

## LAB\_1

Wykonaj poniższe instrukcje ze zrozumieniem – jak coś nie wiesz szukaj w opisie języka R.

1. Użycie `c()` do wprowadzania danych. Wprowadzanie danych 74 122 235 111 292 111 211 133 156 79 z użyciem `c()` pod zmienną o nazwie `whales`:

```
> whales = c(74, 122, 235, 111, 292, 111, 211, 133, 156, 79)
```

Gdy chcemy wiedzieć co zmienna `whales` zawiera piszemy:

```
> whales
```

2. Dane wektora `c()` muszą być tego samego typu i tak poniżej mamy wektor ciągów znakowych (łańcuchów).

```
> Simpsons = c("Homer", "Marge", "Bart", "Lisa", "Maggie")
```

Nadanie nazw wprowadzanym danym poprzez `names()`.

```
> names(simpsons) = c("dad", "mom", "son", "daughter1", "daughter 2")
```

```
> names(simpsons)
```

```
> simpsons
```

```
dad      mom      son      daughter 1 daughter 2
```

```
"Homer" "Marge" "Bart" "Lisa"      "Maggie"
```

3. Wzór na wariancję przedstawia się następująco:

$$\text{VAR}(x) = \frac{(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \dots + (x_n - \bar{x})^2}{n - 1}$$

Wariancja to suma średnich kwadratu odległości liczb z danego zbioru od średniej z liczb tego zbioru. Liczymy to w następujący sposób:

```
> x = c(2,3,5,7,11) #zbiór liczb
```

```
> xbar = mean(x) # średnia
```

```
> x-xbar #odległość liczby od średniej
```

```
> (x-xbar)^2 #kwadrat odległości liczby od średniej
```

```
> sum((x-xbar)^2) #suma kwadratów odległości liczb od średniej
```

```
> n = length(x)
```

```
> n
```

```
> sum((x-xbar)^2)/(n-1) #wariancja
```

### Wykres słupkowy

```
> beer=scan() # scan() pozwala na wczytanie z klawiatury, z której wpisujemy
```

```
3 4 1 1 3 4 3 3 1 3 2 1 2 1 2 3 2 3 1 1 1 1 4 3 1
```

Powyższe dane oznaczają np. preferencje 25 studentów odnośnie marki piwa 1, 2, 3 i 4.

Dla tych danych uzyskujemy wykresy słupkowe częstotliwości występowania marek i proporcje.

```
> barplot(table(beer), # częstotliwość
```

```
+ xlab="beer", ylab="frequency") #jedynek jest 10, dwójek 4, trójek – 8, czwórek -3
```

```
> barplot(table(beer)/length(beer), # proporcje 0.4, 0.16, 0.32, 0.12
```

```
+ xlab="beer", ylab="proportion") # Znak "+" jest symbolem kontynuacji komendy.
```

```

> sales = c(45,44,46) # sprzedaż kwartalna
> names(sales) = c("John","Jack","Suzy") # użycie nazw dla danych
> barplot(sales, main="Sales", ylab="Thousands") # wykres słupkowy
> barplot(sales, main="Sales", ylab="Thousands",
+ ylim=c(42,46), xpd=FALSE)

> barplot(central.park$MAX, # mamy tu maksymalną wartość temperatury w ciągu dnia
+ names.arg=1:31,          # liczba dni to 31
+ xlab="day", ylab="max. temp.") # na osi x mamy dni, na osi y mamy max temperaturę

```

Zapisanie wykresu słupkowego pod zmienną bp i potem wykreślenie go jako tekst.

```

> our.data = c(1,2,2.5); names(our.data)=1:4
> bp = barplot(our.data)
> text(bp, our.data, labels = our.data, pos = 1)

```

### Wykresy ciasteczkowe

```

> sales
John Jack Suzy #nazwy danych
45 44 46 #wartości danych
> pie(sales, main="sales") #wykres ciasteczkowy - kołowy

```

### Wykres punktowy (dla danych zapisanych w sales).

```

> dotchart(sales,xlab="Amount of sales")

```

dane dwuwymiarowe

	Child	
	buckled	unbuckled
Parent		
buckled	56	8
unbuckled	2	16

Tworzenie powyższej tabeli odbywa się w następujący sposób

```

> rbind(c(56,8),c(2,16)) # tworzenie tablicy wierszami
> cbind(c(56,2),c(8,16)) # tworzenie tablicy kolumnami
> x = matrix(c(56,2,8,16),nrow=2) #korzystając z macierzy
> x

```

Przypisanie nazw wierszom i kolumnom dokonuje się poniżej

```

> rownames(x) = c("buckled","unbuckled")
> colnames(x) = c("buckled","unbuckled")
> x

```

### WYKRESY SŁUPKOWE

```

> barplot(x, xlab="Parent", main="Child seat-belt usage")
> barplot(x, xlab="Parent", main="Child seat-belt usage",beside=TRUE)
|> barplot(x,main="Child seat belt usage",legend.text=TRUE)

```

Porównanie dwóch zbiorów danych.

Wykres skrzyneczek

```
> pl = c(0, a, a, 2, 4, 5, 14, 14, 14, 13, 17, 17, 15)
> ep = c(0, 6, 7, 9, 11, 13, 16, 16, 16, 17, 18, 20, 21)
> boxplot(pl,ep, names=c("placebo", "ephedra"))
```

Zestawienie funkcji gęstości

```
> plot(density(pl),ylim=c(0,0.07), main="densityplots of ep and pi")
> lines(density(ep), lty=2)
```

Rozkład prawdopodobieństwa

```
> k = 0:4 #zakres wartości, które mogą być wylosowane
> p=c(1,2,3,2,1)/9 #prawdopodobieństwo przypisane wylosowanym wartościom
> plot(k,p,type="h",xlab="k", ylab="probability",ylim=c(0,max(p))) #wykres szpilkowy
> points(k,p,pch=16,cex=2) #kulki u szczytu
```

Użycie sample() do generowania liczb losowych.

```
> k = 0:2 # zakres wartości
> p = c(1,2,1)/4 #prawdopodobieństwa
> sample(k,size=1,prob=p) #funkcja sample generuje losowe wartości
> sample(k,size=1,prob=p)
> sample(1:6,size=1) +sample (1:6, size=1)
> sample(1: 6, size=1)+sample (1 : 6, size=1)
> sample(0:1,size=10,replace=TRUE)
> sample(1:6,size=10,replace=TRUE)
> sample( 1: 6, size=10,replace=TRUE) + sample( 1 : 6, size=10,replace=TRUE)
> sample(rep(0:1,c(3200,6800)),size=10,replace=T) # zer jest 3200, a jedynek 6800
> sample(0:1,size=10,replace=T,prob=c( 1-.62, .62)) #zero z prawd 0.38 i jedynek z 0.62
```

Zmienna o rozkładzie normalnym i zmienna standardowa  $z = (X - \mu) / \sigma$

```
> pnorm(1.5, mean=0,sd=1) #zmienna losowa standardowa
> pnorm(4.75, mean=4,sd=1/2) # zmienna o rozkładzie normalnym i wartości średniej 4.0
```

Zadanie:

1. Wprowadź dwa zbiory po 20 cyfr z przedziału od 1 do 7. Zbadaj ich funkcję gęstości.

2. Wprowadź dwa zbiory po 15 cyfr w sposób losowy. Oblicz średnią dla każdego ze zbioru. Uzyskaj wykresy skrzyneczkowe oraz słupkowe dla każdego dla tych zbiorów.