

SPRAWOZDANIE

Zajęcia: Analiza procesów uczenia

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium 1

13 czerwca 2020

Temat: Podstawy języka R

Wariant: 1

Adres repozytorium: <https://github.com/Konradbor/APU/tree/master/1>

Konrad Boroń
Informatyka II stopień,
stacjonarne,
7 semestr,
Gr. 1A

1. Polecenie:

Do zmiennej a podstaw wartość wyrażenia $2 * \exp(5)$. Do zmiennej b podstaw podwojona wartość zmiennej a. Wywołaj funkcję sprawdzającą, która z wartości zmiennych jest większa.

2. Wprowadzane dane:

```
> a <- 2*exp(5)
> b <- 2*a
> max(a,b)
```

3. Wynik działania:

Values	
a	296.826318205153
b	593.652636410306

```
> max(a,b)
[1] 593.6526
```

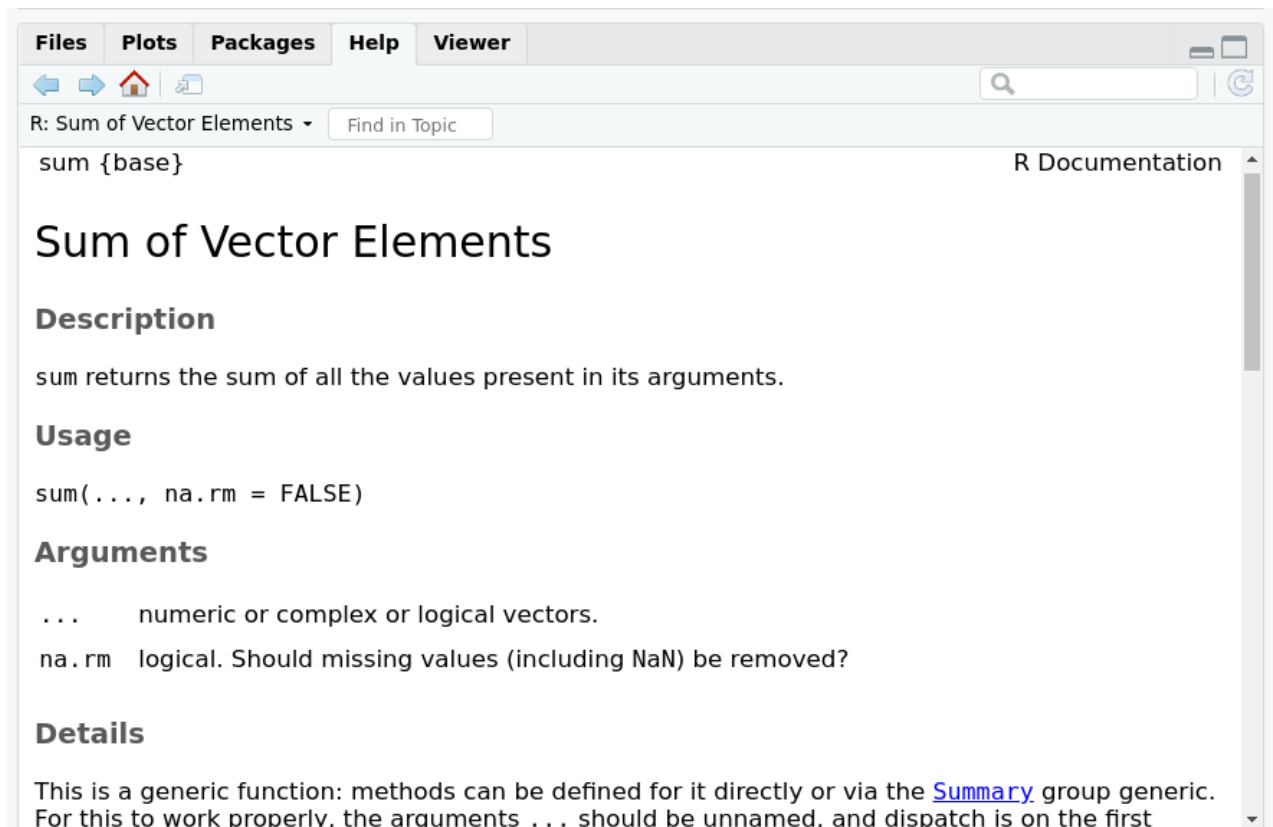
1. Polecenie:

Uruchom i przeczytaj dokumentację dla funkcji sum().

2. Wprowadzane dane:

```
> help(sum)
```

3. Wynik działania:



1. Polecenie:

Stwórz wektor a zawierający liczby od 15 do 25. Policz sumę liczb zawartych w wektorze.

2. Wprowadzane dane:

```
> a <- 15:25  
> sum(a)
```

3. Wynik działania:

```
> a  
[1] 15 16 17 18 19 20 21 22 23 24 25  
> sum(a)  
[1] 220
```

1. Polecenie:

Wyświetl wszystkie funkcje zawierające frazę sum w swojej nazwie.

2. Wprowadzane dane:

```
> apropos("sum", mode = "function")
```

3. Wynik działania:

```
> apropos("sum", mode = "function")
[1] ".colSums"           ".rowSums"           ".rs.callSummary"    ".rs.summarizeDir"
[5] ".tryResumeInterrupt" "colSums"            "contr.sum"          "cumsum"
[9] "format.summaryDefault" "print.summary.table" "print.summary.warnings" "print.summaryDefault"
[13] "rowsum"             "rowsum.data.frame" "rowsum.default"      "rowSums"
[17] "sum"                "summary"           "Summary"             "summary.aov"
[21] "summary.connection" "summary.data.frame" "Summary.data.frame"  "summary.Date"
[25] "Summary.Date"       "summary.default"   "Summary.difftime"    "summary.factor"
[29] "Summary.factor"     "summary.glm"       "summary.lm"          "summary.manova"
[33] "summary.matrix"     "Summary.numeric_version" "Summary.ordered"    "summary.POSIXct"
[37] "Summary.POSIXct"    "summary.POSIXlt"   "Summary.POSIXlt"     "summary.proc_time"
[41] "summary.srcfile"    "summary.srcref"    "summary.stepfun"     "summary.table"
[45] "summary.warnings"  "summaryRprof"
```

1. Polecenie:

Ustaw dowolny katalog roboczy. Następnie stwórz zmienną a zawierającą łańcuch znaków "smartfony Samsung". Zapisz zmienną a z obszaru roboczego do pliku w katalogu roboczym. Następnie usuń zmienną a. Sprawdź wartość zmiennej a (powinno jej brakować). Na końcu wczytaj plik ze zmienną a i sprawdź jej wartość.

2. Wprowadzane dane:

```
> a = "smartfony Samsung"
> save.image("~/Dokumenty/APU/1/e.RData")
> remove(a)
> load("~/Dokumenty/APU/1/e.RData")
```

3. Wynik działania:



1. Polecenie:

Zainstaluj i załaduj pakiet gridExtra, który umożliwia m.in. ładną wizualizację danych tabelarycznych. Następnie przy pomocy dokumentacji pakietu znajdź funkcje do wizualizacji danych tabelarycznych. Użyj jej na pierwszych 10 wierszach.

szach zbioru danych mtcars.

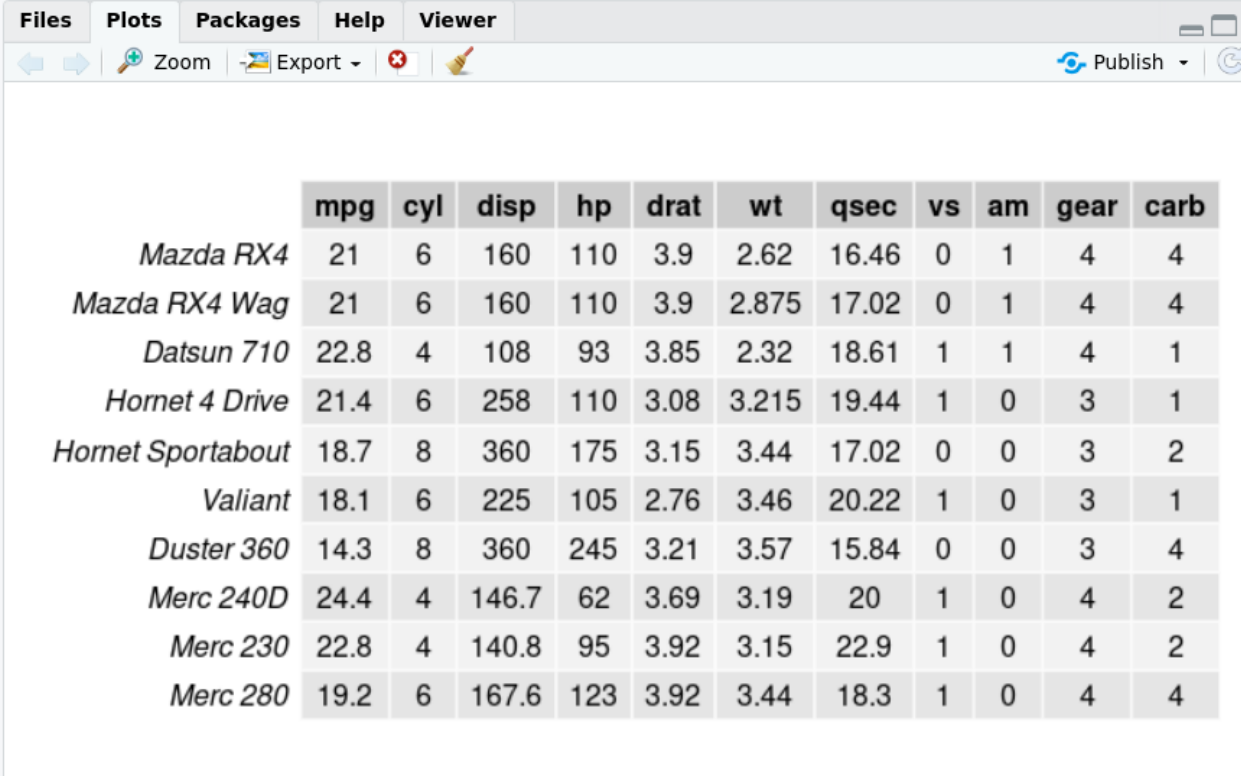
2. Wprowadzane dane:

```
> install.packages("gridExtra")
> library(gridExtra)
> help(package = "gridExtra")
> g <- tableGrob(head(mtcars,10))
> grid.draw(g)
```

Błąd w poleceniu 'grid.draw(g)':nie udało się znaleźć funkcji
↪ 'grid.draw'

```
> library(grid)
> grid.draw(g)
```

3. Wynik działania:



	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
<i>Mazda RX4</i>	21	6	160	110	3.9	2.62	16.46	0	1	4	4
<i>Mazda RX4 Wag</i>	21	6	160	110	3.9	2.875	17.02	0	1	4	4
<i>Datsun 710</i>	22.8	4	108	93	3.85	2.32	18.61	1	1	4	1
<i>Hornet 4 Drive</i>	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
<i>Hornet Sportabout</i>	18.7	8	360	175	3.15	3.44	17.02	0	0	3	2
<i>Valiant</i>	18.1	6	225	105	2.76	3.46	20.22	1	0	3	1
<i>Duster 360</i>	14.3	8	360	245	3.21	3.57	15.84	0	0	3	4
<i>Merc 240D</i>	24.4	4	146.7	62	3.69	3.19	20	1	0	4	2
<i>Merc 230</i>	22.8	4	140.8	95	3.92	3.15	22.9	1	0	4	2
<i>Merc 280</i>	19.2	6	167.6	123	3.92	3.44	18.3	1	0	4	4

1. Polecenie:

Stwórz wektor zawierający ciąg liczb 100, 96, 92, . . . 20.

2. Wprowadzane dane:

```
> a <- seq(100,20,-4)
```

3. Wynik działania:

```
> a
```

```
[1] 100 96 92 88 84 80 76 72 68 64 60 56 52 48  
↪ 44 40 36 32 28 24 20
```

1. Polecenie:

Stwórz wektora a z liczbami od 9 do 5 oraz wektor b z liczbami od 11 do 16. Utwórz nowy wektory d bedacy polaczeniem wektora b i a (w takiej kolejności). Wyświetl go.

2. Wprowadzane dane:

```
> a <- seq(9,5)  
> b <- seq(11,16)  
> d <- c(b, a)
```

3. Wynik działania:

```
> d  
[1] 11 12 13 14 15 16 9 8 7 6 5
```

1. Polecenie:

Stwórz wektor nazwa zawierajay nazwy 10 smartfonów Samsung z systemem Android 8 i osmiordzieniowym procesorem. Potem stwórz wektory wyświetlacz, pamięć_RAM i pamięć_wbudowana, aparat_foto, cena, liczba_opinii zawierajace kolejno dane 10 smartfonów. Następnie stwórz ramke danych smartfony złożona z wektorów nazwa, wyświetlacz, pamięć_RAM, pamięć_wbudowana, aparat_foto, cena oraz liczba_opinii. Wylicz średnia cene smartfonów.

2. Wprowadzane dane:

```
> nazwa <- c("Galaxy J2 Core (2020)",  
+           "Galaxy Xcover FieldPro",  
+           "Galaxy A2 Core",  
+           "Galaxy View2",  
+           "Galaxy M30",  
+           "Galaxy M20",  
+           "Galaxy M10",  
+           "Galaxy Tab Advanced2",  
+           "Galaxy Tab A 8.0 (2018)",  
+           "Galaxy A6s")  
> wyświetlacz <- c(5.0,  
+                 5.1,  
+                 5.0,  
+                 17.3,
```

```

+             6.4,
+             6.3,
+             6.22,
+             10.1,
+             8.0,
+             6.0)
> pamięć_RAM <- c(1,
+               4,
+               1,
+               3,
+               3,
+               3,
+               2,
+               3,
+               2,
+               6)
> pamięć_wbudowana <- c(
+   16,
+   64,
+   8,
+   64,
+   32,
+   32,
+   16,
+   32,
+   32,
+   64
+ )
> aparat_foto <- c(8,
+               12,
+               5,
+               0,
+               13,
+               13,
+               13,
+               8,
+               8,
+               12)
>   cena <- c(80,
+             1020,
+             120,
+             660,
+             300,
+             300,

```

```

+           135,
+           200,
+           130,
+           300)
> liczba_opinii <- c(17,
+                   48,
+                   36,
+                   50,
+                   316,
+                   358,
+                   107,
+                   8,
+                   40,
+                   86)
> ramka <- data.frame(nazwa, wyświetlacz, pamięć_RAM,
+                     ↪ pamięć_wbudowana,
+                     aparat_foto, cena, liczba_opinii)

```

3. Wynik działania:

```

> mean(ramka$cena)
[1] 324.5

```

1. Polecenie:

Do stworzonej w poprzednim zadaniu ramki danych smartfonów dodaj wpis zawierający dane nowego smartfonu. Wylicz średnia ceny ponownie.

2. Wprowadzane dane:

```

> ramka <- rbind(ramka, data.frame(nazwa = "Galaxy A9 (2018)",
+   ↪   wyświetlacz = 6.3, pamięć_RAM = 6, pamięć_wbudowana = 64,
+   aparat_foto = 24, cena = 359, liczba_opinii = 320))

```

3. Wynik działania:

```

> mean(ramka$cena)
[1] 327.6364

```

1. Polecenie:

Korzystając z ramki danych smartfony dodaj nową kolumnę określając ocenę klientów. Wpisz do kolumny odpowiednio oceny w skali od 0 do 5 krok 0.5. Dodana kolumna powinna się automatycznie przekonwertować do cech jakościowych

(tzw. factors). Wylicz średnia ceny każdej oceny.

2. Wprowadzane dane:

```
> ramka$ocena <- factor(c(3, 5, 5, 4, 4, 4, 4, 5, 4.5, 5, 4.5))
```

3. Wynik działania:

```
> mean(as.numeric(levels(ramka$ocena)[ramka$ocena]))  
[1] 4.363636
```

1. Polecenie:

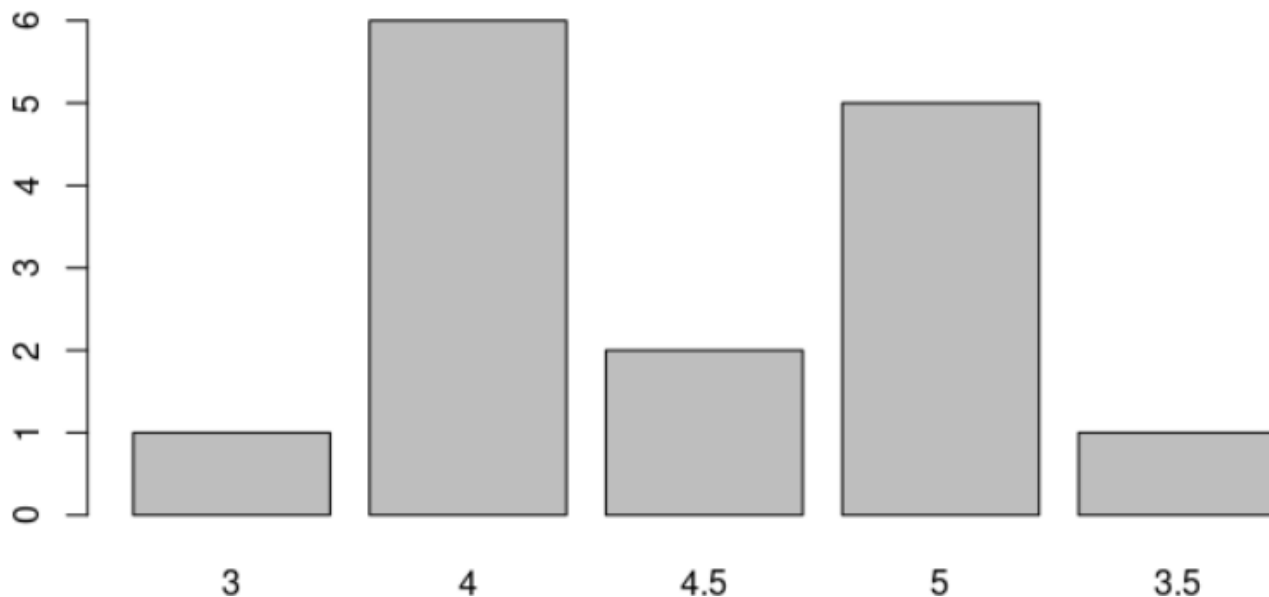
Do ramki danych smartfony dodaj kolejne 4 smartfony. Narysuj na wykresie słupkowym liczebność reprezentantów każdej z ocen klientów.

2. Wprowadzane dane:

```
ramka <- rbind(ramka, data.frame(nazwa = c("Galaxy A7 (2018)",  
"Galaxy Note9",  
"Galaxy J6+",  
"Galaxy J4 Core"  
),  
wyświetlacz = c(6.0,6.4,6.0,6.0),  
pamięć_RAM = c(4,6,3,1),  
pamięć_wbudowana = c(64,128,32,16),  
aparat_foto = c(12,12,10,5),  
cena = c(309,820,230,150),  
liczba_opinii = c(223,1243,198,87),  
ocena = factor(c(4,5,4,3.5)))  
)
```

3. Wynik działania:

```
> barplot(table(ramka$ocena))
```



1. Polecenie:

Wykorzystując ramkę danych smartfony pokaż procentowy udział każdej oceny przy pomocy wykresu kołowego oraz wachlarzowego.

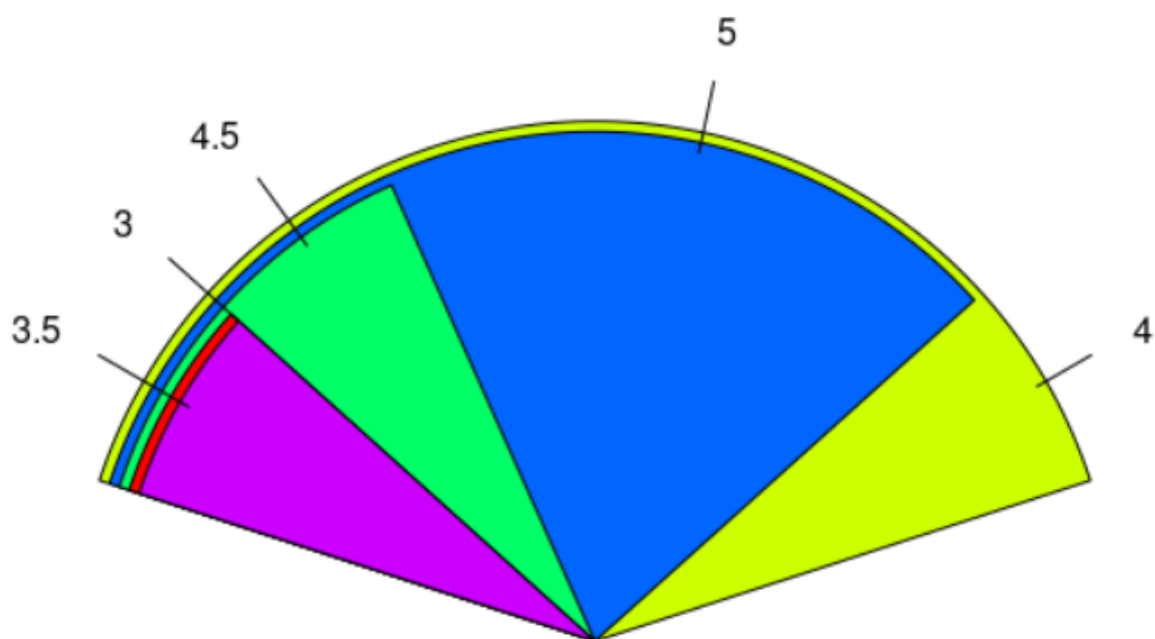
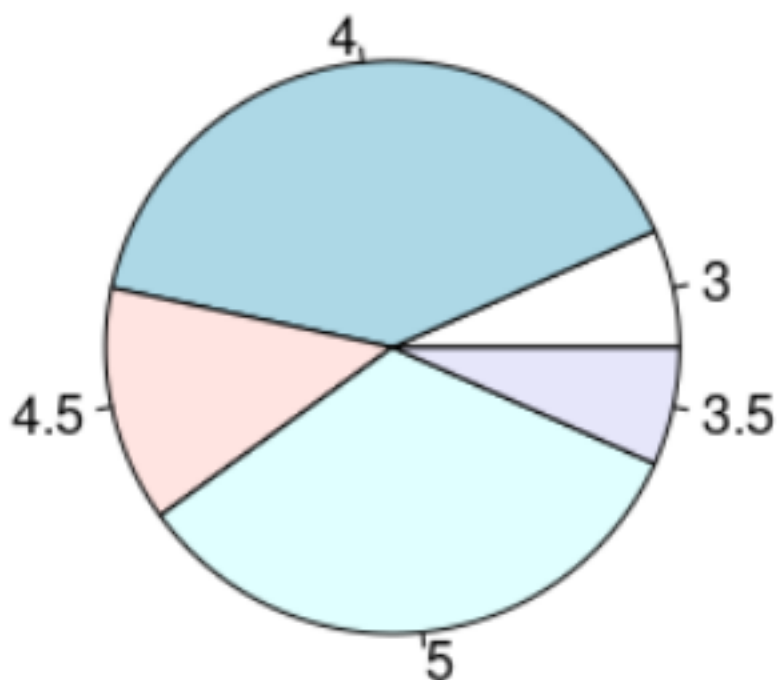
2. Wprowadzane dane:

```
>install.packages("plotrix")
>library(plotrix)
> prop.table(table(ramka$ocena))
```

3	4	4.5	5	3.5
0.06666667	0.40000000	0.13333333	0.33333333	0.06666667

3. Wynik działania:

```
> pie(prop.table(table(ramka$ocena)))
> fan.plot(prop.table(table(ramka$ocena)), labels =
  ↪ names(prop.table(table(ramka$ocena))))
```



1. Polecenie:

Do ramki danych smartfony dodaj nowa kolumn status_opinii z wartościami: "nie ma", "mniej 50 opinii", "50-100 opinii", "wiecej 100 opinii" w zależności od liczby opinii. Zamień dodana kolumnę na cechy jakościowe. Następnie przy pomocy wykresu kołowego wyrysuj procentowy udział smartfonów o konkretnym statusie opinii.

2. Wprowadzane dane:

```
> status_opinii <-  
  ifelse(ramka$liczba_opinii == 0, "nie ma",  
  ifelse(ramka$liczba_opinii < 50, "mniej niż 50 opinii",  
  ifelse(ramka$liczba_opinii <= 100, "50-100 opinii", "więcej niż  
    100 opinii")))  
> status_opinii <- factor(status_opinii)  
> ramka$status_opinii <- status_opinii
```

3. Wynik działania:

```
> pie(prop.table(table(ramka$status_opinii)))
```



1. Polecenie:

Wykorzystując ramkę danych smartfony stwórz zdanie o każdym z smartfonów postaci: nazwa + " ma ocene klientów " + ocena_klientów + "bo ma liczbę opinii" + liczba_opinii. Plus oznacza konkatenację łańcuchów i wartości.

2. Wprowadzane dane:

```
> for (i in 1:nrow(ramka)) {  
+   row <- ramka[i,]  
+   print(paste(row$nazwa, "ma ocene klientow", row$ocena, "bo  
+     ma liczbę opinii", row$liczba_opinii))  
+ }
```

3. Wynik działania:

```
[1] "Galaxy J2 Core (2020) ma ocene klientow 3 bo ma liczbe
↪ opinii 17"
[1] "Galaxy Xcover FieldPro ma ocene klientow 5 bo ma liczbe
↪ opinii 48"
[1] "Galaxy A2 Core ma ocene klientow 5 bo ma liczbe opinii 36"
[1] "Galaxy View2 ma ocene klientow 4 bo ma liczbe opinii 50"
[1] "Galaxy M30 ma ocene klientow 4 bo ma liczbe opinii 316"
[1] "Galaxy M20 ma ocene klientow 4 bo ma liczbe opinii 358"
[1] "Galaxy M10 ma ocene klientow 4 bo ma liczbe opinii 107"
[1] "Galaxy Tab Advanced2 ma ocene klientow 5 bo ma liczbe
↪ opinii 8"
[1] "Galaxy Tab A 8.0 (2018) ma ocene klientow 4.5 bo ma liczbe
↪ opinii 40"
[1] "Galaxy A6s ma ocene klientow 5 bo ma liczbe opinii 86"
[1] "Galaxy A9 (2018) ma ocene klientow 4.5 bo ma liczbe opinii
↪ 320"
[1] "Galaxy A7 (2018) ma ocene klientow 4 bo ma liczbe opinii
↪ 223"
[1] "Galaxy Note9 ma ocene klientow 5 bo ma liczbe opinii 1243"
[1] "Galaxy J6+ ma ocene klientow 4 bo ma liczbe opinii 198"
[1] "Galaxy J4 Core ma ocene klientow 3.5 bo ma liczbe opinii
↪ 87"
```

1. Polecenie:

Zachować ramke danych w pliku .csv. Załadować ramke danych z pliku .csv

2. Wprowadzane dane:

```
> write.csv(ramka, "smartfony.csv")
> read.csv("smartfony.csv")
```

3. Wynik działania:

```
> read.csv("smartfony.csv")
  X nazwa wyświetlacz pamięć_RAM pamięć_wbudowana aparat_foto cena liczba_opinii ocena status_opinii
1 1 Galaxy J2 Core (2020) 5.00 1 16 8 80 17 3.0 mniej niż 50 opinii
2 2 Galaxy Xcover FieldPro 5.10 4 64 12 1020 48 5.0 mniej niż 50 opinii
3 3 Galaxy A2 Core 5.00 1 8 5 120 36 5.0 mniej niż 50 opinii
4 4 Galaxy View2 17.30 3 64 0 660 50 4.0 50-100 opinii
5 5 Galaxy M30 6.40 3 32 13 300 316 4.0 więcej niż 100 opinii
6 6 Galaxy M20 6.30 3 32 13 300 358 4.0 więcej niż 100 opinii
7 7 Galaxy M10 6.22 2 16 13 135 107 4.0 więcej niż 100 opinii
8 8 Galaxy Tab Advanced2 10.10 3 32 8 200 8 5.0 mniej niż 50 opinii
9 9 Galaxy Tab A 8.0 (2018) 8.00 2 32 8 130 40 4.5 mniej niż 50 opinii
10 10 Galaxy A6s 6.00 6 64 12 300 86 5.0 50-100 opinii
11 11 Galaxy A9 (2018) 6.30 6 64 24 359 320 4.5 więcej niż 100 opinii
12 12 Galaxy A7 (2018) 6.00 4 64 12 309 223 4.0 więcej niż 100 opinii
13 13 Galaxy Note9 6.40 6 128 12 820 1243 5.0 więcej niż 100 opinii
14 14 Galaxy J6+ 6.00 3 32 10 230 198 4.0 więcej niż 100 opinii
15 15 Galaxy J4 Core 6.00 1 16 5 150 87 3.5 50-100 opinii
```