Rešitve - lastnosti testov

Nataša Kejžar

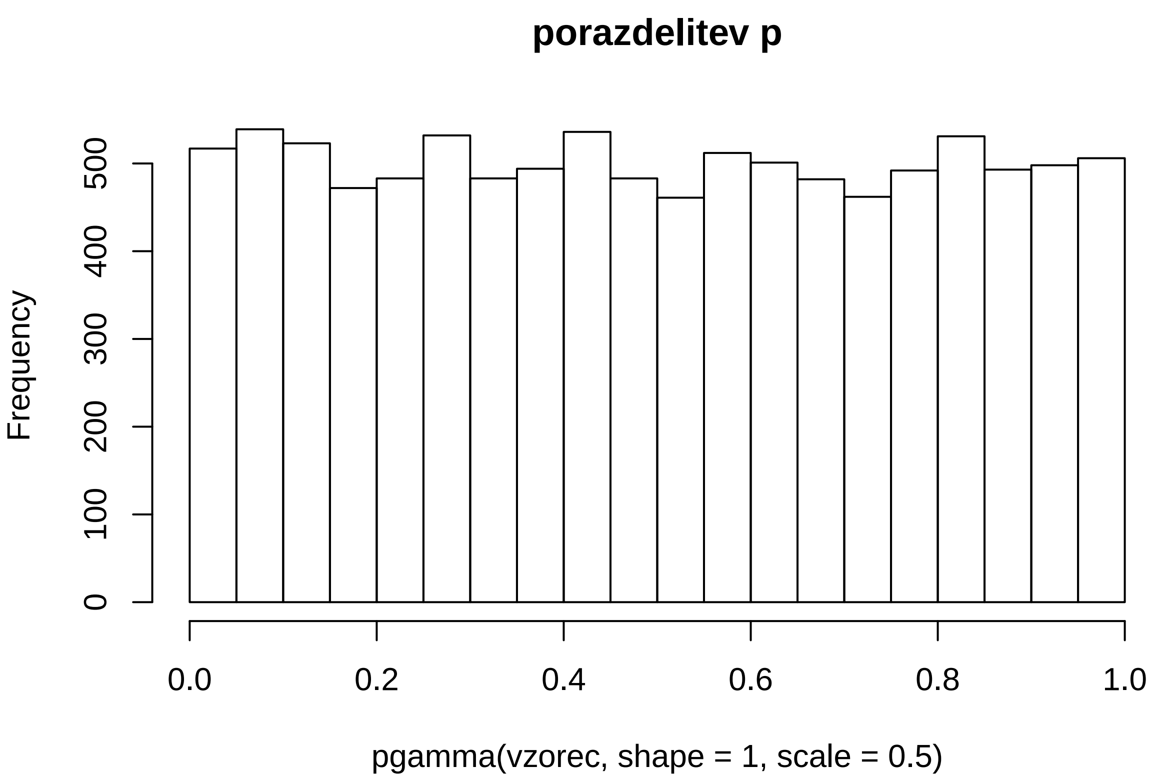
Naloga 1

a. Vemo torej $F_X(X) = Z$, lahko zapišemo tudi

$$F_Z(z) = P(Z \leq z) = P(F_X(X) \leq z) = P(X \leq F_X^{-1}(z)) = F_X(F_X^{-1}(z)) = z$$

Tretji korak smo lahko naredili, ker velja, da je F_X^{-1} definiran za vse vrednosti X in F strogo naraščajoča funkcija.

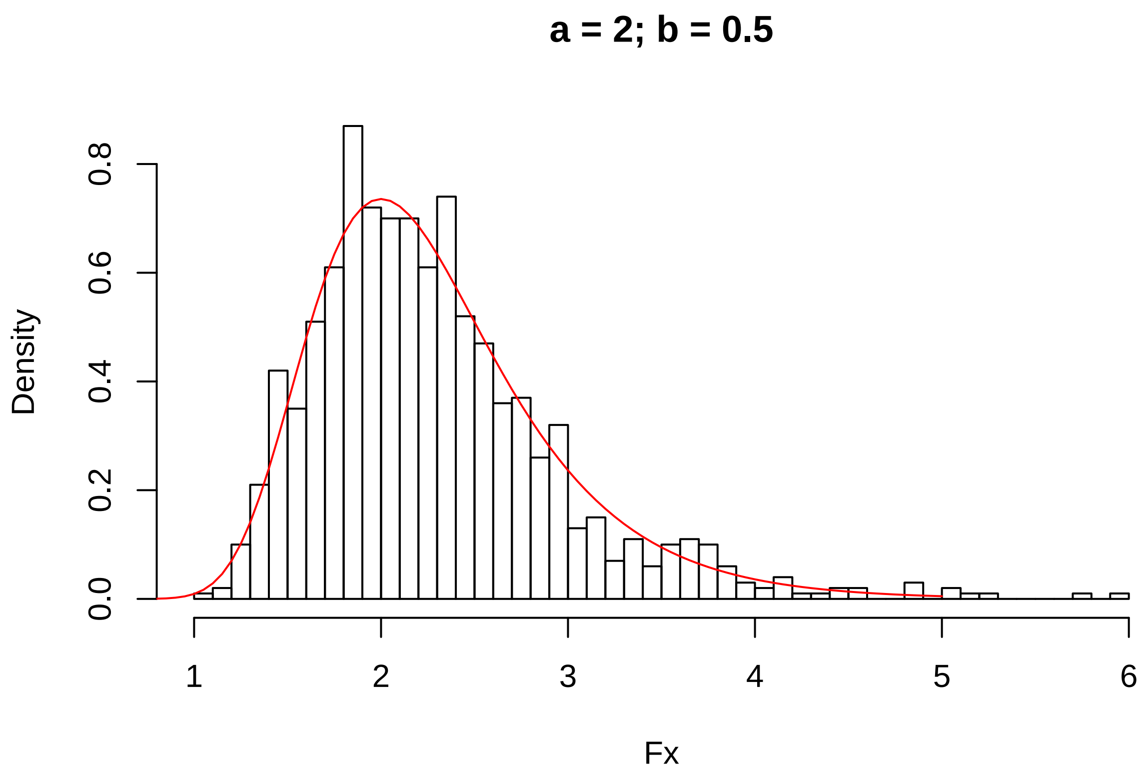
```
vzorec = rgamma(10000,shape=1,scale=0.5)
hist(pgamma(vzorec,shape=1,scale=0.5),
     main="porazdelitev p")
```



Naloga 2 - Gumbel

```
x = runif(1000, min=0, max=1)
a=2
b=0.5
Fx = a-b*log(log(1/x))
hist(Fx, main = paste("a = ", a, "; b = ", b, sep = ""),breaks=50,freq=FALSE)
# preverimo s funkcijo iz R
```

```r
# install.packages("ordinal")
library(ordinal)
curve(dgumbel(x,location=2,scale=0.5),from=0,to=5,add=TRUE,col="red")
```



Naloga 3 - Exp

Najprej izračunamo $F(x)$ iz gostote:

$$
\begin{aligned}
F(x) &= \int_0^x \lambda e^{-\lambda u} du \\
&= \lambda \cdot \frac{-1}{\lambda} \cdot e^{-\lambda u}\Big|_0^x = -e^{-\lambda x} + e^0 = 1 - e^{-\lambda x}
\end{aligned}
$$

Sedaj zamenjamo $F(x) = y$ in x:

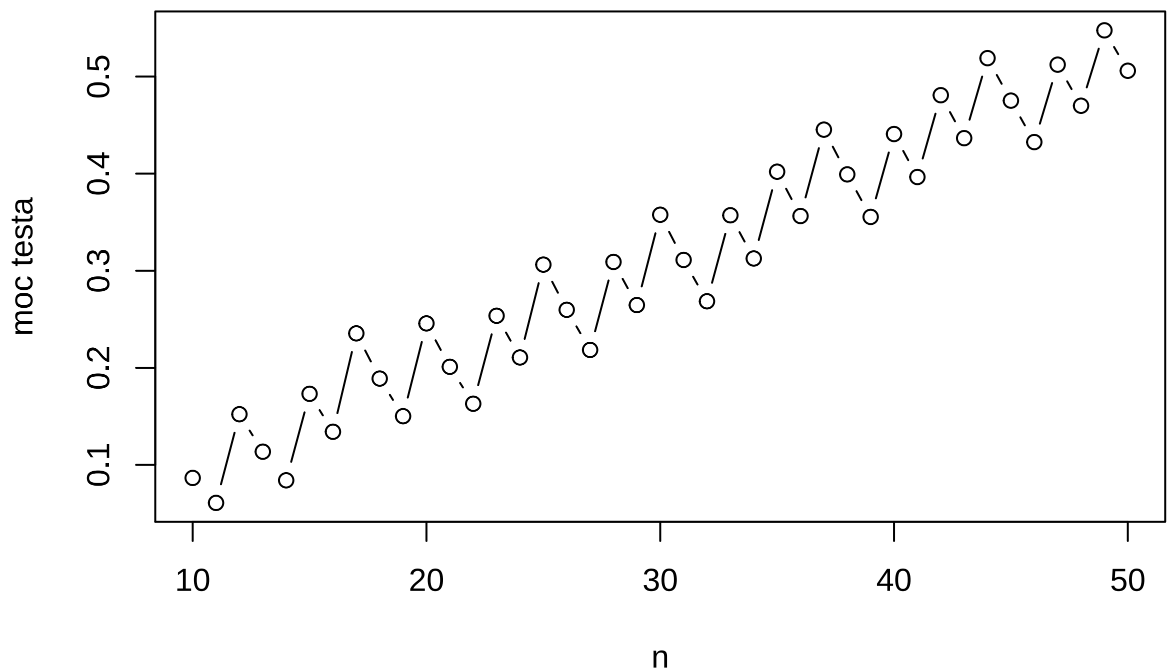
$$
\begin{aligned}
x &= 1 - e^{-\lambda y} \\
1 - x &= e^{-\lambda y} \\
\log(1 - x) &= -\lambda y \\
y &= -\frac{\log(1 - x)}{\lambda} \\
F^{-1}(x) &= -\frac{\log(1 - x)}{\lambda}
\end{aligned}
$$

Naloga 4 - kadilci

```r
# H_0: pi = 0.5
# H_A: pi != 0.5 (0.65)
# testna statistika: sum(X_i) binomska(n, pi)

pi=0.5

# moc testa
moc = numeric(length(10:50))
for(n in 10:50){
  meja1 = qbinom(0.025,size=n,prob=pi)
  # spodnjo mejo dobimo za 1 previsoko
  meja2 = qbinom(0.975,size=n,prob=pi)
  moc[n-9] = pbinom(meja1-1,size=n,prob=0.65,lower.tail=TRUE) +
    pbinom(meja2,size=n,prob=0.65,lower.tail=FALSE)
}
plot(10:50,moc,type="b",xlab="n",ylab="moc testa")
```



Naloga 5 - kadilci II

```r
# H_0: piZ = piM
# var = p*(1-p)*(1/n1 + 1/n2)
se = sqrt((23/40*(1-23/40))*(1/22+1/18))
testna= (0.5 - 12/18)/se
pnorm(testna,mean=0,sd=1)*2 # dvostranska hipoteza
```

```
## [1] 0.2887759
```

```r
### moc testa
n=40
meja1 = qnorm(0.975)
meja2 = qnorm(0.025)
# simuliraj 1 vrednost p
testnaS <- function(p1,p2,n1,n2){
```

```
  p = (p1*n1 + p2*n2)/(n1+n2)
  se = sqrt((p*(1-p))*(1/n1+1/n2))
  testna = (p1-p2)/se
  pvrednost = pnorm(testna,lower.tail=(testna<0))
  return(pvrednost*2) # dvostranska
}
testnaS(0.5,12/18,22,18)
```

```
## [1] 0.2887759
```

```
ponovi=1000
pvrednosti = numeric(ponovi)
n1=22
n2=18
for(i in 1:ponovi){
  # generiraj vzorec iz H_A
  vzorec = c(rbinom(1,size=n1,prob=0.5),rbinom(1,size=n2,prob=0.55))
  # izracunaj p (ali zavrnem) pod H_0
  pvrednosti[i] = testnaS(vzorec[1]/n1,vzorec[2]/n2,n1,n2)
}
sum(pvrednosti < 0.05)/ponovi
```

```
## [1] 0.078
```

```
# 6.6 %

# druga varianta teoreticno

p1=0.5
p2=0.55
# SE izracunamo kot vsoto dveh
p = (p1*n1 + p2*n2)/(n1+n2)
se = sqrt((p*(1-p))*(1/n1+1/n2))

pnorm(meja1,0.05/se,lower.tail=FALSE) +
  pnorm(meja2,0.05/se,lower.tail=TRUE) # 6.28%
```

```
## [1] 0.06143987
```

Naloga 6 - Bush-Al Gore

Spremenljivki X_A (število Američanov, ki v vzorcu podpirajo Al Gore-a) in X_B (število Američanov, ki v vzorcu podpirajo Busha) je med seboj odvisno: $X_A + X_B \geq 2207$. Ker pri statističnih testih težimo po neodvisnih spremenljivkah (npr. za iid opazovanja, katerih porazdelitev je dokaj lepa, bo za dovolj velik vzorec vsota porazdeljena po normalni porazdelitvi). Spremenljivki X_A in X_B lahko zapišemo kot vsoti n Bernulijevo porazdeljenih spremenljivk (obe sta porazdeljeni po binomski porazdelitvi $Bin(n, p_{A\text{ali}B})$), saj vemo, da velja za vsako opazovanje:

$$X_{Ai} = \begin{cases} 1, & \text{človek je za Al Gore-a} \\ 0, & \text{sicer (za Busha ali neodločen)} \end{cases}$$

in

$$X_{Bi} = \begin{cases} 1, & \text{človek je za Busha} \\ 0, & \text{sicer (za Al Gore-a ali neodločen)} \end{cases}$$

Vemo, da sta opazovanji spremenljivk X_{Ai} in X_{Bi}) odvisni in lahko zavzameta 3 le različne vrednosti: (1,0), (0,1) ali (0,0). Uvedemo lahko novo spremenljivko razliko $D_i = X_{Bi} - X_{Ai}; \quad i = 1, \ldots n$. D_i so potem iid. Za D_i vemo, da velja:

$$
\begin{aligned}
E(D_i) = E(X_{Bi} - X_{Ai}) = E(X_{Bi}) - E(X_{Ai}) = p_B - p_A \overset{H_0}{=} 0 \\
var(X_{Bi} - X_{Ai}) =?
\end{aligned}
$$

Izračunajmo vzorčno varianco:

$$
\begin{aligned}
s^2_{D_i} =& \frac{1}{n-1} \sum_{i=1}^{n} (D_i - \bar{D})^2 \\
=& \frac{1}{n-1} \left[\sum_{i=1}^{n} D_i^2 - 2\bar{D} \sum_{i=1}^{n} D_i + \sum_{i=1}^{n} \bar{D}^2 \right] \\
=& \frac{1}{n-1} \left[\sum_{i=1}^{n} D_i^2 - n\bar{D}^2 \right] \\
=& \frac{1}{n-1} \left[\sum_{i=1}^{n} (X_{Bi} - X_{Ai})^2 - n \left(\sum_{i=1}^{n} (X_{Bi} - X_{Ai}) \right)^2 \right] \\
=& \frac{1}{n-1} \left[\sum_{i=1}^{n} X_{Bi}^2 - 2 \sum_{i=1}^{n} X_{Bi} X_{Ai} + \sum_{i=1}^{n} X_{Ai}^2 - n \left(\sum_{i=1}^{n} X_{Bi} - \sum_{i=1}^{n} X_{Ai}) \right)^2 \right] \\
=& \frac{1}{n-1} \left[\sum_{i=1}^{n} X_{Bi} - 0 + \sum_{i=1}^{n} X_{Ai} - n \left(\sum_{i=1}^{n} X_{Bi} - \sum_{i=1}^{n} X_{Ai}) \right)^2 \right]
\end{aligned}
$$

Ker vemo, da je $\frac{1}{n} \sum_{i=1}^{n} X_{Bi} = \hat{p}_B$ nepristranska cenilka za oceno populacijskega deleža, potem lahko zapišemo:

$$
\begin{aligned}
s^2_{D_i} =& \frac{1}{n-1} \left[n\hat{p}_B + n\hat{p}_A - n \left(\hat{p}_B - \hat{p}_A \right) \right)^2 \right] \\
=& \frac{n}{n-1} \left[\hat{p}_B + \hat{p}_A - \hat{p}_B^2 - \hat{p}_A^2 + 2\hat{p}_A\hat{p}_B \right]
\end{aligned}
$$

Za testno statistiko (po CLI) torej vzamemo

$$
\begin{aligned}
T =& \frac{\bar{D} - 0}{\sqrt{\frac{s^2_{D_i}}{n}}} \\
=& \frac{\hat{p}_B - \hat{p}_A - 0}{\sqrt{\frac{1}{n-1} \left(\hat{p}_B - \hat{p}_B^2 + \hat{p}_A - \hat{p}_A^2 + 2\hat{p}_A\hat{p}_B \right)}} \\
T \sim& N(0,1)
\end{aligned}
$$

```
T = (0.47-0.44)/(sqrt(1/(2207-1)*(0.47-0.47$^2$+0.44-0.44$^2$+2*0.47*0.44)))
pvrednost=pnorm(T,lower.tail=FALSE)}  # enosmerni test
```

Ne moremo zavrniti H_0. Ocenjeno varianco lahko izračunamo tudi po formuli $\frac{1}{n-1}(\sum D_i^2 - n\hat{D}^2)$.

```
pB = 0.47
pA = 0.44
n = 2207
testna = (pB-pA)/(sqrt(1/(n-1)*(pB+pA-(pB-pA)^2)))
pnorm(testna,lower.tail=FALSE)
```

```
## [1] 0.06972951
```

Naloga 7 - dietetičarka

Vemo, da pri testu t na moč testa vpliva razlika med H_0 in H_A, velikost standardne napake in stopnja značilnosti. Prva in tretja količina sta v našem primeru konstantni, spreminjamo torej lahko le velikost standardne napake. Velja, da večja, kot je SE, manjša bo moč testa za konstantno razliko in α. V našem primeru je testna statistika

$$T = \frac{\bar{X_1} - \bar{X_2}}{s_p \cdot \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

Imenovalec predstavlja SE in v primeru enakih varianc je s_p kar polovica vsote ocenjenih varianc obeh skupin. Edini del, na katerega torej imamo vpliv, je $\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}$. Zanj želimo, da je čimmanjši. To bo dosegel v svojem minimumu. Tega se lotimo z iskanjem ekstremnih točk (odvajanje, izenačenje z 0).

$$\begin{aligned}
A =& \frac{1}{n_1} + \frac{1}{n_2} \quad \text{za min odvajamo po } n_1,\ n_2 \text{ zapišemo kot } n - n_1 \\
\frac{\partial A}{\partial n_1} =& -\frac{1}{n_1^2} + \frac{1}{(n-n_1)^2} \quad \text{izenačimo z 0, pomnošimo z imenovalcema} \\
0 =& n_1^2 - (n-n_1)^2 \\
0 =& n(n-2n_1)
\end{aligned}$$

Rešitev je $n_1 = n/2$, kar nam da minimum (to preverimo z drugim odvodom, ki mora biti pozitiven). }

```
# moc testa
# n1, n2 - test t neodvisna vzorca,
# razlika v velikosti skupin - vpliv na moc

# funkcija za izracun p-vrednosti
pttest <- function(n1,n,muA,mu0=0,sd0=15){
  x1 = rnorm(n1,mu0,sd0)
  x2 = rnorm(n-n1,mu0+muA,sd0)
  x = c(x1,x2)
  skupina = as.factor(c(rep(1,n1),rep(2,n-n1)))
  return(t.test(x~skupina,var.equal=TRUE)$p.value)
}

N=100
N1 = seq(10,90,by=10)
N2 = 100-N1
### velikost testa (muA == 0)
```

```r
runs=1000
rez0 = NULL
for(i in 1:runs){
  temp = sapply(N1,FUN=pttest,n=N,muA=0)
  rez0 = rbind(rez0,temp)
}
rez0 = as.data.frame(rez0)
names(rez0) = N1
# velikost za razlicne vrednosti n1 in n2
velikosttesta = apply(rez0,2,FUN=function(x){sum(x<0.05)/runs})
velikosttesta
```

```
##    10    20    30    40    50    60    70    80    90
## 0.058 0.048 0.063 0.054 0.058 0.055 0.052 0.044 0.060
```

```r
# moc testa (muA != 0)
rez = NULL
for(i in 1:runs){
  temp = sapply(N1,FUN=pttest,n=N,muA=10)
  rez = rbind(rez,temp)
}
moctesta = apply(rez,2,FUN=function(x){sum(x<0.05)/runs})
moctesta
```

```
## [1] 0.517 0.759 0.860 0.886 0.916 0.877 0.863 0.783 0.506
```

```r
### kaksno velikost vzorca naj dieteticarka izbere?
rezD = NULL
N=seq(70,90,by=2)
N1 = N/2
for(i in 1:runs){
  temp = NULL
  for(j in N1){
    temp = c(temp,pttest(n1=j,n=2*j,muA=10))
  }
  rezD = rbind(rezD,temp)
}
moctesta2 = apply(rezD,2,FUN=function(x){sum(x<0.05)/runs})
names(moctesta2) = N1
N[min(which(moctesta2 > 0.8))] # 74 enot
```

```
## [1] 72
```

```r
# s knjiznico
#install.packages("pwr")
library(pwr)
pwr.t.test(d=10/15,power=0.8,sig.level=0.05,
           type="two.sample",alternative="two.sided")
```

```
##
##      Two-sample t test power calculation
##
##               n = 36.30569
##               d = 0.6666667
##       sig.level = 0.05
##           power = 0.8
```

```
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

Naloga 8

Pod ničelno domnevo morajo biti vrednosti p enakomerno porazdeljene, grafi kažejo, da to velja za vse, razen za drugi test. Med temi testi sedaj izberemo tistega, ki ima največjo moč - najbolj je v levo zamaknjen tretji graf, to je torej naša izbira.

Naloga 9

a. Izberemo vrednosti mu_1, σ, n in Δ. Generiramo podatke:

```r
a <- rnorm(n,mu1,sigma)
b <- rnorm(n,mu1+delta,sigma)
```

Preverimo enakost s testom z. Shranimo vrednost p, ponovimo postopek 1000x in preštejemo, kolikokrat je bila vrednost $p < \alpha$. Delež takih ponovitev je ocena moči. b.

$$
\begin{aligned}
1-\beta &= P_{\mu_1-\mu_2=\Delta}\left(\frac{\bar{X}_1-\bar{X}_2}{\sigma_p\sqrt{2/n}} > z_{1-\alpha/2}\right) + P_{\mu_1-\mu_2=\Delta}\left(\frac{\bar{X}_1-\bar{X}_2}{\sigma_p\sqrt{2/n}} < z_{\alpha/2}\right) \\
&= P_{\mu_1-\mu_2=\Delta}\left(\frac{\bar{X}_1-\bar{X}_2}{\sigma_p\sqrt{2/n}} - \frac{\Delta}{\sigma_p\sqrt{2/n}} > z_{1-\alpha/2} - \frac{\Delta}{\sigma_p\sqrt{2/n}}\right) + \\
&\quad + P_{\mu_1-\mu_2=\Delta}\left(\frac{\bar{X}_1-\bar{X}_2}{\sigma_p\sqrt{2/n}} - \frac{\Delta}{\sigma_p\sqrt{2/n}} < z_{\alpha/2} - \frac{\Delta}{\sigma_p\sqrt{2/n}}\right) \\
&= P_{\mu_1-\mu_2=\Delta}\left(T > z_{1-\alpha/2} - \frac{\Delta}{\sigma_p\sqrt{2/n}}\right) + P_{\mu_1-\mu_2=\Delta}\left(T < z_{\alpha/2} - \frac{\Delta}{\sigma_p\sqrt{2/n}}\right)
\end{aligned}
$$

kjer velja $T \sim N(0,1)$. c. Ker je razlika Δ pozitivna, bo velik predvsem prvi del gornjega izraza, drugi del lahko zanemarimo. Večja kot bo vrednost $\frac{\Delta}{\sigma_p\sqrt{2/n}}$, večja bo moč. Očitno je torej, da moč raste z večjo vrednostjo razlike Δ in večjim vzorcem (n), ter je manjša, če je σ_p večja, torej če je večja tudi populacijska varianca. Moč je seveda odvisna tudi od mejne vrednosti in s tem od α, večja vrednost α, bo pomenila večjo mejo zavrnitve in s tem manjšo moč.

Naloga 10 - mutacija gena

a. $H_0 : p < 0,9$, enosmerni test, testiramo mejno vrednost, statistični test bo zavračal za velike vrednosti $\hat{p}$.
b. Za določeno velikost vzorca n najprej generiramo n opazovanj iz $Ber(0.22)$. Tako dobimo vzorčno število bolnic z mutacijo in število brez. Za število bolnic z mutacijo generiramo opazovanja iz $Ber(0.96)$ za ostale pa iz $Ber(0.93)$ (oziroma $Bin(n_m, 0.96)$ in $Bin(n_b, 0.93)$)

```r
p0 = 0.9
pA = 0.95
### n = ?
moci = NULL
sekv = 100:200
for(n in sekv){
  meja = qbinom(0.95,prob=p0,size=n)
```

```r
  moc = pbinom(meja,prob=pA,size=n,lower.tail=FALSE)
  moci = c(moci,moc)
}
sekv[min(which(moci>0.8))] # 179
```

```
## [1] 179
```

```r
# drugi del
ppM = 0.96
ppB = 0.93
pM = 0.22

pA = pM*ppM + (1-pM)*ppB
moci = NULL
sekv = 100:400
for(n in sekv){
  meja = qbinom(0.95,prob=p0,size=n)
  moc = pbinom(meja,prob=pA,size=n,lower.tail=FALSE)
  # oziroma generiramo vzorec 22%, 78% ...
  moci = c(moci,moc)
}
sekv[min(which(moci>0.8))] # 356
```

```
## [1] 356
```

```r
pwr.p.test(h=2*asin(sqrt(pA)) - 2*asin(sqrt(p0)),
           power=0.8,sig.level=0.05,alternative="greater")
```

```
##
##      proportion power calculation for binomial distribution (arcsine transformation)
##
##               h = 0.1344349
##               n = 342.0927
##       sig.level = 0.05
##           power = 0.8
##     alternative = greater
```