Analiza przeżycia
Raport 1

Romana Żmuda

6 grudnia 2020

Spis treści

1 Zadanie do sprawozdania - Część 1 **3**
1.1 Zadanie 1 3
1.2 Zadanie 2 5
1.3 Zadanie 3 5
1.4 Zadanie 4 6

2 Zadania do sprawozdania - Część 2 **9**
2.1 Zadanie 1 9
2.2 Zadanie 2 10

3 Zadania do sprawozdania - Część 3 **12**
3.1 Zadanie 1 12
3.2 Zadanie 2 13

4 Zadania do sprawozdania - Część 4 **15**
4.1 Zadanie 1 15
4.2 Zadanie 2 15

1 Zadanie do sprawozdania - Część 1

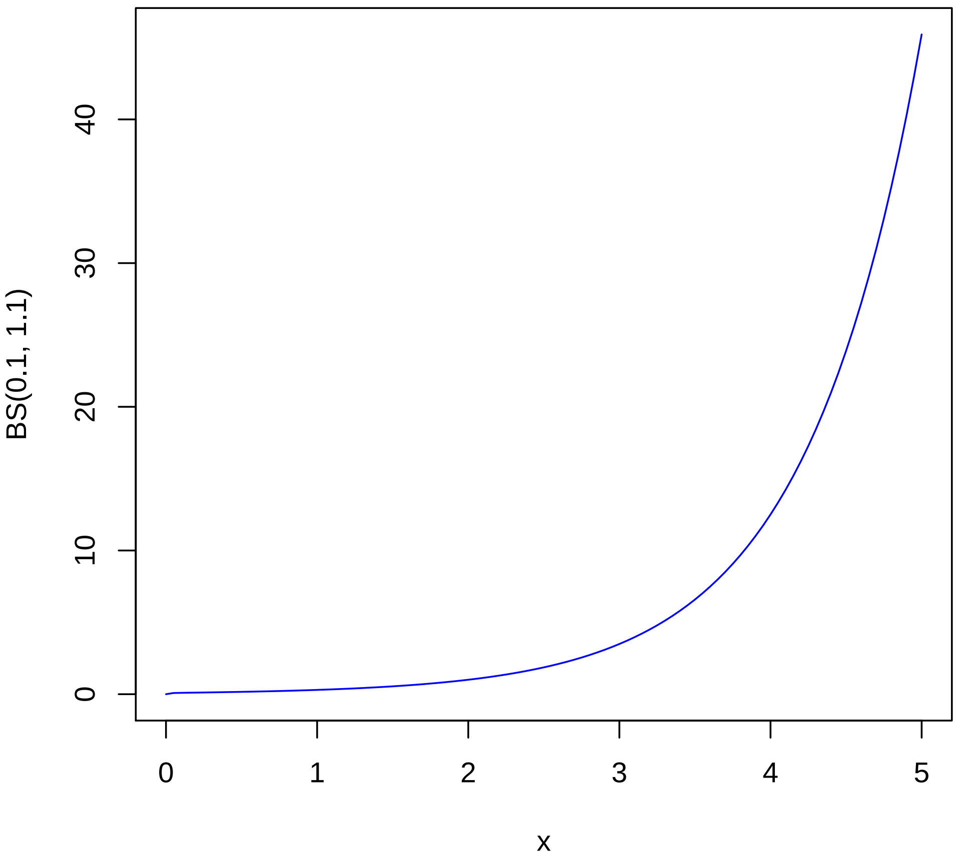
1.1 Zadanie 1

Najpierw wyznaczny odpowiednie funkcje definiujące gęstość, dystrybuantę, funkcję przeżycia oraz funkcję hazardu dla podanego rozkładu.

```
> BS_gestosc <- function(t, alpha, beta){
+   alpha*beta*exp(alpha)*(t^(beta-1))*exp(t^beta)*exp(-alpha*exp(t^(beta)))
+ }
> BS_dystrybuanta <- function(t, alpha, beta){
+   1-exp(alpha*(1-exp(t^beta)))
+ }
> BS_przezycia <- function(t, alpha, beta){
+   exp(alpha*(1-exp(t^beta)))
+ }
> BS_hazard <- function(t, alpha, beta){
+   BS_gestosc(t,alpha,beta)/BS_przezycia(t, alpha, beta)
+ }
>
```

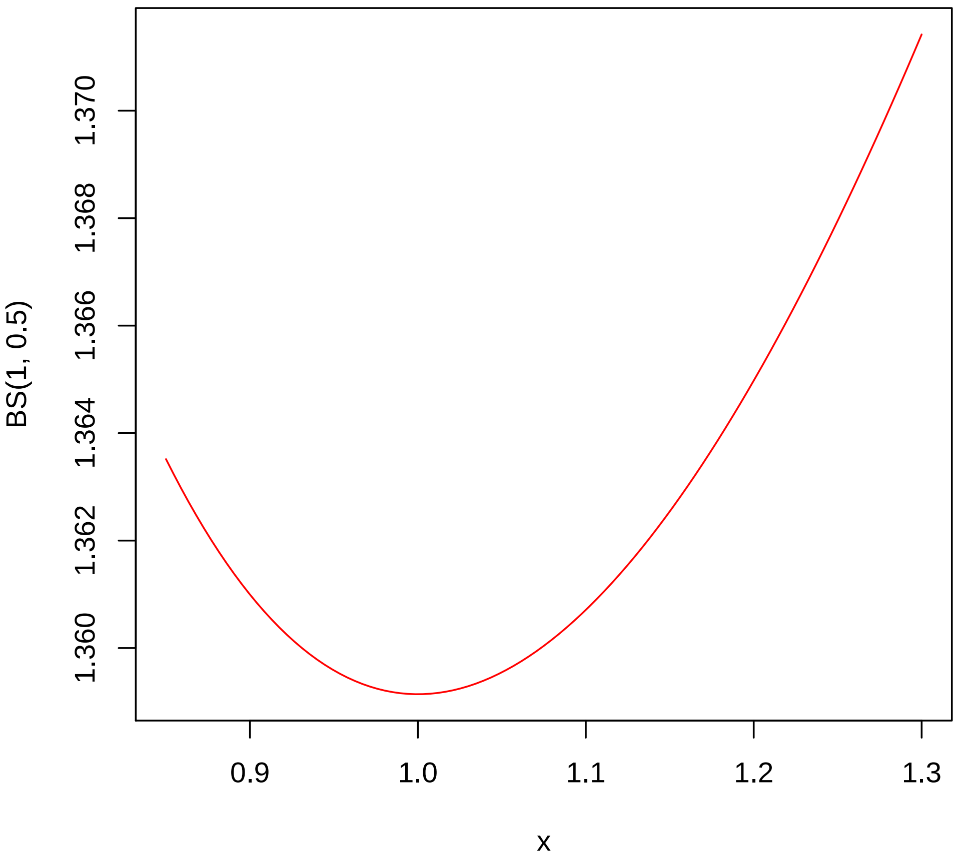
Zadaniem jest narysowanie 2 wykresów funkcji hazardu rozkładu $BS(\alpha, \beta)$, gdzie $\alpha > 0, \beta > 0$ są parametrami kształtu.
Biorąc $\beta >= 1$ dostaniemy rosnącą funkcje hazardu, poniżej znajduje się przykład(zobacz rysunek 1), natomiast dla $0 < \beta < 1$ (zobacz rysunek 2) będzie miała kształt wannowy.

Wykres niemalejącej funkcji hazardu 1, wtedy mamy rosnącą intensywnośći awarii dla rozkładu BS przy parametrach $\alpha = 0.1, \beta = 1.1$.



Rysunek 1: Wykresy funkcji hazardu $BS(0.1, 1.1)$

Wykres wannowy funkcji hazardu 2:



Rysunek 2: Wykresy funkcji hazardu $BS(1, 0.5)$

1.2 Zadanie 2

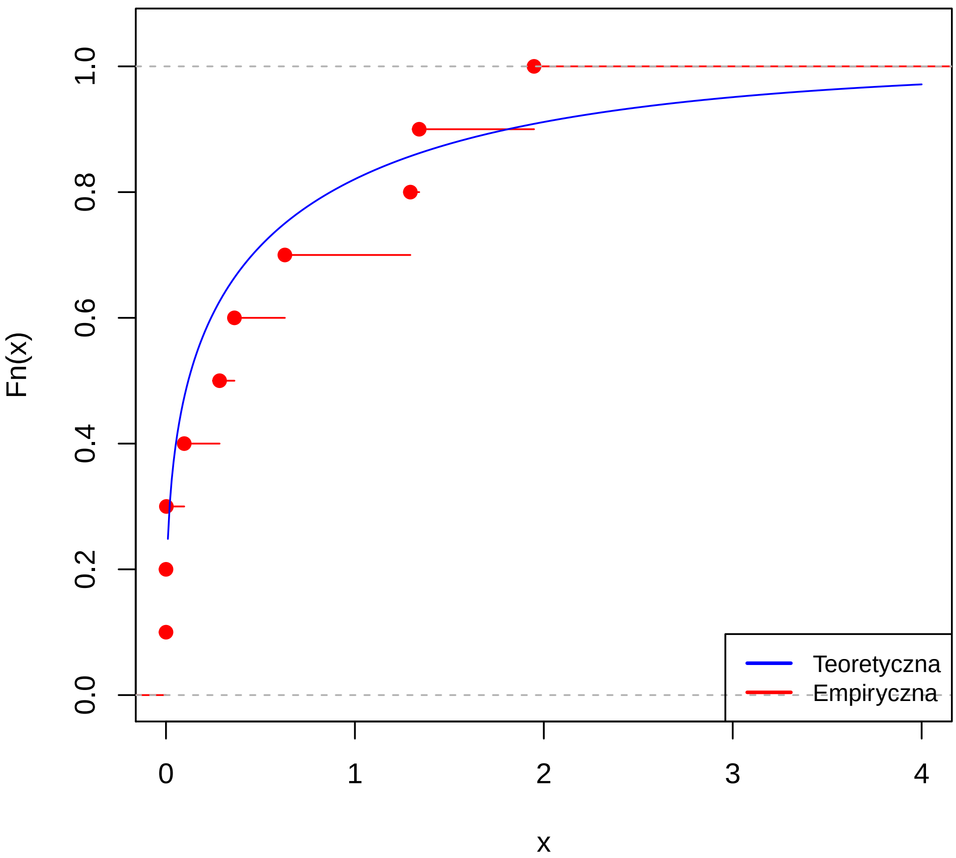
Napisanie programu generującego n zmiennych z rozkładu $BS(n, \alpha, \beta)$ metodą odwróconej dystrybuanty.

```
> #Funkcja na odwróconą dystrybuantę
> BS_odwrocona_dystrybuanta <- function(t, alpha, beta){
+     log(1-(log(1-t)/alpha))^(1/beta)
+ }
> BS_wygenerowane <- function(n, alpha, beta){
+   set <- runif(n)
+   BS_odwrocona_dystrybuanta(set, alpha, beta)
+ }
```

1.3 Zadanie 3

W tym zadaniu naszym celem jest narysowanie dystrybuanty rozkłady $BS(n, \alpha, \beta)$ oraz dystybuanty empirycznej. Wygenerujmy sobie $n = 10 zmiennych, gdzie \alpha = 1, \beta = 0.3$.

```
> #empiryczna
> wygenerowane_empiryczna<-BS_wygenerowane(10, 1, 0.3)
> #teoretyczna
> summary_set <- seq(0.01, 4, 0.01)
> wygenerowane_teoretyczna<-BS_dystrybuanta(summary_set, 1, 0.3)
>
```



Rysunek 3: Wykresy dystrybuanty teoretycznej i empirycznej

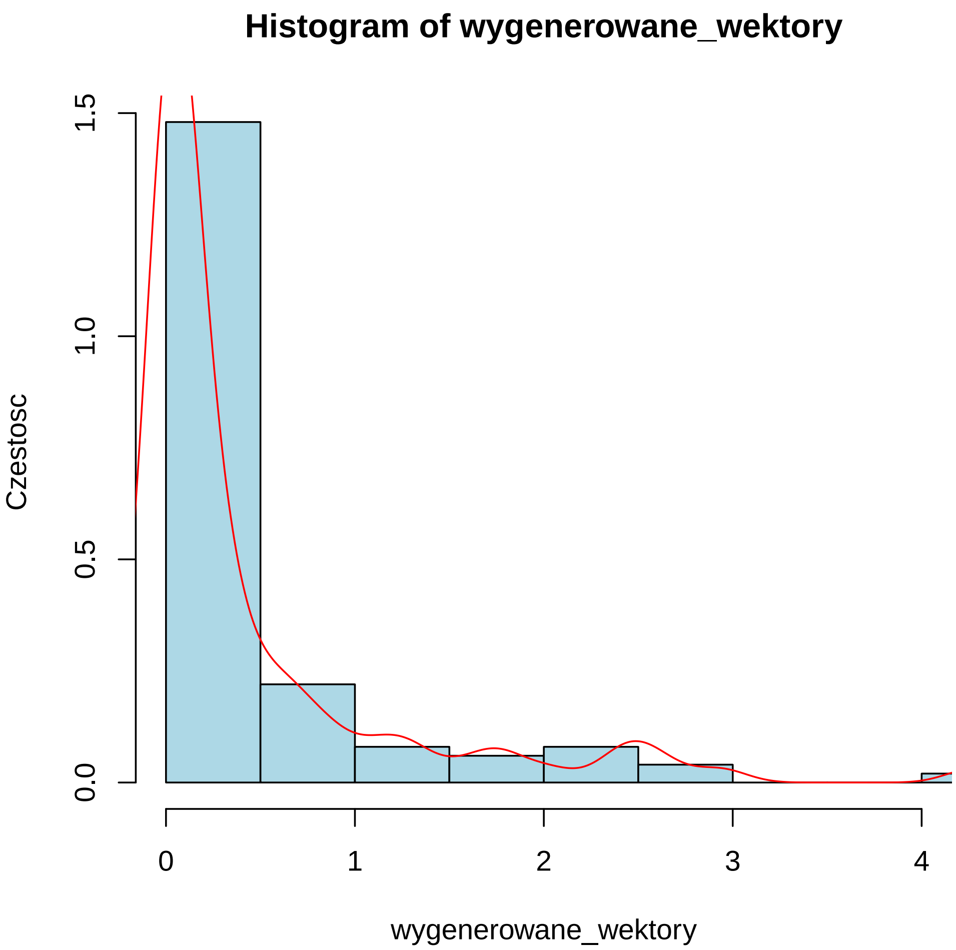
1.4 Zadanie 4

W ostatnim zadaniu z tej serii generujemy 100 liczb z rozkladu $BS(1, 0.3)$, następnie wyznaczamy wartości statystyk opisowych, przedstawiamy to w tabeli 1.

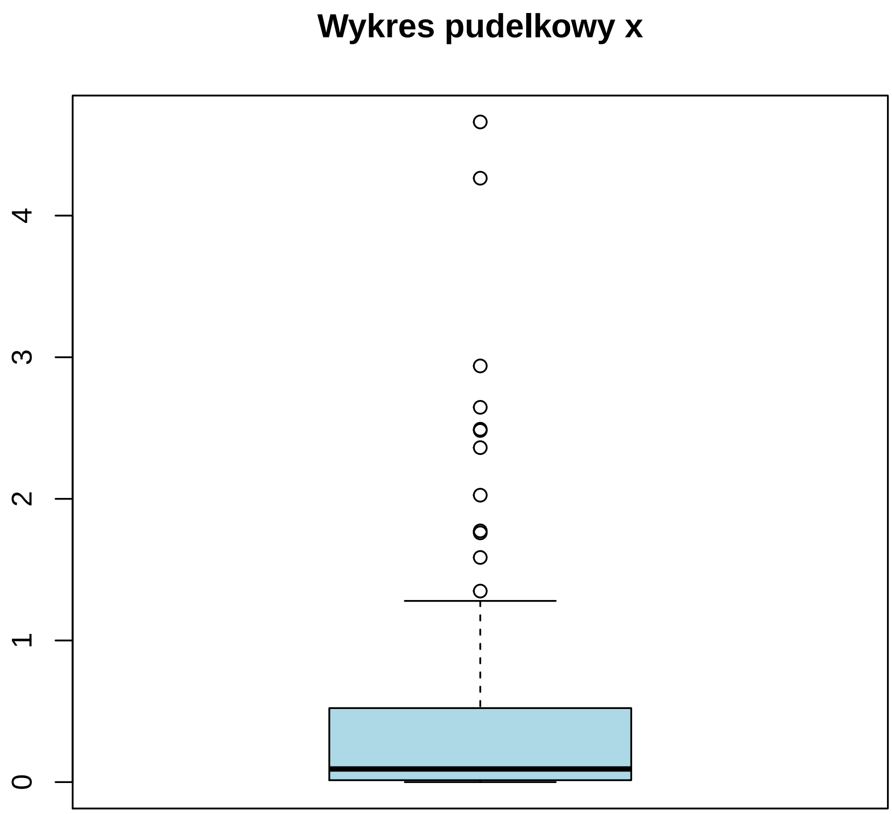
Tabela 1: Statystyki opisowe dla wygenerowanych liczb

Min	1st Qu	Mediana	Średnia	3d Qu.	Max	Odch.Stand.
0.00	0.01	0.09	0.48	0.52	4.66	0.88

Następnie na wykresach 4, 5 ilustrujemy ich gęstość oraz wykres pudełkowy w formie boxplota.



Rysunek 4: Histogram wygenerowanych zmiennych i ich gęstość



Rysunek 5: Wykres pudełkowy dla wektora obserwacji x

2 Zadania do sprawozdania - Część 2

2.1 Zadanie 1

Punkt A

Zadaniem jest napisanie programu do generowania n zmiennych cenzurowanych I-go typu z rozkładu wykładniczego z parametrem skali ϑ :

```
> Ityp <- function(theta, n, t){
+   x <- rexp(n, rate = 1/theta)
+   y <- vector()
+   for(i in 1:n){
+     if(x[i] <= t){
+       y[i] <- x[i]
+       }
+     else{
+       y[i] <- t
+     }
+   }
+   return(sort(y, decreasing = FALSE))
+ }
> #Przykladowe wygenerowanie zmiennych cenzurowanych
> #I-go typu dla n = 20 oraz t= 0.6
> Ityp(1, 20, 0.6)
```

```
 [1] 0.03304506 0.10240974 0.19925143 0.26115560 0.38007843 0.40035156
 [7] 0.60000000 0.60000000 0.60000000 0.60000000 0.60000000 0.60000000
[13] 0.60000000 0.60000000 0.60000000 0.60000000 0.60000000 0.60000000
[19] 0.60000000 0.60000000
```

Punkt B

Tym razem będziemy generować m zmiennych II-typu, gdzie $m <= n$. Zmienna m jest z góry narzuconą wartością, która reprezentuje liczbę m obserwacji (mp. awarii).

```
> IItyp <- function(theta, n, m){
+  x <- rexp(n, rate = 1/theta)
+  x_m <- sort(x)[m]
+  y <- pmin(x, x_m)
+  return(sort(y, decreasing = FALSE))
+ }
> #Przykladowe wygenerowanie zmiennych cenzurowanych
> #II-go typu dla n = 20 oraz m = 16
> IItyp(1, 20, 10)
```

```
 [1] 0.01706082 0.02978903 0.03131683 0.07327441 0.15448145 0.17839997
 [7] 0.25631880 0.26216740 0.27889726 0.43572621 0.43572621 0.43572621
[13] 0.43572621 0.43572621 0.43572621 0.43572621 0.43572621 0.43572621
[19] 0.43572621 0.43572621
```

Punkt C

W ostatnim punkcie generujemy zmienne z losowego cenzurowania dla parametru η będącego parametrem skali.

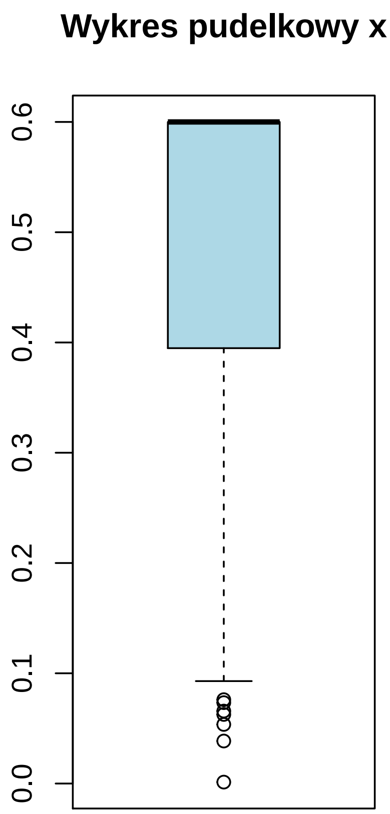
```
> losowecenzurowanie <- function(theta, n, eta){
+  x <- rexp(n, rate = 1/theta)
+  z <- rexp(n, rate = 1/eta)
+  y <- vector()
+  for( i in 1:n){
+    y[i] <- min(x[i], z[i])
+  }
+   return(y)
+ }
> #Przykladowe wygenerowanie zmiennych cenzurowanych
> #losowo dla n = 20 oraz eta = 0.5
> losowecenzurowanie(1, 20, 0.5)
```

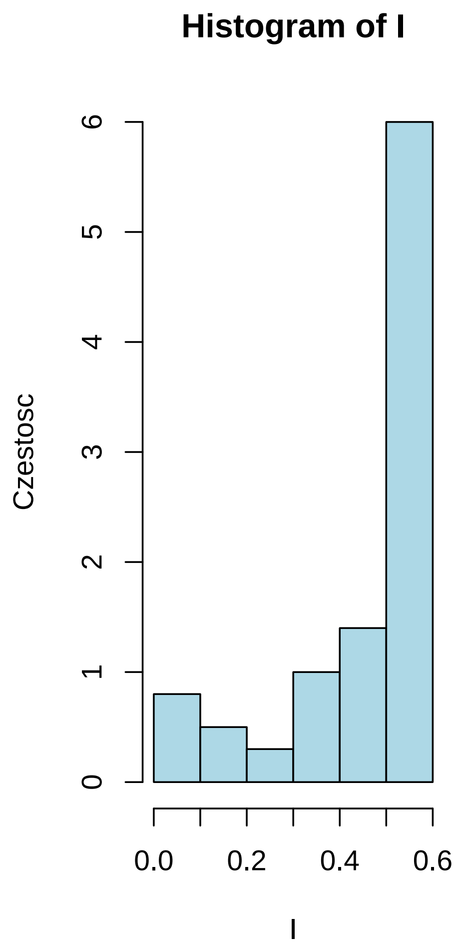
```
 [1] 0.10988521 0.39519606 0.34106283 0.46325342 0.39099735 0.09166469
 [7] 0.33217579 0.03900255 0.04660011 0.12837773 0.64158365 0.38387703
[13] 0.55224555 0.04913994 0.02040859 0.01922899 0.06305624 1.48859212
[19] 0.54267788 0.17974490
```

2.2 Zadanie 2

Wybieram zmienne z cenzurowania I-go typu, używając funkcji do generowania zmiennych z zadania 1, opisuję je używając wartości rozsądnych, wykorzystam do tego boxplota i histogram, który wskaże mi możliwe sensowne statystyki opisowe.

```
> I <- Ityp(1, 100, 0.6)
```





Rysunek 6: Wykres pudełkowy i histogram dla wektora obserwacji x

Tabela 2 przedstawia sensowne wartości statystyki opisowej obliczonych dla wygenerowanych zmiennych I typu cenzurowania.

Tabela 2: Statystyki opisowe dla typu I cenzurowania

Min	1st Qu	Mediana	Średnia	3d Qu.	Max
0.00	0.40	0.60	0.48	0.60	0.60

Jedynymi sensownymi statystykami opisowymi są kwartyle niskiego rzędu. Średnia nie jest sensowną statystyką opisową. Wszystki kwartyle powyższej zadanego t_0 również nie mają sensu. Mediana nie ma sensu jeśli więcej niż połowa danych jest cenzurowanych.

3 Zadania do sprawozdania - Część 3

3.1 Zadanie 1

Punkt A

W tym zadaniu mamy podać oszacowanie największej wiarygodności średniego czasu do pierwszej awarii dla 20 komponentów przy czym 10 z nich uległo awarii w podanych momentach, jest to I typ cenzorowania

```
> x<-c(0.497, 0.638, 0.703, 0.839, 0.841, 0.950, 1.054, 1.103, 1.125, 1.495)
> #dane prawostronnie cenzurowane I - typu
> n <- 20
> t0 <- 2 #2 lata
> R <- length(x)
> #estymator NW parametru 1/ni
> T1 <- sum(x) + t0*(n - R)
> #statystyka dla estymatora T1
> ni <- T1/R
> #oszacowany czas do pierwszej awarii metodą NW
> ni
```

```
[1] 2.9245
```

Punkt B

Następnie mamy do wyznanczenia przedział ufności na poziomie 99%

```
> przedzial_ufnosci <- binom.confint(R, n, conf.level = 0.99)
> # przedzial ufności dla 1/ni
> przedzial_ufnosci$theta_lower <- -log(1-przedzial_ufnosci$lower)/t0
> przedzial_ufnosci$theta_lower
```

```
 [1] 0.1441396 0.1191372 0.1353732 0.1138875 0.1227888 0.1373065 0.1340611
 [8] 0.1335219 0.1334997 0.1778363 0.1441396
```

```
> przedzial_ufnosci$theta_upper <- -log(1-przedzial_ufnosci$upper)/t0
> # przedzialy dla ni
> ni_lower <- data.frame("lower" = 1/przedzial_ufnosci$theta_upper)
> ni_upper <- data.frame("upper" =1/przedzial_ufnosci$theta_lower)
> x <- data.frame("metoda" = przedzial_ufnosci$method, "x" = przedzial_ufnosci$x, "n"
> przedzial_ufnosci_ni <-cbind.data.frame(x, ni_lower, ni_upper)
>
```

Odpowiednia tablica (3) z przedziałami ufności dla odpowiadającym im metodą:

Tabela 3: Przedziały ufności

	metoda	x	n	lower	upper
1	agresti-coull	10.00	20.00	1.44	6.94
2	asymptotic	10.00	20.00	1.29	8.39
3	bayes	10.00	20.00	1.39	7.39
4	cloglog	10.00	20.00	1.49	8.78
5	exact	10.00	20.00	1.31	8.14
6	logit	10.00	20.00	1.40	7.28
7	probit	10.00	20.00	1.38	7.46
8	profile	10.00	20.00	1.38	7.49
9	lrt	10.00	20.00	1.38	7.49
10	prop.test	10.00	20.00	1.66	5.62
11	wilson	10.00	20.00	1.44	6.94

3.2 Zadanie 2

Punkt A

W tym zadaniu mamy podać oszacowanie największej wiarygodności średniego czasu do pierwszej awarii dla n = 20 komponentów przy czym zakładamy, że obserwujemy tylko do 10 pierwszych awarii, jest to II typ danych cenzurowanych.

```
> x<-c(0.497, 0.638, 0.703, 0.839, 0.841, 0.950, 1.054, 1.103, 1.125, 1.495)
> #dane prawostronnie cenzurowane I - typu
> n <- 20
> m <- 10
> t0 <- 2 #2 lata
> sum <- sum(x)
> x_m <- x[m]
> #estymator NW parametru 1/ni
> T2 <- sum + (n - m)*x_m
> #statystyka dla estymatora T1
> ni_II <- T2/m
> #oszacowany czas do pierwszej awarii metodą NW
> ni_II

[1] 2.4195
```

Punkt B

Następnie mamy do wyznanczenia przedział ufności na poziomie 99%, więc $\alpha = 0.01$ nadal przy cenzurowaniu II typu

```
> a <- qgamma(0.01, shape = m, scale = 1/m)
> b <- qgamma(0.99, shape = m, scale = 1/m )
> ni_II <- T2/m
> ni_lower_II <- data.frame("lower" = ni_II/b)
> ni_upper_II <- data.frame("upper" = ni_II/a)
> przedzial_ufnosci_ni_II <- data.frame(ni_lower_II, ni_upper_II)
> przedzial_ufnosci_ni_II
```

```
       lower    upper
1 1.288125 5.858071
```

Odpowiednia tablica (4) z przedziałem ufności dla danych cenzurowanych II typu:

Tabela 4: Przedział ufności typu II

	lower	upper
1	1.29	5.86

4 Zadania do sprawozdania - Część 4

4.1 Zadanie 1

Mamy do rozważenia sytuację, taką że czas w latach do pierwszej awarii komponentu pewnego typu ma rozkład wykładniczy. Przeprowadzono test na 20 komponentach tego typu i zanotowano czas do wystąpienia pierwszej awarii. W ciągu dwóch lat obserwacji awarii uległo 10 komponentów. Mamy do czynienia z danymi cenzurowanymi I rodzaju. Określmy podstawowe zmienne wynikające z tego typu zmienych cenzurowanych, wprowadzamy takie samo nazewnictwo, jak na wykładach:

```
> x<-c(0.497, 0.638, 0.703, 0.839, 0.841, 0.950, 1.054, 1.103, 1.125, 1.495)
> #dane prawostronnie cenzurowane I - typu
> n <- 20
> t0 <- 2 #2 lata
> R <- length(x)
> #estymator NW parametru 1/ni
> T1 <- sum(x) + t0*(n - R)
> #statystyka estymatora T1
> ni <- T1/R
> theta_0 <- 1/2.9
```

Mamy to rozważenia hipotezę H_0 : Po analizie funkcji wiarygodności(nałożenie logarytmu i obliczenie pochodnej), zauważamy że funkcja jest rosnąca na przedziale $(0, max\theta)$, gdzie $max\theta = R/T_1$.

```
> theta_max <- R/T1
> theta_max <- data.frame("theta_max" = theta_max)
> theta_0 <- data.frame("theta_zero" = theta_0)
> thety <- data.frame(theta_max, theta_0)
>
```

Odpowiednia tablica (5) ukazująca potrzebne wartości do następnej analizy:

Tabela 5: Wartości

	theta_max	theta_zero
1	0.34	0.34

Analizując iloraz $\lambda(x)$ dostajemy, że supremum na zbiorze $theta_0$ to $\mathrm{L}(max\theta)$, natomiast na zbiorze $theta$ supremum również $\mathrm{L}(max\theta)$. Wstawiając do definicji $\lambda(x)$ dostajemy że $\lambda(x) = 1$. Ostatecznie hipotezę uznajemy za wiarygodną, co oznacza dla nas że hipoteza H_0 przyjmujemy.

4.2 Zadanie 2

W tym zadaniu rozważamy podobną sytuację z tym, że obserwacje prowadzono do 10 awarii. Zauważamy, że podane zmienne są teraz cenzurowanymi II rodzaju. Tak jak w przykładzie wyżej definiujemy odpowiednie wartości:

```
> x<-c(0.497, 0.638, 0.703, 0.839, 0.841, 0.950, 1.054, 1.103, 1.125, 1.495)
> #dane prawostronnie cenzurowane I - typu
> n <- 20
> m <- 10
> t0 <- 2 #2 lata
> sum <- sum(x)
> x_m <- x[m]
> #estymator NW parametru 1/ni
> T2 <- sum + (n - m)*x_m
> #statystyka dla estymatora T1
> theta_max2 <- T2/m
>
```

Najpierw wyznaczamy iloraz $\lambda(x)$ odpowiadający wartości supremum funkcji wiarygodnościna zbiorze θ_0 przez supremum funkcji ilorazowu wiarygodności na całym zbiorze θ. Po obliczeniach podobnych do pierwszego zadania tylko z inną funkcją wiarygodności i statystyką dostajemy, że tymi supremami są $L(\theta_0)$ oraz $L(m/T2)$. Wtedy nasz test przyjmuje postać $\varphi(x) = 1$, gdy $\lambda < \lambda_0 \bigvee 0$, gdy $\lambda >= \lambda_0$. Poniżej wyliczam statystykę $\lambda(x)$:

```
> lambda<-(1/2^10*(exp(-1/2.9*T2))/(1/theta_max2^10*exp(-1/theta_max2*T2)))
```

Następnie wyznaczamy wartości krytyczne korzystając z twierdzenia Wirksa:

```
> lambda0<-exp(-qchisq(1-0.01, df=1)/2)
```

Sprawdzamy wcześniej zdefiniowaną hipotezę:

```
> test<-function(lambda0){
+   if(lambda < lambda0){
+     print("Odzucamy hipotezę H_0")
+
+   }
+   else if(lambda >= lambda0){
+     print("Przyjmujemy hipotezę H_0")
+   }
+ }
> test(lambda0)

[1] "Przyjmujemy hipotezę H_0"

> library(survival)
> library(survminer)
> library(ggplot2)
> dane3<-read.delim("CzasDoDializy.csv", header = TRUE, ";")
> dane3$Czas <- as.numeric(dane3$Czas)
> surv_object1<-Surv(dane3$Czas, dane3$Cenzura)
> fit1 <- survfit(surv_object1 ~ dane3$Arg25Pro, data = dane3,
+                 type = "kaplan-meier")
> g1<-ggsurvplot(fit1, data = dane3, conf.int = FALSE)
> g1
```