

SPRAWOZDANIE

Zajęcia: Analiza procesów uczenia

Prowadzący: prof. dr hab. inż. Vasyl Martsenyuk

Laboratorium Nr 1 Data 27.02.2023 Temat: "Podstawy języka R" Wariant 6	Maksymilian Grygiel Informatyka II stopień, stacjonarne, Semestr I, gr.1a
---	--

Link do repozytorium: <https://github.com/Maksiolo20/APU>

Zadania:

Wariant zadania: 4

(a) Do zmiennej a podstaw wartość wyrażenia $4 * \sin(\pi)$. Do zmiennej b podstaw potrójną wartość zmiennej a. Wywołaj funkcję sprawdzającą, która z wartości zmiennych jest większa.

```
> a <- 4*sin(pi)
> b <- 3*a
> min(a,b)
[1] 4.898425e-16
```

(b) Uruchom i poczytaj dokumentację dla funkcji max().

```
> help(max)
```

Maxima and Minima

Description

Returns the (regular or **parallel**) maxima and minima of the input values.

`pmax*()` and `pmin*()` take one or more vectors as arguments, recycle them to common length and return a single vector giving the *'parallel'* maxima (or minima) of the argument vectors.

Usage

```
max(..., na.rm = FALSE)
min(..., na.rm = FALSE)

pmax(..., na.rm = FALSE)
pmin(..., na.rm = FALSE)

pmax.int(..., na.rm = FALSE)
pmin.int(..., na.rm = FALSE)
```

(c) Stwórz wektor `a` zawierający liczby od 90 do 115. Policz średnią kwadratów liczb zawartych w wektorze.

```
> vector <- rep(90:115)
> square <- sum(vector*vector)
> square <- square/length(vector)
```

square	10562.5
--------	---------

(d) Wyświetl wszystkie funkcje zawierające frazę `max` w swojej nazwie

```
> apropos("max", mode = "function")
[1] "cummax"      "max"         "max.col"     "mem.maxNSize" "mem.maxVSize" "pmax"
[7] "pmax.int"    "promax"      "varimax"     "which.max"
```

(e) Ustaw dowolny katalog roboczy. Następnie stwórz zmienną `a` zawierającą łańcuch znaków "lodówka z największą pojemnością". 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.

```
> a <- "lodówka z największą pojemnością"
> save(a, file = "lodowka.RData")
> remove(a)
> a
Error: object 'a' not found
> load("lodowka.RData")
> a
[1] "lodówka z największą pojemnością"
```

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

```
> install.packages("gridExtra")
> library(gridExtra)
> help(package="gridExtra")
> grid.table(trees[1:10,])
```

	Girth	Height	Volume
1	8.3	70	10.3
2	8.6	65	10.3
3	8.8	63	10.2
4	10.5	72	16.4
5	10.7	81	18.8
6	10.8	83	19.7
7	11	66	15.6
8	11	75	18.2
9	11.1	80	22.6
10	11.2	75	19.9

(g) Stwórz wektor zawierający ciąg liczb 1000, 998, 996, ..., 850.

```
> vector2 <- seq(1000, 850, -2)
> vector2
 [1] 1000 998 996 994 992 990 988 986 984 982 980 978 976 974 972 970 968 966
[19] 964 962 960 958 956 954 952 950 948 946 944 942 940 938 936 934 932 930
[37] 928 926 924 922 920 918 916 914 912 910 908 906 904 902 900 898 896 894
[55] 892 890 888 886 884 882 880 878 876 874 872 870 868 866 864 862 860 858
[73] 856 854 852 850
```

(h) Stwórz wektor a z liczbami od 30 do 5 oraz wektor b z liczbami od 11 do 23. Utwórz nowy wektor d będący połączeniem wektora b i a (w takiej kolejności). Wyświetl go.

```
> a <- seq(30, 5)
> b <- seq(11, 23)
> d <- c(b, a)
> d
 [1] 11 12 13 14 15 16 17 18 19 20 21 22 23 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14
[31] 13 12 11 10 9 8 7 6 5
```

(i) Stwórz wektor nazwa zawierający nazwy 10 lodówek. Potem stwórz wektory pojemnosc_uzytkowa_chlodziarki, pojemnosc_uzytkowa_zamrazarki, cena, liczba_opinii zawierające kolejno dane 10 lodówek. Następnie stwórz ramkę danych lodówki złożoną z wektorów pojemnosc_uzytkowa_chlodziarki, pojemnosc_uzytkowa_zamrazarki, cena, liczba_opinii. Wylicz średnią cenę lodówek

```
> nazwa <- c("lodowka1", "lodowka2", "lodowka3", "lodowka4", "lodowka5", "lodowka6", "lodowka7", "lodowka8", "lodowka9", "lodowka10")
> pojemnosc_uzytkowa_chlodziarki <- c("250", "260", "240", "200", "180", "280", "250", "245", "220", "215")
> pojemnosc_uzytkowa_zamrazarki <- c("80", "70", "85", "75", "70", "90", "60", "100", "80", "75")
> ilosc_opinii <- c("10", "5", "3", "12", "11", "4", "0", "8", "30", "23")
> cena <- c(3400, 2200, 3100, 6300, 3600, 3000, 4000, 2800, 9000, 3900)
> save.image("C:/Users/Maksio/Laptop/Documents/APU/APU1.Rdata")
> lodowki <- data.frame(nazwa, pojemnosc_uzytkowa_chlodziarki, pojemnosc_uzytkowa_zamrazarki, cena, ilosc_opinii)
> mean(lodowki$cena)
[1] 4130
```

(j) Do stworzonej w poprzednim zadaniu ramki danych lodówek dodaj wpis zawierający dane nowej lodówki. Wylicz średnią ceny ponownie.

```
> lodowki_backup <- lodowki
> nowa_lodowka <- data.frame
> nowa_lodowka <- data.frame(nazwa="nowaLodowka", pojemnosc_uzytkowa_chlodziarki = "300", pojemnosc_uzytkowa_zamrazarki="90", cena = 5500, ilosc_opinii = "13")
> lodowki <- rbind(lodowki, nowa_lodowka)
> mean(lodowki$cena)
[1] 4254.545
```

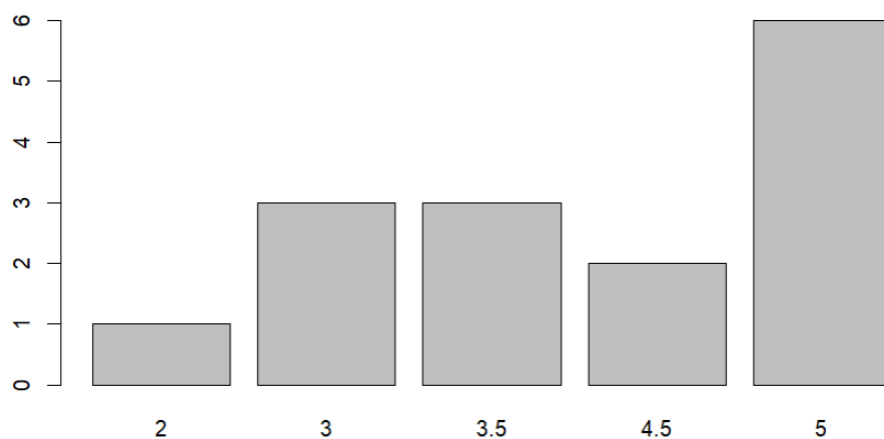
(k) Korzystając z ramki danych lodówki dodaj nową kolumnę określając ocenę klientów. Wpisz do kolumny odpowiednio oceny w skali od 0 do 5 krok 0.5. Dodana kolumna powinna się automatycznie przekonwertować do cech jakościowych (tzw. factors). Wylicz średnią ceny każdej oceny.

```
> ocena <- c(3, 5, 4.5, 5, 5, 5, 3.5, 3, 5, 2, 3)
> lodowki_backup <- lodowki
> lodowki <- cbind(lodowki, ocena)
> tapply(lodowki$cena, lodowki$ocena, mean)
2200 2800 3000 3100 3400 3600 3900 4000 5500 6300 9000
2200 2800 3000 3100 3400 3600 3900 4000 5500 6300 9000
>
```

(l) Do ramki danych aparaty dodaj kolejne 4 lodówki. Narysuj na wykresie słupkowym liczebność reprezentantów każdej z ocen klientów

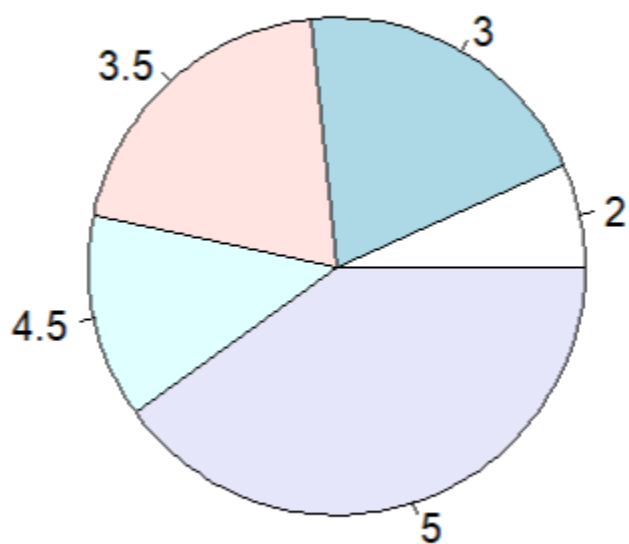
```
> lodowka_nowa_2_1 <- data.frame(nazwa = "lodowkaNowa2_1", pojemnosc_uzytkowa_chlodziarki="200", pojemnosc_uzytkowa_zamrazarki = "65", cena = 4500, ilosc_opinii = "38", ocena = 5)
> lodowka_nowa_2_2 <- data.frame(nazwa = "lodowkaNowa2_2", pojemnosc_uzytkowa_chlodziarki="180", pojemnosc_uzytkowa_zamrazarki = "60", cena = 4000, ilosc_opinii = "15", ocena = 3.5)
> lodowka_nowa_2_3 <- data.frame(nazwa = "lodowkaNowa2_3", pojemnosc_uzytkowa_chlodziarki="280", pojemnosc_uzytkowa_zamrazarki = "90", cena = 6000, ilosc_opinii = "1", ocena = 4.5)
> lodowka_nowa_2_4 <- data.frame(nazwa = "lodowkaNowa2_4", pojemnosc_uzytkowa_chlodziarki="240", pojemnosc_uzytkowa_zamrazarki = "90", cena = 8000, ilosc_opinii = "8", ocena = 4.5)
> lodowki <- rbind(lodowki, lodowka_nowa_2_1)
> lodowki <- rbind(lodowki, lodowka_nowa_2_2)
> lodowki <- rbind(lodowki, lodowka_nowa_2_3)
> lodowki <- rbind(lodowki, lodowka_nowa_2_4)
```

```
> liczebnosc <- table(lodowki$ocena)
> barplot(liczebnosc)
```

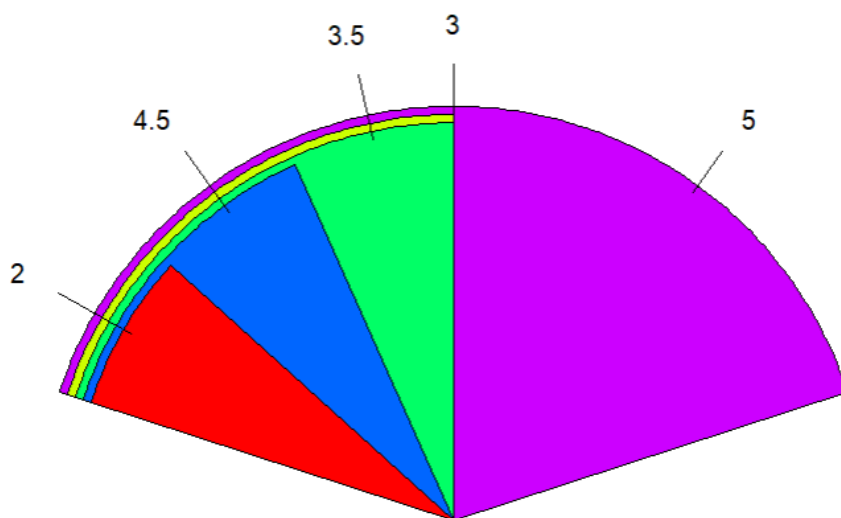


(m) Wykorzystując ramkę danych lodowki pokaż procentowy udział każdej oceny przy pomocy wykresu kołowego oraz wachlarzowego.

```
> barplot(liczebnosc)
> procenty <- liczebnosc/sum(liczebnosc)
> pie(procenty)
```



```
> library(plotrix)
> fan.plot(liczebnosc, labels = names(liczebnosc))
```

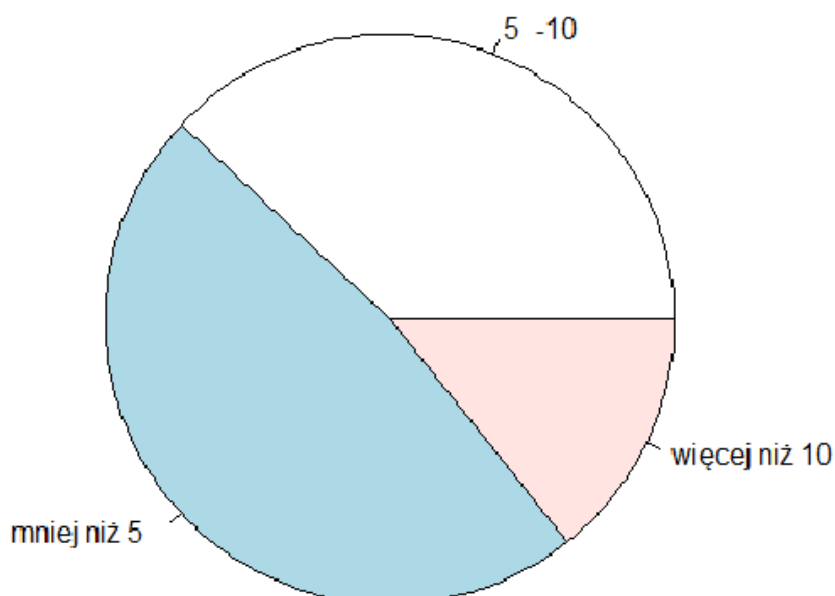


(n) Do ramki danych lodówki dodaj nową kolumnę status_opinii z wartościami: “nie ma”, “mniej 50 opinii”, “50-100 opinii”, “więcej 100 opinii” w zależności od liczby opinii. Zamień dodaną kolumnę na cechy jakościowe. Następnie przy pomocy wykresu kołowego wyrysuj procentowy udział lodówek o konkretnym statusie opinii. W zadaniu zamieniono kryteria – granice ustawiono na “mniej niż 5 opinii” “5-10 opinii”, “więcej niż 10 opinii”, by lepiej się zgadzało z danymi w zadaniu:

```

> lodowki[, "status_opinii"] <- ifelse(lodowki$liczba_opinii <= 0, "nic", "mniej niż 5")
> lodowki[, "status_opinii"] <- ifelse(lodowki$liczba_opinii <= 5, "mniej niż 5", "5-10")
> lodowki$status_opinii <- ifelse(lodowki$liczba_opinii <= 10, lodowki$status_opinii, "więcej niż 10")
Error in `<-data.frame`(`*tmp*`, status_opinii, value = logical(0)) :
  zamiana ma 0 wierszy, dane mają 11
> lodowki[, "status_opinii"] <- ifelse(lodowki$liczba_opinii <= 10, lodowki$status_opinii, "więcej niż 10")
> pie(table(lodowki$status_opinii))

```



(o) Wykorzystując ramkę danych lodowki stwórz zdanie o każdej z lodówek postaci: nazwa + " ma ocenę klientów " + ocena_klientów + " bo ma liczbę opinii" + liczba_opinii. Plus oznacza konkatencję łańcuchów i wartości.

```

> paste(lodowki$nazwa, "ma ocenę klientów", lodowki$ocena, "bo ma liczbę opinii", lodowki$liczba_opinii)
[1] "lodowka1 ma ocenę klientów 3 bo ma liczbę opinii 10"
[2] "lodowka2 ma ocenę klientów 5 bo ma liczbę opinii 5"
[3] "lodowka3 ma ocenę klientów 4.5 bo ma liczbę opinii 3"
[4] "lodowka4 ma ocenę klientów 5 bo ma liczbę opinii 12"
[5] "lodowka5 ma ocenę klientów 5 bo ma liczbę opinii 11"
[6] "lodowka6 ma ocenę klientów 5 bo ma liczbę opinii 4"
[7] "lodowka7 ma ocenę klientów 3.5 bo ma liczbę opinii 0"
[8] "lodowka8 ma ocenę klientów 3 bo ma liczbę opinii 8"
[9] "lodowka9 ma ocenę klientów 5 bo ma liczbę opinii 30"
[10] "lodowka10 ma ocenę klientów 2 bo ma liczbę opinii 23"
[11] "nowaLodowka ma ocenę klientów 3 bo ma liczbę opinii 13"

```

(p) Zachować ramkę danych w pliku .csv. Załadować ramkę danych z pliku .csv

```

> write.csv(lodowki, "lodowki_data.csv")
> dane <- read.csv("lodowki_data.csv")

```

Wnioski:

1. Laboratorium wprowadziło w podstawy języka programowania R oraz jego podstawowe zastosowania - działanie i tworzenie ramek danych

2. Poznano strukturę i składnię języka R oraz podstawowe operacje na danych.
3. Laboratorium było bardzo pomocne w zdobyciu umiejętności pracy z danymi i zaprojektowania prostych analiz w języku R.