

Funkce pro import dat:

`c()` použijeme, pokud chceme do R dostat několik hodnot (řekněme do 10).

`scan()` tehdy, pokud máme pár sloupečků, nebo naopak pokud máme velmi rozsáhlý datový soubor (o desetitisících řádkách či sloupcích) a import pomocí `read.table()` je příliš pomalý - potom je třeba se řídit nápovědou.

`read.table()` apod. pro import běžných tabulkových dat.

Funkce	hlavní argumenty	popis argumentů
<code>c()</code>		
<code>scan()</code>	<code>what =</code>	"numeric" (nebo 0), "character" (nebo ""), "logical", apod.
<code>read.table()</code>	<code>file =</code>	cesta k souboru, např. "c:/Dokumenty/tab.txt", nebo "clipboard"
<code>read.delim()</code>	<code>header =</code>	TRUE/FALSE, zda dataset má hlavičku
<code>read.delim2()</code>	<code>sep =</code>	separátor hodnot, např. "", ",", " nebo tabulátor "\t"
<code>read.csv()</code>	<code>dec =</code>	decimální oddělovač: "." nebo ",", "
<code>read.csv2()</code>	<code>row.names =</code>	pořadí nebo jméno sloupce obsahujícího jména řádků
	<code>as.is =</code>	vektor identifikující sloupce, jež nemají být převedeny na faktor
	<code>stringsAsFactors =</code>	TRUE/FALSE, zda mají být textové řetězce převedeny na faktory
	<code>fill =</code>	mají být prázdné buňky nahrazeny NA?

	<code>header =</code>	<code>sep =</code>	<code>dec =</code>
<code>read.table()</code>	FALSE	""	."
<code>read.delim()</code>	TRUE	"\t"	."
<code>read.delim2()</code>	TRUE	"\t"	","
<code>read.csv()</code>	TRUE	","	."
<code>read.csv2()</code>	TRUE	";"	","

Export dat:

`write()` – pouze pro matice a vektory. Je potřeba specifikovat, do kolika sloupců jsou hodnoty uspořádány (default je 1 pro textové objekty a 5 pro ostatní) a matici musíme předem otočit, protože funkce zapisuje po řádcích.

`write.table()` – pro dataframy, preferovaný.

`write.csv()` a `write.csv2()` pro jednodušší export do excelu, ale jsou neflexibilní, není možné měnit argumenty

Funkce	hlavní argumenty	popis argumentů
write()	<code>file =</code> <code>ncolumns =</code> <code>sep =</code>	cesta souboru do kolika sloupců má být výsledek uspořádán separátor
write.table()	<code>file =</code> <code>quote =</code> <code>sep =</code> <code>dec =</code> <code>row.names =</code> <code>col.names =</code>	cesta souboru TRUE/FALSE zda mají být textové řetězce uzavřeny v uvozovkách separátor desetinný oddělovač TRUE/FALSE zda mají být exportována i jména řádků / sloupců

Uložení a načtení vybraných objektů pro další práci v R

Provedeme pomocí funkce `save()`, do níž vyjmenujeme objekty, které chceme uložit, a uvedeme jméno souboru (nebo jeho plnou cestu, pokud ho chceme uložit jinam než do pracovního adresáře). Například pro uložení objektů 'spe' a 'env' do soubory 'mydata.RData' v pracovním adresáři použijeme:

```
save(spe, env, file= 'mydata.RData')
```

Pokud máme jména ukládaných objektů ve formě textového vektoru (například výstupem z funkce `ls()`), vložíme je do argumentu `list=`. Například, kdybychom chtěli uložit všrchny objekty pracovního prostředí, použijeme:

```
save(list = ls(), file='mydata.RData')
```

Přípona není nijak zásadní a může být v podstatě jakákoliv, je ale zvykem pro R pracovní prostředí používat koncovku ".RData".

Načtení provedeme příkazem `load()` – stejně jako bychom načítali pracovní prostředí uložené příkazem `save.image()` (případně pomocí menu File/Save Workspace...):

```
load("mydata.RData")
```

Opět, pokud nemáme soubor, který chceme načíst, v pracovním adresáři, musíme specifikovat celou jeho cestu, např.:

```
load("C:/Dokumenty/mydata.RData")
```