Regresní analýza dat
01REAN - Cvičení 01

Jiri Franc

Czech Technical University
Faculty of Nuclear Sciences and Physical Engineering
Department of Mathematics

Úvod do R

Osnova dnešního cvičení:

- O čem a k čemu slouží R.
- Kde stáhnout a jak spustit R.
- Užitečné Editory pro práci s R.
- První kroky v R.
- Základní pravděpodobnost a statistika v R.
- Načtení a zobrazení dat.

Co je a k čemu slouží R?

- R je software zaměřený na použití při statistickém počítání, analýze dat a grafickém zobrazení.
- R má počátek a kořeny v jazyku S, který se vyvíjel již od šedesátých let v Bellových laboratořích.
- R na rozdíl od S byl odpočátku koncipován jako open source projekt.
- R a jeho zdrojové kódy jsou k dispozici jako volně šiřitelný software s podmínkami Free Software Foundation's GNU General Public License.

Kde získat R?

R distribuuje Comprehensive R Archive Network (CRAN) a lze volně stáhnout ze stránek projektu: http://cran.r-project.org, kde je k dispozici pro nejrozšířenější operační systémy:

- **Microsoft Windows**:
  http://cran.r-project.org/bin/windows/base/
- **MacOS**:
  http://cran.r-project.org/bin/macosx/
- **Linux**:
  http://cran.r-project.org/bin/linux/

k dispozici jsou jak binární soubory, tak všechny zdrojové kódy i nástroje pro vytváření vlastních knihoven.

Balíčky a knihovny v R

V R, kromě funkcí, které jsou obsaženy v samotném základním programu (base), existuje mnoho rozšiřujících balíčků (package).

Balíček je souborem R funkcí, dat a je kompilován do kódu, který se uložen do knihovny (library).

Základní balíčky nainstalované spolu s R:

```
> (getOption("defaultPackages"))
[1]"datasets" "utils" "grDevices" "graphics" "stats" "methods"
```

nebo

```
> sessionInfo()
R version 3.1.2 (2014-10-31)
Platform: x86_64-w64-mingw32/x64 (64-bit)
locale:
[1] LC_COLLATE=Czech_Czech Republic.1250 LC_CTYPE=Czech_Czech Republic
[3] LC_MONETARY=Czech_Czech Republic.1250 LC_NUMERIC=C
[5] LC_TIME=Czech_Czech Republic.1250
attached base packages:
[1] stats graphics grDevices utils datasets methods base
loaded via a namespace (and not attached):
[1] tools_3.1.2
```

Balíčky a knihovny v R

Vznik a aktualizace nových balíčků:
https://cran.r-project.org/web/packages/available_packages_by_date.html

Balíček lze stáhnout a nainstalovat při připojení k internetu přímo z jednoho z mnoha úložišť na světě (v ČR provozuje cz.nic).
Zjistit jaké balíčky mám již nainstalovány (jaké používám knihovny).

```
library()
```

Nápověda k instalaci balíčků

```
> ?install.packages
```

Balíčky a knihovny v R

Příklad:
Instalace balíčku "car"- Companion to Applied Regression.

```
install.packages("car")
```

Načtení knihovny (balíčku) umožní používání funkcí v ní obsažených.

```
library(car)
```

Nápověda k dané knihovně

```
library(help = "car")
```

V R-studiu lze balíček vyhledat v databázi a nainstalovat pouhým "klikáním".
Lze také stáhnout binární soubory k balíčkům, upravovat je a instalovat z místního úložiště. Viz volby ve funkci `install.packages`.

Základní Nápověda a Dokumentace

Základní formy dokumentace pro R jsou:

- Online nápověda, která je součástí základní distribuce a každého balíčku.
- Elektronický manuál v pdf ke každému balíčku.
- Knihy popisující základní funkce v R.
- Články popisující často nové funkce v R.

Manuál k jednotlivým balíčkům

Manuál k jednotlivým balíčkům lze stáhnout ve formátu pdf z příslušné webové stránky.
Například: pro balíček "car": https://cran.r-project.org/web/packages/ca

Nebo zkrácenou verzi popisující funkcionality balíčku otevřít přímo z konzole:

Zjištění všech dostupných:

```
vignette(all = T)
```

a otevření vybrané dostupné:

```
vignette("embedding", package = "car")
```

Manuál jednotlivých funkcí

Manuál k dané funkci je k nalezení buď přímo v manuálu příslušného balíku, nebo opět přímo z konzole.

Buď pomocí funkce `help`

```
help("mean")
```

nebo zkráceně

```
?mean
```

Pro nápovědu k danému balíku (né konkrétní funkci používáme)

```
help(package = "car")
```

R Manuály: edited by the R Development Core Team

Základní úvod a nezbytné informace jsou ve volně dostupných maunálech vydávaných a aktualizovaných přímo R Development Core Teamem.

- **An Introduction to R**: úvod do R spolu se statistickou analýzu dat a grafikou.
- **R Data Import/Export**: popisuje možnosti importu a exportu dat a rozšiřující balíčky k tomu určené.
- **Writing R Extensions**: popisuje jak vytvořit vlastní balíček, jak psát nápovědy a jak využívat jiných jazyků.
- **R Installation and Administration**.
- **The R language definition**.
- **R Internals**.
- **The R Reference Index**.

Další online zdroje k R

Online a další dokumentace:

- Quick-R http://www.statmethods.net/index.html
- **R-bloggers** -novinky, zprávy a návody od cca 750 R bloggerů.
- FAQ na stránkách R-projektu: http://CRAN.R-project.org/faqs.html
- Novinky na stránkách R-projektu: http://CRAN.R-project.org/doc/Rnews/
- Stackoverflow: https://stackoverflow.com/questions/tagged/r
- kaggle - zdroj dat a návodů pro práci s daty nejen v R: https://www.kaggle.com

Komunikace s R - Editory

- Nejednoduší způsob: po jednom řádku přímo v konzoli.
- Víceúčelové editory: Emacs, Kate, RWinEdt, atd..
- Speciální editory:
  - Tinn-R
  - **R-studio**
    -doporučeno,
    -je nainstalováno jak na školních počítačích tak na citrixu
    https://sf.fjfi.cvut.cz/Citrix/StoreWeb/
    - stahujte volně z:
    https://www.rstudio.com/products/rstudio/download/

Aritmetika v R

```
> ## Aritmetika - R jako kalkulacka
> 40 + 2                  # sčítání
[1] 42
> 44 - 2                  # odčítání
[1] 42
> 6  * 7                  # násobení
[1] 42
> 294 / 7                 # dělení (mezery mohou a nemu
[1] 42
> 42^7                    # mocnina
[1] 230539333248
> sqrt(1764)              # druhá odmocnina
[1] 42
> 230539333248^(1/7)      # (od)mocnina, závorky nutné
[1] 42
```

Aritmetika v R

```
> #Konstanty a funkce
> exp(1)                  # eulerovo cislo
[1] 2.718282
> exp(42)                 # exponenciela
[1] 1.739275e+18
> pi                      # konstanta pi
[1] 3.141593
> sin(pi/2)               # sinus
[1] 1
> log(exp(1))             # prirozený logaritmus
[1] 1
> ln(exp(1))              # funkci ln R nezna !
Error: could not find function "ln"
> factorial(42)           # 42!
[1] 1.405006e+51
> choose(5, 2)            # 5 nad 2 = kombinacni cislo
[1] 10
```

Změna počtu zobrazených číslic

```
> 10/3
[1] 3.333333
> 10/3
[1] 3.333333
> options(digits = 15) #chcili zobrazit vice cislic
> 10/3
[1] 3.33333333333333
> options(digits = 7)  # zpet na 7
```

Práce s proměnnými v R

```
> ### Prace s promennymi
> x <- 42  # ulozeni hodnoty do
> x        # vytisteni hodnoty v promenne ulozene
[1] 42
> x = 42   # nelze pouzit v kombinaci s jinym prikazem
> y <- 3   # uložení do jiné proměnné
> x + y
[1] 45
z <- x + y
# vyzkousejte print(z <- x + y) a print(z = x + y)

> ### Vymazani promennych
> ls()           # vypis pouzivanych a definovanych objektu
[1] "x" "y" "z"
> rm(list=ls()) # vymaze vsechn objekty
> ls()
character(0)
```

Datové typy

```
> sqrt(-1)                # není definováno
[1] NaN
> sqrt(-1+0i)             # je definováno
[1] 0+1i
> sqrt(as.complex(-1)) # podobně
[1] 0+1i
> (0 + 1i)^2              # umí
[1] -1+0i
> typeof((0 + 1i)^2)
[1] "complex"
> x <- (0 + 1i)^2
> x
[1] -1+0i
> y <- as.numeric(x)
> y
[1] -1
> class(y)
[1] "numeric"
> class(x)
[1] "complex"
> y == x
[1] TRUE
```

Datové typy

```
a <- c(1,2,5.3,6,-2,4)                    # numeric vektor
b <- c("one","two","three")               # character vektor
c <- c(TRUE,TRUE,TRUE,FALSE,TRUE,FALSE # logical vektor
d <- matrix(1:20, nrow=5,ncol=4)          # matice
```

Dále potkáme a budeme se jim věnovat více příští hodinu:
`Arrays, Data Frames, Lists, Factors`
Vyzkoušejte další možnosti zadávání a práce s vektory:

```
seq(from = 1, to = 5)
seq(from = 2, by = -0.1, length.out = 4)
1:5
[1] 1 2 3 4 5
x <- c(74, 31, 95, 61, 76, 34, 23, 54, 96)
x[2:4]
x[c(1, 3, 4, 8)]
x[-c(1, 3, 4, 8)]
LETTERS[1:5]
letters[-(6:24)]
```

Datové typy

Vyzkoušíme práci s data frame - načteme data trees

```
#### data frame
trees
head(trees)      # prvnich par radku
summary(trees)   # prehled promennych
table(trees[,"Girth"])
str(trees)

Girth            # nezna
trees$Girth      # takto nahlidneme na prvni promenou
G = trees$Girth # uz zname G
attach(trees)
Girth
detach(trees)
```

Datové typy

Vyzkoušíme práci s data frame - načteme data trees

```
## Lze pracovat jen s nekterými sloupci, radky databaze

trees[1,]        # jen 1. radek
trees[-1,]        # bez 1. radku
trees[c(2,3),]         # jen konkretni radky
trees[,c("Girth","Volume")]
trees[!trees$Volume,] # vykricnik  znamena negaci
trees[trees$Height>60,] # jen radky splnujici podminku
# podobne pro sloupce
```

Úkoly

- Spočtěte hodnotu funkce $f(x \mid \mu, \sigma^2)$ v bodě $x = 2$ pro všechny kombinace parametrů $\mu = [0, 2]$ a $\sigma = [1, 2]$

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

  Zkuste najít v R zabudovanou funkci která je k tomuto výpočtu určená a použijte ji.

- Spočtěte hodnotu funkce $P(k \mid n, p)$ pro parametry $n = 5$, $p = 0.4$ a pro hodnoty $k = [0, 1, \ldots, 5]$.

$$P(k, n, p) = \binom{n}{k} p^k (1-p)^{n-k}$$

  Zkuste najít v R zabudovanou funkci která je k tomuto výpočtu určená a použijte ji.

Úkoly

- Zapiště předchozí výpočet pomocí vektoru a poté pomocí funkce for (smyčka)

```
for (k in 0:5){
#zde prijde kod pro vypocet
}
```

- Zkuste napisat vlastní funkci pro výpočet pravděpodobnostní funkce binomického rozdělení.

Úkoly

- Načtěte si data `trees` z base balicku datasets.
- Spočtěte střední hodnotu výšky a objemu stromů.
- Spočtěte BMI index stromu za předpokladu, že objemová hmotnost dřeva všech uvedených stromů je konstantní a rovna 900 kg / m^3, Výsledek ulože do tabulky trees jako novou proměnnou.
- Koukněte na tabulku (`table`) vypočtených hodnot, rozdělte ji na 3 skupiny - a každému záznamu přiřaďte proměnnou typu faktor.
  př:

```
trees$obezita.stromu = ...
  factor(trees$BMI,levels=c("hubeny","normalni","tlusty")))
```