

Statistická inference I

Téma 10: Normální model - Věrohodnost

Veronika Bendová

bendova.veroonika@gmail.com

Princip věrohodnosti

- $f(\mathbf{x}|\theta)$... hustota rozdělení $\mathcal{L}(\theta)$
- $L(\theta|\mathbf{x})$... funkce věrohodnosti
 - $L(\theta|\mathbf{x}) = c(\mathbf{x})f(\mathbf{x}|\theta)$, kde $c \in \mathbb{R}$ je nezávislá na θ a $f(\mathbf{x}|\theta) = \prod_{i=1}^n f(x_i|\theta)$.
- $\ell(\theta|\mathbf{x})$... logaritmus funkce věrohodnosti
 - $\ell(\theta|\mathbf{x}) = \ln(L(\theta|\mathbf{x})) = \ln(c(\mathbf{x})) + \ln(f(\mathbf{x}|\theta))$
- $S(\theta)$... skóre funkce (první derivace logaritmu funkce věrohodnosti)
 - $S(\theta) = \frac{\partial}{\partial \theta} \ell(\theta|\mathbf{x})$
- $\hat{\theta}$... MLE odhad parametru θ
 - lze jej získat maximalizací fce věrohodnosti $L(\theta|\mathbf{x})$ resp. log. fce věrohodnosti $\ell(\theta|\mathbf{x})$
 - odpovídá kořenu skóre funkce $S(\theta)$; ten můžeme vypočítat explicitně (je-li to možné), nebo vhodnou numerickou metodou. Pro nalezení jednorozměrného parametru θ použijeme např. Newton-Raphsonovu metodu, metodu sečen, nebo metodu bisekce.
- $\mathcal{I}(\hat{\theta})$... Fisherova míra informace (mínus druhá derivace logaritmu věrohodnostní funkce vyjádřená v MLE odhadu parametru θ)
 - $\mathcal{I}(\hat{\theta}) = -\frac{\partial^2}{\partial \theta^2} \ell(\theta|\mathbf{x})|_{\theta=\hat{\theta}}$
- $\text{Var}[\hat{\theta}]$... MLE odhad rozptylu odhadu parametru θ
 - $\text{Var}[\hat{\theta}] = \frac{1}{\mathcal{I}(\hat{\theta})}$

Dataset 6: 03-paired-means-clavicle2.txt

- Datový soubor obsahuje osteometrické údaje o délkách klíčních kostí (*clavicula*). Data pochází z anglického souboru dokumentovaných skeletů (Parsons, 1916). V souboru se nachází délky klíčních kostí na pravé a levé straně těla v párovém uspořádání. Jednotlivé kosti bez druhostranné kosti nebyly do souboru zařazeny.
- Přehled proměnných v datasetu:
 - id ... ID jedince;
 - sex ... pohlaví jedince (*m* - muž, *f* - žena);
 - length.L ... délka klíční kosti z levé strany (v mm);
 - length.R ... délka klíční kosti z pravé strany (v mm).

Příklad 10.1 Maximálně věrohodné odhady parametrů μ a σ^2 v normálním modelu

Načtěte datový soubor 03-paired-means-clavicle2.txt. Necht' náhodná proměnná X popisuje délku klíční kosti z levé strany u mužů. Za předpokladu, že náhodná veličina X pochází z normálního rozdělení, tj. $X \sim N(\mu, \sigma^2)$:

1. Odvod'te
 - a. tvar jádra věrohodnostní funkce $L((\mu, \sigma^2)^T | x)$ normálního modelu;
 - b. tvar jádra logaritmu věrohodnostní funkce $\ell((\mu, \sigma^2)^T | x)$ normálního modelu;
 - c. skóre funkci pro parametr μ + MLE odhad parametru μ ;
 - d. skóre funkci pro parametr σ^2 + MLE odhad parametru σ^2 ;
 - e. tvar Fisherovy informační matice.
2. Dosazením do vzorců stanovte přesnou hodnotu maximálně věrohodného odhadu parametrů μ a σ^2 , tj. $\hat{\mu}$ a $\hat{\sigma}^2$.
3. Pomocí maximalizace logaritmu věrohodnostní funkce $\ell((\mu, \sigma^2)^T | x)$ normálního modelu nalezněte maximálně věrohodný odhad parametrů μ a σ^2 . Maximalizaci proved'te
 - a. pomocí funkce `optim()`;
 - b. pomocí vlastnoručně naprogramované dvourozměrné Newton-Raphsonovy metody `NRnorm()`;
 - c. pomocí vlastnoručně naprogramované Broydenovy metody `BMnorm()`.
4. Vykreslete (a) vrstevnicový diagram; (b) 3D-diagram logaritmu dvourozměrné věrohodnostní funkce normálního modelu spolu s maximálně věrohodnými odhady parametrů μ a σ^2 odhadnutými pomocí funkce `optim()`. K vykreslení použijte (a) funkce `image()` a `contour()`, (b) funkci `persp()`.

Řešení příkladu 10.1

1. Odvod'te

a. tvar jádra věrohodnostní funkce $L((\mu, \sigma^2)^T | x)$ normálního modelu

b. tvar jádra logaritmu věrohodnostní funkce $\ell((\mu, \sigma^2)^T | x)$ normálního modelu

c. skóre funkci pro parametr μ + MLE odhad parametru μ

d. skóre funkci pro parametr σ^2 + MLE odhad paramertu σ^2

e. tvar Fisherovy informační matice

2. Dosazením do vzorců stanovte přesnou hodnotu maximálně věrohodného odhadu parametrů μ a σ^2 , tj. $\hat{\mu}$ a $\hat{\sigma}^2$.

Poznámka k pseudokódu: V celém pseudokódu této prezentace je proměnnými **sigma** i **sigma2** myšlena hodnota parametru σ^2 . Název **sigma** používáme zejména proto, aby naprogramované funkce byly přehlednější. Taktéž proměnná **theta[2]** ve funkci **lnormo()** reprezentuje parametr σ^2 .

```
1 data <- read.delim(...) # nacteni dat
2 cla.L <- data[... , ...] # vyber delek leve kl. kosti muzu
3 data <- na.omit(...) # odstraneni NA hodnot
4 n <- ... # delka vektoru cla.L
5 mu <- ... # MLE odhad parametru mu
6 sigma2 <- ... # MLE odhad parametru sigma ^ 2
7 tab <- data.frame(...) # souhrnna tabulka vysledku
```

	mu	sigma2
1	153.6	96.96

8
9

3. Pomocí maximalizace logaritmu věrohodnostní funkce $\ell((\mu, \sigma^2)^T | x)$ normálního modelu nalezněte maximálně věrohodný odhad parametrů μ a σ^2 . Maximalizaci proveďte
- pomocí funkce `optim()`

```
10 lnormo <- function(theta, x){ # log. veroh. fce rozd. N(mu, sigma ^ 2)
11   n <- ... # delka vektoru x
12   like <- ... # log. fce veroh. rozdeleni N(theta[1], theta[2])
13   return(...) # vystup; promenna like
14 }
15 OPTtheta <- optim(c(145, 100), lnormo, x = cla.L,
16                   control = list(fnscale = -1)) # MLE odhad vektoru
17   # parametru theta = (mu, sigma ^ 2) ^ T
18 max.m <- OPTtheta$par[1] # MLE odhad parametru mu
19 max.s <- OPTtheta$par[2] # MLE odhad parametru sigma ^ 2
```

```
      mu    sigma2
1 153.6043 96.97404
```

20
21

- b. pomocí vlastnoručně naprogramované dvourozměrné Newton-Raphsonovy metody NRnorm();

```
22 lnorm <- function(mu, sigma, x){ # log. veroh. fce rod. N(mu, sigma ^ 2)
23   n <- ... # delka vektoru x
24   like <- ... # logaritmus veroh. fce rozd. N(mu, sigma ^ 2)
25   return(...) # vystup; promenna like
26 }
27
28 snorm <- function(mu, sigma, x){ # vektor skore funkci U
29   n <- # delka vektoru x
30   skore.mu <- ... # skore fce pro par. mu
31   skore.sigma <- # skore fce pro par. sigma ^ 2
32   U <- c(..., ...) # vektor U = (S(mu), S(sigma ^ 2)) ^ T
33   return(...) # vystup; vektor U
34 }
```

```

35 hnorm <- function(mu, sigma, x){ # Fisherova inf. matice I
36   n <- ... # delka vektoru x
37   I11 <- ... # I_{11}
38   I12 <- ... # I_{12} = I_{21}
39   I22 <- ... # I_{22}
40   I <- matrix(...) # Fisherova informaci matice I
41   return(...) # vystup; matice I
42 }
43
44 NMnorm <- function(x01, x02, t){
45   # iniciacni faze funkce; x = (mu, sigma ^ 2) ^ T
46   x <- c(x01, x02) # poc. nastaveni vektoru x v kroku 1
47   f1 <- ... # poc. nastaveni f1 v kroku 1: f1 = 0
48   f2 <- lnorm(...) # poc. nastaveni f2 v kroku 1: f2 = l(x[1], x[2] | t)
49   k <- ... # poc. nastaveni pocitadla k: k = 0
50
51   # telo funkce; proces probiha, dokud podminka while(...) plati
52   while(abs(f1 - f2) > 0.00001 & k < 1000){
53     k <- ... # zvetseni pocitadla o 1
54     U <- snorm(..., t) # vektor skore fci U = (S(x[1]), S(x[2]))^T
55     H <- hnorm(..., t) # FIM v hodnotach mu = x[1] a sigma^2 = x[2]
56     x <- x + solve(H) %*% U # aktualizace vektoru x
57     f1 <- f2 # aktualizace f1
58     f2 <- lnorm(..., t) # aktualizace f2: log. veroh. fce v novem x
59   }
60   return(list(mu = x[1], sigma2 = x[2], like = f2,
61             k = k)) # vystup; seznam o 4 polozkach
62 }
63 NMres <- NMnorm(x01 = ..., x02 = ..., cla.L) # MLE odhad (mu, sigma^2)^T
64   # x01 a x02 je treba zvolit vhodne, napr. x01 = 150 a x02 = 90
65 max.nm <- NMres$... # MLE odhad par. mu
66 max.ns <- NMres$... # MLE odhad par. sigma ^ 2

```

	mu	sigma2
1	153.6	96.36093

67

68

c. pomocí vlastnoručně naprogramované Broydenovy metody BMnorm()

```
69 BMnorm <- function(x10, x11, x20, x21, t){
70   # iniciacni faze funkce; x1 = (mu, sigma ^ 2)^T v kroku 1;
71   # x2 = (mu, sigma ^ 2)^T v kroku 2
72   x1 <- c(x10, x20) # poc. nastaveni vektoru x v kroku 1
73   x2 <- c(x11, x21) # poc. nastaveni vektoru x v kroku 2
74   f1 <- lnorm(x1[1], x1[2], t) # log. veroh. fce v kroku 1
75   f2 <- lnorm(...) # log. veroh. fce v kroku 2
76   k <- ... # poc. nastaveni pocitadla k: k = 0
77   B <- diag(2) # poc. nastaveni matice B (jednotkova m. dimenze = 2x2)
```

```

78 # telo funkce; proces probiha, dokud podminka while(...) plati
79 while(abs(f1 - f2) > 0.00001 & k < 1000){
80   k <- ... # zvetseni pocitadla o 1
81   U1 <- snorm(x1[1], x1[2], t) # vektor skore fci v kroku 1
82   U2 <- snorm(...) # vektor skore fci v kroku 2
83   B <- B + c(((...) - B %*% (...)) / sum(...) ^ 2) %*% t(c(...))
84   # aktualizace matice B
85   x <- (x2 - solve(B) %*% U2) # vypocet vektoru x
86   f1 <- ... # aktualizace f1 (nyni f2)
87   f2 <- lnorm(...) # aktualizace f2 (log. veroh. fce ve vektoru x)
88   x1 <- ... # aktualizace x1 (nyni x2)
89   x2 <- ... # aktualizace x2 (nyni x)
90 }
91 return(list(mu = x[1], sigma2 = x[2], like = f2,
92           k = k)) # vystup; seznam o 4 polozkach
93 }
94
95 BMres <- BMnorm(..., t = cla.L) # MLE odhad (mu, sigma ^ 2)^T
96 # x10, x11, x20 a x21 je treba zvolit vhodne, napr. 147, 157, 90 a 100.
97 max.bm <- BMres$mu # MLE odhad par. mu
98 max.bs <- BMres$sigma2 # MLE odhad par. sigma ^ 2

```

	mu	sigma2
1	153.6039	96.46132

99
100

Tabulka: Odhady parametrů μ a σ^2 normálního rozdělení

	μ	σ^2
exaktní výpočet	153.600000	96.960000
funkce optim()	153.604298	96.974037
Newton-Raphsonova metoda	153.600000	96.360928
Broydenova metoda	153.603912	96.461318

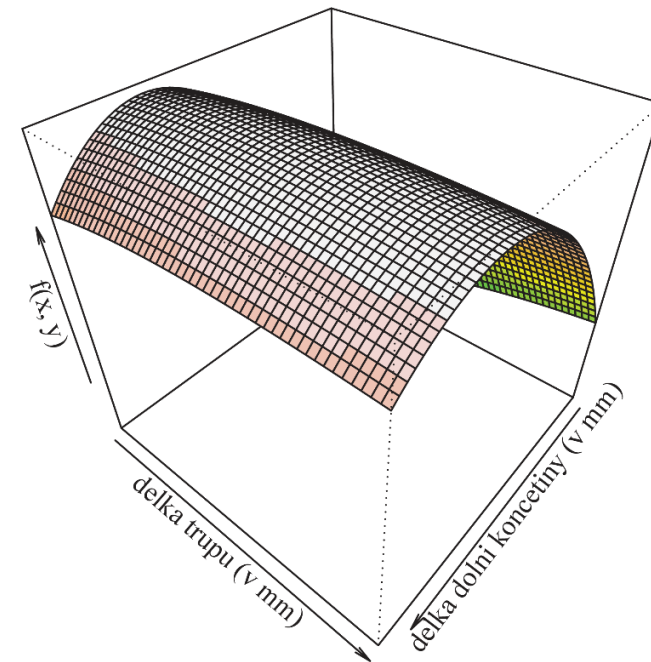
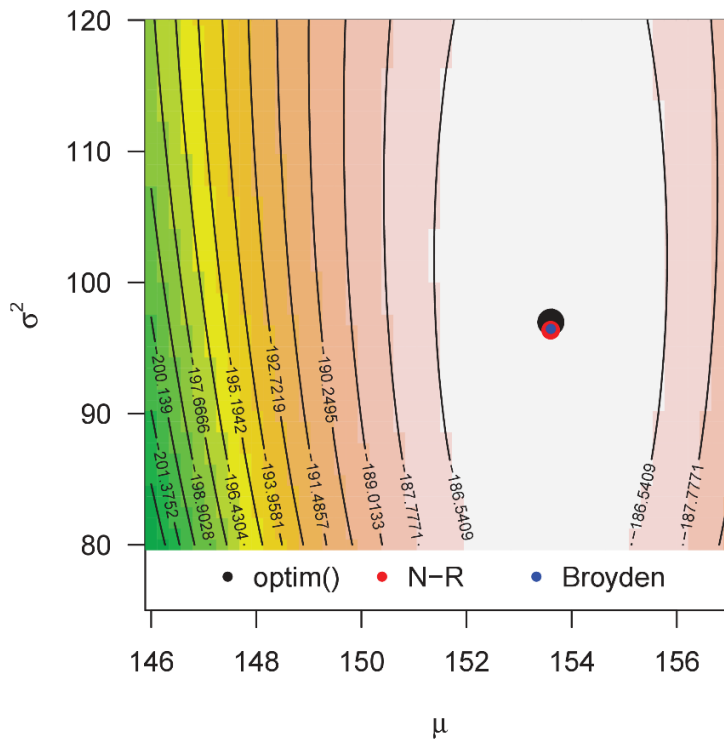
3. Vykreslete (a) vrstevnicový diagram; (b) 3D-diagram logaritmu dvourozměrné věrohodnostní funkce normálního modelu spolu s MLE odhady parametrů μ a σ^2 odhadnutými pomocí všech tří funkcí. K vykreslení použijte (a) funkce `image()` a `contour()`, (b) funkci `persp()`.

```
101 N <- 50
102 mu <- seq(...) # posl. bodu osy x; od 146 do 157 o delce N
103 sigma2 <- seq(...) # posl. bodu osy y; od 80 do 120 o delce N
104 M <- matrix(NA, N, N) # prazdna matice dimenze (N x N)
105 for(i in 1:length(mu)){
106   for(j in 1:length(sigma2)){
107     M[i, j] <- lnorm(mu[i], sigma2[j], cla.L)
108   }
109 } # hodnoty log. veroh. fce
110
111 k <- ... # pocet barev; k = 15
112 par(...) # okraje grafu 4, 4, 2, 2
113 image(..., asp = F, ylim = c(75, 120), col = terrain.colors(...),
114        xlab = expression(...), ylab = expression(...),
115        breaks = seq(min(M), max(M), length = k + 1),
116        las = ..., ) # vrstevnicovy diagram log. veroh. fce
117 contour(..., levels = ..., add = T) # konturovy diagram log. veroh. fce
118 points(...) # cerny bod; MLE odhad (mu, sigma ^ 2); fce optim
119 points(...) # cerveny bod MLE odhad (mu, sigma ^ 2); N-R metoda
120 points(...) # modry bod MLE odhad (mu, sigma ^ 2); Br. metoda
121 legend('bottom', horiz = T, ...) # legenda
```

```

122 nrz <- nrow(M) # pocet radku matice M
123 ncz <- ncol(M) # pocet sloupcu matice M
124 color <- terrain.colors(...) # paleta k barev
125 stredy <- (M[-1, -1] + M[-1, -ncz] + M[-nrz, -1] + M[-nrz, -ncz]) / 4
126 # matice stredy site
127 stredy.col <- cut(stredy, k) # rozdel rozpeti hodnot do k ekvidistantnich
128 # intervalu a kazde hodnote prirad interval, do ktereho nalezi
129 par(...) # okraje grafu 1, 1, 1, 1
130 persp(..., col = color[stredy.col], ticktype = ..., xlab = ..., ylab = ...,
131         zlab = ..., , phi = 40, theta = 130)) # 3D-diagram log. veroh. fce

```



Obrázek: (a) Vrstevnicový diagram (vlevo); (b) 3D-diagram (vpravo) logaritmu dvourozměrné věrohodnostní funkce normálního modelu spolu s maximálně věrohodnými odhady parametrů μ a σ^2