2 Bodové a intervalové rozdělení četností

2.1 Jednorozměrné bodové rozdělení četností

Dataset 1: Porodní hmotnost novorozenců

Máme k dispozici údaje o porodní hmotnosti novorozenců z okresní nemocnice získané v období jednoho roku a současně máme k dispozici údaje o počtu starších biologických sourozenců novorozence, pohlaví novorozence a vzdělání matky (Alánová, 2008; soubor 17-anova-newborns.txt).

Popis proměnných v datasetu 1:

- edu.M – vzdělání matky (1 – základní, 2 – střední bez maturity, 3 – střední s maturitou, 4 – vysokoškolské);
- prch.N – počet biologických starších sourozenců (0–8);
- sex.C – pohlaví dítěte (m – muž, f – žena);
- weight.C – porodní hmotnost dítěte (g).

Příklad 2.1. Načtení datového souboru
Načtěte dataset 17-anova-newborns.txt do proměnné data a vypište prvních 5 řádků z načteného souboru. Zjistěte, zda soubor obsahuje neznámé (NA) hodnoty a pokud ano, tak je odstraňte. Potom zjistěte dimenzi datové tabulky data.

Řešení příkladu 2.1
Datový soubor načteme pomocí funkce read.delim(). První pět řádků vypíšeme pomocí příkazu head() se specifikací argumentu n = 5 řádků.

```
1 data <-read.delim('17-anova-newborns.txt')
2 head(data, n = 5)
```

```
  edu.M prch.N sex.C weight.C                                                    3
1     2      0     m     3470                                                    4
2     2      0     m     3240                                                    5
3     2      0     f     2980                                                    6
4     1      0     m     3280                                                    7
5     3      0     m     3030                                                    8
```

Načtená datová tabulka obsahuje údaje o čtyřech znacích: vzdělání matky (edu.M), počet starších sourozenců novorozence (prch.N), pohlaví novorozence (sex.C) a porodní hmotnost novorozence (weight.C). Pomocí funkce is.na() zjistíme, zda načtený soubor obsahuje neznámé hodnoty.

```
9 sum(is.na(data))
```

```
[1] 24                                                                           10
```

Počet NA hodnot v datovém souboru $n_{NA} = 24$. Po bližším prozkoumání datového souboru můžeme zjistit, že chybí celkem 13 hodnot v proměnné edu.M, 5 hodnot v proměnné prch.N a 6 hodnot v proměnné weight.C. Neznámé hodnoty odstraníme ze souboru pomocí funkce na.omit(). Ke zjištění dimenze tabulky použijeme příkaz dim().

```
11 data <- na.omit(data)
12 dim(data)
```

```
[1] 1382    4                                                                    13
```

Tabulka data má po odstranění NA hodnot 1382 řádků a čtyři sloupce. Celkem tedy máme k dispozici údaje o 1382 objektech, přičemž u každého objektu máme záznamy o čtyřech znacích. ♣

Příklad 2.2. Úprava datového souboru
Z popisu datasetu 1 víme, že počet starších sourozenců u sledovaných novorozenců se pohybuje v rozsahu 0–8 sourozenců. V následující analýze se zaměřte pouze na novorozence, kteří mají maximálně dva starší sourozence. Tyto novorozence rozdělte podle porodní hmotnosti do tří kategorií: nizka – hmotnost novorozence je nižší než 2500 g; norma – hmotnost novorozence se pohybuje v rozmezí 2500–4200 g; vysoka – hmotnost novorozence je vyšší než 4200 g. Dále upravte označení jednotlivých variant znaku $X = \textit{vzdělání matky}$ tak, aby bylo na první pohled zřejmé, jakého nejvyššího vzdělání bylo u matky dosaženo (1 – ZS, 2 – SS, 3 – SSm, 4 – VS).

Řešení příkladu 2.2
Z tabulky data nejprve vyselektujeme novorozence s žádným, jedním nebo dvěma staršími sourozenci.

```
data <- data[data$prch.N <= 2, ]
dim(data)
```

```
[1] 1276    4
```

V proměnné data nám nyní zůstalo 1276 novorozenců. Nyní vložíme do tabulky data novou proměnnou weight.K, která bude podle porodní hmotnosti novorozence weight.C nabývat hodnoty 1 – nizka, 2 – norma, 3 – vysoka.

```
data$weight.K[data$weight.C < 2500] <- 1
data$weight.K[data$weight.C >= 2500 & data$weight.C <= 4200] <- 2
data$weight.K[data$weight.C > 4200] <- 3
head(data)
```

```
  edu.M prch.N sex.C weight.C weight.K
1     2      0     m     3470        2
2     2      0     m     3240        2
3     2      0     f     2980        2
4     1      0     m     3280        2
5     3      0     m     3030        2
6     2      1     m     3650        2
```

Nově vytvořenou proměnnou weight.K převedeme pomocí funkce factor() na proměnnou typu faktor, což je speciální typ proměnné, umožňující přiřazení názvů k číselným hodnotám. Díky tomuto převodu můžeme nyní pomocí argumentu labels jednotlivé kategorie proměnné weight.K pojmenovat.

```
data$weight.K <- factor(data$weight.K, labels = c('nizka', 'norma', 'vysoka'))
head(data)
```

```
  edu.M prch.N sex.C weight.C weight.K
1     2      0     m     3470    norma
2     2      0     m     3240    norma
3     2      0     f     2980    norma
4     1      0     m     3280    norma
5     3      0     m     3030    norma
6     2      1     m     3650    norma
```

Analogickým způsobem nyní pojmenujeme kategorie znaku $X = \textit{vzdělání matky}$.

```
data$edu.M <- factor(data$edu.M, labels = c('ZS', 'SS', 'SSm', 'VS'))
head(data)
```

```
  edu.M prch.N sex.C weight.C weight.K
1    SS      0     m     3470    norma
2    SS      0     m     3240    norma
3    SS      0     f     2980    norma
4    ZS      0     m     3280    norma
5   SSm      0     m     3030    norma
6    SS      1     m     3650    norma
```

♣

Příklad 2.3. Variační řada
Vytvořte variační řadu znaku X = *vzdělání matky* a variační řadu znaku Y = *porodní hmotnost novorozence*.

Řešení příkladu 2.3
Zaměřme se nejprve na znak X = *vzdělání matky*. Znak má celkem čtyři varianty: základní vzdělání, střední vzdělání, střední vzdělání s maturitou a vysokoškolské vzdělání. Variační řada je tabulka obsahující pro každou (j-tou) variantu znaku X (a) absolutní četnost n_j; (b) relativní četnost p_j; (c) absolutní kumulativní četnost N_j; (d) relativní kumulatiní četnost F_j.

Absolutní četnost varianty ZS získáme aplikováním funkce sum() na logický výraz edu == 'ZS'. Výraz edu == 'ZS' vytvoří nový vektor obsahující hodnoty 1 na pozici, kde se ve vektoru edu vyskytovala hodnota ZS, a nuly na ostatních pozicích. Aplikováním funkce sum() na tento vektor nul a jedniček získáme četnost výskytu výrazu ZS ve vektoru edu. Analogicky získáme hodnoty absolutních četností pro varianty SS, SSm a VS.

```
edu <- data$edu.M
n1 <- sum(edu == 'ZS')
n2 <- sum(edu == 'SS')
n3 <- sum(edu == 'SSm')
n4 <- sum(edu == 'VS')
nj <- c(n1, n2, n3, n4)
```

Relativní četnosti jednotlivých variant znaku X získáme jako podíl absolutních četností variant ku celkovému počtu 1276 objektů v souboru. Pomocí funkce cumsum() aplikované na vektor absolutních (resp. relativních) četností získáme vektor absolutních (resp. relativních) kumulativních četností.

```
n  <- sum(nj)
pj <- nj / n
Nj <- cumsum(nj)
Fj <- cumsum(pj)
```

Pomocí příkazu data.frame() vytvoříme požadovanou variační řadu, přičemž argumentem row.names specifikujeme názvy řádků variační řady. Výslednou tabulku zobrazíme zaokrouhlenou na čtyři desetinná místa (funkce round() se specifikací argumentu digits = 4). Poznamenejme, že zaokrouhlení se projeví ve výpisu tabulky, ovšem původní hodnoty uložené v proměnné edu.var.r zůstávají nezaokrouhleny.

```
edu.name  <- c('ZS', 'SS', 'SSm', 'VS')
edu.var.r <- data.frame(nj, pj, Nj, Fj, row.names = edu.name)
round(edu.var.r, digits = 4)
```

```
     nj     pj   Nj     Fj
ZS  347 0.2719  347 0.2719
SS  424 0.3323  771 0.6042
SSm 425 0.3331 1196 0.9373
VS   80 0.0627 1276 1.0000
```

Interpretace výsledků: Datový soubor obsahuje údaje o celkovém počtu 1276 novorozenců s maximálně dvěma staršími sourozenci, přičemž v 347 případech (27.19 %) bylo nejvyšší dosažené vzdělání matky základní, v 424 případech (33.23 %) bylo nejvyšší dosažené vzdělání matky středoškolské bez maturity, apod. Celkem 771 (60.42 %) matek novorozenců získalo středoškolské vzdělání bez maturity, nebo nižší, celkem 1196 (93.73 %) matek novorozenců získalo středoškolské vzdělání s maturitou, nebo nižší.

Zaměřme se nyní na znak Y = *porodní hmotnost novorozence*. Protože variační řadu má smysl sestrojovat pouze pro kategoriální znak, použijeme k vytvoření variační řady proměnnou weight.K. Znak Y má tři varianty: nízká porodní hmotnost, norma a vysoká porodní hmotnost.

Variační řadu můžeme sestrojit analogickým postupem jako výše, nebo použitím funkce variacni.rada(), která je k dispozici v RSkriptu Sbirka-AS-I-2018-funkce.R, jenž vznikl pro potřeby této publikace. RSkript načteme pomocí příkazu source(). Názvy řádků variační řady specifikujeme argumentem row.names ve funkci variacni.rada(). Výslednou tabulku opět zobrazíme zaokrouhlenou na čtyři desetinná místa.

```
source('Sbirka-AS-I-2018-funkce.R')
wei <- data$weight.K
```

```
wei.name  <- c('nizka', 'norma', 'vysoka')
wei.var.r <- variacni.rada(wei, row.names = wei.name)
round(wei.var.r, digits = 4)
```

```
        nj     pj   Nj     Fj
nizka  240 0.1881  240 0.1881
norma  993 0.7782 1233 0.9663
vysoka  43 0.0337 1276 1.0000
```

Interpretace výsledků: Porodní hmotnost novorozenců v datovém souboru s maximálně dvěma staršími sourozenci, se v 993 případech (77.82 %) pohybovala v normě, v 240 případech (18.81 %) byla nižší než norma a v 43 případech (3.37 %) byla vyšší než norma. Celkem 240 novorozenců (18.81 %) mělo porodní hmotnost nižší než norma, 1233 novorozenců (96.63 %) mělo porodní hmotnost nižší nebo rovnu normě a 1276 novorozenců (100 %) mělo porodní hmotnost vysokou, v normě, nebo nižší. ♣

Příklad 2.4. Sloupcový diagram absolutních četností
Nakreslete sloupcový diagram absolutních četností pro znak X = *vzdělání matky* a pro znak Y = *porodní hmotnost novorozence*.

Řešení příkladu 2.4
Zaměřme se nejprve na znak X = *vzdělání matky*. Sloupcový diagram absolutních četností vykreslíme pomocí funkce barplot(). Konstrukci grafu začneme vykreslením prázdného grafu s připravenými popisky. Prvním uvedeným argumentem je vektor absolutních četností. Argumentem col (resp. border) zvolíme barvu výplně (resp. ohraničení) sloupců jako bílou. Argumentem ylim stanovíme rozsah měřítka osy y na hodnoty 0–500, argumenty xlab a ylab změníme popisky osy x a y. Argumentem names můžeme specifikovat názvy jednotlivých sloupců v grafu a konečně argumentem las změníme směr popisků měřítka osy y z vertikálních na horizontální.

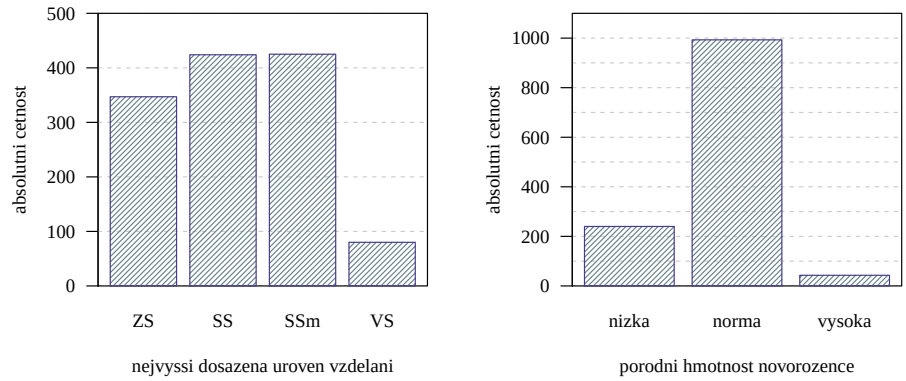
Do grafu doplníme referenční čáry pomocí funkce abline(). Argumentem h specifikujeme vykreslení horizontálních čar v posloupnosti čísel 0, 100, ..., 500, šedou barvou (argument col) a přerušovanou čárou (argument lty). Kolem grafu obkreslíme černý rámeček příkazem box() se specifikací argumentu bty = 'o'.

Nakonec do grafu dokreslíme příkazem barplot() sloupce. Přidání sloupců do stávajícího grafu nastavíme argumentem add. Stanovením hodnoty F u argumentu axes potlačíme opětovné vypsání měřítek osy x a osy y. Barvu výplně a ohraničení sloupců zvolíme v odstínu modré. Argumentem density nastavíme šrafování výplně sloupců s intenzitou hustoty čar 20.

Obdobným postupem získáme sloupcový diagram absolutních četností pro znak Y = *porodní hmotnost novorozence*.

```
# Vzdelani matky
barplot(edu.var.r$nj, col = 'white', border = 'white', ylim = c(0, 500),
        xlab = 'nejvyssi dosazena uroven vzdelani', ylab = 'absolutni cetnost',
        names = edu.name, las = 1)
abline(h = seq(0, 500, by = 100), col = 'grey80', lty = 2)
box(bty = 'o')
barplot(edu.var.r$nj, add = T, axes = F, col = 'lightblue4',
        border = 'slateblue4', density = 30)

# Porodni hmotnost novorozencu
barplot(wei.var.r$nj, col = 'white', border = 'white', ylim = c(0, 1100),
        xlab = 'porodni hmotnost novorozence', ylab = 'absolutni cetnost',
        names = wei.name, las = 1)
abline(h = seq(0, 1100, by = 100), col = 'grey80', lty = 2)
box(bty = 'o')
barplot(wei.var.r$nj, add = T, axes = F, col = 'lightblue4',
        border = 'slateblue4', density = 30)
```



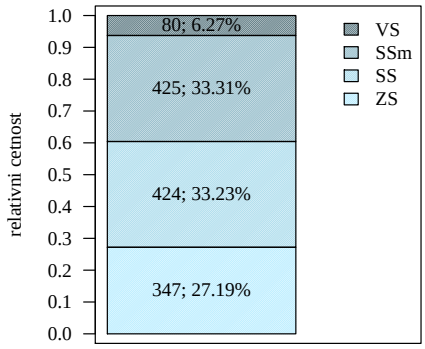
♣

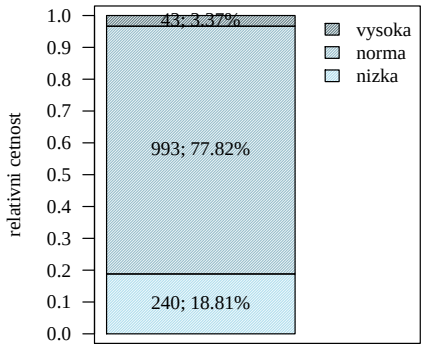
Příklad 2.5. Sloupcový diagram relativních četností
Nakreslete sloupcový diagram relativních četností pro znak $X =$ *vzdělání matky* a pro znak $Y =$ *porodní hmotnost novorozence*.

Řešení příkladu 2.5
Zaměřme se nejprve na znak $X =$ *vzdělání matky*. Sloupcový diagram relativních četností vykreslíme pomocí funkce rel.barplot(), která je k dispozici v RSkriptu Sbirka-AS-I-2018-funkce.R. Tento RSkript jsme načetli v rámci příkladu 2.3 příkazem source().

Prvním argumentem ve funkci rel.barplot() je vektor absolutních četností. Argumentem col (resp. names) specifikujeme barvy (resp. názvy) příslušné jednotlivým kategoriím. Pomocí dalších argumentů stanovíme hustotu šrafování výplně (density), rozsah osy x (xlim) a popisek osy x (xlab). Analogicky sestrojíme sloupcový diagram relativních četností pro znak $Y =$ *porodní hmotnost novorozence*.

```
90  # Vzdelani matky
91  c.blue <- c('lightblue1', 'lightblue2', 'lightblue3', 'lightblue4')
92  rel.barplot(edu.var.r$nj, col = c.blue, names = edu.name,
93              density = 80, xlim = c(0.2, 1.8), xlab = 'vzdelani matky')
94  box(bty = 'o')
95
96  # Porodni hmotnost novorozence
97  rel.barplot(wei.var.r$nj, col = c.blue[2:4], xlim = c(0.2, 1.8),
98              names = wei.name, xlab = 'porodni hmotnost novorozence' )
99  box(bty = 'o')
```


1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
relativni cetnost
80; 6.27%
425; 33.31%
424; 33.23%
347; 27.19%
VS
SSm
SS
ZS
vzdelani matky


1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
relativni cetnost
43; 3.37%
993; 77.82%
240; 18.81%
vysoka
norma
nizka
porodni hmotnost novorozence

♣

2.2 Dvourozměrné bodové rozdělení četností

Příklad 2.6. Kontingenční tabulka absolutních a relativních četností
Zaměřte se nyní na oba znaky X = *vzdělání matky* a Y = *porodní hmotnost novorozence* najednou. Z předchozího textu víme, že znak X má čtyři varianty, znak Y má tři varianty. Celkem tedy můžeme získat $4 * 3 = 12$ různých kombinací variant znaků X a Y. Sestrojte kontingenční tabulku absolutních četností a kontingenční tabulku relativních četností znaků X a Y.

Řešení příkladu 2.6
Kontingenční tabulka absolutních četností bude tabulka o velikosti $(4+1) \times (3+1) = 5 \times 4$ ve tvaru

	nizka	norma	vysoka	suma
ZS	n_{11}	n_{12}	n_{13}	$n_{1.}$
SS	n_{21}	n_{22}	n_{23}	$n_{2.}$
SSm	n_{31}	n_{32}	n_{33}	$n_{3.}$
VS	n_{41}	n_{42}	n_{43}	$n_{4.}$
suma	$n_{.1}$	$n_{.2}$	$n_{.3}$	n

kde n_{jk}, $j = 1, \ldots, 4$ a $k = 1, \ldots, 3$ je *simultánní absolutní četnost* j-té varianty znaku X a k-té varianty znaku Y, $n_{j.}$ (resp. $n_{.k}$) je *marginální absolutní četnost* j-té varianty znaku X (resp. k-té varianty znaku Y) a n je celkový počet objektů v datovém souboru.

Kontingenční tabulku simultánních absolutních četností KT.abs získáme příkazem table(). Následně dopočítáme vektor absolutních marginálních četností nj. znaku X. K tomu využijeme funkci apply() se specifikací argumentů FUN = sum a MARGIN = 1 (aplikuj funkci sum na všechny řádky tabulky KT.abs). Funkce apply() s takto zadanými argumenty sečte všechny hodnoty v každém řádku tabulky KT.abs. Vektor nj. připojíme k tabulce KT.abs příkazem cbind().

Analogicky dopočítáme vektor marginálních četností $(n_{.k})$ znaku Y, přičemž nastavíme argument MARGIN = 2 (aplikuj funkci sum na všechny sloupce tabulky KT.abs). Vektor n.k připojíme k tabulce Tab.K příkazem rbind().

```
KT.abs  <- table(edu, wei)
nj.     <- apply(KT.abs, MARGIN = 1, FUN = sum)
KT.abs  <- cbind(KT.abs, suma = nj.)
n.k     <- apply(KT.abs, MARGIN = 2, FUN = sum)
(KT.abs <- rbind(KT.abs, suma = n.k))
```

```
     nizka norma vysoka suma
ZS      75   264      8  347
SS      79   325     20  424
SSm     73   341     11  425
VS      13    63      4   80
suma   240   993     43 1276
```

Interpretace výsledků: V datovém souboru se vyskytuje celkem 75 novorozenců s maximálně dvěma staršími sourozenci, kteří mají nízkou porodní hmotnost a jejichž matka má základní vzdělání a 341 novorozenců, jejichž porodní hmotnost je v normě a jejichž matka má středoškolské vzdělání s maturitou, apod.

Tabulku relativních četností získáme vydělením tabulky absolutních četností KT.abs celkovým počtem objektů ve studii. Výslednou tabulku zobrazíme zaokrouhlenou na čtyři desetinná místa.

```
KT.rel <- KT.abs / n
round(KT.rel, digits = 4)
```

```
      nizka  norma vysoka   suma
ZS   0.0588 0.2069 0.0063 0.2719
SS   0.0619 0.2547 0.0157 0.3323
SSm  0.0572 0.2672 0.0086 0.3331
VS   0.0102 0.0494 0.0031 0.0627
suma 0.1881 0.7782 0.0337 1.0000
```

Interpretace výsledků: V datovém souboru se vyskytuje celkem 5.88 % novorozenců s maximálně dvěma staršími sourozenci, kteří mají nízkou porodní hmotnost a jejichž matka má základní vzdělání, 26.72 % novorozenců, jejichž porodní hmotnost je v normě a jejichž matka má středoškolské vzdělání s maturitou, apod. ♣

Příklad 2.7. Kontingenční tabulka řádkově a sloupcově podmíněných relativních četností
Zaměřte se nyní opět na oba znaky X = *vzdělání matky* a Y = *porodní hmotnost novorozence* najednou. Vytvořte kontingenční tabulku řádkově podmíněných relativních četností k-té varianty znaku Y, $k = 1, \ldots, 3$ za předpokladu pevně stanovené j-té varianty znaku X, $j = 1, \ldots, 4$. Dále vypočtěte kontingenční tabulku sloupcově podmíněných relativních četností j-té varianty znaku X, $j = 1, \ldots, 4$ za předpokladu pevně stanovené k-té varianty znaku Y, $k = 1, \ldots, 3$.

Řešení příkladu 2.7
Kontingenční tabulka řádkově podmíněných relativních četností nám dává relativní zastoupení všech možných variant znaku Y = *porodní hmotnost novorozence* ve výběru objektů s jednou konkrétní variantou znaku X. V takové tabulce uvažujeme vždy jeden řádek jako celek, a tedy součet relativních četností v každém řádku je roven 1.

Při výpočtu tabulky řádkově podmíněných relativních četností vyjdeme z tabulky simultánních absolutních četností, kterou získáme analogicky jako v příkladu 2.6 pomocí funkce table(). Aplikováním funkce prop.table() s argumentem margin = 1 na kontingenční tabulku KT.abs získáme tabulku řádkově podmíněných relativních četností. Hodnoty tabulky si zobrazíme zaokrouhlené na čtyři desetinná místa (round()).

```
KT.abs   <- table(edu, wei)
KT.rel.r <- prop.table(KT.abs, margin = 1)
round(KT.rel.r, digits = 4)
```

```
     wei
edu    nizka  norma vysoka
  ZS  0.2161 0.7608 0.0231
  SS  0.1863 0.7665 0.0472
  SSm 0.1718 0.8024 0.0259
  VS  0.1625 0.7875 0.0500
```

Interpretace výsledků: Ze všech novorozenců v datovém souboru, kteří mají maximálně dva starší sourozence a jejichž matka má dokončené středoškolské vzdělání zakončené maturitou, má 17.18 % nízkou porodní hmotnost, 2.59 % vysokou porodní hmotnost a 80.24 % novorozenců má porodní hmotnost v normě. Ze všech novorozenců v datovém souboru s maximálně dvěma staršími sourozenci, jejichž matka má dokončené vysokoškolské vzdělání, má 16.25 % nízkou porodní hmotnost, 5.00 % vysokou porodní hmotnost a 78.75 % novorozenců má porodní hmotnost v normě.

Kontingenční tabulka sloupcově podmíněných relativních četností nám dává relativní zastoupení všech možných variant znaku X = *vzdělání matky* ve výběru objektů s jednou konkrétní variantou znaku Y. V takové tabulce představuje vždy jeden sloupec celek, a tedy součet relativních četností v každém sloupci je roven 1.

Při konstrukci tabulky sloupcově podmíněných relativních četností vyjdeme opět z tabulky simultánních absolutních četností. Aplikováním funkce prop.table() s argumentem margin = 2 na kontingenční tabulku KT.abs získáme tabulku sloupcově podmíněných relativních četností.

```
# Sloupcove podminene relativni cetnosti
KT.rel.s <- prop.table(KT.abs, margin = 2)
round(KT.rel.s, digits = 4)
```

```
     wei
edu    nizka  norma vysoka
  ZS  0.3125 0.2659 0.1860
  SS  0.3292 0.3273 0.4651
  SSm 0.3042 0.3434 0.2558
  VS  0.0542 0.0634 0.0930
```

Interpretace výsledků: Ze všech novorozenců v datovém souboru, kteří mají maximálně dva starší sourozence a jejichž porodní hmotnost byla nízká, se 31.25 % narodilo matkám s ukončeným základním vzděláním. Ze všech

novorozenců v datovém souboru, kteří mají maximálně dva starší sourozence a jejichž porodní hmotnost byla v normě, se 32.73 % se narodilo matkám s dokončeným středoškolským vzděláním bez maturity a 34.34 % se narodilo matkám se středoškolským vzděláním ukončeným maturitou. ♣

2.3 Jednorozměrné intervalové rozdělení četností

Dataset 2: Délkově-šířkové rozměry lebky egyptské populace

Z archivních materiálů (Schmidt, 1888; soubor 01-one-sample-mean-skull-mf.txt) máme k dispozici původní kraniometrické údaje o délce a šířce lebky ze starověké egyptské populace. Současně máme k dispozici průměrné hodnoty obou rozměrů, hodnoty směrodatné odchylky a počty případů vzorku novověké egyptské populace (délka lebky: $\bar{x}_m = 177.568\,\text{mm}$, $\bar{x}_f = 171.962\,\text{mm}$; $s_m = 7.526\,\text{mm}$, $s_f = 7.052\,\text{mm}$; $n_m = 88$, $n_f = 52$ a šířka lebky: $\bar{x}_m = 136.402\,\text{mm}$, $\bar{x}_f = 131.038\,\text{mm}$; $s_m = 6.411\,\text{mm}$, $s_f = 5.361\,\text{mm}$; $n_m = 88$, $n_f = 52$).

Popis proměnných v datasetu 2:

- id – pořadové číslo;
- pop – populace (egant – egyptská starověká);
- sex – pohlavie (m – muž, f – žena);
- skull.L – největší délka mozkovny (mm), t.j. přímá vzdálenost kraniometrických bodů *glabella* a *opisthocranion*;
- skull.B – největší šířka mozkovny (mm), t.j. vzdálenost obou kraniometrických bodů *euryon*.

Příklad 2.8. Načtení datového souboru
Načtěte dataset 01-one-sample-mean-skull-mf.txt a vypište první čtyři řádky z načteného souboru. Prozkoumejte, zda soubor obsahuje neznámé hodnoty a případně je ze souboru odstraňte. Potom zjistěte dimenzi datové tabulky.

Řešení příkladu 2.8
Datový soubor načteme příkazem read.delim(). První čtyři řádky vypíšeme pomocí příkazu head() se specifikací argumentu n = 4.

```
data <- read.delim('01-one-sample-mean-skull-mf.txt')
head(data, n = 4)
```

```
  id   pop sex skull.L skull.B
1 416 egant   m      188      145
2 417 egant   m      172      139
3 420 egant   m      176      138
4 421 egant   m      184      128
```

Načtená datová tabulka obsahuje jednu identifikační proměnnou id a údaje o čtyřech znacích: populaci (pop), pohlaví skeletu (sex), největší délce mozkovny (skull.L) a největší šířce mozkovny (skull.B). Pomocí funkce is.na() zjistíme, zda načtený soubor obsahuje neznámé hodnoty.

```
sum(is.na(data))
```

```
[1] 5
```

Počet neznámých hodnot v datovém souboru $n_{NA} = 5$. Podívejme se nyní, kde přesně se v souboru NA hodnoty vyskytují.

```
data[apply(is.na(data), MARGIN = 1, FUN = sum) > 0, ]
```

```
     id   pop sex skull.L skull.B
38  477 egant   m      NA      NA
110 554 egant   m     183      NA
222 456 egant   f      NA      NA
```

Funkce is.na() nám označí číslem 1 pozice, na kterých se v tabulce data vykytují NA hodnoty, a číslem 0 pozice, na kterých se NA hodnoty nevyskytují. Získáme tedy tabulku nul a jedniček. V této tabulce potom vypočítáme řádkové součty nul a jedniček (funkce apply() s argumenty MARGIN = 1 a FUN = sum). Protože číslem 1 je označena pozice v řádku, na které se vyskytlo NA pozorování, bude součet nul a jedniček v řádku s NA pozorováním větší než 0. Pomocí logického operátoru > a podmnožinového operátoru [,] potom vypíšeme z tabulky data pouze ty

řádky, pro něž byl řádkový součet větší než 0, čímž získáme řádky s výskytem NA hodnot. Vidíme, že hodnoty chybí celkem u tří objektů, přičemž u dvou objektů chybí oba délkové rozměry a u jednoho objektu chybí pouze údaj o největší šířce mozkovny.

```
[1] 325   5
```

Po odstranění řádků obsahujících NA pozorování (funkce na.omit()) nám zůstala datová tabulka o velikosti 325 řádků a pěti sloupců. Celkem tedy máme údaje o 325 skeletech, přičemž u každého skeletu máme záznamy o jedné identifikační proměnné a čtyřech znacích. ♣

Příklad 2.9. Histogram
V následujících dvou příkladech se zaměříme primárně na znak X = *největší šířka mozkovny* u skeletů mužského pohlaví. Proveďte prvotní náhled na tento znak sestrojením histogramu.

Řešení příkladu 2.9
Z tabulky data nejprve vyselektujeme údaje o největší šířce mozkovny pro muže. Dále zjistíme, kolik takovýchto údajů máme k dispozici (pomocí funkce length()) a v jakém rozmezí se naměřené hodnoty pohybují (funkce range()).

```
skull.BM  <- data[data$sex == 'm', 'skull.B']
(n.M      <- length(skull.BM))
```

```
[1] 216
```

```
range(skull.BM)
```

```
[1] 124 149
```

Celkem máme údaje o největší šířce mozkovny u 216 mužských skeletů. Naměřené hodnoty se pohybují v rozmezí 124–149 mm.

Jelikož je sledovaný znak X spojitého typu, je potřeba naměřené hodnoty roztřídit do stejně dlouhých *třídicích intervalů*. V praxi to znamená, že vytvoříme intervaly pokrývající svým rozsahem celou reálnou osu, tj.

$$(\infty; u_1\rangle\ ,\ (u_1; u_2\rangle,\ \ldots,\ (u_r; u_{r+1}\rangle,\ (u_{r+1}; \infty),$$

kde

$(u_j; u_{j+1}\rangle$, $j = 1, \ldots, r$, je j-tý třídicí interval. Krajní intervaly $(\infty; u_1\rangle$ a $(u_{r+1}; \infty)$ jako třídicí intervaly neuvažujeme, nikdy neobsahují žádné naměřené hodnoty a slouží pouze jako doplněk k třídícím intervalům. Počet třídicích intervalů se mění v závislosti na počtu naměřených hodnot. Přesný počet třídicích intervalů r v konkrétním případě stanovíme pomocí tzv. Sturgesova pravidla

$$r \approx 1 + 3.3 \log_{10} n, \tag{1}$$

kde n je počet naměřených hodnot.

```
(r <- round(1 + 3.3 * log10(n.M)))
```

```
[1] 9
```

Podle Sturgersova pravidla je optimální počet třídicích intervalů pro znak X = *největší šířka mozkovny* roven 9. Minimální naměřená hodnota znaku X je 124, maximální naměřená hodnota je 149. Optimální šířku jednoho třídicího intervalu spočítáme odečtením minimální hodnoty 124 od maximální hodnoty 149, vydělením tohoto rozdílu optimálním počtem třídidích intervalů a zaokrouhlením výsledku na nejbližší vyšší celé číslo. Toto specifické zaokrouhlení provedeme příkazem ceiling().

```
(d <- ceiling((149 - 124) / r))
```

```
[1] 3
```

Optimální šířka jednoho třídicího intervalu pro znak X je 3 mm. Vynásobíme-li počet třídicích intervalů optimální šířkou jednoho intervalu, zjistíme, že celkový rozsah třídicích intervalů je $9 \times 3 = 27$. Rozsah hodnot 124–149 je však pouze 25. Proto dolní hranici prvního třídicího intervalu stanovíme jako $u_1 = 123$. Protože šířka jednoho třídicího intervalu má být rovná 3, budou další hranice stanoveny jako $u_2 = 126$, $u_3 = 129$, ..., $u_9 = 150$. Pomocí funkce seq() vytvoříme vektor hranic třídicích intervalů (hranice) a vektor středů třídicích intervalů (centr). Oba vektory využijeme při tvorbě histogramu pro znak $X =$ *největší šířka mozkovny* pro muže.

```
161 hranice <- seq(123, 150, by = 3)
162 centr   <- seq(124.5, 148.5, by = 3)
```

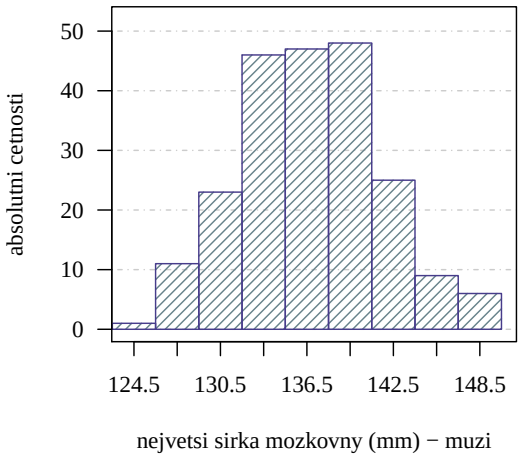
Histogram vykreslíme pomocí funkce hist(). Konstukci histogramu zahájíme přípravou prázdného grafu s připravenými popisky. Prvním argumentem bude vektor naměřených hodnot znaku X (skull.B). Argumentem col (resp. border) zvolíme barvu výplně (resp. ohraničení) sloupců jako bílou. Argumentem ylim stanovíme měřítko osy y v rozsahu 0–52 a specifikací argumentu axes = F zakážeme vykreslení měřítek os x a y. Argumenty xlab a ylab změníme popisky osy x a osy y a specifikací argumentu main = '' odstraníme nadpis grafu.

Do grafu dokreslíme referenční čáry pomocí funkce abline(). Argumentem h specifikujeme vykreslení horizontálních čar v posloupnosti čísel 0, 10, ..., 60, šedou barvou (argument col) a čerchovanou čárou (argument lty = 4). Dále kolem grafu obkreslíme černý rámeček pomocí příkazu box() se specifikací argumentu bty = 'o'.

Nyní do grafu dokreslíme příkazem hist() požadovaný histogram. Přidání histogramu do stávajícího grafu nastavíme specifikací argumentu add = T. Roztřídění naměřených hodnot do třídicích intervalů s hranicemi stanovenými dle našich preferencí nastavíme specifikací argumentu breaks = hranice . Barvu výplně (col) a ohraničení sloupců (border) zvolíme v odstínu modré. Argumentem density nastavíme šrafování výplně sloupců s intenzitou hustoty čar 20.

Nakonec do grafu doplníme měřítko osy x tak, aby, zobrazené měřítko, podle zavedené konvence, uvádělo středy třídicích intervalů. K tomu nám dopomůže funkce axis() s argumenty side = 1 a at = centr. Měřítko osy y doplníme specifikací argumentu side = 2 ve funkce axis(). Zobrazení popisků měřítka osy y v horizontálním směru změníme argumentem las.

```
163 hist(skull.BM, col = 'white', border = 'white',
164      ylim = c(0, 52),  axes = F,
165      xlab = 'nejvetsi sirka mozkovny (mm) - muzi',
166      ylab = 'absolutni cetnosti', main = '')
167 abline(h = seq(0, 60, by = 10), col = 'grey80', lty = 4)
168 box(bty = 'o')
169
170 hist(skull.BM, add = T, breaks = hranice,
171      col = 'lightblue4', border = 'slateblue4', density = 20)
172 axis(side = 1, at = centr)
173 axis(side = 2, las = 1)
```

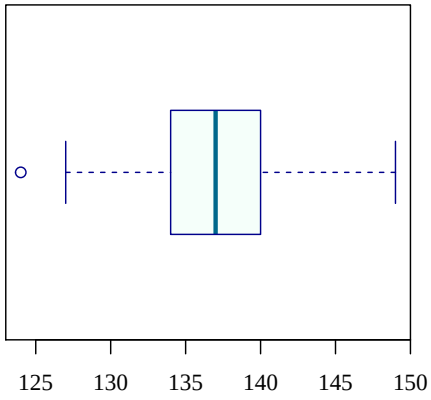


♣

Příklad 2.10. Krabicový diagram
Sestrojte krabicový diagram pro znak X = *největší šířka mozkovny* u skeletů mužského pohlaví.

Řešení příkladu 2.10
Krabicový diagram znaku X vykreslíme příkazem boxplot(). Prvním argumentem bude vektor naměřených hodnot znaku X (skull.BM). Argumentem type = 2 nastavíme výpočet hranic krabice pomocí jednoduchého výpočtu analogickému ručnímu výpočtu bez zbytečných aproximací. Argument horozintal změní polohu grafu ze svislé na vodorovnou. Barvu výplně grafu (col), hranice grafu (border) i čáru uprostřed grafu (medcol) reprezentující polohu mediánu (viz kapitola **??**) zvolíme opět v odstínech modré. Popisek osy x změníme argumentem xlab.

```
boxplot(skull.BM, type = 2, horizontal = T,
        col = 'mintcream', border = 'darkblue', medcol = 'deepskyblue4',
        xlab = 'nejvetsi sirka mozkovny (mm) - muzi')
```



♣

Příklad 2.11. Histogram a krabicový diagram
V tomto příkladu se zaměříme na znak Y = *největší délka mozkovny* u skeletů mužského pohlaví. Proveďte prvotní náhled na tento znak pomocí histogramu a krabicového diagramu.

Řešení příkladu 2.11
Z tabulky data nejprve vyselektujeme údaje o největší délce mozkovny pro muže a zjistíme, kolik takovýchto údajů máme k dispozici a v jakém rozmezí pohybují naměřené hodnoty.

```
skull.LM  <- data[data$sex == 'm', 'skull.L']
(n.M      <- length(skull.LM))
```

```
[1] 216
```

```
range(skull.LM)
```

```
[1] 164 199
```

Celkem máme údaje o největší délce mozkovny u 216 mužských skeletů. Naměřené hodnoty se pohybují v rozmezí 164–199 mm. Jelikož znak *největší délka mozkovny* pro skelety mužského pohlaví je spojitého typu, rozdělíme opět data do vhodného počtu stejně širokých třídicích intervalů. Počet intervalů stanovíme pomocí Sturgesova pravidla.

```
(r <- round(1 + 3.3 * log10(n.M)))
```

```
[1] 9
```

Optimální počet třídicích intervalů pro znak Y = *největší délka mozkovny* u skeletů mužského pohlaví je roven 9. Minimální naměřená hodnota znaku Y je 164, maximální naměřená hodnota je 199. Optimální šířku jednoho třídicího intervalu spočítáme odečtením minimální hodnoty 164 od maximální hodnoty 199, vydělením tohoto rozdílu optimálním počtem třídidích intervalů a zaokrouhlením na nejbližší vyšší celé číslo (funkce ceiling()).

```
(d <- ceiling((199 - 164) / r))
```

```
[1] 4
```

Optimální šířka jednoho třídicího intervalu pro znak Y je 4 mm. Vynásobíme-li počet třídicích intervalů optimální šířkou jednoho intervalu, zjistíme, že celkový rozsah třídicích intervalů je $9 \times 4 = 36$. Rozsah hodnot 164–199 je však pouze 35. Proto dolní hranici prvního třídicího intervalu stanovíme jako $u_1 = 163$. Jelikož šířka jednoho třídicího intervalu má být rovná 4, budou další hranice stanoveny jako $u_2 = 167$, $u_3 = 171$, ..., $u_9 = 199$.

Nyní již můžeme vytvořit histogram pro znak Y = *největší délka mozkovny* u skeletů mužského pohlaví. Pomocí funkce seq() vytvoříme nejprve posloupnost hranic třídicích intervalů (hranice) a posloupnost středů každého třídicího intervalu (centr). Histogram vykreslíme pomocí funkce hist(). Konstukci histogramu opět zahájíme přípravou prázdného grafu s připravenými popisky. Do grafu zaneseme horizontální referenční čáry (abline()) a okolo grafu obkreslíme černý rámeček (box()). Opětovným použitím příkazu hist() se specifikací argumentu add = T dokreslíme do prázdného grafu požadovaný histogram s námi zvolenými hranicemi (argument breaks = hranice). Nakonec doplníme do grafu měřítko osy x, resp. y (funkce axis() se specifikací argumentu side = 1, resp. side = 2).

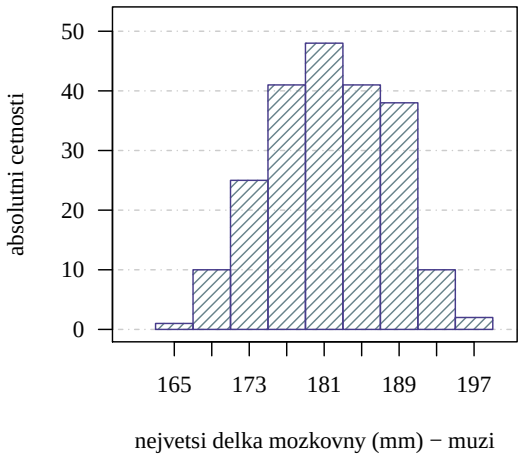
Krabicový diagram vykreslíme příkazem boxplot().

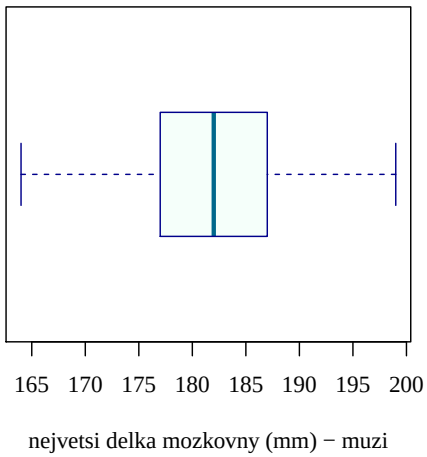
```
# Histogram
hranice <- seq(163, 199, by = 4)
centr   <- seq(165, 197, by = 4)

hist(skull.LM, col = 'white', border = 'white',
     ylim = c(0, 52),  axes = F,
     xlab = 'nejvetsi delka mozkovny (mm) - muzi',
     ylab = 'absolutni cetnosti', main = '')
abline(h = seq(0, 60, by = 10), col = 'grey80', lty = 4)
box(bty = 'o')

hist(skull.LM, add = T, breaks = hranice,
     col = 'lightblue4', border = 'slateblue4', density = 20)
axis(side = 1, at = centr)
axis(side = 2, las = 1)

# Krabicový diagram
boxplot(skull.LM, type = 2, horizontal = T,
        col = 'mintcream', border = 'darkblue', medcol = 'deepskyblue4',
        xlab = 'nejvetsi delka mozkovny (mm) - muzi')
```





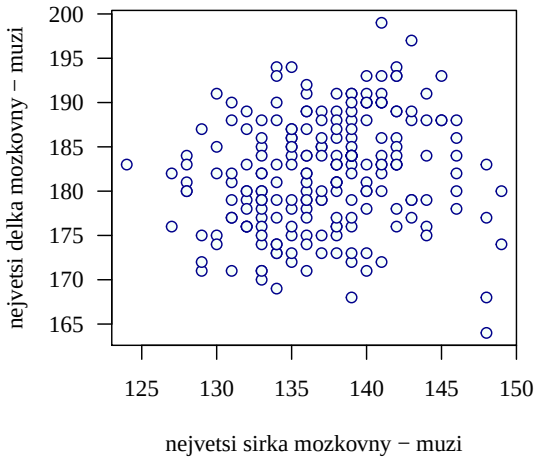
♣

2.4 Dvourozměrné intervalové rozdělení četností

Příklad 2.12. Dvourozměrný tečkový diagram
Zaměřte se nyní na oba znaky X = *největší šířka mozkovny* a Y = *největší délka mozkovny* u skeletů mužského pohlaví najednou. Vytvořte dvourozměrný tečkový diagram reprezentující vztah mezi znaky X a Y.

Řešení příkladu 2.12
Dvourozměrný tečkový diagram vykreslíme příkazem plot(). Prvními dvěma argumenty jsou vektory naměřených hodnot (skull.BM a skull.LM). Argumentem pch = 21 nastavíme vykreslení specifického typu bodů, které mají kulatý tvar a u nichž je možné nastavit barvu obrysu (argument col) a barvu výplně bodů (argument bg). Argumenty xlab a ylab změníme popisky osy x a osy y, argumentem las změníme směr popisků měřírka osy y ze svislých na vodorovné.

```
206 plot(skull.BM, skull.LM, pch = 21, col = 'darkblue', bg = 'mintcream',
207      xlab = 'nejvetsi sirka mozkovny - muzi',
208      ylab = 'nejvetsi delka mozkovny - muzi', las = 1)
```



♣

2.5 Příklady k samostatnému procvičování

Příklad 2.13. Opakování: Načtení datového souboru
Načtěte dataset 17-anova-newborns.txt. Ze souboru odstraňte neznámé hodnoty. V následující analýze se zaměřte pouze na novorozence, kteří mají maximálně dva starší sourozence. Tyto novorozence rozdělte podle porodní hmotnosti do tří kategorií: nizka – hmotnost novorozence je nižší než 2500 g; norma – hmotnost novorozence se pohybuje v rozmezí 2500–4200 g; vysoka – hmotnost novorozence je vyšší než 4200 g. Nakonec upravte označení jednotlivých variant znaku $X =$ *počet starších sourozenců* (0 – zadny, 1 – jeden, 2 – dva). Vypište prvních 6 řádků z upraveného souboru a zjistěte dimenzi datového souboru.

Řešení příkladu 2.13

```
  edu.M prch.N sex.C weight.C weight.K
1     2  zadny     m     3470    norma
2     2  zadny     m     3240    norma
3     2  zadny     f     2980    norma
4     1  zadny     m     3280    norma
5     3  zadny     m     3030    norma
6     2  jeden     m     3650    norma
```

Po odstranění NA hodnot má datová tabulka celkem 1276 řádků a 5 sloupců. Celkem tedy máme k dispozici údaje o pěti znacích u 1276 novorozenců.

♣

Příklad 2.14. Variační řada Vytvořte variační řadu znaku $X =$ *počet starších sourozenců*. Výsledky variační řady interpretujte.

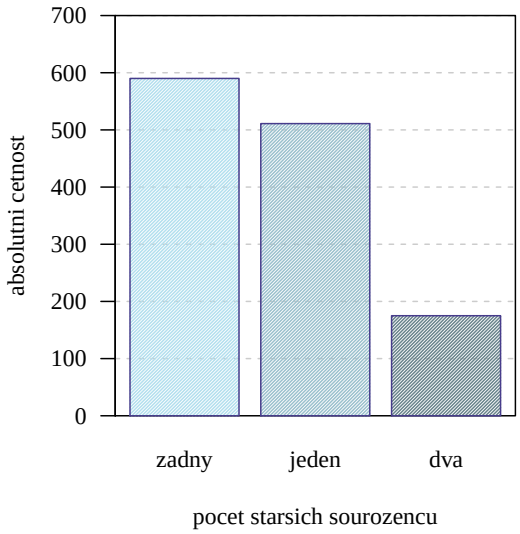
Řešení příkladu 2.14

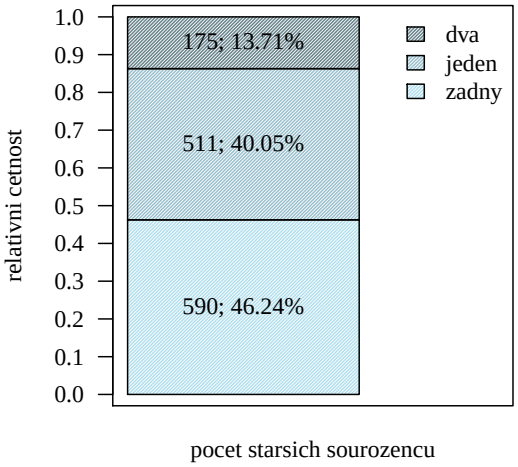
```
       nj     pj   Nj     Fj
zadny 590 0.4624  590 0.4624
jeden 511 0.4005 1101 0.8629
dva   175 0.1371 1276 1.0000
```

Interpretace výsledků: Z celkového počtu 1276 novorozenců je 590 novorozenců (46.24 %) prvorozených. Z celkového počtu 1276 novorozenců je 1101 (86.29 %) novorozenců prvorozených nebo druhorozených. ♣

Příklad 2.15. Sloupcový diagram absolutních a relativních četností
Nakreslete sloupcový diagram absolutních četností a sloupcový diagram relativních četností pro znak $X =$ *počet starších sourozenců*.

Řešení příkladu 2.15


absolutni cetnost
700
600
500
400
300
200
100
0
zadny
jeden
dva
pocet starsich sourozencu


relativni cetnost
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
175; 13.71%
511; 40.05%
590; 46.24%
dva
jeden
zadny
pocet starsich sourozencu

♣

Příklad 2.16. Kontingenční tabulka absolutních a relativních četností
Zaměřme se nyní na oba znaky X = *počet starších sourozenců* a Y = *porodní hmotnost novorozence* najednou. Sestrojte kontingenční tabulku absolutních četností a kontingenční tabulku relativních četností znaků X a Y. Hodnoty v tabulkách interpretujte.

Řešení příkladu 2.16
Kontingenční tabulka absolutních četností

```
      nizka norma vysoka suma
zadny   123   456     11  590
jeden    91   399     21  511
dva      26   138     11  175
suma    240   993     43 1276
```

Kontingenční tabulka relativních četností

```
       nizka  norma vysoka   suma
zadny 0.0964 0.3574 0.0086 0.4624
jeden 0.0713 0.3127 0.0165 0.4005
dva   0.0204 0.1082 0.0086 0.1371
suma  0.1881 0.7782 0.0337 1.0000
```

Interpretace výsledků: V datovém souboru se vyskytuje 123 (9.64 %) prvorozených novorozenců s nízkou porodní hmotností, 399 (31.27 %) druhorozených novorozenců, jejichž porodní hmotnost je v normě a 11 (0.86 %) novorozenců s dvěma staršími sourozenci a vysokou porodní hmotností. ♣

Příklad 2.17. Kontingenční tabulka řádkově a sloupcově podmíněných relativních četností
Vytvořte kontingenční tabulku řádkově podmíněných relativních četností k-té varianty znaku Y = *porodní hmotnost novorozenců*, $k = 1, \ldots, 3$, za předpokladu pevně stanovené j-té varianty znaku X = *počet starších sourozenců*, $j = 1, \ldots, 4$. Dále vypočtěte kontingenční tabulku sloupcově podmíněných relativních četností j-té varianty znaku X, $j = 1, \ldots, 4$, za předpokladu pevně stanovené k-té varianty znaku Y, $k = 1, \ldots, 3$. Hodnoty v tabulkách interpretujte.

Řešení příkladu 2.17
Kontingenční tabulka řádkově podmíněných relativních četností

```
       wei
prch     nizka  norma vysoka
  zadny 0.2085 0.7729 0.0186
  jeden 0.1781 0.7808 0.0411
  dva   0.1486 0.7886 0.0629
```

Interpretace výsledků: Ze všech prvorozených novorozenců v datovém souboru má 20.85 % nízkou porodní hmotnost, 1.86 % vysokou porodní hmotnost a 77.29 % má porodní hmotnost v normě.

Kontingenční tabulka sloupcově podmíněných relativních četností

```
       wei
prch     nizka  norma vysoka
  zadny 0.5125 0.4592 0.2558
  jeden 0.3792 0.4018 0.4884
  dva   0.1083 0.1390 0.2558
```

Interpretace výsledků: Ze všech novorozenců v datovém souboru, kteří mají porodní hmotnost v normě, je 45.92 % prvorozených, 40.18 % druhorozených a 13.90 % má dva starší sourozence.

♣

Příklad 2.18. Načtení datového souboru
Načtěte dataset 01-one-sample-mean-skull-mf.txt a vypište jeho prvních šest řádků. Ze souboru odstraňte NA hodnoty a zjistěte dimenzi datové tabulky.

Řešení příkladu 2.18

```
240 head(data, n = 6)
```

```
   id   pop sex skull.L skull.B                                                  241
1 416 egant   m      188      145                                                 242
2 417 egant   m      172      139                                                 243
3 420 egant   m      176      138                                                 244
4 421 egant   m      184      128                                                 245
5 422 egant   m      183      139                                                 246
6 423 egant   m      177      143                                                 247
```

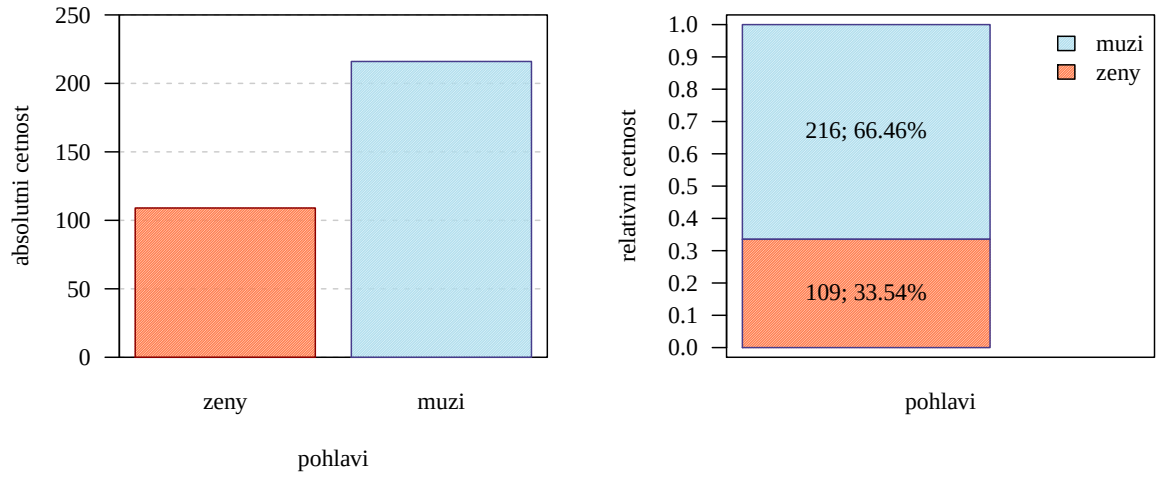
Po odstranění NA hodnot disponuje datová tabulka 325 řádky a 5 sloupci. Celkem máme k dispozici údaje o 325 skeletech, přičemž u každého skeletu máme záznam o jedné identifikační proměnné a čtyřech znacích. ♣

Příklad 2.19. Variační řada, sloupcový diagram absolutních (resp. relativních) četností
Zaměřte se nyní na kategoriální znak X = *pohlaví*. Pro tento znak vytvořte variační řadu a sestrojte soupcový diagram absolutních četností a sloupcový diagram relativních četností. Výsledky variační řady interpretujte. Zamyslete se nad tím, zda je možné na základě současného datového souboru sestrojit kontingenční tabulku simultánních absolutních (resp. relativních) četností. Jaké kroky by bylo potřeba podniknout, aby sestrojení tabulek bylo možné?

Řešení příkladu 2.19

```
      nj     pj  Nj     Fj                                                       248
zeny 109 0.3354 109 0.3354                                                      249
muzi 216 0.6646 325 1.0000                                                      250
```

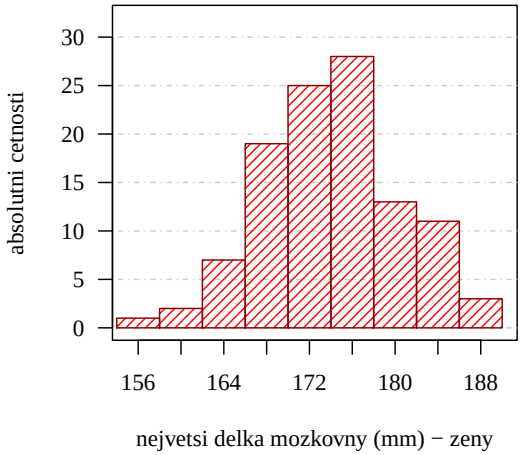
Interpretace výsledků: V datovém souboru se vyskytuje celkem 325 skeletů, z čehož 109 (33.54 %) skeletů je ženského pohlaví a 216 (66.46 %) skeletů je mužského pohlaví.

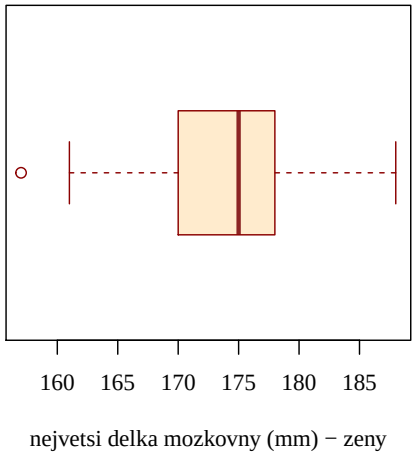


Odpověď na otázku: K sestrojení kontingenční tabulky simultánních absolutních (resp. relativních) četností potřebujeme dva znaky kategoriálního typu. Protože v databázi máme pouze jeden znak kategoriálního typu (*pohlaví*), museli bychom druhý znak zajistit kategorizací jedné ze spojitých proměnných, tedy buď proměnné *největší délka mozkovny* nebo proměnné *největší šířka mozkovny*. ♣

Příklad 2.20. Histogram a krabicový diagram
Zaměřte se na znak $X =$ *největší délka mozkovny* u skeletů ženského pohlaví a proveďte prvotní náhled na tento znak. Pomocí Sturgesova pravidla určete optimální počet třídicích intervalů, následně optimální délku každého třídicího intervalu a stanovte hranice jednotlivých třídicích intervalů. Vykreslete histogram a krabicový diagram pro znak *největší délka mozkovny* u skeletů ženského pohlaví.

Řešení příkladu 2.20
Optimální počet třídidích intervalů pro *největší délku mozkovny* u skeletů ženského pohlaví je podle Sturgesova pravidla roven 8. Optimální šířka každého třídicího intervalu je 4 mm.

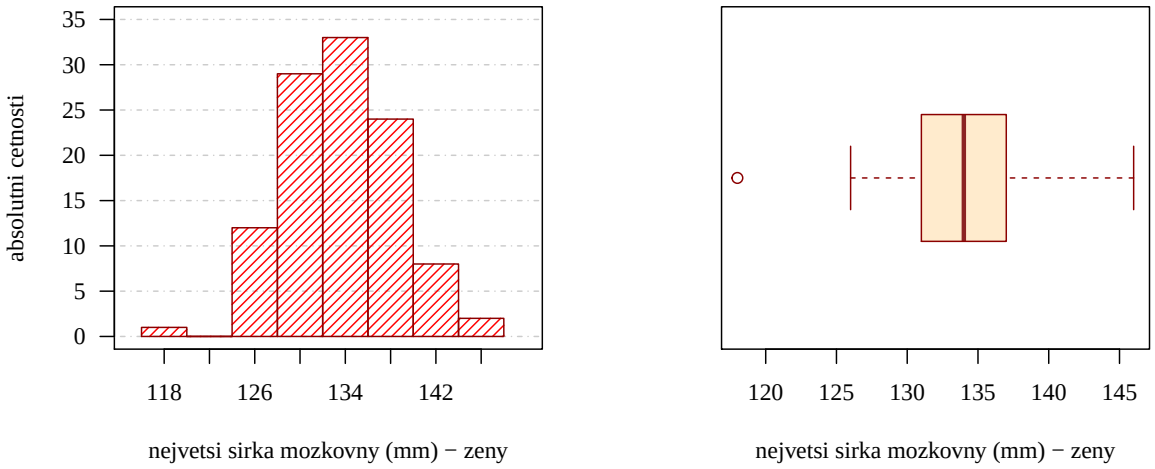




♣

Příklad 2.21. Histogram a krabicový diagram
Zaměřte se na znak $Y =$ *největší šířka mozkovny* u skeletů ženského pohlaví a proveďte prvotní náhled na tento znak. Pomocí Sturgesova pravidla určete optimální počet třídicích intervalů, následně optimální šířku každého třídicího intervalu a stanovte hranice jednotlivých třídicích intervalů. Vykreslete histogram a krabicový diagram pro znak *největší šířka mozkovny* u skeletů ženského pohlaví.

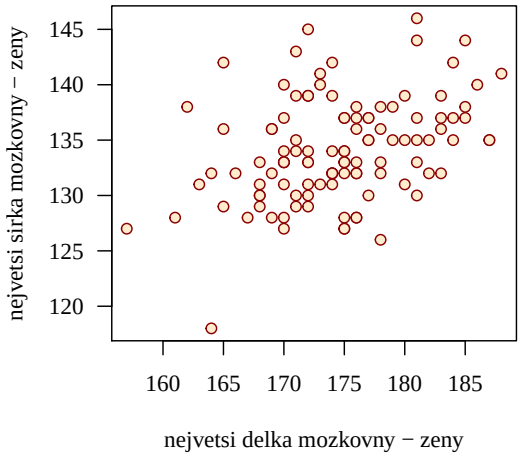
Řešení příkladu 2.21
Optimální počet třídidích intervalů pro *největší šířku mozkovny* u skeletů ženského pohlaví je podle Sturgesova pravidla roven 8. Optimální šířka každého třídicího intervalu je 4 mm.



♣

Příklad 2.22. Dvourozměrný tečkový diagram
Zaměřte se nyní na oba znaky X = *největší délka mozkovny* a Y = *největší šířka mozkovny* u skeletů ženského pohlaví najednou. Vytvořte dvourozměrný tečkový diagram reprezentující vztah mezi znaky X a Y.

Řešení příkladu 2.22



♣