojcica")
```



Статистички тестови

Имамо два независна узорка висина оца и висина мајке. Постављамо нулту хипотезу H_0 да су узорци из расподела који имају исту средњу вредност, против алтернативне хипотез H_1 да немају исту средњу вредност. За ово испитивање користићемо Студентов ***t-test.***
Идеја теста: Ако се две узорачке средине пуно разликују одбацује се нулта хипотеза у корист алтернативне. ***t-test*** је апроксимативни тест, тест статистика је само приближно ***t-***дистибуирана.

```
> t.test(father,mother)

	Welch Two Sample t-test

data:  father and mother
t = 45.645, df = 1785.7, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 4.927221 5.369661
sample estimates:
mean of x mean of y 
 69.23285  64.08441 
```

p- вредност је мала тако да одбацујемо нулту у корист алтернативне(на нивоу значајности 5%).Закључак овог тестирања је да немају исту средњу вредност.

Даље желимо да испитамо да ли су просечне висине дечака и девојчица исте. Постављамо нулту хипотезу H_0 да су узорци из расподела који имају исту средњу вредност, против алтернативне хипотез H_1 да немају исту средњу вредност. За ово испитивање такође ћемо користи Студентов ***t-test.***

```
> t.test(height[sex=="M"],height[sex=="F"])

	Welch Two Sample t-test

data:  height[sex == "M"] and height[sex == "F"]
t = 30.662, df = 895.02, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 4.791018 5.446293
sample estimates:
mean of x mean of y
 69.22882  64.11016
```

p- вредност је мала тако да одбацујемо нулту у корист алтернативне(на нивоу значајности 5%).Закључак овог тестирања је да просечне висине дечака и девојчица нису исте.

Тестови зависности

У зависности од врсте променљиве одлучујемо који тест да користимо.

1. Номиналне променљиве (нема поретка)
   Независност испитујемо помоћу ***Хи квадрат*** теста за независност употребом функције ***chisq.test***.
   Идеја теста : израчунавају се очекиване учесталости под претпоставком о независности. Ако добијене вредности превише одступају од очекиваних, онда одбацујемо нулту хипотезу.
   Испитаћемо независност за променљиве редни број породице и пол ,зато што су то једине категоричке променљиве у бази. Нулта хипотеза је да су променљиве независне против алтернативне да су зависне.

```
> TK<-table(family,sex)
> chisq.test(TK)

	Pearson's Chi-squared test

data:  TK
X-squared = 193.13, df = 196, p-value = 0.5447
```

   p- вредност је велика тако да прихватамо нулту хипотезу (на нивоу значајности 5%). Закључак овог тестирања је су променљиве фамилија и пол независне.

2. Непрекидне променљиве

   Овде претпостављамо да променљиве ,теоретски ,могу узети све вредности из неког интервала. Независност испитујемо помоћу ***Pearson***-овог теста корелације за независност употребом функције ***cor.test***.
   Испитаћемо независност за променљиве висина деце и број деце у породици. Нулта хипот еза је да су променљиве независне против алтернативне да су зависне.

```
> cor.test(height,nkids)

	Pearson's product-moment correlation

data:  height and nkids
t = -3.8298, df = 896, p-value = 0.0001372
```

```
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.19074723 -0.06200411
sample estimates:
       cor
-0.1269101
```

p- вредност је мала тако да одбацујемо нулту у корист алтернативне(на нивоу значајности 5%).Закључак овог тестирања је да су променљиве зависне.

3. Дискретне променљиве

Претпоставимо да се добијена посматрања могу поређати. Нулта хипотеза је да су узорци неколерисани. То испитујемо помоћу ***Spearman*** **-**овог теста корелације за независност употребом функције ***cor.test.*** Испитаћемо корелацију за променљиве висина деце и број деце у породици. Нулта хипотеза је да су променљиве корелиране против алтернативне да нису.

```
> cor.test( height,nkids, method="spearman")


	Spearman's rank correlation rho

data:  height and nkids
S = 134980000, p-value = 0.0003785
alternative hypothesis: true rho is not equal to 0
sample estimates:
       rho
-0.1183672
```

p- вредност је мала тако да одбацујемо нулту у корист алтернативне(на нивоу значајности 5%).Закључак овог тестирања је да променљиве нису корелиране.

Још неки тестови зависности

- ***Mann-Whitney U*** тест независности

H_0 : Узорци су независни
H_1 : Узорци нису независни

```
> wilcox.test(height~sex)

	Wilcoxon rank sum test with continuity correction

data:  height by sex
W = 15256, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
```

Закључак: p- вредност је мала тако да одбацујемо нулту у корист алтернативне хипотезе

- ***Kruskal-Wallis*** -ов модел

H_0 : Узорци су из исте расподеле
H_1 : Узорци нису из исте расподеле

```
> kruskal.test(height~sex)

	Kruskal-Wallis rank sum test

data:  height by sex
Kruskal-Wallis chi-squared = 484.65, df = 1, p-value < 2.2e-16
```

Закључак: p- вредност је мала тако да одбацујемо нулту у корист алтернативне хипотезе

Закључак

На основу различитих графичких и табеларних података из базе, као и на основу резултата тестирања која су обављена ,можемо закључити следеће:

- У истраживању је било више дечака од девојчица
- Најмањи број деце у породици је 1, највећи 15
- Најнижи отац је висок 62.00 ,а највиши 78.5 инча
- Најнижа мајка је висока 58.00 ,а највиша 70.5 инча
- Најниже дете је високо 56.00 ,а највише 79.0 инча
- Највећи број очева има висину 70 инча
- Највећи број мајки има висину 65 инча
- Највећи број деце има висину 66 инча
- Највећи број породица има петоро деце
- Ни један од посматраних узорака нема нормалну расподелу
- Средња вредност висина мајки и очева није једнака
- Просечне висине дечака и девојчица нису једнаке
- Променљиве породица и пол су независне
- Променљиве висина деце и број деце у породици су зависне
- Променљиве висина деце и број деце у породици нису корелиране

Галтонов закључак :

Галтон је написао да је разлика у висини између детета и родитеља пропорционална одступању родитеља од типичних људи у популацији.Односно приметио је да високи родитељи имају високу децу, али у просеку нижу од родитеља. Висину женске деце помножио је бројем 1.08 да би их могао упоредити с висином мушке деце, а висину родитеља дефинисао је као средњу вредност оба родитеља.