

Parancsok

1. Gyakorlat

Munkakönyvtár lekérése / beállítása: `getwd()`, `setwd()`

Súgó: `?parancs`, `help(„parancs”)`, `help.search(„csomag”)`, `RSiteSearch(„csomag”)`, `args(„parancs”)`

Változók:

Értékkadás: `a=5*2`, `(b=2^3)`, `d=„100”`

Típuslekérdezés: `typeof(a)`, `is.numeric(a)`, `is.character(a)`, `str(d)`

Vektorok:

Definiálásuk: `c()` függvény

`seq(1, 9 by = 2)` – Matlabban `(1:2:9)`

Műveletek:

komponensenkénti összeadás: `v1 + v2`

komponensenkénti szorzás: `v1 * v2`

mátrixszorzás: `v1 %*% v2`

Részvektorok képzése szögletes zárójellel: `v[2]`, `v[1:3]`

Listák: `list()` függvény, kb. mint egy C-s struct

`lista = list(a = 5, b = 2, c = c(1, 2, 3))`

Adattagokra hivatkozás:

index: `lista[[1]]`

property neve: `lista$c`

Adattáblázat:

`data.frame(tablazatNeve)` – mezői hivatkozhatók a `$` operátorral

szövegfájlból: `read.table(„data.txt”, header=TRUE)`

pontosvesszővel tagolt .csv fájlból: `read.csv2(„data.csv”)`

Könyvtár betöltése: `library()` parancs

`library(könyvtárNeve)`

Kiírás a konzolra: `cat()` parancs, java szintax (+ helyett ,)

2. Gyakorlat

Táblázat felcsatolása: `attach(tablazat)`

Táblázat függvények:

Sorok (rekordok) száma: `length(tablazat)`

Átlag: `mean(tablazat)`

Szórás: `sd(tablazat)`

Variancia: `var(tablazat)`

Medián: `median(tablazat)`

Kvartilisek: `quantile(tablazat)`, `quantile(tablazat, probs = vektor)`

Legnagyobb érték: `max(tablazat)`

Legkisebb érték: `min(tablazat)`

Interkvartilis terjedelme: `IQR(tablazat)`

Változó mintabeli jellemzői `summary(valtozo)`

Skewness: „ferdeség”, korrigált/korrigálatlan-e az empirikus szórás

`library(moments)` kell hozzá

A ferdeséget adja meg: negatív szám esetén az adathalmaz balra, pozitív esetén jobbra ferde

Az átlag és a medián összefüggése

Grafikonok:

Hisztogram: `hist(vektor)`, `hist(vektor, breaks = 5)`

Boxdiagram: `boxplot(vektor)`

Oszlopdigram: `barplot(vektor)`
Kördiagram: `pie(vektor)`
Gyakorisági értékek:
Gyakoriság: `table(változónév)`
Relatív gyakoriság: `prop.table(vektor)`
Hiányzó értékek törlése a táblázatból:
`tablazat2 = tablazat[!is.na(tablazat)]`
Hiányzó értékek száma: `length(tablazat)-length(tablazat2)`
Táblázat készítése: N változó átlagos értéke K változó szerinti bontásban, hiányzó értékek nélkül:
`tapply(N, K, mean, na.rm = TRUE)`
Táblázat leválasztása: `detach(input)`

3. Gyakorlat

Student eloszlás (képletgyűjteményben omega) függvénye: `qt()`
`qt(alfa, df)`, ahol alfa = konfidencia intervallum/2, df = elemszám -1

T-próba:

$S = (\text{várhatóÉrték} - \mu_0) / \sqrt{\text{variancia} / \text{elemszám}}$
 $= (\text{várhatóÉrték} - \mu_0) / (\text{szoras} / \sqrt{\text{elemszam}})$

$S_0 = qt(\text{alfa}/2, df = \text{elemszam}-1)$

A nullhipotézist elfogadjuk, ha $(-S_0 \leq S \leq S_0)$ igaz, különben elvetjük.

De ha nem akarunk gondolkodni / ocsmány sztochás képleteket használni: `t.test()`

Default t-próba: `t.test(x, y = NULL, alternative = c(„two sided”, „less”, „greater”), mu = 0, paired = FALSE, var.equal = FALSE, conf.level = 0.95)`

x: vektor

y: vektor (opcionális)

alternative: string, a nullhipotézist jelöli (opcionális)

mu: szám, a megadott átlag (kétmintás próbánál az átlagok különbségének) értéke

paired: boolean, páros t-próbánál igaz (opcionális)

var.equal: boolean, a két variancia egyenlő-e (opcionális, ne foglalkozzunk vele)

conf.level: konfidencia szint (= 1-szignifikancia szint)

Elfogadjuk a nullhipotézist, ha az outputban a p-value NAGYOBB mint a szignifikancia szint.

4. Gyakorlat