

Семинарска работа по предметот Бизнис статистика

Стефан Везенкоски 233152

Извор на податочното множество: <https://www.kaggle.com/datasets/gchan357/human-glaucoma>
(го скартив податочното множество и во прилог ќе ви испратам excel документ со точната бројка на вредности со која работев)

Работам со податочното множество каде се прикажуваат податоци за пациенти кои имаат глауком. Се состои од следите осум обележја:

- Glaucoma
- Age
- ocular_pressure
- MD
- PSD: promenlivost na standardot na defektot
- GHT
- cornea_thickness: debelina na roznica
- RNFL4

Јас работам и правам детални статистички анализи за следниве две обележја:

- PSD
- cornea_thickness

1. ПРВ ДЕЛ

Табела со распределба на честоти за PSD:							
> print(freq_table_df_PSD)							
Интервали	Средни_точки	фреквенција	Релативна_фреквенција	Релативна_фреквенција_проенти	Кумулативна_фреквенција	Кумулативна_фреквенција_проенти	
1 [1.18,2.02)	1.599474	57	0.163323782	16.3323782	57	16.33238	
2 [2.02,2.86)	2.438421	62	0.177650430	17.7650430	119	34.09742	
3 [2.86,3.7)	3.277368	39	0.111747851	11.1747851	158	45.27221	
4 [3.7,4.54)	4.116316	15	0.042979943	4.2979943	173	49.57020	
5 [4.54,5.37)	4.955263	14	0.040114613	4.0114613	187	53.58166	
6 [5.37,6.21)	5.794211	13	0.037249284	3.7249284	200	57.30659	
7 [6.21,7.05)	6.633158	12	0.034383954	3.4383954	212	60.74499	
8 [7.05,7.89)	7.472105	11	0.031518625	3.1518625	223	63.89685	
9 [7.89,8.73)	8.311053	10	0.028653295	2.8653295	233	66.76218	
10 [8.73,9.57)	9.150000	9	0.025787966	2.5787966	242	69.34097	
11 [9.57,10.4)	9.988947	16	0.045845272	4.5845272	258	73.92550	
12 [10.4,11.2)	10.827895	26	0.074498567	7.4498567	284	81.37536	
13 [11.2,12.1)	11.666842	16	0.045845272	4.5845272	300	85.95989	
14 [12.1,12.9)	12.505789	17	0.048710602	4.8710602	317	90.83095	
15 [12.9,13.8)	13.344737	14	0.040114613	4.0114613	331	94.84241	
16 [13.8,14.6)	14.183684	9	0.025787966	2.5787966	340	97.42120	
17 [14.6,15.4)	15.022632	5	0.014326648	1.4326648	345	98.85387	
18 [15.4,16.3)	15.861579	2	0.005730659	0.5730659	347	99.42693	
19 [16.3,17.1]	16.700526	2	0.005730659	0.5730659	349	100.00000	

Табела на честоти за обележјето PSD

1. ТАБЕЛИ ЗА РАСПРЕДЕЛБА НА ЧЕСТОТИ ЗА PSD и cornea_thickness

1.1 Определба на број на интервали (k) (за обележјето cornea_thickness) :

```
# Пресметка на бројот на интервали
num_intervals_cornea <- round(sqrt(length(cornea_thickness_values))) # k = 18
```

Објаснување:

1. length(cornea_thickness_values) враќа број на елементи од обележјето (n=349)
2. sqrt – квадратен корен
3. round() – заокружува на цел број

num_intervals_cornea = $\sqrt{349} \approx 19$ – број на интервали

1.2 Ширина на интервалите (w)

```
# Пресметка на ширината на интервалите за cornea_thickness
range_cornea <- max(cornea_thickness_values) - min(cornea_thickness_values)
width_cornea <- range_cornea / num_intervals_cornea
```

Ширина на интервал: $w \geq R / k$.

Во range_cornea се зачувува рангот (најголем – најмал елемент од множеството)

width_cornea (променлива за ширина) = ранг / бројот на интервали ;

180 / 19 \approx 9

1.3 . ИНТЕРВАЛИ

```
intervals_cornea <- seq(min(cornea_thickness_values), max(cornea_thickness_values),
by = width_cornea)
```

Со помош на seq() се генерираат вредности од минималната до максималната вредност од множеството и секоја следна вредност ќе биде зголемена за ширината w. (w = 9)

1.4 Табела на распределба на честоти

```
freq_table_cornea <- cut(cornea_thickness_values, breaks = intervals_cornea, right
= FALSE, include.lowest = TRUE)
freq_table_cornea <- table(freq_table_cornea)
```

Создава табела со фреквенциите на секој интервал

1.5 Пресметка на средни точки на интервалите

```
midpoints_cornea <- (intervals_cornea[-length(intervals_cornea)] +
intervals_cornea[-1]) / 2
```

intervals_cornea[-length(intervals_cornea)] – почеток на секој интервал

intervals_cornea[-1] – крајот на секој интервал

Првиот и последниот интервал се делат со 2 за да се добие СРЕДНАТА ТОЧКА НА ИНТЕРВАЛОТ

1.6 Пресметка на релативни и кумулативни фреквенции (и Пресметка на релативни и кумулативни фреквенции во %)

```
rel_freq_cornea <- prop.table(freq_table_cornea)
cum_freq_cornea <- cumsum(freq_table_cornea)
```

rel_freq_cornea <- prop.table(freq_table_cornea) - пресметка на релативна фреквенција на податоците со prop.table() која ги дели секоја фреквенција со n

cum_freq_cornea <- cumsum(freq_table_cornea) - пресметува кумулативна фреквенција со функцијата cumsum() која ги собира фреквенциите по редослед.

```
rel_freq_percent_cornea <- rel_freq_cornea * 100
cum_freq_percent_cornea <- cum_freq_cornea / sum(freq_table_cornea) * 100
```

Релативната фреквенција за секој интервал / 100 за да се добие релативната фреквенција во %
Кумулативната фреквенција / n * 100 за да се добие кумулативна фрек. Во %

1.7 Креирање на табелата со распределба на честоти за cornea_thickness

```
freq_table_df_cornea <- data.frame(
  Интервали = names(freq_table_cornea),
  Средни_точки = midpoints_cornea,
  Фреквенција = as.numeric(freq_table_cornea),
  Релативна_фреквенција = as.numeric(rel_freq_cornea),
  Релативна_фреквенција_проценти = as.numeric(rel_freq_percent_cornea),
  Кумулативна_фреквенција = as.numeric(cum_freq_cornea),
  Кумулативна_фреквенција_проценти = as.numeric(cum_freq_percent_cornea)
)
```

freq_table_df_cornea <- data.frame() – за организирање податоци во табела. Секој претходно добиен податок за секој интервал се зачувува во посебна променлива и на крајот со print() се печати табелата.

На крајот се прикажува табелата која содржи:

- Интервали каде што соодветните податоци се поделени на одреден број на интервали и соодветна ширина на интервалите.
- Пресметани се средните точки на секој од интервалите(во децимали)
- Фреквенција за секој од интервалите
- Релативна честота за секој од интервалите (и релативна честота во %)
- Кумулативна честота за секој од интервалите (и кумулативна честота во %)

Табела со распределба на честоти за cornea_thickness:

```
> print(freq_table_df_cornea)
```

	Интервали	Средни точки	Фреквенција	Релативна_фреквенција	Релативна_фреквенција_проценти	Кумулативна_фреквенција	Кумулативна_фреквенција_проценти
1	[445,454)	449.7368	4	0.011461318	1.1461318	4	1.146132
2	[454,464)	459.2105	1	0.002865330	0.2865330	5	1.432665
3	[464,473)	468.6842	5	0.014326648	1.4326648	10	2.865330
4	[473,483)	478.1579	17	0.048710602	4.8710602	27	7.736390
5	[483,492)	487.6316	14	0.040114613	4.0114613	41	11.747851
6	[492,502)	497.1053	9	0.025787966	2.5787966	50	14.326648
7	[502,511)	506.5789	23	0.065902579	6.5902579	73	20.916905
8	[511,521)	516.0526	38	0.108882521	10.8882521	111	31.805158
9	[521,530)	525.5263	31	0.088825215	8.8825215	142	40.687679
10	[530,540)	535.0000	27	0.077363897	7.7363897	169	48.424069
11	[540,549)	544.4737	38	0.108882521	10.8882521	207	59.312321
12	[549,559)	553.9474	37	0.106017192	10.6017192	244	69.914040
13	[559,568)	563.4211	26	0.074498567	7.4498567	270	77.363897
14	[568,578)	572.8947	29	0.083094556	8.3094556	299	85.673352
15	[578,587)	582.3684	19	0.054441261	5.4441261	318	91.117479
16	[587,597)	591.8421	15	0.042979943	4.2979943	333	95.415473
17	[597,606)	601.3158	13	0.037249284	3.7249284	346	99.140401
18	[606,616)	610.7895	2	0.005730659	0.5730659	348	99.713467
19	[616,625]	620.2632	1	0.002865330	0.2865330	349	100.000000

```
>
```

Добиената табела за обележјето **cornea_thickness**

Исто така податоците се претставени и со соодветни хистограми и полигони.

2. Креирање на хистограми и полигони

1.1 ФУНКЦИЈА ЗА КРЕИРАЊЕ НА ХИСТОГРАМИ И ПОЛИГОНИ

```
2 # Функција за создавање на хистограм и полигон
3 create_hist_and_polygon <- function(values, variable_name, color_hist,
4   color_polygon) {
5   num_intervals <- round(sqrt(length(values)))
6   range_values <- max(values) - min(values)
7   width <- range_values / num_intervals
8
9   # Хистограм
10  hist_data <- hist(values, breaks = num_intervals, plot = FALSE)
11
12  # Создавање на хистограм
13  hist_plot <- ggplot(data = data.frame(x = values), aes(x = x)) +
14    geom_histogram(binwidth = width, fill = color_hist, color = "black") +
15    labs(x = paste(variable_name, "Вредности"), y = "Фреквенција", title =
16      paste("Хистограм на", variable_name)) +
17    theme_minimal()
18
19  # Подготовка на податоци за полигон
20  x.axis <- c(min(hist_data$breaks), hist_data$mids, max(hist_data$breaks))
21  y.axis <- c(0, hist_data$counts, 0)
22
23  # Создавање на полигон
24  polygon_plot <- ggplot() +
25    geom_polygon(aes(x = x.axis, y = y.axis), fill = color_polygon, alpha = 0.5)
26  +
```

```

24     labs(x = paste(variable_name, "Вредности"), y = "Густина", title =
paste("Густински полигон на", variable_name)) +
25     theme_minimal()
26
27     return(list(hist_plot = hist_plot, polygon_plot = polygon_plot))
28 }

```

Дефинирана функција **create_hist_and_polygon** со четири аргументи: values(податоци), variable_name(име на променлива), color_hist (боја на хистограм) , color_polygon (боја на полигон)

Во функцијата се пресметани бр. на интервали и ширина (по ист принцип како погоре)

Создавање на хистограм:

```

29     hist_data <- hist(values, breaks = num_intervals, plot = FALSE)
30
31     geom_histogram(binwidth = width, fill = color_hist, color = "black") +
32     labs(x = paste(variable_name, "Вредности"), y = "Фреквенција", title =
paste("Хистограм на", variable_name)) +
33     theme_minimal()

```

Креирање на хистограмот. Ggplot се користи за визуелизација на хистограмот, со geom_histogram се задава ширина и боја и со labs се поставуваат оски и насловот.

Наслов: Хистограм за Cornea Thickness

X оска: Cornea Thickness вредности

Y оска: Фреквенција

Подготовка на податоци за полигон

x.axis <- c(min(hist_data\$breaks), hist_data\$mids, max(hist_data\$breaks)) – граници за инт. по x оска

y.axis <- c(0, hist_data\$counts, 0) – граници за инт. по y оска, 0 за почеток.

СОЗДРАВАЊЕ НА ПОЛИГОН

```

polygon_plot <- ggplot() +
  geom_polygon(aes(x = x.axis, y = y.axis), fill = color_polygon, alpha = 0.5) +
  labs(x = paste(variable_name, "Вредности"), y = "Густина", title =
paste("Густински полигон на", variable_name)) +
  theme_minimal()

```

По ист принцип како за креирање на хистограм

Со

```
return(list(hist_plot = hist_plot, polygon_plot = polygon_plot))
```

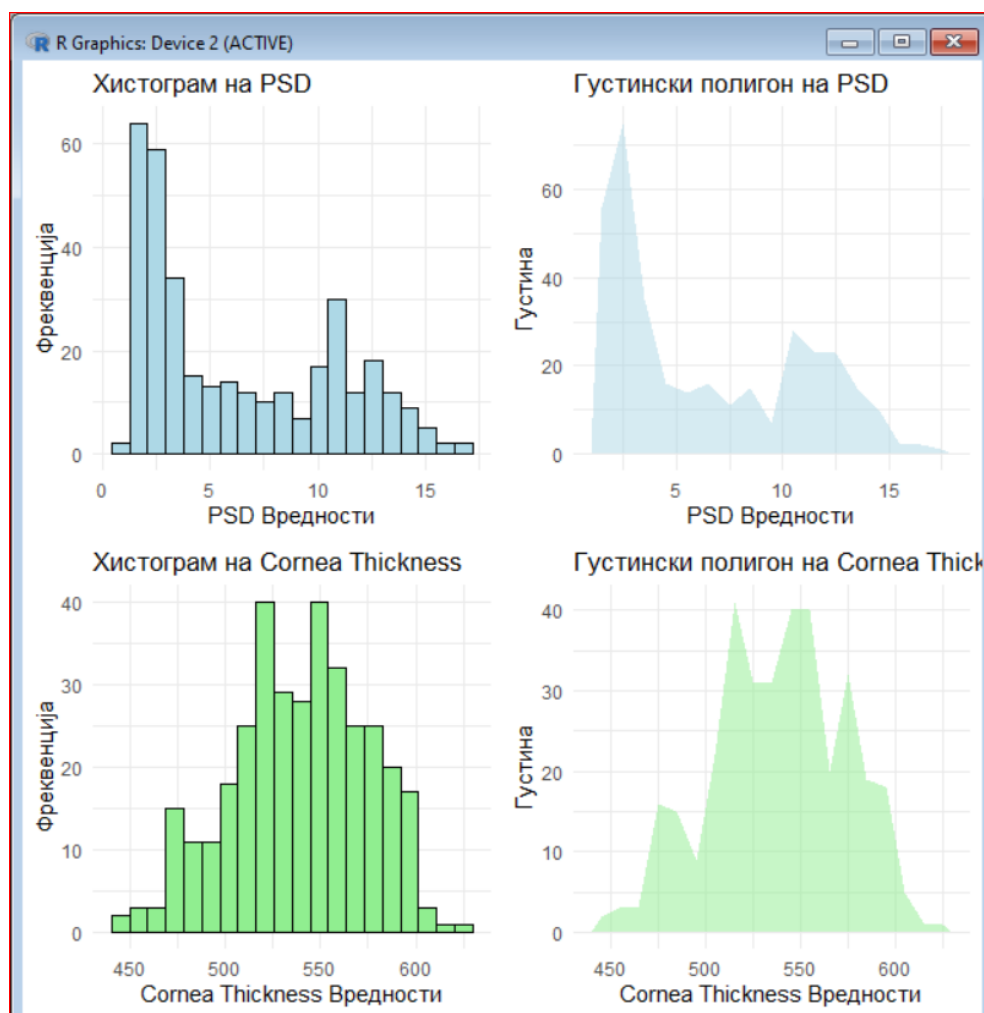
ни се враќа резултатот од функцијата со графички приказ

Потоа следи повик на функцијата за посебните обечежја. Пример за Cornea_thickness:

```
cornea_plots <- create_hist_and_polygon(cornea_thickness_values, "Cornea Thickness", "lightgreen", "lightgreen")
```

Печатење:

```
print(cornea_plots$hist_plot)  
print(cornea_plots$polygon_plot)
```



Скриншот од добиени графички прикази за хистограм и полигон за PSD и Cornea Thickness

3. Креирање на стебло-лист дијаграм за PSD

```
# Стебло-лист дијаграм за PSD -----  
---  
myStem <- function(x, leftDigits, rounding = 1) {  
  data = data.table("x" = x)  
  data[, left := floor(x / 10^leftDigits)]  
  data[, right := (round(x - left * 10^leftDigits, rounding)) * 10^rounding]  
  data = data[, paste(sort(right), collapse = " "), by = left]  
  data[, out := paste(left, " | ", V1), by = left]  
  cat(data$out, sep = "\n")  
}  
  
cat("\nСтебло-лист дијаграм за PSD:\n")  
myStem(PSD_values, 0, 2)  
#  
_____
```

MyStem – променлива за стебло-лист дијаграмот

function(x, leftDigits, rounding = 1){} – фунцкија

x – вредноста која треба да се добие во стебло-лист дијаграмот

```
data[, left := floor(x / 10^leftDigits)]  
data[, right := (round(x - left * 10^leftDigits, rounding)) * 10^rounding]
```

Со ова се извлекува стеблото и листот.

Left – пресметка на стеблото со делење на вредноста на x со $10^{\text{leftDigits}}$ и заокружување со floor.

Right – пресметување на листовите со одземање на стеблото и заокружување со rounding децимални места. Резултатот се множи со 10^{rounding} за да се прикажат листовите како цели броеви.

```
data = data[, paste(sort(right), collapse = " "), by = left]  
data[, out := paste(left, " | ", V1), by = left]  
cat(data$out, sep = "\n")
```

Сортирање на листовите од најмал кон најголем (by left to right) и креирање на конечен приказ на стебло-лист дијаграмот

```

Стебло-лист дијаграм за PSD:
> myStem(PSD_values, 0, 2)
1 | 18 25 28 29 33 36 38 39 40 41 41 42 43 44 46 46 47 48 48 53 55 57 59 60 62 62 63 63 66 67 68 68 69 70 70 72 77 77 78 79 81 81 81 82 82 83 83 84 85 86 88 92 94 96
2 | 0 0 1 2 3 3 4 4 5 6 7 8 10 14 14 14 14 14 16 16 17 18 18 18 19 23 23 24 27 28 28 29 29 29 31 31 32 33 33 36 40 42 44 48 51 52 52 55 56 56 57 58 60 61 63 64 67 71 75 80 80 81 83 88
3 | 0 2 2 5 6 6 6 8 11 13 16 21 22 23 23 28 28 31 35 36 41 48 50 56 60 67 69 76 78 88 96 97 97 98 99
12 | 0 0 1 2 6 6 11 28 32 34 35 41 51 54 55 56 63 64 70 75 76 79 83 94 96
4 | 3 4 12 17 27 34 35 60 65 71 74 77 80 83 85 95
11 | 1 6 8 12 13 14 17 18 18 20 23 25 25 26 27 29 46 58 64 66 74
10 | 2 6 8 9 19 24 28 32 33 37 40 40 40 48 51 56 57 60 62 67 73 73 86 90 93 93 96 96
14 | 29 35 51 51 51 55 69 71 90 96
8 | 6 8 21 27 33 37 38 55 70 73 77 79 82 91 94
5 | 5 11 15 19 30 46 50 53 54 78 86 90 93 99
13 | 2 2 11 20 29 31 41 63 66 67 69 73 88 97 99
7 | 6 28 40 48 52 59 69 69 78 78 83
6 | 1 14 15 19 28 50 57 65 65 77 85 85 91 92 96 97
16 | 17 90
9 | 11 12 35 55 91 94 99
15 | 7 59
17 | 12
> #
>

```

Стебло-лист дијаграм за PSD

4. Графици на расејување за податоците од двете обележја и објаснување за врската помеѓу двете обележја

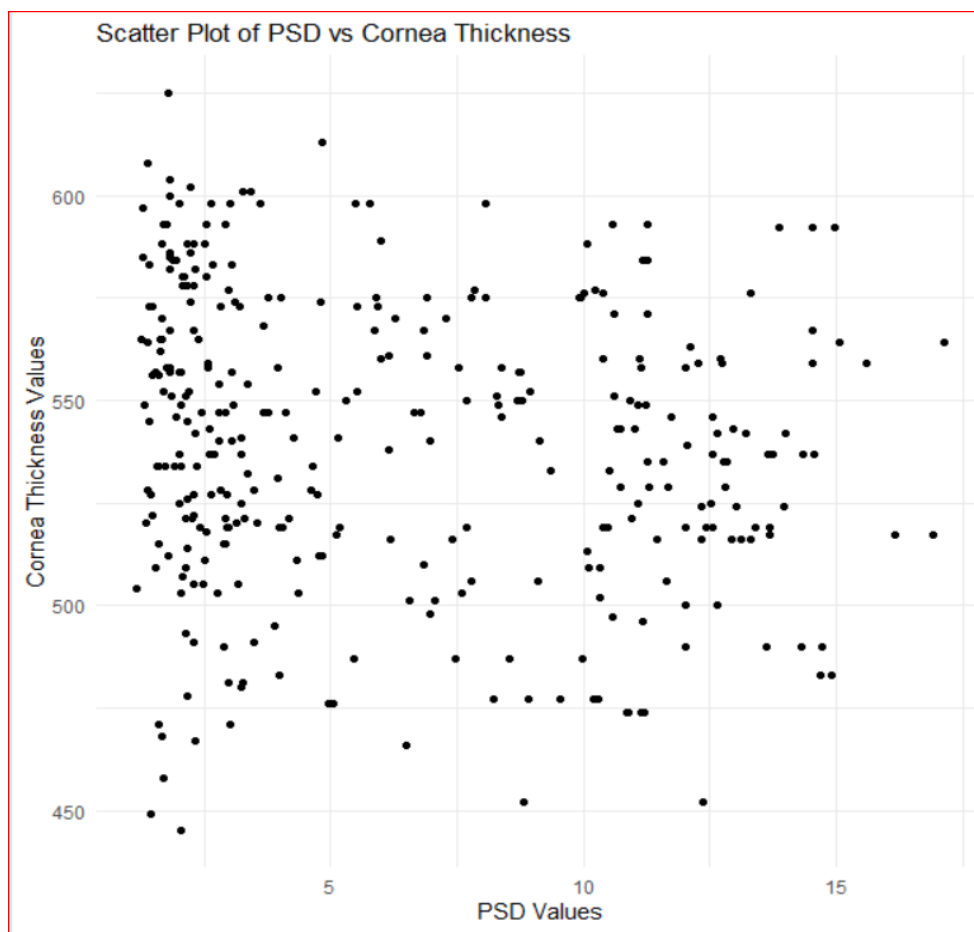


График на расејување за PSD и Cornea Thickness


```
# 2. Графика на расејување за двете обележја-----
scatter_plot <- ggplot(податоци, aes(x = PSD_values, y = cornea_thickness_values))
+
  geom_point() +
  labs(x = "PSD Values", y = "Cornea Thickness Values", title = "Scatter Plot of
PSD vs Cornea Thickness") +
  theme_minimal()
```

Дискусија за врската помеѓу двете обележја: од графикот може да се забележи дека постои **негативна линеарна корелација** помеѓу двете обележја. Коефициентот на корелација r е -0.1501, што укажува на слаба негативна корелација помеѓу двете обележја.

5. Мода, медијана и просек на податоците

```
# 3. Мода, медијана и просек за cornea_thickness -----
---

cornea_mode <-
as.numeric(names(table(cornea_thickness_values))[which.max(table(cornea_thickness_v
alues))])
cornea_median <- median(cornea_thickness_values)
cornea_mean <- mean(cornea_thickness_values)

cat("\nМода, медијана и просек за cornea_thickness:\n")
print(paste("Мода:", cornea_mode))
print(paste("Медијана:", cornea_median))
print(paste("Просек:", cornea_mean))
```

```
cornea_mode <-
as.numeric(names(table(cornea_thickness_values))[which.max(table(cornea_thickness_v
alues))])
```

Наоѓа мода со помош на `which.max(table(cornea_thickness_values))` која наоѓа која вредност од множеството на обележјето се повторува најчесто.

```
cornea_median <- median(cornea_thickness_values)
```

Наоѓа која вредност е на средина (сортирано)

```
cornea_mean <- mean(cornea_thickness_values)
```

Наоѓа просек на податоците

Мода, медијана и просек за cornea_thickness:

Мода: **519**

Медијана: **541**

Просек: **539.002865329513**

Мода, медијана и просек за PSD:

Мода: **2.14 (се повторува 5 пати)**

Медијана: **4.65**

Просек: **6.29962750716332**

6. Кварталите, опсегот и интеркварталниот распон на податоците

```
# 4. Квартали, опсег и интерквартален распон за корнеалната дебелина-----
cornea_quantiles <- quantile(cornea_thickness_values, probs = c(0.25, 0.5, 0.75))
cornea_range <- max(cornea_thickness_values) - min(cornea_thickness_values)
cornea_IQR <- IQR(cornea_thickness_values)

cat("\nКвартили, опсег и интерквартален распон за cornea_thickness:\n")
print("Квартили:")
print(cornea_quantiles)
print("Опсег:")
print(cornea_range)
print("Интерквартален распон:")
print(cornea_IQR)
```

```
cornea_quantiles <- quantile(cornea_thickness_values, probs = c(0.25, 0.5, 0.75))
```

Пресметка на квартили Q1, Q2 (медијана) и Q3

```
cornea_range <- max(cornea_thickness_values) - min(cornea_thickness_values)
```

Пресметка на опсег (ранг) на податоците од обележјето.
Најголем – најмал елемент

```
cornea_IQR <- IQR(cornea_thickness_values)
```

Пресметка на интерквартален распон (IQR)
 $Q3 - Q1 = IQR$

Квартали, опсег и интерквартален распон за cornea thickness:

Квартили: 25% 50% 75%

516 541 565

Опсег: **180**

Интерквартален распон(IQR): 49 (КВАРТИЛ 3(75%) – КВАРТИЛ 1(25%))

565-516 = 49

Квартили, опсег и интерквартален распон за PSD:

Квартили: 25% 50% 75%

2.29 4.65 10.57

Опсег: **15.94**

Интерквартален распон: $10.57 - 2.28 = 8.29$

7. Дисперзија и стандардна девијација

```
# 5. Пресметка на дисперзија и стандардна девијација-----
cornea_variance <- var(cornea_thickness_values)
cornea_sd <- sd(cornea_thickness_values)

cat("\nДисперзија и стандардна девијација за cornea_thickness:\n")
print(paste("Дисперзија:", cornea_variance))
print(paste("Стандардна девијација:", cornea_sd))
#
```

Пресметка на дисперзија

```
cornea_variance <- var(cornea_thickness_values)
```

`var(cornea_thickness_values)` функцијата ја пресметува дисперзијата и ја зачувува во променливата `cornea_variance` (пр. за `cornea_thickness`)

Пресметка на стандардна девијација:

```
cornea_sd <- sd(cornea_thickness_values)
```

√ од дисперзијата

$$s = \sqrt{\frac{\sum (x - \bar{x})^2}{n - 1}}$$

→ Формула за стандардна девијација

Дисперзија и стандардна девијација за cornea thickness:

Дисперзија: **1245.30746303066**

(пресметана со формулата која е користена на факултет)

Стандардна девијација: $\sqrt{1245.30746303066} = 35.2889141662174$

Дисперзија и стандардна девијација за PSD:

Дисперзија: **19.5440053206205**

Стандардна девијација: **4.42086024667377"**

8. Коефициент на корелација

```
# 7. Пресметка на коефициент на корелација за PSD и cornea_thickness-----
correlation_coefficient <- cor(PSD_values, cornea_thickness_values)

cat("\nКоефициент на корелација помеѓу PSD и cornea_thickness:\n")
print(correlation_coefficient)
# _____
_____
```

Коефициент на корелација помеѓу PSD и cornea_thickness:

-0.1500741 (слаба негативна врска)

ВТОР ДЕЛ

1. Интервал на доверба за PSD:

$\bar{X} \pm z_{\alpha/2} \frac{s}{\sqrt{n}}$ → Формулата за пресметување на инт. на доверба

```
interval_doverba <- function(volume) {  
  CI <- 0.95 # INTERVAL NA SIGURNOST(%) od 95%  
  alfa <- 1 - CI # nivoto na prifatлива greshka(%)  
  MoE <- qnorm(1 - alfa / 2) * sd(volume) / sqrt(length(volume))  
  min.interval <- mean(volume) - MoE  
  max.interval <- mean(volume) + MoE  
  
  # pecati interval  
  print(c(min.interval, max.interval))  
  print(mean(volume))  
}  
  
sample_size <- min(1000, length(PSD_values))  
interval_doverba(sample(PSD_values, size = sample_size))  
print("-----")  
print("-----")
```

interval_doverba е функција за пресметување на интервалот на доверба на просекот

```
MoE <- qnorm(1 - alfa / 2) * sd(volume) / sqrt(length(volume))
```

Пресметка на маргинална грешка Z

```
min.interval <- mean(volume) - MoE  
max.interval <- mean(volume) + MoE
```

Пресметка на долен и горен интервал на доверба

Објаснување:

95% интервал на доверба

$$\text{Алфа} = 1 - 0.95 = 0.05$$

$$\text{Алфа} / 2 = 0.025$$

$$Z \approx 1.96$$

Пресметка на интервалот: (min.interval, max.interval)

(5.835815, 6.763440) : интервалот на доверба, што значи дека со **95% сигурност**, просекот на PSD_values е помеѓу **5.835815** и **6.763440**

6.299628: Ова е просечната вредност на примерокот што е избран.

2. Хипотези за тестируваниот параметар

```
# VTOR DEL, zadaca 2 -----
-----
# hipotezi-----
# H0: EX = 5
# Ha: EX != 5
EX <- 5
alfa <- 0.05
gr <- qnorm(1 - alfa / 2) # se naogja granicata
print("Интервал на доверба:")
print(c(-1 * gr, gr)) # se pecati intervalot (leva i desna granica)
z <- ((mean(PSD_values) - EX) / sd(PSD_values)) * sqrt(length(PSD_values)) #
presmetka za z statistikata
print(z)

# proverka koja hipoteza e točna
if(z > (-1 * gr) & z < gr) {
  print("H0: T. H0 се прифаќа")
} else {
  print("Ha: T. H0 се отфрла")
}
print("-----")
print("-----")
# -----
```

```
gr <- qnorm(1 - alfa / 2) # se naogja granicata
```

Граница за алфа /2

```
print("Интервал на доверба:")  
print(c(-1 * gr, gr)) # se pecati intervalot (leva i desna granica)
```

Пресметка на интервалот на доверба

```
z <- ((mean(PSD_values) - EX) / sd(PSD_values)) * sqrt(length(PSD_values)) #  
presmetka za z statistikata
```

$$Z_0 = \frac{\bar{X} - \mu_0}{S} \sqrt{n}$$

-> формула за пресметка

```
# proverka koja hipoteza e točna  
if(z > (-1 * gr) & z < gr) {  
  print("H0: T. H0 се прифаќа")  
} else {  
  print("Ha: T. H0 се отфрла")  
}
```

Овој услов проверува дали тест статистиката z спаѓа во интервалот на доверба

Објаснување:

$EX = 5$

$H_0: EX = 5$ (просекот е 5)

$H_a: EX \neq 5$ (просекот не е 5)

Алфа = 0.05

Ниво на значајност = $1 - \alpha/2 = 0,025$

Критична вредност за Z тестот: 1.96

$(-1.96, 1.96)$ – критичен домен

Пресметана вредност за Z: 5.491928

Z не припаѓа во критичниот домен со што H_0 отфрла и се прифаќа H_a .

3. Тест за распределба

```
test za raspredelba-----
# Тестирање на нормалност за PSD
print("Непараметарски тест(тест за распределба):")
print("H0: Обележјето PSD има нормална распределба.")
print("H0: Обележјето PSD нема нормална распределба.")

# Тестирање на нормалност за PSD
shapiro_test_PSD <- shapiro.test(PSD_values)
cat("\nРезултати од Шапиро-Вилковиот тест за нормалност за PSD:\n")
print(shapiro_test_PSD)

# Објаснување
if (shapiro_test_PSD$p.value < 0.05) {
  cat("Нултата хипотеза ( $H_0$ ) се отфрла: Обележјето PSD нема нормална
распределба.\n")
} else {
  cat("Нултата хипотеза ( $H_0$ ) не се отфрла: Обележјето PSD има нормална
распределба.\n")
}
```

```
print("H0: Обележјето PSD има нормална распределба.")
print("H0: Обележјето PSD нема нормална распределба.")
```

```
# Тестирање на нормалност за PSD
shapiro_test_PSD <- shapiro.test(PSD_values)
cat("\nРезултати од Шапиро-Вилковиот тест за нормалност за PSD:\n")
print(shapiro_test_PSD)
```

shapiro.test(PSD_values) – функција која прави Shapiro-Wilk тест за нормална распределба на обележјето PSD.

```
if (shapiro_test_PSD$p.value < 0.05) {
  cat("Нултата хипотеза ( $H_0$ ) се отфрла: Обележјето PSD нема нормална
распределба.\n")
} else {
  cat("Нултата хипотеза ( $H_0$ ) не се отфрла: Обележјето PSD има нормална
распределба.\n")
}
```

Проверка дали р вредноста од тестот е помала од 0.05. Ако е помала, тоа значи дека нултата хипотеза (H_0) се отфрла, во спротивно др прифаќа.

Објаснување:

Ќе тестираме дали обележјето X – PSD, има нормална распределба. Се поставуваат следните хипотези:

H_0 : X има нормална распределба

H_a : X нема нормална распределба

Се користи Shapiro-Wilk тест за нормална распределба. $W = 0.87568$. За ниво на значајност алфа 0.05 а р-вредноста се добива дека е $3.665e-16$. Па бидејќи р-вредноста е помала од алфа, нултата хипотеза се отфрла. Односно обележјето X нема нормална распределба.

Се печати : Нултата хипотеза (H_0) се отфрла: Обележјето PSD нема нормална распределба.

4. Тест за независност

```
# VTOR DEL, zadaca 4 -----  
-----  
# test na hipotezi za nezavisnost-----  
if ("вашата_колона_1" %in% colnames(податоци) && "вашата_колона_2" %in%  
colnames(податоци)) {  
  chisq_test <- chisq.test(table(податоци$вашата_колона_1,  
податоци$вашата_колона_2))  
  print(chisq_test)  
} else {  
  cat("Chi-squared тестот не е применлив.\n")  
}  
  
# print("-----  
-----")
```

Се печати: Chi-squared тестот не е применлив.

5. Регресиона анализа

```
# VTOR DEL, zadaca 5 -----  
-----  
# linearna regresija-----  
plot(PSD_values, cornea_thickness_values)  
  
# создавање на модел за линерна регресија  
model <- lm(cornea_thickness ~ PSD, data = податоци)  
abline(model, col='red') # цртање на линија на регресија  
  
r <- cor(податоци$PSD, податоци$cornea_thickness)  
print(r)  
  
# за x = 15  
pred <- predict(model, newdata = data.frame(PSD = 15), level = 0.95, interval =  
"prediction")  
print(pred)
```

```
r <- cor(податоци$PSD, податоци$cornea_thickness)
```

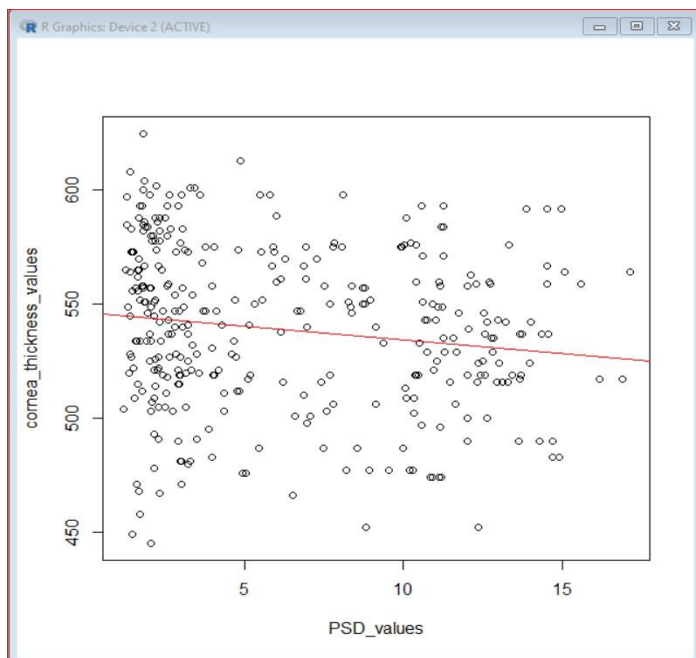
Пресметка на корелација

cor() пресметува корелација помеѓу PSD и cornea_thickness

```
# за x = 15  
pred <- predict(model, newdata = data.frame(PSD = 15), level = 0.95, interval =  
"prediction")  
print(pred)
```

predict() предвидува вредности на зависната променлива (cornea_thickness) врз основа на нови вредности на независната променлива (PSD)

$X(15) = 528.5803$: Оваа вредност (528.5803) е предвидената вредност за дебелината на роговица (Y) кога PSD е 15. Тоа е просечната предвидена вредност од моделот.



Права на регресија за двете обележја

