

Az R használata pénzügyi matematikához

Az R statisztikai programcsomag letölthető az alábbi címről:

<http://www.r-project.org>

Az RStudio letölthető az alábbi címről:

<https://www.rstudio.com/products/rstudio/download/>

Alapműveletek, elemi függvények az R-ben

- ▶ $+$, $-$, $*$, $/$
- ▶ hatványozás: $^$, pl.: 2^3
- ▶ exponenciális függvény: `exp(2)`
- ▶ természetes (e) alapú logaritmus: `log(3)`
- ▶ 10-es alapú logaritmus: `log10(100)`
- ▶ a -alapú logaritmus: `log(x, a)`
- ▶ trigonometrikus függvények: `sin`, `cos`, `tan`, pl.: `sin(pi/2)`

Értékadás

- ▶ $a < -4 + 5$
- ▶ $B2 = \text{sqrt}(a)$ (sqrt: négyzetgyök-vonás)
- ▶ $a - B2$

A változónevek:

- ▶ nem kezdődhetnek számmal
- ▶ nem tartalmazhatnak szóközt
- ▶ case sensitive

Vektorok

A vektorok elemei azonos típusúak: numeric, character, logical, ...

Típus lekérdezése: `typeof(v)`

Vektor definiálása:

- ▶ `v<-c(0,1,-2.4,8)`
- ▶ `vektor=1:30` (vagy fordítva: `w=21:17`)

Vektorok kiegészítése, összefűzése:

- ▶ `X=c(v,3.15)`
- ▶ `y2=c(X,w)`

Vektorokhoz kapcsolódó parancsok:

- ▶ `length(v)`: a `v` vektor hossza
- ▶ `min(v)` és `max(v)`: a vektor legkisebb és legnagyobb eleme
- ▶ `sort(v)`: a `v` vektor elemeit sorbarendezi
- ▶ `sample(v)`: az elemeket véletlenszerű sorrendben sorolja fel
- ▶ `sum(v)`, `prod(v)`: a vektor elemeinek összege, ill. szorzata

Műveletek vektorokkal

Az `R` a vektorokkal koordinátánként végzi a műveleteket!

- ▶ `v=c(1,0,0,1,5)`
- ▶ `w=3:5`
- ▶ `2*v+w`

Itt `v` és `w` nem ugyanolyan hosszúak. Az `R` a rövidebb vektort ciklikusan ismétli, amíg olyan hosszú nem lesz, mint a másik vektor.

Az elemi függvények is elemenként hajtódnak végre: `exp(0:3)`

Mivel egyenlő `2*1:5`?

Hivatkozás a vektorok elemeire:

`v=1:17`

- ▶ `v[5]`
- ▶ `v[2:5]`
- ▶ `v[-1]`
- ▶ `v[-c(4,9,12)]`

Elágazások, ciklusok

- ▶ **if** elágazás

`if(logikai kifejezés) utasítás else utasítás`

Példa: `if (d>=0) d else 0` (ez éppen a d pozitív részének kiszámítása)

- ▶ **ifelse** parancs

`ifelse(v,t,f)`: ha a v logikai vektor igaz, akkor t -vel, egyébként f -fel tér vissza

Példa: `ifelse(x>0,x,-x)` eredménye az $\text{abs}(x) = |x|$ vektor

- ▶ **for** ciklus

`for (változó in vektor) utasítás`

Példa: `s=0, for (i in 1:10) s=s+i`

Vagy: `> v4=c(1,3,5,7)`

`> for (i in v4) print(i^2)`

Függvények

Függvény definiálása:

```
fv<-function(arg1,arg2,...)utasítások
```

Példa: `negyzet<-function(x) x^2`

Ezek után használhatjuk a `negyzet(-5)` parancsot.

Példa: a pozitív rész függvény

```
pozitivresz = function(x) {if (x>=0) x else 0}
```

Grafikus parancsok

Új függvény ábrázolása:

- ▶ `plot()`

Példa: `plot(sin,-pi,2*pi)`

A már meglévő ábra változtatása:

- ▶ `points()`: pontokat tesz rá az ábrára

Példa: `points(1:3,c(0,0,0.5),col="red",pch=2)`

- ▶ `abline()`: egy új egyenest tesz az ábrára

Példa: `abline(h=1,col="blue",lty=3)`

(A `help(abline)` parancs kiírja az `abline` fv használatát.)

- ▶ `title()`: címet ad a grafikonnak

Példa: `title("Hullamvonal")`

Nevezetes eloszlások

Az adott eloszlás neve előtt:

- ▶ d: sűrűségfüggvény, vagy felvételi valószínűség
- ▶ p: eloszlásfüggvény
- ▶ q: kvantilisok
- ▶ r: véletlen számok az adott eloszlásból

Példák nevezetes eloszlások használatára:

- ▶ binomiális eloszlás: `dbinom(3,size=12,prob=0.2)`
- ▶ Poisson eloszlás: `ppois(16,lambda=12)`
- ▶ egyenletes eloszlás: `runif(10,min=1,max=3)`
- ▶ exponenciális eloszlás: `pexp(2,rate=1/3)`
- ▶ normális eloszlás: `dnorm(84,mean=72,sd=15)`
- ▶ χ^2 -eloszlás: `qchisq(.95,df=7)`
- ▶ Student (t)-eloszlás: `qt(.95,df=5)`

FinCal package

A FinCal csomag alkalmas jelenérték- és jövőérték-számításra.

Töltsük be a csomagot:

- ▶ `library(FinCal)`

Vizsgáljuk meg (pl. a `help()` segítségével) és próbáljuk ki a következő parancsokat:

- ▶ `fv`, `pv`
- ▶ `pmt`
- ▶ `n.period`