

Sprawozdanie 1

Analiza przeżycia

Marta Stankiewicz (282244) Kacper Szmigielski (282255)

Spis treści

1	Lista 1	2
1.1	Zadanie 1	2
1.2	Zadanie 2	4
1.3	Zadanie 3	4
1.4	Zadanie 4	5
1.5	Zadanie 5	5
2	Lista 2	6
2.1	Zadanie 1	6
2.2	Zadanie 2	7
2.3	Zadanie 3	7
3	Lista 3	8
3.1	Zadanie 1a	8
3.2	Zadanie 1b	9
3.3	Zadanie 2	10
3.4	Zadanie 3	11
4	Lista 4	13
4.1	Zadanie 1	13
4.2	Zadanie 2	14
4.3	Zadanie 3	16

Spis rysunków

1	Wykresy Weibulla	4
2	Badanie rozkładu histogramów	5

Spis tabel

1	Statystyki dla rozkładu Weibulla	5
2	Statystyki opisowe dla wygenerowanych danych cenzurowanych	7
3	Statystyki opisowe dla danych dotyczących leków	7
4	Realizacja przedziału ufności dla średniego czasu do remisji choroby	10
5	Oszacowania parametrów $\hat{\theta}$ i $\hat{\mu}$ oraz przedziały ufności $[T_L, T_U]$ dla leków A i B	11
6	Oszacowania obciążenia oraz błędów średniokwadratowych dla $\hat{\theta}$ oraz $\tilde{\theta}$. .	13
7	Porównanie mocy testu dla różnych wartości θ i liczebności prób ($n = 20, 50$)	15

1 Lista 1

1.1 Zadanie 1

Gęstość funkcji

$$f(t, \alpha, \beta, \gamma) = \frac{\alpha\gamma}{\beta} \left(\frac{t}{\beta}\right)^{\alpha-1} \left[1 - \exp\left(-\left(\frac{t}{\beta}\right)^{\alpha}\right)\right]^{\gamma-1} \exp\left(-\left(\frac{t}{\beta}\right)^{\alpha}\right) \mathbf{1}_{(0,\infty)}(t).$$

```
density_weib <- function(x, alpha, beta, gamma) {  
  dens <- ifelse(  
    x > 0,  
    (alpha * gamma / beta) *  
    (x / beta)^(alpha - 1) *  
    (1 - exp(-(x / beta)^alpha))^(gamma - 1) *  
    exp(-(x / beta)^alpha),  
    0  
  )  
  return(dens)  
}
```

Dystrybuanta funkcji

$$F(t, \alpha, \beta, \gamma) = \begin{cases} \left[1 - \exp\left(-\left(\frac{t}{\beta}\right)^\alpha\right) \right]^\gamma, & t \geq 0, \\ 0, & t < 0. \end{cases}$$

```
distrib_weib <- function(x, alpha, beta, gamma){
  dist <- (1-exp(-(x/beta)^alpha))^gamma
  return(dist)
}
```

Funkcja hazardu

$$h(t, \alpha, \beta, \gamma) = \frac{f(t, \alpha, \beta, \gamma)}{1 - F(t, \alpha, \beta, \gamma)} = \frac{f(t, \alpha, \beta, \gamma)}{S(t)},$$

```
haz_weib <- function(x, alpha, beta, gamma){
  haz <- density_weib(x, alpha, beta, gamma) /
    (1 - distrib_weib(x, alpha, beta, gamma))
  return(haz)
}
```

Funkcja kwantylowa (odwrotnej dystrybucyjności)

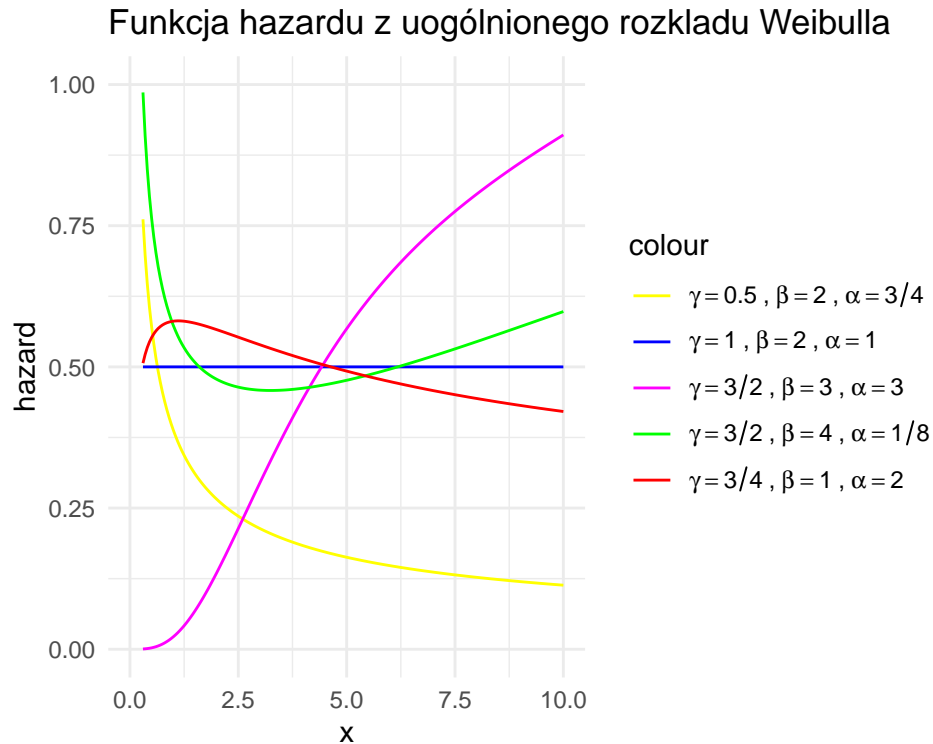
$$Q(p; \alpha, \beta, \gamma) = \beta \left[-\ln(1 - p^{1/\gamma}) \right]^{1/\alpha}, \quad 0 < p < 1,$$

```
quant_weib <- function(p, alpha, beta, gamma){
  quant <- beta * (-log(1 - p^(1 / gamma)))^(1 / alpha)
  return(quant)
}
```

Funkcje zostały napisane zgodnie z definicjami

1.2 Zadanie 2

1.2.1 Wykresy Weibulla



Rysunek 1: Wykresy Weibulla

Wykresy dla odpowiednich parametrów są zgodne z przedstawionymi na laboratoriach

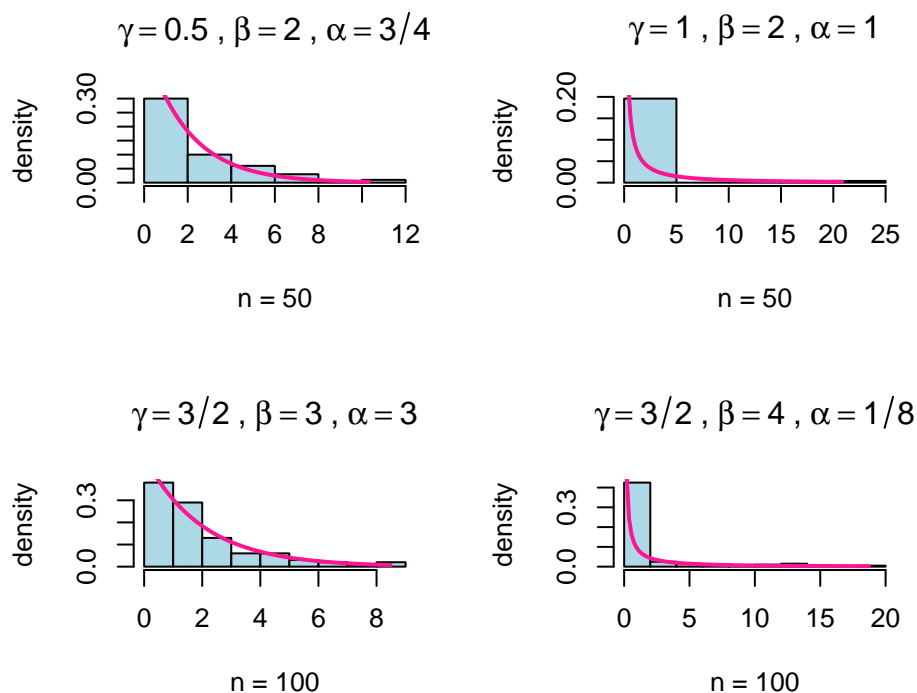
1.3 Zadanie 3

Dane są generowane przy użyciu metody odwrotnej dystrybucyjnej (u nas funkcji kwantylowej)

```
number_generator <- function(n, alpha, beta, gamma){  
  u <- runif(n) # losujemy z U(0,1)  
  x <- quant_weib(u, alpha, beta, gamma)  
  return(x)  
}
```

1.4 Zadanie 4

1.4.1 Histogramy oraz gęstości



Rysunek 2: Badanie rozkładu histogramów

Dla $n = 100$ histogramy są lepiej dopasowane do wykresów gęstości rozkładów. Co jest zgodne z intuicją, biorąc pod uwagę fakt, że dane są generowane z tych rozkładów.

1.5 Zadanie 5

1.5.1 Podstawowe wskaźniki opisowe

Tabela 1: Statystyki dla rozkładu Weibulla

Args	Mean	Med_emp	Med	SD	Q1_em	Q1_tr	Q3_em	Q3_tr	Min	Max	rozstęp
Zestaw 1	2.17	1.40	1.39	2.15	0.63	0.58	3.04	2.77	0.05	10.32	10.27
Zestaw 2	0.90	0.03	0.06	3.02	0.00	0.00	0.74	0.78	0.00	20.85	20.85
Zestaw 3	1.99	1.39	1.39	1.85	0.73	0.58	2.49	2.77	0.02	8.49	8.47
Zestaw 4	1.37	0.04	0.06	3.35	0.00	0.00	0.64	0.78	0.00	18.72	18.72

Zestaw 1 : $\alpha = 1 \beta = 2 \gamma = 1 n = 50$ | Zestaw 3 : $\alpha = 1 \beta = 2 \gamma = 1 n = 100$

Zestaw 2 : $\alpha = \frac{1}{2}$ $\beta = 2$ $\gamma = \frac{3}{8}$ $n = 50$ | Zestaw 4 : $\alpha = \frac{1}{2}$ $\beta = 2$ $\gamma = \frac{3}{8}$ $n = 100$

Dla większej próby widać dokładniejsze kwantyle bardziej zbliżone do ich teoretycznej wartości, niż dla próby mniejszej. Jedynie w zestawie pierwszym, Q3 jest bliższe teoretycznej wartości niż w zestawie 3, jest to zapewne spowodowane losowością danych oraz dość małą próbą (50 i 100).

2 Lista 2

2.1 Zadanie 1

Funkcja została napisana zgodnie z wytycznymi zadania

```
#-----Generatory-----
gen1 <- function(n, alpha, lambda, t0){
  y <- runif(n)
  x <- - (1/lambda)*log(1- y^(1/alpha))

  delta <- ifelse(x <= t0, 1, 0)
  x_obs <- pmin(x, t0)

  return(data.frame(x = x_obs, delta = delta))
}

gen2 <- function(n, alpha, lambda, m) {
  y <- runif(n)
  x <- - (1/lambda)*log(1- y^(1/alpha))

  x <- sort(x)
  x_cenz <- x[m]

  x <- c(x[1:m], rep(x_cenz, n-m))
  delta <- c(rep(1,m), rep(0, n-m))

  return(data.frame(x, delta))
}

gen3 <- function(n, alpha, lambda, eta){
  y <- runif(n)
  x <- - (1/lambda)*log(1- y^(1/alpha))
  c <- rexp(n, rate=1/eta)
  x <- pmin(x, c)
  delta <- ifelse(x==c, 0, 1)
```

```

return(data.frame(x, delta))
}
#-----

```

2.2 Zadanie 2

2.2.1 Podstawowe wskaźniki opisowe dla danych cenzurowanych

Tabela 2: Statystyki opisowe dla wygenerowanych danych cenzurowanych

	Q1	Q2	Q3	RANGE	IQR	CENSORED	NCENSORED
Cenz 1 typu	0.4996	0.7673	1.5158	1.6515	1.0162	5	15
Cenz 2 typu	0.4463	1.1395	1.1807	1.1195	0.7344	8	12
Cenz los	0.3205	0.4799	0.8105	4.3109	0.4899	7	13

Z tabeli widać, że sposób cenzurowania wpływa na charakterystykę danych.

Dla cenzurowania typu pierwszego wartości kwartylowe są umiarkowane, a rozstęp (RANGE = 1.6515) i IQR = 1.0162 wskazują na dość duże zróżnicowanie danych. W porównaniu z innymi typami, dane te są bardziej rozproszone niż przy cenzurowaniu typu drugiego, ale mniej zróżnicowane niż przy cenzurowaniu losowym.

Cenzurowanie typu drugiego daje najwyższą medianę (Q2 = 1.1395), co sugeruje przesunięcie wartości w stronę większych obserwacji.

Z kolei cenzurowanie losowe ma najniższe kwartyle i najwyższy zakres, co wskazuje na większe rozproszenie i obecność mniejszych wartości w próbie.

Różnice w liczbie danych ocenzurowanych i nieocenzurowanych między typami są niewielkie, więc każdy rodzaj cenzurowania wpływa głównie na rozkład, a nie na licznosc danych.

2.3 Zadanie 3

2.3.1 Podstawowe wskaźniki opisowe dla danych cenz

Tabela 3: Statystyki opisowe dla danych dotyczących leków

	Q1	Q2	Q3	RANGE	IQR	CENSORED	NCENSORED
lek A	0.3753	0.9711	1.0000	0.9665	0.6247	10	10
lek B	0.3329	0.8456	1.0000	0.9621	0.6671	10	10

Na podstawie tabeli widać, że Q2 i zakres (range) są niższe w grupie B, co może wskazywać na szybsze działanie leku oraz mniejsze zróżnicowanie wyników związane z indywidualnymi predyspozycjami biologicznymi badanych osób. W obu grupach wartość Q3 wynosi 1, co jest zrozumiałe, ponieważ połowa obserwacji została ocenzurowana.

3 Lista 3

3.1 Zadanie 1a

```
X_A <- c(0.03345514, 0.08656403, 0.08799947, 0.24385821,
         0.27755032, 0.40787247, 0.58825664, 0.64125620,
         0.90679161, 0.94222208, rep(1,10))
X_B <- c(0.03788958, 0.12207257, 0.20319983, 0.24474299,
         0.30492413, 0.34224462, 0.42950144, 0.44484582,
         0.63805066, 0.69119721, rep(1, 10))

t <- 1
n <- 20
R_A <- sum(X_A < t)
R_B <- sum(X_B < t)
T_A <- sum(X_A[1:10]) + t*(n-R_A)
T_B <- sum(X_B[1:10]) + t*(n-R_B)

theta_hat_A <- R_A / T_A
theta_hat_B <- R_B / T_B

theta_wave <- - log(1-R_A/n)/t

mi_hat_A <- 1/theta_hat_A
mi_hat_B <- 1/theta_hat_B
mi_wave <- 1/theta_wave
```

Na podstawie danych opisanych w zadaniu 3 z listy 2, przyjmując, że są one realizacjami zmiennych z rozkładu wykładniczego można wyznaczyć oszacowania największej wiarygodności średniego czasu do remisji choroby dla pacjentów leczonych lekiem A oraz pacjentów leczonych lekiem B. W tym celu skorzystano ze wzorów podanych na Wykładzie 3:

$$\hat{\vartheta} = \frac{R}{T_1}$$

gdzie

$$T_1 = \sum_{i=1}^R X_{(i)} + t_0(n - R)$$

gdzie zmienna losowa R jest losową liczbą obserwacji niecenzurowanych (kompletnych) w przedziale $(0, t_0]$. Oszacowanie NW średniego czasu do remisji choroby pacjentów leczonych lekiem A wynosi 0.7034 natomiast oszacowanie to w przypadku leku B wynosi 0.743 . Na podstawie uzyskanych oszacowań średniego czasu do remisji, lek A wydaje się lepszym wyborem, gdyż charakteryzuje się krótszym (mniejszym) średnim czasem do wystąpienia remisji w porównaniu z lekiem B.

3.2 Zadanie 1b

```
library(binom)
T.L.1.A <- binom.confint(R_A, n, methods = 'exact',
                        conf.level = 1-0.05)$lower
T.U.1.A <- binom.confint(R_A, n, methods = 'exact',
                        conf.level = 1-0.05)$upper

T.L.A1 <- - log(1-T.L.1.A)/t
T.U.A1 <- - log(1-T.U.1.A)/t

upp_A_1 <- 1/T.L.A1
low_A_1 <- 1/T.U.A1

T.L.2.A <- binom.confint(R_A, n, methods = 'exact',
                        conf.level = 1-0.01)$lower
T.U.2.A <- binom.confint(R_A, n, methods = 'exact',
                        conf.level = 1-0.01)$upper

T.L.A2 <- - log(1-T.L.2.A)/t
T.U.A2 <- - log(1-T.U.2.A)/t

upp_A_2 <- 1/T.L.A2
low_A_2 <- 1/T.U.A2

T.L.1.B <- binom.confint(R_B, n, methods = 'exact',
                        conf.level = 1-0.05)$lower
T.U.1.B <- binom.confint(R_B, n, methods = 'exact',
                        conf.level = 1-0.05)$upper

T.L.B1 <- - log(1-T.L.1.B)/t
T.U.B1 <- - log(1-T.U.1.B)/t

upp_B_1 <- 1/T.L.B1
low_B_1 <- 1/T.U.B1

T.L.2.B <- binom.confint(R_B, n, methods = 'exact',
                        conf.level = 1-0.01)$lower
T.U.2.B <- binom.confint(R_B, n, methods = 'exact',
                        conf.level = 1-0.01)$upper

T.L.B2 <- - log(1-T.L.2.B)/t
T.U.B2 <- - log(1-T.U.2.B)/t
```

```
upp_B.2 <- 1/T.L.B2
low_B.2 <- 1/T.U.B2
```

3.2.1 Tabela przedziałów

Tabela 4: Realizacja przedziału ufności dla średniego czasu do remisji choroby

Lek	α	T_L	T_U
A	0.05	0.768	3.1506
A	0.01	0.656	4.0720
B	0.05	0.768	3.1506
B	0.01	0.656	4.0720

Otrzymane w tabeli przedziały ufności dla średniego czasu do remisji są identyczne i nie pozwalają na odróżnienie skuteczności leków A i B. Jest to spowodowane faktem, że w obu grupach odnotowano taką samą liczbę nieocenzurowanych zdarzeń ($R_A = R_B$), przy tych samych n oraz t .

3.3 Zadanie 2

```
fun <- function(X, m, n, alpha) {
  Xs <- sort(X)
  T2 <- sum(Xs[1:m]) + (n - m)*Xs[m]
  theta.hat <- m/T2
  mi.hat <- 1/theta.hat

  q.m.1 <- qgamma(alpha / 2, shape = m, rate = m)
  q.m.2 <- qgamma(1 - alpha / 2, shape = m, rate = m)

  T.L <- m*q.m.1/T2
  T.U <- m*q.m.2/T2

  return(list(
    theta.hat = theta.hat,
    mi.hat = mi.hat,
    TL = T.L, TU = T.U
  ))
}
```

3.3.1 Tabela oszacowania

Tabela 5: Oszacowania parametrów $\hat{\theta}$ i $\hat{\mu}$ oraz przedziały ufności $[T_L, T_U]$ dla leków A i B

Lek	α	$\hat{\theta}$	$\hat{\mu}$	T_L	T_U
A	0.05	0.7332	1.3638	0.3516	1.2527
A	0.01	0.7332	1.3638	0.2725	1.4664
B	0.05	0.9643	1.0371	0.4624	1.6474
B	0.01	0.9643	1.0371	0.3584	1.9284

Średni czas do remisji ($\hat{\mu}$) jest krótszy dla leku B (1.0371) niż dla leku A (1.3638), co sugeruje, że lek B może działać szybciej. Jednak przedziały ufności dla obu leków są dość szerokie i częściowo się pokrywają, więc różnice nie są jednoznaczne. Warto zauważyć, że w zależności od przyjętego typu cenzurowania wnioski mogą się różnić. Przy założeniu cenzurowania I-go typu lek A wydaje się nieco lepszy, natomiast przy cenzurowaniu II-go typu lepsze wyniki uzyskuje lek B.

3.4 Zadanie 3

```
set.seed(123)
M <- 10000
theta <- 1
t.values <- c(0.5, 1, 2)
n.values <- c(10, 30)

gen1 <- function(n, alpha=1, lambda=theta, t0){
  y <- runif(n)
  x <- - (1/lambda)*log(1- y^(1/alpha))

  delta <- ifelse(x <= t0, 1, 0)
  x_obs <- pmin(x, t0)

  return(data.frame(x = x_obs, delta = delta))
}

symulacja <- function(n, t0, theta){
  repeat {
    X <- gen1(n, t0=t0, lambda=theta)
    R <- sum(X$delta)
    if (R < n) break # akceptujemy tylko próby, gdzie jest cenzurowanie
  }
}
```

```

}
T1 <- sum(X$x[which(X$delta ==1)]) + t0*(n - R)

theta.hat <- R/T1
theta.wave <- - log(1 - R/n)/t0

return(c(theta.hat, theta.wave))
}

results <- data.frame(
  n = numeric(),
  t0 = numeric(),
  Bias_hat = numeric(),
  MSE_hat = numeric(),
  Bias_wave = numeric(),
  MSE_wave = numeric()
)

for (n in n.values) {
  for (t0 in t.values) {

    est <- replicate(M, symulacja(n, t0, theta))

    theta.hat <- est[1, ]
    theta.wave <- est[2, ]

    bias.hat <- mean(theta.hat - theta)
    mse.hat <- mean((theta.hat - theta)^2)

    bias.wave <- mean(theta.wave - theta)
    mse.wave <- mean((theta.wave - theta)^2)

    results <- rbind(results, data.frame(
      n = n,
      t0 = t0,
      Bias_hat = bias.hat,
      MSE_hat = mse.hat,
      Bias_wave = bias.wave,
      MSE_wave = mse.wave
    ))
  }
}

```

3.4.1 Tabela bias mse

Tabela 6: Oszacowania obciążenia oraz błędów średniokwadratowych dla $\hat{\theta}$ oraz $\tilde{\theta}$

n	t_0	$\text{Bias}(\hat{\theta}, \theta)$	$\text{MSE}(\hat{\theta}, \theta)$	$\text{Bias}(\tilde{\theta}, \theta)$	$\text{MSE}(\tilde{\theta}, \theta)$
10	0.5	0.0683	0.3098	0.0804	0.3345
10	1.0	0.0581	0.1843	0.0814	0.2200
10	2.0	-0.0529	0.0810	-0.0808	0.0624
30	0.5	0.0230	0.0912	0.0251	0.0939
30	1.0	0.0249	0.0579	0.0332	0.0678
30	2.0	0.0230	0.0415	0.0531	0.0685

W przeprowadzonych symulacjach zauważalne jest, że wraz ze wzrostem liczby obserwacji n zarówno obciążenia, jak i błędy średniokwadratowe dla obu estymatorów systematycznie maleją, co wskazuje na poprawę ich dokładności. Zauważalne jest jednak to, że estymator $\hat{\theta}$ wypada korzystniej w porównaniu z estymatorem $\tilde{\theta}$, gdyż charakteryzuje się mniejszym obciążeniem oraz niższymi wartościami MSE.

4 Lista 4

4.1 Zadanie 1

Dysponujemy danymi cenzurowanymi I-go typu, które są realizacjami zmiennych losowych z rozkładu wykładniczego. Poniżej znajduje się zadeklarowana funkcja, której wartością dla tego typu danych jest wartość poziomu krytycznego w teście ilorazu wiarygodności do testowania hipotez dwustronnych, lewostronnych oraz prawostronnych.

```
wart_poz_kryt <- function(r, s, n, t0, theta0, rodzaj = "two.sided"){  
  
  theta_hat <- r / (s + t0*(n-r))  
  
  L <- function(theta){  
    L <- (theta^r)*exp(-theta*(s + t0*(n-r)))  
    return(L)  
  }  
  
  L_hat <- L(theta_hat)  
  L_theta0 <- L(theta0)
```

```

if (rodzaj=="two.sided"){
  lambda_obs <- L_theta0 / L_hat # funkcja osiagga sup dla est. MLE
} else if (rodzaj == "less") {
  if (theta_hat < theta0){
    lambda_obs <- L_theta0 / L_hat
  } else {
    lambda_obs <- 1
  }
} else {
  if (theta_hat > theta0){
    lambda_obs <- L_theta0 / L_hat
  } else {
    lambda_obs <- 1
  }
}

p_value <- 1 - pchisq(-2*log(lambda_obs), df = 1)

return (p_value)
}

```

4.2 Zadanie 2

```

set.seed(123)
M <- 10000
theta0 <- 1/2
n <- c(20,50)
t0 <- 2
theta <- c(0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.5, 2)

gen1 <- function(n, alpha=1, lambda=theta, t0){
  y <- runif(n)
  x <- - (1/lambda)*log(1- y^(1/alpha))

  delta <- ifelse(x <= t0, 1, 0)
  x_obs <- pmin(x, t0)

  return(data.frame(x = x_obs, delta = delta))
}

```

```

wyniki <- list()

for (n in n) {
  moc <- numeric(length(theta))
  p_val <- numeric(M)

  for (k in seq_along(theta)){
    lambda <- theta[k]

    for (j in 1:M){
      x_wylos <- gen1(n=n, lambda=lambda, t0 = t0)
      x_obs <- x_wylos$x
      r <- sum(x_wylos$delta)
      s <- sum(x_wylos$x[x_wylos$delta==1])
      p_val[j] <- wart_poz_kryt(r=r, s=s, n=n, t0=t0,
                              theta0=theta0, rodzaj = "two.sided")
    }
    moc[k] <- 1/M * sum(p_val[1:M]<0.05)
  }
  p_val_theta0 <- numeric(M)
  for (j in 1:M){
    x_wylos <- gen1(n = n, lambda = theta0, t0 = t0)
    r <- sum(x_wylos$delta)
    s <- sum(x_wylos$x[x_wylos$delta == 1])
    p_val_theta0[j] <- wart_poz_kryt(r = r, s = s, n = n, t0 = t0,
                                     theta0 = theta0, rodzaj = "two.sided")
  }

  rozmiar_hat <- mean(p_val_theta0 < 0.05)

  wyniki[[paste0("n=", n)]] <- data.frame(theta = theta, moc = moc)
}

```

4.2.1 Tabela mocy

Tabela 7: Porównanie mocy testu dla różnych wartości θ i liczebności prób ($n = 20, 50$)

θ	Moc ($n = 20$)	Moc ($n = 50$)
0.2	0.8015	0.9942
0.3	0.3966	0.7591
0.4	0.1258	0.2306

θ	Moc (n = 20)	Moc (n = 50)
0.5	0.0535	0.0523
0.6	0.1071	0.1818
0.7	0.2389	0.5081
0.8	0.4242	0.8050
0.9	0.6258	0.9442
1.0	0.7699	0.9907
1.5	0.9945	1.0000
2.0	1.0000	1.0000

Na podstawie wyników symulacji możemy zauważyć, że wartości oszacowania mocy są wyższe dla $n = 50$ niż dla $n = 20$ dla każdego $\theta \neq \theta_0$, co oznacza, że zwiększenie liczby obserwacji poprawia zdolność testu do wykrywania rzeczywistych różnic w parametrze.

Korzystając ze wzoru pokazanego na Wykładzie 4 tj.

$$\beta_\varphi = \sup_{\vartheta \in \Theta_0} \beta_\varphi(\vartheta)$$

gdzie $\beta_\varphi(\vartheta)$ oznacza funkcję mocy testu φ ,

otrzymujemy rozmiar testu. W przypadku hipotezy dwustronnej oznacza to, że sprawdzamy wartość oszacowania mocy testu dla $\theta = \theta_0$. U nas: dla $n = 20$ rozmiar testu wynosi 0.0535 natomiast dla $n = 50$ jest on równy 0.0535.

4.3 Zadanie 3

```

czas_A <- c(0.03345514, 0.08656403, 0.08799947, 0.24385821,
            0.27755032, 0.40787247, 0.58825664, 0.64125620,
            0.90679161, 0.94222208, rep(1, 10))

czas_B <- c(0.03788958, 0.12207257, 0.20319983, 0.24474299,
            0.30492413, 0.34224462, 0.42950144, 0.44484582,
            0.63805066, 0.69119721, rep(1, 10))

s_A <- sum(czas_A[1:10])
r <- 10
s_B <- sum(czas_B[1:10])
n <- 20
t0 <- 1
theta0 <- 1

p_wart_A <- wart_poz_kryt(r=r, s=s_A, n=n, t0=t0,

```



```
theta0=theta0, rodzaj="two.sided")  
p_wart_B <- wart_poz_kryt(r=r, s=s_B, n=n, t0=t0,  
                           theta0=theta0, rodzaj="two.sided")
```

Dla danych dotyczących leku A otrzymano wartość p-value równą $p_A = 0.2374$, natomiast dla leku B $p_B = 0.323$.

Przy poziomie istotności $\alpha = 0.05$ w obu przypadkach $p > \alpha$, dlatego nie ma podstaw do odrzucenia hipotezy zerowej $H_0 : \theta = 1$.

Oznacza to, że nie ma statystycznie istotnych dowodów na to, że średni czas do remisji choroby w grupie A lub w grupie B różni się od 1.