

# Modele i wnioskowanie statystyczne

## Wprowadzenie do R

Konrad Jędrzejewski

ISE PW

24 października 2012

# Informacje ogólne

- R jest pakietem (środowiskiem) przeznaczonym do zaawansowanych obliczeń statystycznych.
- Licencja GNU GPL – całkowicie bezpłatny.
- Źródła: <http://www.r-project.org>.
- CRAN (*Comprehensive R Archive Network*).
- Platformy: Windows, Linux, Unix, MacOS.
- Język R jest językiem interpretowanym, a nie kompilowanym.
- R Commander.

# Środowisko

- Konsola
  - >
  - +
- Rozróżnialność małych wielkich liter
- Mechanizm „strzałek” – poprzednie komendy
- Pomoc
  - **help**(nazwa),
  - ?nazwa
  - **apropos**(nazwa)
  - ??nazwa

# Środowisko

- Funkcje – wywołanie.
  - nazwa(arg1, arg2, arg3 = wartość)
  - `x = 1:100; y = rnorm(100); plot(x, y, type = "l")`
  - **args**(legend)
  - **example**(legend)
- # komentarz
- Przydatne funkcje
  - **ls()**, **objects()**
  - **rm()**
  - **print**("napis")
  - **print**(dane)

# Środowisko - pakiety

- Pakiety
  - **library()** – lista pakietów zainstalowanych
  - **search()** – lista pakietów załadowanych
- Załadowanie pakietu
  - **library(nazwa\_pakietu)**
  - **require(nazwa\_pakietu)**
- Usunięcie pakietu (z pamięci)
  - **detach(package:nazwa\_pakietu)**

# Własne skrypty, funkcje

- Katalog roboczy
  - **getwd()**, **setwd("nazwa\_katalogu")**
  - **dir()**, **list.files()**
- Uruchamianie własnych skryptów
  - **source("nazwa\_skryptu")**
- Dołączanie własnych funkcji
  - **source(„nazwa\_pliku")**
  - **ls()**
    - „funkcja1\_z\_nazwa\_pliku”      „funkcja2\_z\_nazwa\_pliku”
  - **rm("funkcja1\_z\_nazwa\_pliku")**

# Wczytywanie i zapisywanie danych

- Pobieranie z plików:
  - dane = **scan**('c:/plik.txt')
  - dane = **read.table**('plik.txt', **header** = T)
  - dane = **read.csv**('Zeszyt1.csv', **sep** = ";" **header** = T, **dec** = ',')
  - **names**(dane), **rownames**(dane), **dimnames**(dane)
  - **write**(x, 'plik.txt')
  - **write.table**(dane, **file** = 'plik.txt'), **write.csv**()
- Edycja/zmiana danych
  - **edit**(dane), **fix**(dane)
- Pobieranie z pakietów
  - **data**(uspop, **package** = 'datasets')

# Typy danych

- Numeryczny
  - 2.345
  - 3.5e-15
- Znakowy
  - 'a', "abc", \n, \t
- Zespólny
  - $x = 3 + 4i$       **Mod(x), Arg(x), Re(x), Im(x)**
- Logiczny
  - TRUE\T, FALSE\F



# Struktury danych

- Wektor
- Tablica/macierz
- Faktor (factor)
- Lista
- Ramka (dataframe)

# Wektor

- Tworzenie
  - `x = c(1,2,3,4)`, `x <- c(1,2,3,4)`
  - `x = c("bdb","db","dst","bdb")`
  - `x = c(TRUE,FALSE,TRUE,FALSE)` ; `y = c(T,F,T,F)`
- Indeksowanie wektorów
  - `x[3]`
  - `x[2:4]`
  - `x[c(2,5,8)]`
- Operacje arytmetyczne

# Generowanie wektorów

- **:**
  - `x = 1:100`
  - `x = 100:1`
- **seq()**
  - `x = seq(0, 5, by = 0.25)`
  - `x = seq(0, 5, length = 10)`
- **rep()**
  - `x = rep(c(1,2,3), 4)`
  - `x = rep(c(1,2,3), each = 4)`

# Tablica

- Tablica (array) jest wektorem zawierającym dodatkowe dane określające uporządkowanie elementów w tablicy.
- **dim()**
  - `x = 1:20`
  - `dim(x) = c(4,5)`
  - `attributes(x)`
  - `dimnames(x) = list(letters[1:4],LETTERS[1:5])`

# Tablica

- **matrix()**
  - `matrix(1:20, 4, 5)`
- **array()**
  - `array(1:20,c(4,5))`
- **rbind(), cbind()**
  - `x= rbind(1:3,4:6); y = cbind(1:3, 4:6)`
- Mnożenie macierzy
  - `z = x%*%y`
- Indeksowanie
  - `x[2,3], x[1:3,1:2], x[2,], x[,3]`

# Faktor (factor)

- Faktor (factor) jest strukturą przechowującą oprócz szeregu danych informacje o powtórzeniach takich samych wartości oraz zbiorze unikalnych wartości.
- **factor()**
  - faktor = **factor**(c(2,3,4), levels=1:5)
  - punkty = **factor**(c(95,56,74,80,52,99,35,74))
  - oceny = **factor**(c("bdb","dst","db","dst","bdb","ndst","db"))
- **levels()**
  - **levels**(oceny)
- **table()**
  - **table**(oceny)

# Lista

- Lista (list) jest uporządkowanym zbiorem elementów różnego typu.
  - Lista = **list**("Jan","Kowalski",1990,"Warszawa","TRUE")
  - Lista = **list**(imie="Jan",nazwisko="Kowalski",rok\_ur=1990,zam="Warszawa",stud="TRUE")
- Wybór z listy
  - Lista\$nazwisko
  - Lista[2][1]
- Dodawanie
  - Lista\$imie[2] = „Jakub”; Lista\$nazwisko[2] = "Nowak"; ...
  - Lista2 = **list**(imie=c("Jan","Piotr"),nazwisko=c("Kowalski","Nowak"),rok\_ur=c(1991,1995),zam=c("Warszawa","Poznan"),stud=c(T,F))

# Ramka

- Ramka (dataframe) jest macierzą, w której poszczególne kolumny mogą zawierać wartości różnego typu.
- Tworzenie
  - `ramka = data.frame(LETTERS[1:6], seq(10,60, by = 10), seq(10,60, by = 10) > 35)`
  - `names(ramka) = c("Litera","Punkty"," Punkty > 35")`
- Wybór z ramki
  - `ramka[3,] ; ramka[,2]`
  - `ramka$Punkty`
  - `ramka$Litera[2]`
  - `ramka$"Punkty > 35"`



# Generowanie liczb losowych

- **sample()**
  - **sample(1:6, 4, replace = T)**
- Rozkłady zmiennych – binom, geom, hyper, pois, norm, unif, exp, chisq, f, t, beta, gamma
- Przedrostki: r, d, p, q
  - **x = rnorm(100, mean = 2, sd = 3)**
  - **dnorm(0, mean = 0, sd = 1) ; dunif(0.4, min=0, max=2)**
  - **pnorm(0, mean = 0, sd = 1) ; punif(0.4, min=0, max=2)**
  - **qnorm(0.95, mean = 0, sd = 1); qunif(0.4, min=0, max=2)**
- Podstawowe funkcje
  - **mean(), sd(), var(), median(), quantile()**

# Wykresy

- Podstawowe funkcje
  - **plot()**,
  - **plot(x,y, xlab = "opis x", ylab = "opis y", main = "tytul")**
  - kolejne dane na wykresie - **points()**, **lines()**
  - **hist(rnorm(1000))**
  - **pie(1:6, labels = LETTERS[1:6])**
  - **grid()**, **title()**, **legend()**
- Zarządzanie oknami
  - **dev.new()**, **x11()**, **dev.off()**, **dev.cur()**, **dev.set(nr\_device)**

# Wykresy

- Zapisywanie
  - **> jpeg('rys.jpg')**
    - > plot(x,y)**
    - > dev.off()**
  - **> plot(x,y)**
    - > dev.copy(png, 'rys.png')**
    - > dev.off()**
  - **> dev.print(pdf, 'rys.pdf')**

# Programowanie w R

- Instrukcja warunkowa
- Pętle
- Skrypty
- Funkcje

# Instrukcja warunkowa

- **if**(warunek) wyrażenie
- **if**(warunek) wyrażenie1 **else** wyrażenie2
- **ifelse**(warunek, a, b)
- **switch**(zmienna, wartosc1 = akcja1, wartosc2 = akcja2, ...)
- Operatory logiczne: **&**, **|**, **!**, **xor(x,y)**, **==**, **!=**, **<**, **>**, **<=**, **<=**, **isTRUE(x)**, **&&**, **||**.

# Pętle

- **for**(licznik **in** start:koniec) wyrażenie
- **while**(warunek) wyrażenie
- **repeat** wyrażenie
- **break**
- **next**
- Przykłady

# Skrypty

- **source("nazwa\_skryptu.R")**

- Przykład

Skrypt1.R

```
x = 1:100
```

```
y = x^2
```

```
plot(x, y, type="l")
```

```
grid()
```

```
> source("Skrypt1.R")
```

# Funkcje

- nazwa\_funkcji = **function**(arg1, arg2, arg3 = wartość)  
{ciało funkcji}
- Zwracane wartości - ostatnia linia
- **return()**
- Przeciążanie funkcji
- **stop()**
- **warning()**