Základy biostatistiky, 1. cvičení

Poslední aktualizace: 15. února 2013

Začátek

- 1. Na disku J: založte složku J:\biostat a zkopírujte do ní
 - soubor V:\zvara\biostat\cvic01.pdf (popis cvičení)
 - soubor V:\zvara\biostat\cvic01.R (skripty)
 - soubor V:\zvara\cestina.pdf
 - složku V:\zvara\biostat\data (data)
- 2. Otevřete soubory cestina.pdf a cvic01.pdf.
 - Soubor cestina.pdf ukazuje, jak na běžné české klávesnici vygenerovat některé znaky, s nimiž pracuje R.
 - Soubor cvic01.pdf popisuje první cvičení, podle tohoto návodu budeme pracovat.
- 3. Spusťte program R z nabídky programů (verze SDI). Otevře se okno, kterému budeme říkat konzole. Nahoře je nabídka, kterou můžeme využívat při práci. Do konzole můžeme za vyvolávací znak > psát jednotlivé příkazy ve formě textu. Pokud bychom příkaz neukončili a odřádkovali pomocí Enter předčasně, program reaguje tak, že místo nerovnítka > uvede +. Použijeme-li tlačítko ESC, program bude napsané ignorovat a ohlásí se pomocí > nebo +. Konzole si pamatuje naše příkazy. Znovu je můžeme postupně zobrazit pomocí kurzorové šipky směřující nahoru.
- 4. Nastavte v R pracovní adresář pomocí jednoho z následujících způsobů
 - Napište a odešlete pomocí Enter příkaz setwd("J:\biostat")
 - V horní nabídce zvolte postupně File Change dir a pomocí myši najděte složku J:\biostat
- 5. Zkontrolujte nastavení pracovního adresáře pomocí getwd(). Odpověď by měla znít J:\biostat
- Spusťte Commander příkazem library (Rcmdr). Otevře se nové okno, které má tři části.

7. Pracovní skript cvic01.R umístíme do horního okna (Script Window) tak, že v horní nabídce Commanderu zvolíme postupně
File Open script file ...

Práce ve skriptovém okně

- Znak # uvozuje komentář. Program takto chápe celý zbytek řádku. Doporučuje se vkládat vlastní komentáře.
- Abychom o komentáře (a nejen o ně) nepřišli, doporučuje se skriptový soubor čas od času ukládat pomocí Ctrl+S.
- Pokud píšeme na jeden řádek více příkazů, je třeba oddělit je středníkem. Vhodnější je psát každý příkaz na nový řádek, středníky pak nejsou nutné.
- Když chceme, aby program provedl příkaz, stačí umístit kurzor kamkoliv do příslušného řádku a použít kombinaci Ctrl+R nebo myší klepnout na Submit.
- Několik řádků s příkazy můžeme nechat provést najednou tak, že je označíme jako blok (myší při stisknutém levém tlačítku) a použijeme buď Ctrl+R nebo Submit.
- Nápovědu k libovolnému příkazu získáme pomocí help(prikaz) nebo pomocí ?prikaz.
- Commander je pomůcka, která pomůže volbou z horní nabídky připravit většinu příkazů, které budeme potřebovat. Program takto připravené příkazy hned také provede.
- Jednotlivé příkazy se po spuštění zkopírují do výstupního okna (Outupt Window), v němž se objeví také odpověď programu.
- V dolním okénku (Messages) se objevují hlášení: červeně chybová hlášení, zeleně varování, modře běžné informace.
- Již provedené příkazy můžeme provádět opakovaně, můžeme je podle potřeby nejprve upravovat.
- Při vkládání desetinných čísel musíme používat jako oddělovač tečku.

Jednoduché operace

1.	Kalkulačka		#		
	3+2		$\frac{1}{4}$ sčítání		
	3-2		# odčítání		
	3*2		# násobení		
	3/2		# dělení		
	5^2		# druhá mocnina		
	5 Z 5^3		# třotí mocnina		
	$5^{(1/2)}$		# druhé odmogning		
	5(1/2)		# druhá odmocnina		
	sqrt(5)		# druna odnochina # henstente = 2.14		
	p1		$\#$ Konstanta $\pi = 5.14$		
	sin(p1/2)		# sinus		
exp(2) log(2)			# exponenciela (e na neco)		
		~	# prirozeny logaritmus (log, ln)		
	log(100,base=10)		# dekadicky logaritmus (\log_{10} , \lg)		
	$\exp(\log(2))$		# vysla dvojka?		
	factorial(5)		# 5! (faktoriál)		
	choose(5,2)		# kombinační číslo $\binom{3}{2}$		
2. Práce s proměnnými					
	x <- 5	#	uloží hodnotu 5 do proměnné x		
	x = 5	#	uloží hodnotu 5 do $x,$ ale nesmí být uvnitř volání jiné funkce		
x #		#	zobrazí se x		
	y = 3	#			
	x+y	#			
	x-y	#			
	z <- x+y # z = x+y #		součet uloží do z , ale nezobrazí jej		
	<pre>print(z<-x+y)</pre>	#	uloží součet do z a zobrazí jej		
	<pre>print(z=x+y)</pre>	#	chyba, předpokládá, že z je argument funkce print()		
	(z=x+y)	#	uloží do z součet a zobrazí jej		
3.	Co je v pracovním prostoru?				

Co je v pracovním prostoru: ls() # seznam objektů v pracovním prostoru rm(list=ls()) # odstranění (skoro) všech objektů z pracovního prostoru ls() # opravdu nic nezůstalo?

Práce s daty

- 1. Data je vhodné připravit a zkontrolovat ve vhodném tabulkovém procesoru.
- 2. Pro přenos dat do R je vhodný univerzální textový formát označovaný zpravidla příponou csv. Při načítání takových dat do R je třeba přihlédnout zejména k tomu, jaký oddělovač je použit pro desetinná čísla. V českém prostředí to bývá čárka, v anglickém tečka.
- 3. Datový soubor deti23.csv připravený v českém prostředí načteme do Commanderu pomocí horní nabídky postupnou volbou
 Data Import data from test file, clipboard, or file
 a dále
 Enter name for data set Deti23
 Field Separator Other Specify: ; (středník)
 Decimal-Point Character Comma [,]
 Pomocí myši zvolíme ve složce data soubor deti23.csv.
- 4. Zkontrolujte, že se v informačním okénku objevila zpráva o počtu vložených řádků a sloupců a že nad skriptovým okénkem se původní označení
 vložených řádků a sloupců a že nad skriptovým okénkem se původní označení
 vložených řádků a sloupců a že nad skriptovým okénkem se původní označení
- 5. Alternativní možnost:
 Deti23 <- read.csv2("./data/Deti23.csv", header=TRUE)
 Poklepat na tlačítko Data set: a myší vybrat zvolený soubor.
- 6. Datový soubor si prohlédneme klepnutím na tlačítko <u>View data set</u> nebo pomocí příkazu **print (Deti23)**.
- K jednotlivým proměnným se dostaneme pomocí operátoru \$, např. Deti23\$vaha nebo Deti23\$delka Chybou skončí příkazy vaha nebo delka.
- Proměnné uvnitř datového souboru (databáze, data.frame) zpřístupníme pomocí příkazu attach(), který datový soubor zkopíruje na místo, kde do něj R vidí: attach(Deti23)

```
vyska
Deti23$vyska
```

Seznam míst (datových souborů, knihoven (package), pracovních prostorů, kde ${\sf R}$ postupně hledá význam identifikátorů (slov), zveřejní funkce

search() (proved'te)

Pozor: Pokud nyní datový soubor nějak upravíme, na zmíněné kopii se změna neprojeví.

- 9. Kopii datového souboru můžeme odstranit pomocí detach(Deti23) (proveďte)
 Ověříme, že datový soubor Deti23 se již na vyhledávací cestě nenachází: search().
 Abychom mohli opět pracovat jen s názvy proměnných, musíme po případné úpravě datového souboru znovu použít příkaz attach() (to počká).
- 10. Jako novou proměnnou vyrobíme délku dítěte, ale vyjádřenou v metrech. Pomocí nabídky Commanderu toho dosáhneme postupnou volbou
 Data Manage variables in active data set Compute new variable ...
 New variable name delka.m
 Expression to compute delka/100
 Téhož výsledku můžeme dosáhnout příkazem
 Deti23 <- transform(Deti23, delka.m = delka / 100)
- 11. O úspěšném vytvoření nové proměnné najdeme informaci v informačním okénku Messages:, o tomtéž se můžeme přesvědčit, když klepneme na tlačítko View data set: nebo když si necháme data vypsat pomocí příkazu Deti23\$delka.m
- 12. Jako novou proměnnou vyrobte index obezity BMI, který je dán jako poměr váhy v kilogramech a čtverce výšky v metrech, tedy
 bmi = vaha/delka.m².
 Ověřte, zda se vám to podařilo.
- 13. Podobně vytvořte proměnnou dobaPlnolet, která udává kolik roků byla matka v době porodu plnoletá. Nezapomeňte ověřit, zda se to podařilo.
- 14. Pro další práci si znovu zpřístupníme proměnné souboru Deti23 příkazem attach(Deti23)
- 15. Jen pátou hodnotu proměnné bmi vytiskneme pomocí bmi [5], souvislou řadu od 3. do 6. hodnoty příkazem bmi [3:6]. Nesouvislou řadu 3., 5. a 8. hodnoty dostaneme pomocí příkazu bmi [c(3,5,8)]. Seznam požadovaných hodnot musí být jediným argumentem v hranatých závorkách, proto příslušné pořadové indexy musíme spojít do balíčku pomocí funkce c().

- 16. Datový soubor jako je Deti23 si můžeme představit jako tabulku hodnot, v níž rozlišujeme řádky a sloupce.
- 17. Jednotlivé hodnoty můžeme volat jako Deti23[2,3] (druhá hodnota třetí proměnné vekMatky) nebo stejnou hodnotu jako Deti23[3, "vekMatky"].
- 18. Několik hodnot délek v centimetrech i v metrech můžeme (třeba pro kontrolu) vypsat pomocí Deti23[c(1:3,9),c("delka.m", "delka")]
- 19. Všechny hodnoty věku otců a matek dostaneme pomocí Deti23[,c("vekOtce", "vekMatky")], podobně všechny údaje o 15. dítěti pomocí Deti23[15,].

Numerické popisné statistiky

1. Ověřte v tlačítku Data set: , zda máte aktivní datový soubor Deti23

2.	Popisné statistiky pro v	věk matek	získáme volbou			
	Statistics Summaries	Numerical summaries				
	Variables (pick one	or more)	vekMatky			
	Přímý výpočet téhož:					
	<pre>summary(vekMatky)</pre>	#	míry polohy			
	length(vekMatky	#	počet hodnot n			
	mean(vekMatky	#	průměr			
	median(vekMatky	#	medián			
	<pre>sort(vekMatky</pre>	#	variační řada			
	var(vekMatky	#	rozptyl			
	sd(vekMatky)	#	směr. odchylky			
	min(vekMatky)	#	minimum			
	max(vekMatky)	#	maximum			
	range(vekMatky)	#	minimum a maximum			
	IQR(vekMatky)	#	(mezi)kvartilové rozpětí			
	quantile(vekMatky,probs=c(0,25,50,75,100)/100)					
	-	#	některé kvantily			

3. Nyní porovnáme váhu chlapců a děvčat ve 12. měsíci věku postupnou volbou Statistics Summaries Numerical summaries Variables (pick one or more) vekMatky Sumarize by groups: Gender

4. Samostatně vyšetřete hodnoty BMI.

Ukončení práce

 Upravený datový soubor uložíme v erkovém formátu, abychom jej příště mohli snadno znovu zavést. V horní nabídce Commanderu zvolíme postupně
 Data Active data set Save active data set ...

a kurzorem připravíme uložení do adresáře data.

2. Uložte skriptový soubor (při standardním ukončení možnost jeho uložení systém připomene). Při zavírání Commander nabídne uložení skriptového souboru pod označením RCommander.R a všechny výstupy pod označním RCommander.txt Jména těchto souborů doporučuji upravit, použít například označení příslušného cvičení.