4.1

povinné parametry

parametry s defaultním nastavením

Řešení (a) i (b) v jednom:

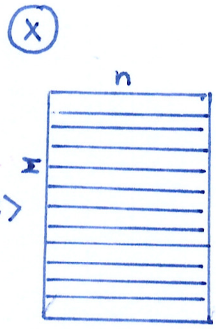
```
KLWald <- function(mu0, main, n, M=500..., mu1=..., mu2=mu1, sigma1=...,
          sigma2=sigma1, p=0.9, alpha = 0.05) {
```

defaultní nastavení - příprava pro (c) a (d)

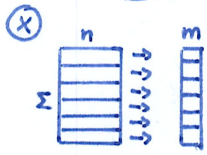
Generování dat

```
  M <- 500; n <- 5; p <- 0.9
  X <- matrix(NA, M, n)
  for(i in 1:M){
    bin <- rbinom(n,1,p)
    X[i,][bin == 1] <- rnorm(sum(bin==1), mu1, sigma1)
    X[i,][bin == 0] <- rnorm(sum(bin==0), mu2, sigma2)
  }
```

μ_1, σ_1, μ_2, σ_2



Příprava podkladů pro histogram



m <- apply(..., ..., ...) ... vektor průměrů (length(m) = 500)

zW <- $\frac{\bar{x}-\mu}{\sigma}\sqrt{n}$ ... vektor 500 realizací test. statistik z_W

↳ histogram

$Z_W \overset{H_0}{\sim} N(0,1)$

d <- hist(zW, plot=F)$dens ... výšky sloupců histogramu (využijeme později)

xfit <- seq(min(zW)-10, max(zW)+10, length=...)

yfit <- dnorm(...) ... hustota N(0,1) nad xfit

akt.hl.vyzn <- $\frac{\sum_{i=1}^{M} I(|zW| > u_{1-\alpha/2})}{M} = \hat{\alpha}$

sum() abs(zW) > qnorm()

počet IS, které obsahují $\mu = 20$ je stejný jako počet zW, které jsou v absolutní hodnotě větší než $u_{1-\alpha/2}$. (Vyplývá z propojení testu pomocí IS a testu pomocí krit. oboru.)

akt.p.pokryti <- $1-\hat{\alpha}$

Histogram

automaticky ohlídáme, aby se do grafu vešel histogram i ∩

```
  hist(zW, prob=T, ylim = c(0, max(yfit, d)), xlim= ..., density= ...,
       xlab=' ', ylab= ..., main= ..., las= ..., col=..., ...) ... histogram
  box(...)
  mtext(expression(z_W), ...) ... zW popisek pod histogramem
  mtext(bquote(paste(1-alpha == .(1-alpha), '; ',
               1-widehat(alpha) == .(akt.p.pokryti))), ...)
  mtext(main, ...) ... X~N(20,100); n=5
  lines(xfit, yfit, ...) ... ∩
}
```

```
KLWald(mu0=20, main='X~N(20,100); n=5', n=5)
```

! Nezapomeňte u každé situace uvést závěr, zda je test konservativní nebo liberální. =)

(a) Data pochází z $N(146, 10^2)$, ale my testujeme $H_0: \mu = 150$ oproti $H_1: \mu \neq 150$.
(Analogie reálné situace, kde skutečnou hodnotu μ neznáme (může být klidně 146), a přesto nějakou H_0 (např. $H_0: \mu = 150$) testujeme.)

(*) Za platnosti H_0: $Z_W = \dfrac{\bar{X} - \mu_0}{\sigma}\sqrt{n} \overset{H_0}{\sim} N(0,1)$

$H_0: \mu = \mu_0$ (150)

test. stat. je stejná   rozdělení test. stat. se liší

Pokud H_0 neplatí: $Z_W = Z_{W,\lambda} = \dfrac{\bar{X} - \mu_0}{\sigma}\sqrt{n} \sim N(\lambda, 1)$ … necentrální normální rozdělení s **parametrem necentrality** λ.

Zdůvodnění:

Pokud H_0 neplatí: $Z_{W,\lambda} = Z_W + \lambda = \underbrace{\dfrac{\bar{X} - \mu}{\sigma}\sqrt{n}}_{\text{Obecný tvar } Z_W \text{ (statistika)}} + \underbrace{\dfrac{\mu - \mu_0}{\sigma}\sqrt{n}}_{\text{konstanta } \lambda} = \dfrac{\bar{X} - \mu + \mu - \mu_0}{\sigma}\sqrt{n} = \dfrac{\bar{X} - \mu_0}{\sigma}\sqrt{n} = Z_W$

statistika, číslo, číslo, číslo, číslo, číslo, číslo, číslo

(Pokud $\mu = \mu_0$, dosazujeme za μ hodnotu μ_0 a získáváme $\frac{\bar{X} - \mu_0}{\sigma}\sqrt{n}$. Viz (*).
Zde ale nepředpokládáme, že $\mu = \mu_0$.)

$N(0,1) + \lambda \rightarrow N(\lambda, 1)$

$H_0: \mu = \mu_0$ $H_1: \mu \neq \mu_0$

$H_0: \mu = 150$ $H_1: \mu \neq 150$

(a) $\mu = 146$, $\sigma^2 = 10^2$, $n = 50$

rozdeleni.zW ← function(mu, mu0= … (150), sigma = … (10), sigma2 = sigma, M= …, n= …, p=0.9, main='', pozice='topleft') {

X ← matrix(NA, M, n) // M ← 2000, n ← 50

for (…) {
… }

⇒ (Σ, n)

Viz předchozí příklad ((…) rnorm(sum(bin==1), mu, sigma)
(…) rnorm(sum(bin==0), mu, sigma2))

Opět si kód nachystáme i pro směs, která nás čeká v (c) a (d).

$\bar{x}$ … m ← apply(…) vektor průměrů o délce 2000

z_W … $\dfrac{\bar{x} - \mu_0}{\sigma}\sqrt{n}$ vektor test. statistik (2000)

λ … $\dfrac{\mu - \mu_0}{\sigma}\sqrt{n}$ … posunutí (par. necentrality λ) … číslo (délka 1)

xfit ← seq(…) min(zW)−6 až max(zW)+6

yfit ← dnorm(…) $N(0,1)$

lfit ← dnorm(…) $N(\lambda, 1)$

#U směsi křivka $N(\lambda, 1)$ nesuperponuje histogram Zw dokonale. Zamyslete se, proč, a co z toho pro nás do praxe vyplývá.

hist(zW, breaks=20, ylim=c(0, max(yfit)+0.05),
  xlim=c(min(zW)-5, max(zW)+5), prob=T, …)

relativní škála osy y!!!
bez ní nebudou křivky hustot viditelné!

box(…) □

mtext(expression(…), …) ← Z_W

mtext(bquote(…), …) ← $n=50$, $\mu_0=150$ (automaticky)

mtext(main, …) ← $X \sim N(150, 100)$ (**main** můžeme zadat jako jeden ze vstupů funkce (nebo zautomatizovat. Jak chcete =)).

lines(xfit, yfit, …)

lines(xfit, lfit, …)

odstraní rámeček okolo legendy   (nepovinné; zmenší písmo v legendě)

legend(pozice, fill= …, bty='n', cex=0.9, legend= …, …)

}

(d) rozdeleni.zw(mu=155, sigma2=30, main='X~ 0.9 N(155, 100) + 0.1 N(155, 900)', pozice='topright')

4.3

a) $H_{01}: \mu = \mu_0$ $\quad H_{11}: \mu \neq \mu_0$ $\quad W = (-\infty; u_{\alpha/2}\rangle \cup \langle u_{1-\alpha/2}; \infty)$

Síla: $\beta^*_{11}(\mu) = 1-\beta_{11}(\mu) = Pr(H_0 \text{ zamítáme} \mid H_0 \text{ neplatí})$ — síla je tedy pravděpodobnost

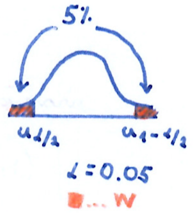
na tohle se zaměříme — Toto neovlivníme

H_0 zamítáme, pokud $\frac{\bar{X}-\mu_0}{\sigma}\sqrt{n} \in W$. Tedy v tomto případě zapíšeme sílu $\beta^*_{11}(\mu)$ jako:

$$\beta^*_{11}(\mu) = 1-\beta_{11}(\mu) = Pr\Big(\underbrace{\frac{\bar{X}-\mu_0}{\sigma}\sqrt{n} < u_{\alpha/2}}_{(-\infty;\, u_{\alpha/2}\rangle}\Big) + Pr\Big(\underbrace{\frac{\bar{X}-\mu_0}{\sigma}\sqrt{n} > u_{1-\alpha/2}}_{\langle u_{1-\alpha/2};\, \infty)}\Big) = (\dots) \text{ viz teorie k příkladu}$$

$$= \Phi\Big(u_{\alpha/2} - \frac{\mu-\mu_0}{\sigma}\sqrt{n}\Big) + \Phi\Big(u_{\alpha/2} + \frac{\mu-\mu_0}{\sigma}\sqrt{n}\Big)$$

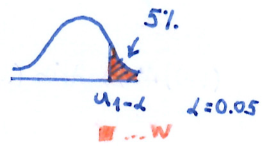
distr. fce $N(0,1)$: Φ … pnorm(), u … qnorm()



b) $H_{02}: \mu \le \mu_0$ $\quad H_{12}: \mu > \mu_0$ (pravostranná) $\quad W = \langle u_{1-\alpha}; \infty)$

$$\beta^*_{12}(\mu) = 1-\beta_{12}(\mu) = Pr\Big(\underbrace{\frac{\bar{X}-\mu_0}{\sigma}\sqrt{n} > u_{1-\alpha}}_{\langle u_{1-\alpha};\, \infty)}\Big) = (\dots) = \Phi\Big(u_\alpha + \frac{\mu-\mu_0}{\sigma}\sqrt{n}\Big)$$

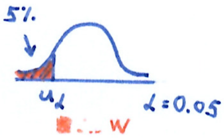
distr. fce $N(0,1)$: Φ … pnorm(), u … qnorm()



c) $H_{03}: \mu \ge \mu_0$ $\quad H_{13}: \mu < \mu_0$ (levostranná) $\quad W = (-\infty; u_\alpha\rangle$

$$\beta^*_{13}(\mu) = 1-\beta_{13}(\mu) = Pr\Big(\frac{\bar{X}-\mu_0}{\sigma}\sqrt{n} < u_\alpha\Big) = (\dots) = 1-\Phi\Big(u_{1-\alpha} + \frac{\mu-\mu_0}{\sigma}\sqrt{n}\Big)$$

distr. fce $N(0,1)$: Φ … pnorm(), u … qnorm



- 1) Vytvoříme funkci sila.exact() počítající pro zadané $\mu, \mu_0, \sigma, n, \alpha$ sílu pro libovolnou alternativu.

```
sila.exact <- function(mu, mu0, sigma, n, alpha=0.05, alternative='two.sided'){
   if(alternative == 'two.sided') { sila <- Φ(u_{α/2} - (μ-μ0)/σ √n) + Φ(u_{α/2} + (μ-μ0)/σ √n) }
   if(            == 'greater')   {sila <- Φ(u_α + (μ-μ0)/σ √n)}
   if(            == 'less')      {sila <- 1-Φ(u_{1-α} + (μ-μ0)/σ √n
                      <- 1-pnorm(qnorm(1-alpha) + (mu-mu0)/sigma *
                                * sqrt(n)) }
   return(sila)
}
```

2) Vykreslíme graf pro (a) (oboustranná 'two.sided' alternativa)

```
barva <- c(...) ... vektor 4 barev (např. dodgerblue 4, 3 a 1 a darkturquoise)
mu <- seq(...) ... posl. od 132 do 168 (délka minimálně 500)
n <- c(10, 20, 50, 100)
sila <- sila.exact(mu=mu, mu0=..., sigma=..., n=n[1])
plot(mu, sila, type='n', ylim=c(0,1), xlab=μ, las=...,      # příprava prázdného grafu
     ylab=expression(paste('silofunkce ', 1-beta)), ...) ... prázdný graf s popisky

for(i in 1:4){                          # cyklus: vykreslí 4 silofunkce
  sila <- sila.exact(..., n=n[i])
  lines(mu, sila, col=barva[i])
}
legend(..., col=barva, bty=..., lty=1, lwd=2, legend=paste('n =', n))
  # pozice ('right'); vektor barev; lty=1, lwd=2: v legendě budou plné čáry o tloušťce 2
  # efektivní způsob, jak vypsat n=10, n=20, n=50, n=100
abline(h=0.05, col=..., lty=...) ... vodorovná přerušovaná šedá čára
  # 0.05 = alpha
```

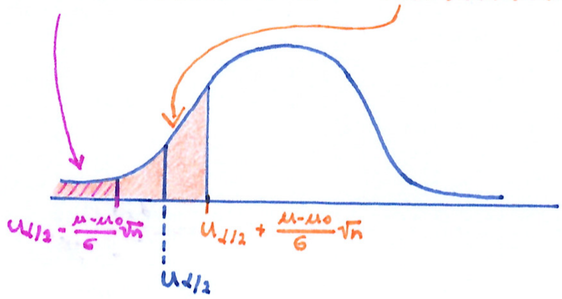
3) Zopakujeme (2) pro (b) (pravostranná 'greater' alternativa) a (c) (levostranná 'less' alternativa).

4.4

$H_{01}: \mu = \mu_0$ $\quad H_{11}: \mu \neq \mu_0$ $\quad W = (-\infty; u_{\alpha/2}\rangle \cup \langle u_{1-\alpha/2}; \infty)$

Exaktní síla: $\beta^*_{11}(\mu) = 1 - \beta_{11}(\mu) = P_r(H_0 \text{ zamítáme} \mid H_0 \text{ neplatí})$

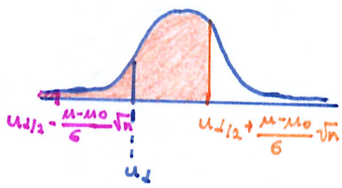
$$= \Phi\left(u_{\alpha/2} - \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right) + \Phi\left(u_{\alpha/2} + \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right)$$ ... viz předchozí příklad



a) $\mu - \mu_0 > 0$ $\quad (\mu > \mu_0)$ $\longrightarrow$ $\frac{\mu-\mu_0}{\sigma}\sqrt{n} = c$ ... kladné

$\uparrow(\mu-\mu_0)$: $\underbrace{\Phi\left(u_{\alpha/2} - \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right)}_{\text{klesá, až je zanedbatelně malé}} + \underbrace{\Phi\left(u_{\alpha/2} + \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right)}_{\text{roste}} \approx \Phi\left(u_{\alpha/2} + \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right)$

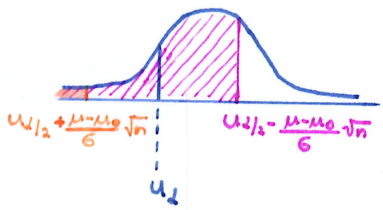
s rostoucím $\mu-\mu_0$:



b) $\mu - \mu_0 < 0$ $\quad (\mu < \mu_0)$ $\longrightarrow$ $\frac{\mu-\mu_0}{\sigma}\sqrt{n} = c$ ... záporné

$\downarrow(\mu-\mu_0)$: $\underbrace{\Phi\left(u_{\alpha/2} - \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right)}_{\text{roste}} + \underbrace{\Phi\left(u_{\alpha/2} + \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right)}_{\text{klesá, až je zanedbatelně malé}} \approx \Phi\left(u_{\alpha/2} - \frac{\mu-\mu_0}{\sigma}\sqrt{n}\right)$

s klesajícím $\mu-\mu_0$:



Rozdíl mezi (a) a (b) je pouze v tom, zda $\mu - \mu_0 > 0$ nebo $\mu - \mu_0 < 0$.
Sjednoťme tedy obě situace pomocí $|\mu - \mu_0|$. Potom $|\mu - \mu_0| > 0$.

$\Rightarrow$ **aproximatická síla:** $\tilde{\beta}^*_{11}(\mu) = \Phi\left(u_{\alpha/2} + \frac{|\mu-\mu_0|}{\sigma}\sqrt{n}\right)$

---

1. Funkci pro výpočet exaktní síly máme z předchozího příkladu ... `sila.exact()`
2. Vytvoříme funkci `sila.aprox()` počítající pro zvolené $\mu, \mu_0, \sigma, n, \alpha$ aproximatickou sílu.

```
sila.aprox <- function(mu, mu0, sigma, n, alpha = 0.05){
    sila <- Φ(u_{α/2} + |μ-μ0|/σ √n)
            pnorm()  qnorm()  abs()
    return(sila)
}
```

3. Vykreslíme graf (globální pohled):

```
mu ← seq( )                              posloupnost od 145 do 155 s délkou např. 1024
sila.ex ← sila.exact (mu=mu, mu0=150,...)  vektor hodnot ex. síly (1024)
sila.ap ← sila.aprox (mu=mu, mu0=150,...)  vektor hodnot aprox. síly (1024)

plot(mu, sila.ex, ylim=c(0,1),...)   graf křivky exaktní síly (modrá plná čára)
lines(...)    křivka aprox. síly - červenou barvou, přerušovanou čarou (lty=5)
legend(...)   legenda
```

4. Vykreslíme graf (lokální pohled):

```
plot(mu, sila.ex, xlim=c(148, 152), ylim=c(-0.1, 0.4), asp=F,...)
lines(-||-)
legend(-||-)
```

4.5 exaktní síla ($\beta^*_{1i}(\mu)$) ... přesná síla, kterou počítáme podle vzorců exaktní síly (definovaná pro obou-, levo- i pravostr. alt.)

aproximatická síla ($\tilde{\beta}^*_{1i}(\mu)$) ... přibližná síla, kterou počítáme pomocí vzorce aprox. síly (definovaná pouze pro oboustr. alt.)

empirická síla ... skutečná síla vypočítaná z reálných nebo nasimulovaných dat (můžeme mluvit i o odhadu exaktní síly z dat, jehož kvalita závisí na kvalitě dat).

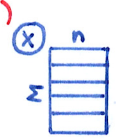
$\hat{\beta}^*_{1i}(\mu)$ — Spočitatelná pro obou-, levo- i pravostrannou alternativu.

$$\hat{\beta}^*_{1i}(\mu) = \frac{\sum_{i=1}^{M} I(H_0 \text{ zamítáme})}{M} \qquad \left(\widehat{SD}[\hat{\beta}^*_{1i}(\mu)] = \sqrt{\frac{\underbrace{\hat{\beta}^*_{1i}(\mu)}_{\text{emp. síla}}\ \underbrace{\hat{\beta}_{1i}(\mu)}_{\text{1-emp. síla (emp. PCHDD)}}}{M}} \right)$$

---

1. Vytvoříme funkci na výpočet empirické síly sila.empir():

```
sila.empir <- function (mu, mu0, sigma, sigma2= sigma, n=100, M=... , alpha=...,
                        p=..., alternative = 'two.sided') {
  X <- matrix (NA, M, n)
  for (i in 1:M){
     ...}
```



M náhodných výběrů o rozsahu n

(viz 4.1, 4.2)

Cyklus naprogramujeme tak, aby generoval i smíšená data (využijeme v části (b), kde počítáme sílu pro směs).

$\bar{x}$ ... `m <- apply(...)` vektor M=1000 průměrů

z_W ... `zW <-` $\frac{\bar{x}-\mu_0}{\sigma}\sqrt{n}$ vektor test. statistik zW (1000)

p-hodnoty ... `p <- cbind (pnorm(zW), 1-pnorm(zW))`

`p <- 2*apply (p, 1, min)` ... vektor M=1000 p-hodnot testů o μ když σ^2 známe

výpočet empirické síly ... (# H_0 zamítáme)/M

```
  if(alternative == 'two.sided') { sila <- sum( p < alpha) / M }
  if(     -||-       'greater'  ) {   -||-   (1-pnorm(zW) < α) -||- }
  if(     -||-       'less'     ) {   -||-   (pnorm(zW) < α) -||- }
  return (sila)
}
```

2. Pomocí funkce sila.empir() vypočítáme hodnoty empirické síly pro $\mu_0 = 150$, $\sigma^2 = 10^2$ postupně pro každé μ $\mu \in (144, 144.5, \dots, 155.5, 156)$, hodnoty SD a dolní a horní hranici chybové úsečky $\hat{\beta}^*_{11}(\mu) \pm \widehat{SD}[\hat{\beta}^*_{11}(\mu)]$

alpha <- 0.05; n <- 100; M <- 1000; mu0 <- 150; sigma <- 10

`mu <- seq( )` ... 144-156 se vzdáleností bodů 0.5 (délka vektoru = 25)

`sila.emp <- NULL` prázdný vektor

(b) sigma2 = 30

```
for (i in 1:length(mu)){
   sila.emp[i] <- sila.empir (mu=mu[i], mu0=..., sigma=..., sigma2= sigma, ...)
}
```

... vektor $\hat{\beta}^*_{11}(\mu)$ o délce 25

SD <- $\sqrt{\frac{\hat{\beta}^*_{11}(\mu)\,(1-\hat{\beta}^*_{11}(\mu))}{M}}$ ... sila.emp   vektor SD (25)

dh <- $\hat{\beta}^*_{11}(\mu)$ - SD   vektor dolní hranice chybové úsečky (25)

hh <- $\hat{\beta}^*_{11}(\mu)$ + SD   -||- horní -||- (25)

3. Pomocí fce sila.exact() vypočítáme hodnoty exaktní síly pro hustou síť $\mu \in \langle 144; 156 \rangle$

zde si to můžeme dovolit, protože výpočet exaktní síly není výpočetně tolik náročný jako výpočet empirické síly

```
mu1 <- seq()
sila.ex <- sila.exact (mu=mu1, mu0=..., sigma=..., n=..., ...)
```

4. Vykreslíme graf (a)

I. `plot(mu, sila.emp, ylim=c(0,1), ...)` křivka empirické síly (plná, modrá)

II. `abline (v= mu0, ...)` svislá referenční šedá přerušovaná čára

III. `abline (h = alpha, ...)` vodorovná referenční šedá přeruš. čára

IV. `lines (mu1, sila.ex, ...)` křivka exaktní síly (tmavě modrá, přerušovaná)

V. `arrows (mu, dh, mu, hh, code=3, angle=90, length= 0.03)`

chybové úsečky I — oboustranná šipka ↕ — úhel kudýlek šipky 90°... I — délka kudýlek

VI. `points(mu, sila.emp., cex=0.9, ...)` zmenší velikost bodů

VII. `legend(...)` ... legenda

5. Vytvoříme tabulku

```
mu.vyber <- c(146, ..., 154)
sila.ex.vyber <- sila.exact( mu= mu.vyber, ... )
sila.emp.vyber <- NULL
for (i in 1:length(mu.vyber)){
   sila.emp.vyber [i] <- sila.empir(mu = mu.vyber[i], ...)}
tab <- data.frame (rbind(sila.ex.vyber, sila.emp.vyber), row.names= ... )
names(tab) <- mu.vyber
```

mu.vyber ... definujeme posloupnost vybraných čísel (n=12)

sila.ex.vyber ... vektor exaktních sil (n=12)

sila.emp.vyber ... příprava prázdného vektoru

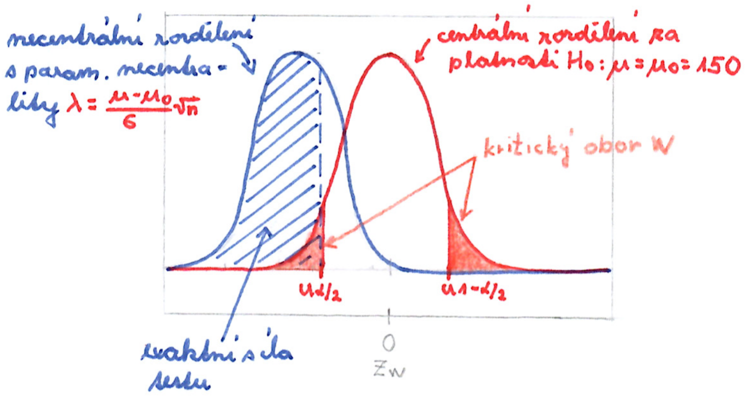
for ... vektor hodnot emp. síly (12)

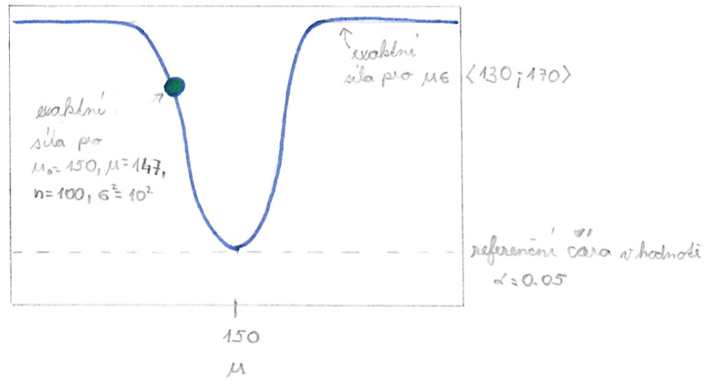
tab ... tabulka exaktních a emp. sil

names(tab) ... změna názvů sloupců

4.6

V tomto příkladu využijeme a propojíme poznatky, které jsme získali v příkladech 4.2, 4.3 a 4.5. Výstupem příkladu je animace, ve které se s měnící se hodnotou μ mění rozdělení test. statistiky Z_W a hodnota svaktní síly. Rozeberme si nejprve, co vidíme na jedné dvojici obrázků.





1. Vytvoříme funkci sila.animace(), která vykreslí 1 dvojici grafů:

sila.animace ← function(mu, mu0, sigma, n, alpha=0.05) {

mu0 ← 150; mu ← 147; sigma ← 10; n ← 100; alpha ← 0.05

a) 1. graf

$\lambda \ldots \frac{\mu-\mu_0}{\sigma}\sqrt{n}$

x ← seq(..) posl. od -10 do 10 o délce min 500

y ← dnorm(...) hustota $N(0,1)$ nad posl. x

l ← dnorm(...) hustota $N(\lambda,1)$ nad posl. x

plot(x, y, ylim=c(-0.05; 0.4), xlab='', type='n', ...) prázdný graf

mtext(expression(...), ...) popisek Z_W

mtext(bquote(...), ...) popisek n=100, μ_0=150, μ=147... randomalizovaný

q1 ← qnorm(...)... kvantil $u_{\alpha/2}$

q2 ← qnorm(...)... kvantil $u_{1-\alpha/2}$

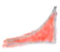
// Funkce polygon(): barva polygonu

polygon(x, y, col=..., density=..)

vektor x-ových souřadnic; vektor y-ových souřadnic; hustota šrafování

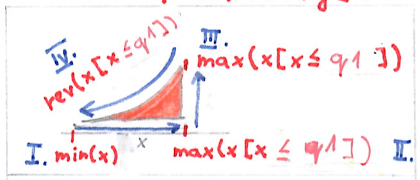
Obdélník:

polygon(x=c(1,2,2,1), y=c(6,6,8,8), col='green', density=20)

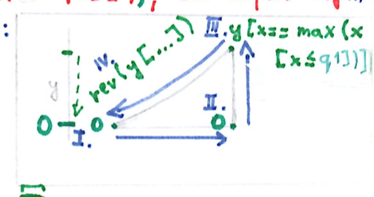


polygon(x=c(min(x), max(x[x<=q1]), max(x[x<=q1]), rev(x[x<=q1])),
y=c(0 , 0 , y[x== max(x[x<=q1])], rev(y[x %in% x[x<=q1]])), col=..., density=...)
col=..., density=...)

x:

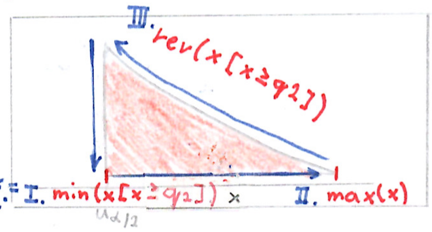


y:

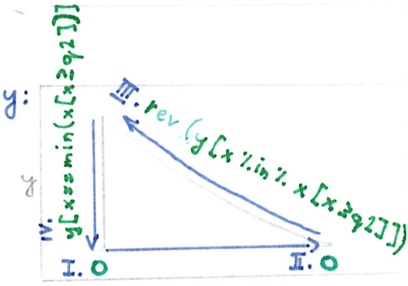


polygon(x=c(...), y=c(...), col=..., density=...)

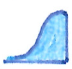
x:



y:

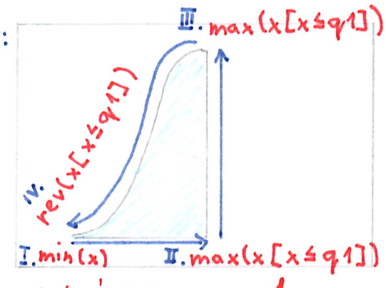


lines(x, y, col=...)



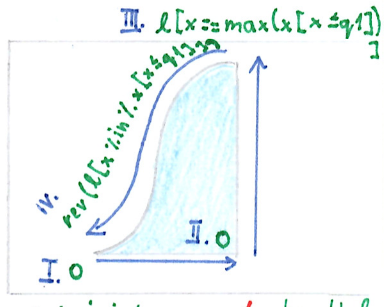
polygon(x=c(...), y=c(...), col=..., density=...)

x:



stejné jako u ◿

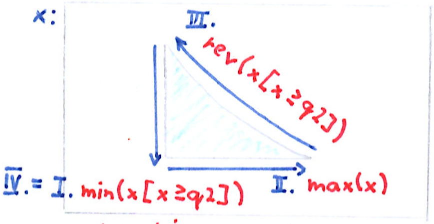
y:



stejné jako u ◿ akorát l je místo y.

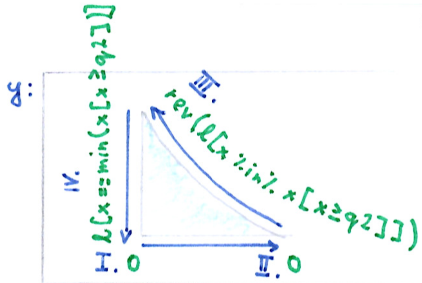
polygon(x=c(...), y=c(...), col=...,
density=...)

x:



stejné jako u △

y:



stejné jako u △ akorát l je místo y

lines(x, l, col=...) legenda vodorovná
legend(..., horiz=T, lwd=..., col=..., bty=..., ...)

b) 2. graf

mu1 ← seq(...) posl. od 130 do 170 o min délce 500
sila.ex ← sila.exact(mu=mu1, mu0=150, sigma=..., n=..., ...) vektor exaktní síly (500)
plot(mu1, sila.ex, ylim=c(-0.1, 1), ...) graf s křivkou exaktní síly
mtext(expression(...), ...) popisek μ
sila.akt ← sila.exact(mu=mu, mu0=150, ...)... hodnota exaktní síly pro jedno konkrétní μ (např. μ=147)
points(mu, sila.akt, ...) vykreslení bodu s exaktní sílou pro vybrané μ (μ=147).
legend(..., lwd=c(2, NA), pch=c(NA, 19), horiz=T, ...) legenda
abline(...) vodorovná referenční čára v bodě α=0.05.

}

2. Vytvoříme animaci

mu ← seq( ) zadaná posl. 140, ..., 146, 146.5, ..., 153.5, 154, 155, ... 160
oopts ←
ani.record ←
saveLatex(for(i in 1:length(mu)){
sila.animace(mu=mu[i], mu0=150, sigma=10, n=100)
})

1. $H_{01}: \mu = \mu_0$ $\quad H_{11}: \mu \neq \mu_0$ ... $n_{min} \geq \dfrac{(u_{\beta^*_{11}(\mu)} - u_{\alpha/2})^2}{|\mu - \mu_0|^2} \sigma^2$

2. $H_{02}: \mu \leq \mu_0$ $\quad H_{12}: \mu > \mu_0$ ... $n_{min} \geq \dfrac{(u_{\beta^*_{12}(\mu)} - u_{\alpha})^2}{(\mu - \mu_0)} \sigma^2$

3. $H_{03}: \mu \geq \mu_0$ $\quad H_{13}: \mu < \mu_0$ ... $n_{min} \geq \dfrac{(u_{\beta^*_{13}(\mu)} - u_{\alpha})^2}{(\mu_0 - \mu)^2} \sigma^2$

qnorm() qnorm()

---

1. Vytvoříme funkci min.n(), která pro zvolené $\mu, \mu_0, \sigma, \alpha$ a β^* (resp. β) vypočítá minimální potřebný rozsah náh. výběru podle zvolené alternativy.

min.n ← function(mu, mu0, sigma = 10, alpha = ..., sila = ..., alternative = ...){

if(alternative == 'two.sided') { n ← $\dfrac{(u_{\beta^*_{11}(\mu)} - u_{\alpha/2})^2}{|\mu - \mu_0|^2} \sigma^2$ }

if(== 'greater') { n ← $\dfrac{(u_{\beta^*_{12}(\mu)} - u_{\alpha})^2}{(\mu - \mu_0)^2} \sigma^2$ }

if(== 'less') { n ← $\dfrac{(u_{\beta^*_{13}(\mu)} - u_{\alpha})^2}{(\mu_0 - \mu)^2} \sigma^2$

← ((qnorm(sila) - qnorm(alpha))^2 / (mu0 - mu)^2*sigma^2 }

return(n)
}

2. Vykreslíme graf ((a) pro oboustrannou alternativu)

sila ← 0.8
mu ← seq() posl. od 145 do 155 po kroku 0.5 (délka 21)
n11 ← min.n(mu=mu, mu0=150, ...) vektor minimálních rozsahů (délka = 21)
plot(mu, n11, pch=21, col=..., bg=..., xlim=c(145, 155), xlab='',
    ylab=expression(n[min]), ...)
mtext(expression(...), ...) popisek μ
mtext(bquote(paste(..., beta, '* = ', .(sila))), ...) automatický popisek $\alpha = 0.05$ $\beta^* = 0.8$

3. Vykreslíme graf ((b) pro pravostr. alt., resp. (c) pro levostr. alt.)

mu ← seq(...) 150.25 – 154 (by = 0.25) resp. 146 – 149.75 (by = 0.25)
n12 ← min.n(...) c(146, 152)
plot(mu, n12, xlim = c(148, 154), ...)
mtext(...) μ
mtext(...) $\alpha = 0.05$, $\beta^* = 0.8$
segments(145 150, 0, 150 155, 0, lty = 2, col = ...)

4. Spočítáme tabulku

mu.vyber ← seq(...) posloupnost vybraných μ (viz. zadání) (12)
n11.v ← min.n(mu.vyber, mu0=150, ...) min. rozsahy pro vybraná μ a oboustr. alt. (12)
n12.v ← min.n(...) min. rozsahy pro vybraná μ a pravostr. alt.
n13.v ← min.n(...) –‖– –‖– levostr. alt.
n12.v ← c(rep(NA,6), n12.v[7:12]) rozsahy vypočítané pro nesmyslné kombinace μ a alternativy nahradíme NA.
n13.v ← c(n13.v[1:6], rep(NA,6)) –‖– –‖–
tab ← data.frame(...) tabulka min. rozsahů pro všechny 3 alternativy.