Uniwersytet Warszawski
Wydział Matematyki, Informatyki i Mechaniki

Aleksandra Urbaniec
Nr albumu: 220775

Analiza przeżycia, teoria i przykład zastosowania w badaniu długości życia pacjentek z rakiem piersi

Praca magisterska
na kierunku MATEMATYKA
w zakresie MATEMATYKI OGÓLNEJ

Praca wykonana pod kierunkiem
dra inż. Przemysława Biecka
Instytut Matematyki Stosowanej i Mechaniki - Zakład Statystyki Matematycznej

czerwiec 2010

Oświadczenie kierującego pracą

Potwierdzam, że niniejsza praca została przygotowana pod moim kierunkiem i kwalifikuje się do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data Podpis kierującego pracą

Oświadczenie autora (autorów) pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.

Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data Podpis autora (autorów) pracy

Streszczenie

W pracy przedstawione jest wprowadzenie do analizy przeżycia, opis modeli, metody estymacji parametrów tych modeli oraz przykład zastosowania w medycynie. Praca składa się z części teoretycznej i praktycznej: przy użyciu danych rzeczywistych modelowana jest długość życia pacjentek z rakiem piersi. Ponadto zostały przeprowadzone symulacje badające własności niektórych testów i estymatorów w analizie przeżycia.

Słowa kluczowe

Analiza przeżycia, Model Cox'a, Test log-rank, Estymator Kaplana-Meiera, Estymator Flemingtona-Harringtona

Dziedzina pracy (kody wg programu Socrates-Erasmus)

11.2 Statystyka

Klasyfikacja tematyczna

46N30, 62P10

Tytuł pracy w języku angielskim

Survival analysis, theory and application in breast cancer study

Spis treści

Wprowadzenie

Tematem prezentowanej pracy są zagadnienia związane z analizą przeżycia. Analiza przeżycia to z definicji zbiór metod statystycznych służący do badania czasu, jaki upłynie do wystąpienia określonego zdarzenia [2]. Z matematycznego punktu widzenia opiera się ona na teorii rachunku prawdopodobieństwa (rozkłady zmiennych, asymptotyka), statystyki (własności estymatorów) oraz optymalizacji (metody iteracyjne optymalizacji funkcji). Obecnie rozwój analizy przeżycia odbywa się w dużej mierze dzięki wykorzystaniu procesów stochastycznych. Pozwala to między innymi na modelowanie czasu między powtarzającymi się zdarzeniami.

Pierwotnie analiza przeżycia była używana do celów aktuarialnych (zagadnienie długości życia ubezpieczonego) oraz przemysłowych (badanie trwałości produktów) [8]. Obecnie jej techniki wykorzystywane są również w zagadnieniach medycznych, ekonomicznych, demograficznych, czy społecznych, jak na przykład: przy estymacji kosztów opieki zdrowotnej [3], analizowaniu czasu do zatrudnienia kobiet po urodzeniu dziecka [7], czy długości życia pacjentów po przeszczepie. W części praktycznej pracy zostało zaprezentowane jedno z zastosowań analizy przeżycia - przy wyjaśnianiu zjawisk medycznych.

Większość narzędzi statystycznych zawiera zaimplementowane funkcje wykorzystywane w analizie przeżycia. Program R dysponuje pakietem *survival*, w programie SAS można korzystać m. in. z PROC PHREG służącej do estymacji modelu Cox'a, podobnie łatwo analizę przeżycia przeprowadza się za pomocą programów SPSS oraz Statistica. W pracy wszelkie obliczenia wykonuję w programie R, w tekście oraz w załączniku ujawnione zostały funkcje używane przy przeprowadzanych analizach.

Praca składa się z trzech części. W rozdziale pierwszym przedstawiona została teoria analizy przeżycia, począwszy od podstawowych pojęć (cenzurowanie zmiennych, funkcja przeżycia, hazard, test log-rank ect.) poprzez estymatory funkcji przeżycia, modele parametryczne, kończąc na nieparametrycznym modelu proporcjonalnego hazardu (modelu Cox'a). Drugi rozdział zawiera wyniki przeprowadzonych przeze mnie symulacji badających własności estymatorów i testów używanych w analizie przeżycia. W kręgu zainteresowania znajdywała się obciążoność nieparametrycznych estymatorów funkcji przeżycia, moc testu log-rank i rozkład estymatorów parametrów w modelach parametrycznych. Ostatni rozdział to analiza danych rzeczywistych, a modelowanym zagadnieniem jest czas do śmierci pacjentek chorych na raka piersi.

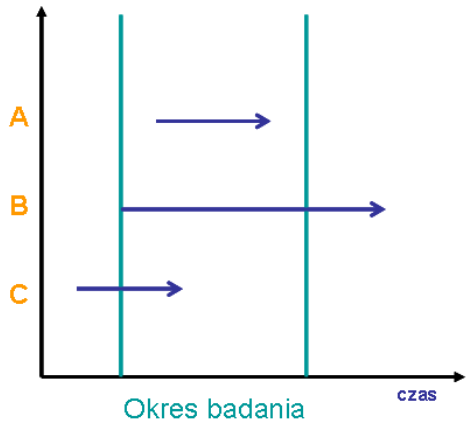
Rozdział 1

Analiza przeżycia - wstęp teoretyczny

1.1. Podstawowe definicje w analizie przeżycia

Analiza przeżycia to w najogólniejszej definicji zbiór metod statystycznych, badających procesy, w których interesujący jest czas jaki upłynie do (pierwszego) wystąpienia pewnego zdarzenia [2]. Tym zdarzeniem może być śmierć pacjenta, następne ocielenie się krowy, czy migracja do innego miasta. Analiza przeżycia ma również zastosowanie w naukach aktuarialnych przy modelowaniu śmierci ubezpieczonego lub momentu wystąpienia szkody.

Pierwszym poważnym problemem, który napotyka się w tego typu analizach jest *cenzurowanie danych*. Zazwyczaj obserwacje poddaje się badaniu w określonym czasie. Dla części obserwacji w tym okresie nie zaobserwuje się szukanego zdarzenia. Wiadomo jednak, że to zdarzenie nastąpi kiedyś w przyszłości. Mówimy wtedy o cenzurowaniu prawostronnym. Cenzurowanie lewostronne ma miejsce wtedy, gdy wiemy, że dane zdarzenie zaobserwowano wcześniej, lecz nie wiemy dokładnie kiedy. Na przykład planujemy przeprowadzić eksperyment badający ile czasu potrzebuje krowa po ocieleniu, by znowu znaleźć się w rui i rozpoczynamy badanie 30 dni po ocieleniu się krowy. Część zwierząt może jednak znaleźć się w rui w ciągu tych 30 dni, są one wtedy lewostronnie cenzurowane. W cenzurowaniu przedziałowym wiadomo, że zdarzenie miało miejsce w jakimś przedziale czasowym, jest on nam jednak bliżej nieznany.



Rysunek 1.1: Różne rodzaje cenzurowania - opracowanie własne.

Posługując się następującym przykładem oraz rysunkiem 1.1 prześledźmy raz jeszcze różne typy cenzurowania: Badamy czas życia pacjenta po operacji przeszczepu serca. Gdy zaobserwowano śmierć pacjenta i znana jest data przeszczepu, można ustalić dokładny czas życia po przeszczepie, nie ma więc cenzurowania. Z taką sytuacją mamy do czynienia w przykładzie A. Cenzurowanie prawostronne występuje wtedy, gdy chory pozostaje przy życiu aż do zakończenia badania - przypadek B. Obserwacja jest cenzurowana lewostronnie (przypadek C), gdy nieznany jest czas operacji przeszczepu (pacjent nie pamięta, zginęła jego dokumentacja medyczna etc.), wiadomo jednak, że operacja nastąpiła nie wcześniej niż w chwili t_0. Jeśli więc znany jest czas śmierci pacjenta (t_1), długość życia po operacji jest nie większa niż $t_1 - t_0$.

Wprowadźmy teraz kluczowe pojęcia w analizie przeżycia. Niech czas do wystąpienia zdarzenia będzie zmienną losową T z rozkładu - f o dystrybuancie F. *Funkcja przeżycia* zadana jest wzorem

$$S(t) = 1 - F(t^-). \tag{1.1}$$

Określa ona jakie jest prawdopodobieństwo, że zdarzenie nie zostanie zaobserwowane do czasu t. Można również zastanawiać się, jaka jest „szansa" zaobserwowania zdarzenia w chwili Δt, jeśli nie zostało ono zaobserwowane do chwili t. Opisuje to *funkcja hazardu*. Gdy czas jest ciągły funkcja hazardu zadana jest wzorem

$$h(t) = \lim_{\triangle t \to 0} \frac{\mathbb{P}(t \leqslant T < t + \triangle t \,|\, T \geqslant t)}{\triangle t}. \tag{1.2}$$

Skumulowany hazard jest natomiast sumą hazardu do chwili t, czyli dla czasu ciągłego jest postaci

$$\Lambda(t) = \int_0^t h(u)\, du, \tag{1.3}$$

a dla czasu dyskretnego

$$\Lambda(t) = \sum_{j:t_j \leqslant t} h(t_j). \tag{1.4}$$

Z powyższych wzorów można wyprowadzić zależność między skumulowaną funkcją hazardu a funkcją przeżycia. Dla czasu ciągłego mamy bowiem

$$h(t) = \lim_{\triangle t \to 0} \frac{\mathbb{P}(t \leqslant T < t + \triangle t \,|\, T \geqslant t)}{\triangle t} = \frac{\lim_{\triangle t \to 0} \frac{\mathbb{P}(t \leqslant T < t + \triangle t)}{\triangle t}}{\mathbb{P}(T \geqslant t)} = \frac{f(t)}{S(t)} = -\frac{S'(t)}{S(t)} = -\frac{d \log[S(t)]}{dt}, \tag{1.5}$$

a stąd

$$\Lambda(t) = \int_0^t h(u)\, du = \log[S(0)] - \log[S(t)] = -\log[S(t)]. \tag{1.6}$$

Ostatecznie więc otrzymujemy zależność

$$S(t) = \exp(-\Lambda(t)). \tag{1.7}$$

1.2. Metody nieparametryczne estymacji funkcji przeżycia

1.2.1. Estymator Kaplana-Meiera

W estymacji metodą Kaplana-Meiera przyjmuje się dyskretny czas ($t \in \{t_1, t_2, \ldots, t_n\}$) oraz cenzurowanie niezależne od czasu przeżycia. Estymator funkcji przeżycia dla chwili t zadany

Czas t_j	Start n_j	Zdarzenia d_j	Cenzurowane $w_j = n_j - n_{j+1} - d_j$	Zbiór ryzyk r_j	Pr. przeżycia $P_j = (r_j - d_j)/r_j$	Funkcja przeżycia $S_j = P_j \times P_{j-1}$
0	31	2	3	$31 - 3 = 28$	$(28 - 2)/28 = 0.93$	$0.93 \times 1.00 = 0.93$
1	26	1	2	$26 - 2 = 24$	$(24 - 1)/24 = 0.96$	$0.96 \times 0.93 = 0.89$
2	23	1	2	$23 - 2 = 21$	$(21 - 1)/21 = 0.95$	$0.95 \times 0.89 = 0.85$
3	20	1	2	$20 - 2 = 18$	$(18 - 1)/18 = 0.94$	$0.94 \times 0.85 = 0.80$
ect.						

Tabela 1.1: Przykład liczenia estymatora Kaplana-Meiera, źródło: [12].

jest następującym wzorem:

$$\hat{S}(t) = \prod_{j:t_j \leqslant t} \frac{r_j - d_j}{r_j}, \text{ dla } 0 \leqslant t \leqslant t_n. \tag{1.8}$$

W powyższej definicji r_j oznacza liczbę obserwacji objętych ryzykiem, zaś d_j liczbę wystąpień zdarzenia w chwili t_j. Wartość r_j dla chwili t_j jest różnicą między liczbą obserwacji, dla których czas do zdarzenia jest równy co najmniej t_{j-1} a liczbą obserwacji, które uległy cenzurowaniu w przedziale czasowym $(t_{j-1}, t_j]$. Przykład liczenia estymatora Kaplana-Meiera, zaczerpnięty z artykułu [12], znajduje się w tabeli 1.1.

1.2.2. Estymator Flemingtona-Harringtona

Innym sposobem estymacji funkcji przeżycia metodą nieparametryczną jest powiązanie jej ze skumulowaną funkcją hazardu ($\Lambda(t)$). Dla estymatora Flemingtona-Harringtona punktem wyjścia jest estymator skumulowanej funkcji hazardu Nelsona–Aalena zadany wzorem

$$\hat{\Lambda}(t) = \sum_{j:t_j \leqslant t} \frac{d_j}{r_j}, \tag{1.9}$$

gdzie d_j i r_j to, tak jak w przypadku estymatora Kaplana-Meiera, liczba zdarzeń i liczba obserwacji objętych ryzykiem w chwili t_j. Korzystając z 1.7, estymator Flemingtona-Harringtona jest więc ostatecznie postaci

$$\hat{S}(t) = \exp(-\hat{\Lambda}(t)). \tag{1.10}$$

1.3. Metody parametryczne estymacji funkcji przeżycia

Może się zdarzyć, że posiadamy dodatkowe informacje na temat badanych zjawisk, na przykład wiemy z jakiej rodziny rozkładów pochodzą. Dla potrzeb analizy przeżycia jest to sytuacja bardzo dla korzystna, pozwala bowiem na prognozowanie przy użyciu dopasowanego rozkładu.

Mając zatem rodzinę rozkładów, z której pochodzą obserwacje, wyznacza się dystrybuantę $F(t)$, a stąd $\lambda(t)$, $\Lambda(t)$ i $S(t)$, korzystając z pokazanych wcześniej zależności między tymi funkcjami. W analizie przeżycia najczęściej stosowane są następujące rozkłady: wykładniczy, Weibulla, Gompertza i log-logistyczny. Z dwóch pierwszych będę korzystała w dalszej części rozdziału, są więc one omówione poniżej.

1.3.1. Rozkład wykładniczy

Funkcja gęstości rozkładu wykładniczego jest następującej postaci:

$$f(t) = \exp(-\lambda t), \text{ dla } t \geqslant 0, \ \lambda > 0. \tag{1.11}$$

Dostajemy więc

$$h(t) = \lambda, \tag{1.12}$$

$$\Lambda(t) = \lambda t, \tag{1.13}$$

$$S(t) = \exp(-\lambda t). \tag{1.14}$$

Zauważmy, że dla tego rozkładu funkcja hazardu jest stała $h(t) = \text{const}$, tzn. „niebezpieczeństwo" wystąpienia zdarzenia nie zależy od czasu. W wielu przypadkach założenie to jest dalekie od rzeczywistości, na przykład przy modelowaniu śmierci ubezpieczonego. W modelach aktuarialnych przyjmuje się zazwyczaj założenie, że ryzyko rośnie z czasem.

1.3.2. Rozkład Weibulla

Funkcję gęstości dla rozkładu Weibulla zapisuje się jako

$$f(t) = k\lambda(t\lambda)^{k-1} \exp\left(-(t\lambda)^k\right), \text{ dla } t \geqslant 0,\ \lambda, k > 0, \tag{1.15}$$

co daje

$$h(t) = k\lambda(t\lambda)^{k-1}, \tag{1.16}$$

$$\Lambda(t) = (t\lambda)^k, \tag{1.17}$$

$$S(t) = \exp\left(-(t\lambda)^k\right). \tag{1.18}$$

Rozkład Weibulla posiada dwa parametry. Parametr λ nazywany jest parametrem skali, natomiast k - parametrem kształtu. Zauważmy, że rozkład wykładniczy jest szczególnym przypadkiem tego rozkładu (dla $k = 1$). Monotoniczność funkcji hazardu zależy od parametru kształtu. Gdy $k = 1$ hazard jest stały (mamy wtedy rozkład wykładniczy), gdy $k > 1$ rosnący, zaś dla $k < 1$ malejący w czasie.

1.4. Testy statystyczne w analizie przeżycia - testowanie istotności różnic

Jednym z podstawowych zagadnień w analizie przeżycia jest testowanie, czy dwie próby mają te same funkcje przeżycia. Podział na próby może odbywać się na podstawie jakiejś zmiennej objaśniającej. Na przykład, w późniejszym rozdziale, podczas analizowania danych o pacjentkach chorych na raka piersi, interesować mnie będzie, czy pacjentki z przerzutami żyją krócej niż te bez przerzutów.

Jeśli nie obserwuje się cenzurowania, do powyższego problemu można stosować standardowe testy porównujące rozkłady w próbach, jak test *Kołomogorowa-Smirnowa*, czy *Sign Test*. W pozostałym (najczęstszym) przypadku wykorzystuje się testy uwzględniające cenzurowanie. Najpopularniejszym testem tego typu jest test *log-rank*. Poniżej zaprezentowana jest konstrukcja tego testu.

Rozważmy dwie grupy obserwacji w momentach czasu $t = 1, 2, \ldots, T$. Hipoteza zerowa zakłada, że obie grupy mają takie same funkcje przeżycia. Dla każdej chwili t wyznacza się zaobserwowaną oraz oczekiwaną liczbę zdarzeń w każdej grupie. Niech dalej:

- N_{1t}, N_{2t} - liczba obserwacji będących w stanie ryzyka w chwili t odpowiednio dla pierwszej i drugiej grupy,
- $N_t = N_{1t} + N_{2t}$ - liczba wszystkich obserwacji będących w stanie ryzyka w chwili t,

- O_{1t}, O_{2t} - liczba zaobserwowanych zdarzeń w chwili t odpowiednio dla pierwszej i drugiej grupy,
- $O_t = O_{1t} + O_{2t}$ - liczba wszystkich zaobserwowanych zdarzeń w chwili t.

Jeśli rzeczywiście obie próby pochodzą z tych samych rozkładów, zdarzenia powinny rozkładać się proporcjonalnie do liczby obserwacji z danej grupy w stosunku do całkowitej liczby obserwacji (O_t), czyli dla grupy j ($j = 1, 2$) oczekiwana liczba zdarzeń w chwili t wynosi

$$E_{jt} = O_t \frac{N_{jt}}{N_t},\ j = 1, 2 \tag{1.19}$$

zaś wariancja

$$V_t = \frac{O_t(N_{1t}/N_t)(N_{2t}/N_t)(N_t - O_t)}{N_t - 1}. \tag{1.20}$$

Ostatecznie proponowana statystyka testowa jest postaci

$$Z = \frac{\sum_{t=1}^{T}(O_{1t} - E_{1t})}{\sqrt{\sum_{t=1}^{T} V_t}}. \tag{1.21}$$

Jeśli hipoteza zerowa o równości krzywych przeżycia jest prawdziwa statystyka Z jest asymptotyczna z centralnym twierdzeniem granicznym.

Alternatywną (zaimplementowaną m. in. w programie R) statystyką testu log-rank jest statystyka Z zadana jako

$$Z = \sum_{j=1}^{k} \frac{\left(\sum_{t=1}^{T}(O_{jt} - E_{jt})\right)^2}{\sum_{t=1}^{T} E_{jt}}, \tag{1.22}$$

gdzie definicje E_{jk}, O_{jk} dla $j = 1, \ldots, k$ rozszerzamy z 2 na k grup. Przy prawdziwości hipotezy zerowej powinna ona zbiegać dla dużych N do rozkładu χ_2 z $k - 1$ stopniami swobody, gdzie k - liczba grup. Statystyka ta jest więc również użyteczna, gdy bada się równość funkcji przeżycia w więcej niż dwóch grupach.

W programie R (pakiet *survial*) znajduje się funkcja *survdiff*(), w której zaimplementowana jest cała rodzina testów G-rho, opartych na statystyce 1.22. W testach tych zdarzenia ważone są wagami $S(t)^{\text{rho}}$, dla $S(t)$ - estymatora Kaplana-Meiera, rho$\in [0, 1]$.

Test log-rank należy do rodziny tych testów przy rho=0 - wszystkie zdarzenia mają tą samą wagę. Dla rho=1 mamy do czynienia z modyfikacją Peto-Peto testu *Gehana-Wilcoxona* - większą wagą objęte są zdarzenia wcześniejsze.

Porównując testy G-rho, test log-rank jest zalecany, gdy śmiertelność w grupach jest proporcjonalna - krzywe przeżycia nie przecinają się. W pozostałych przypadkach można zaobserwować przewagę testów z obserwacjami ważonymi (gdzie zdarzenia wcześniejsze mają większe wagi niż późniejsze). Testy te mogą jednak prowadzić do błędnych wyników dla dużego poziomu cenzurowania we wczesnej fazie badania (źródło [11]).

1.5. Parametryczne modele przeżycia

1.5.1. Modele z przeskalowanym czasem życia

Do parametrycznych modeli przeżycia należą „modele z przeskalowanym czasem życia"(ang. *accelerated failure time models*). Zakłada się w nich różne funkcje przeżycia w zależności od

wektora zmiennych objaśniających x_i. Mówiąc bardziej obrazowo, funkcja przeżycia jest skalowana tak, że czas biegnie inaczej w zależności od obserwacji. Dla obserwacji i mamy

$$S_i(t) = S_0\left(\frac{t}{\phi_i}\right) = S_0(t_0),\ \phi_i > 0 \text{ - const.} \tag{1.23}$$

Zwykle przyjmuje się, że

$$\phi_i = \exp(\beta^T x_i), \tag{1.24}$$

gdzie β to wektor parametrów.

Niech czas do zdarzenia dla obserwacji i będzie zmienną losową T_i, wtedy $T_0 = \frac{T_i}{\phi_i} = T_i \exp(-\beta^T x_i)$ ma stałą dystrybucję oraz

$$\log(T_0) = \log(T_i) - \beta^T x_i. \tag{1.25}$$

W ogólności modele parametryczne są postaci (symbol $\sim$ oznacza równość rozkładów)

$$l(T) \sim \beta^T x + \sigma\varepsilon, \tag{1.26}$$

gdzie $l()$ jest zwykle transformacją liniową, σ – parametrem skali, a ε – zadaną dystrybucją.

Zastanówmy się teraz, jakiej postaci jest model parametryczny dla konkretnych rozkładów. Rozważmy najprostszy przypadek, czyli *rozkład wykładniczy*. Docelowo T_0 powinno być standardowym rozkładem wykładniczym (Exp(1)). Dla i-tej obserwacji T_i jest z rozkładu Exp(λ_i). Przeskalowuje się więc czas o $\phi_i = \frac{1}{\lambda_i}$

$$S_i(t) = 1 - \exp(-\lambda_i t) = S_0(\lambda_i t). \tag{1.27}$$

Z 1.24 oraz 1.25 dostajemy

$$\log(T) \sim \beta^T x + \log(\varepsilon),\ \varepsilon \text{ pochodzi z rozkładu wykładniczego z parametrem } \lambda = 1.^1 \tag{1.28}$$

Podobnie, gdy obserwacje pochodzą z *rozkładu Weibulla* o parametrach λ_i i k

$$S_i(t) = 1 - \exp\left(-(\lambda_i t)^k\right) = S_0(\lambda_i t), \tag{1.29}$$

gdzie S_0 jest funkcją przeżycia dla rozkładu Weibulla(1,k). Model parametryczny z przeskalowanym czasem dla rozkładu Weibulla ma więc postać:

$$\log(T) \sim \beta^T x + \alpha \log(\varepsilon), \tag{1.30}$$

gdzie ε pochodzi z rozkładu Weibulla, $\alpha = \frac{1}{k}$ nazywana jest parametrem skali.

1.5.2. Modele proporcjonalnego hazardu

Innym sposobem estymacji modeli przeżycia są modele proporcjonalnego hazardu, do których należy m. in. model Cox'a (patrz podrozdział 1.6). U ich podstaw leży *założenie proporcjonalnego hazardu*, przy którym funkcję hazardu dla i-tej obserwacji definiuje się wzorem

$$h_i(t) = h_0(t)\exp(\beta^T x_i'), \tag{1.31}$$

gdzie $h_0(t)$ jest bazową funkcją hazardu (jednakową dla wszystkich obserwacji), x_i'-wektorem zmiennych objaśniających dla i-tej obserwacji, a β-szukanymi parametrami. Zauważmy, że dla

[1] Warto zauważyć, że postać parametru ϕ_i zależy od parametryzacji rozkładu. Na przykład, gdy przyjmuje się, że rozkład wykładniczy jest postaci $f(t) = \exp\left(-\frac{t}{\theta_i}\right)$, zachodzi $\phi_i = \theta_i$.

modeli parametrycznych z rozkładem wykładniczym oraz rozkładem Weibulla, oba podejścia są równoważne z dokładnością do przeskalowania parametrów wektora β. To przedstawienie jest jednak o tyle wygodniejsze, że pozwala na szybką interpretację parametrów, jako czynników wpływających na funkcję hazardu, czyli na szansę zaobserwowania zdarzenia w danym momencie.

Dla rozkładu wykładniczego mamy

$$h_i(t) = \lambda_i = h_0(t)\exp(\beta^T x_i') = 1 * \exp(\beta^T x_i') = \exp(\beta^T x_i'). \tag{1.32}$$

Czyli z powyższego oraz $\phi_i = \frac{1}{\lambda_i}$ dostajemy zależność między dwoma podejściami

$$\exp(\beta^T x_i') = \lambda_i = \frac{1}{\phi_i} = \exp(-\beta^T x_i), \tag{1.33}$$

co ostatecznie daje

$$x_i' = -x_i. \tag{1.34}$$

Podobnie dla rozkładu Weibulla

$$h_i(t) = h_0(t)\exp(\beta^T x_i') = k\lambda\left(t\lambda_i\right)^{k-1} = kt^{k-1}\lambda_i^k. \tag{1.35}$$

Podstawiając $h_0(t) = kt^{k-1}$ mamy $\lambda_i^k = \exp(\beta^T x_i')$. W modelu z przeskalowanym czasem życia dla rozkładu Weibulla wyprowadziliśmy w 1.29, że $\phi_i = \frac{1}{\lambda_i}$. Zatem

$$\lambda_i = \exp\left(\frac{\beta^T x_i'}{k}\right) = \frac{1}{\phi_i} = \exp(-\beta^T x_i), \tag{1.36}$$

co ostatecznie daje

$$x_i' = -x_i k = -\frac{x_i}{\alpha}. \tag{1.37}$$

W programie R (w bibliotece *survival*) dostępna jest funkcja *survreg*(), z której będę korzystać w następnych rozdziałach. Służy ona do estymacji modeli parametrycznych z przeskalowanym czasem życia. Dla rozkładów wykładniczego i Weibulla przyjmuje ona odpowiednio modele postaci 1.28 oraz 1.30.

1.5.3. Estymacja parametrów w modelu

Parametryczne modele przeżycia estymuje się, podobnie jak uogólnione modele liniowe (ang. *generalized linear models*), stosując iterowaną ważoną metodę najmniejszych kwadratów (ang. *iteratively reweighted least squares*). Szczególnym traktowaniem powinny być jednak objęte obserwacje cenzurowane. Poniżej zaprezentowany zostanie sposób estymacji modeli parametrycznych (na podstawie [13]) zaimplementowany m. in. we wspomnianej wcześniej funkcji *survreg*().

Niech y będzie wektorem danych (np. logarytmem z czasu zaobserwowanego do zajścia zdarzenia), może zawierać obserwacje cenzurowane. Zakładamy, że dla i-tej obserwacji (oznaczenia, jak w podrozdziale wyżej)

$$z_i = \frac{y_i - \beta^T x_i}{\sigma} \sim f, \tag{1.38}$$

gdzie f - zadana dystrybucja. Funkcja wiarygodności dla y jest więc postaci

$$L = \left(\prod_{i\in \mathrm{nC}} (f(z_i)/\sigma)\right)\left(\prod_{i\in \mathrm{Cl}} \int_{-\infty}^{z_i} f(u)du\right)\left(\prod_{i\in \mathrm{Cp}} \int_{z_i}^{\infty} f(u)du\right)\left(\prod_{i\in \mathrm{Cs}} \int_{z_i^l}^{z_i^r} f(u)du\right), \tag{1.39}$$

gdzie nC, Cl, Cp oraz Cs oznaczają zbiory indeksów odpowiednio dla obserwacji niecenzurowanych, cenzurowanych lewostronnie, prawostronnie oraz przedziałowo, natomiast $z_i^l = \frac{y_i^l - \beta^T x_i}{\sigma}$, $z_i^r = \frac{y_i^l - \beta^T x_i}{\sigma}$, a y_i^l, y_i^r - granice przedziału przy cenzurowaniu przedziałowym. Wtedy logarytm funkcji wiarygodności wyraża się jako

$$l = \sum_{i \in \text{nC}} (g_1(z_i) - \log(\sigma)) + \sum_{i \in \text{Cl}} g_2(z_i) + \sum_{i \in \text{Cp}} g_3(z_i) + \sum_{i \in \text{Cs}} g_4(z_i^l, z_i^r), \tag{1.40}$$

przy oznaczeniach $g_1 = \log(f)$, $g_2 = \log(F)$, $g_3 = \log(1-F)$ oraz $g_4 = \log\left(F(z_i^r) - F(z_i^l)\right)$, a F - dystrybuanta rozkładu f. Niech $\mu = X\beta$ będzie wektorem predyktorów linowych, β- wektorem szukanych parametrów, X - wektorem obserwacji, a N liczbą obserwacji. Pochodne logarytmu funkcji wiarygodności po β_j można zapisać jako

$$\frac{\partial l(\beta, \sigma)}{\partial \beta_j} = \sum_{i=1}^{N} \frac{\partial g^i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \beta_j} = \sum_{i=1}^{N} x_{ij} \frac{\partial g^i}{\partial \mu_i}, \tag{1.41}$$

$$\frac{\partial l(\beta, \sigma)}{\partial \beta_j \beta_k} = \sum_{i=1}^{N} x_{ij} x_{ik} \frac{\partial^2 g^i}{\partial \mu_i^2}, \tag{1.42}$$

gdzie $g^i \in \{g_1, g_2, g_3, g_4\}$ i zależy od tego, czy oraz ewentualnie jak i-ta obserwacja była cenzurowana. Jeśli chodzi o dokładne wzory na $\frac{\partial g_i}{\partial \mu}$, $\frac{\partial g_i}{\partial(\log(\sigma))}$, $\frac{\partial^2 g_i}{\partial \mu^2}$, $\frac{\partial^2 g_i}{\partial \mu \partial(\log(\sigma))}$, $\frac{\partial^2 g_i}{\partial(\log(\sigma))^2}$ znajdują się one m. in. w [13] str. 72.

Potraktujmy na moment parametr σ jako stałą. Iteracyjnie wyznacza się $\hat{\beta}$ tak, że wartość $\hat{\beta}^{(n+1)}$ w $(n+1)$-tym kroku wynosi $\hat{\beta}^{(n)} + \delta^{(n+1)}$, gdzie $\hat{\beta}^{(n)}$ to wartość $\hat{\beta}$ w n-tym kroku, a $\delta^{(n+1)}$ spełnia warunek

$$\left(X^T D^{(n)} X\right) \delta^{(n+1)} = X^T U^{(n)}. \tag{1.43}$$

W powyższym wzorze $D^{(n)}$ jest macierzą diagonalną $N \times N$ mającą na przekątnych $-\frac{\partial^2 g^i}{\partial \hat{\mu}_i^{(n)2}}\left(\hat{z}_i^{(n)}\right)$ $(i = 1, \ldots, N)$, a $U^{(n)}$ to wektor $\left(\frac{\partial g^1}{\partial \hat{\mu}_1^{(n)}}\left(\hat{z}_i^{(n)}\right), \ldots, \frac{\partial g^N}{\partial \hat{\mu}_N^{(n)}}\left(\hat{z}_i^{(n)}\right)\right)^T$, $\left(\hat{z}_i^{(n)}\right) = \frac{y_i - x_i^T \hat{\beta}^{(n)}}{\sigma}$.

Zakłada się, że zachodzi $X\hat{\beta}^{(n)} = \hat{\mu}^{(n)}$, a stąd wynika równość

$$\left(X^T D^{(n)} X\right)\left(\hat{\beta}^{(n)} + \delta^{(n+1)}\right) = X^T D^{(n)} \hat{\mu}^{(n)} + X^T U^{(n)} = \left(X^T D^{(n)}\right)\left(\hat{\mu}^{(n)} + D^{(n)^{-1}} U^{(n)}\right). \tag{1.44}$$

Powyższa procedura (przy stałym σ) jest równoważna iterowanej ważonej metodzie najmniejszych kwadratów, dla której

$$\hat{\beta}^{(n+1)} = \left(X^T W^{(n)} X\right)^{-1} W^{(n)} h^{(n)}, \tag{1.45}$$

dla $W^{(n)}$ macierzy wag, w tym przypadku również $D^{(n)}$, oraz zmiennej objaśnianej równej $h^{(n)} = \hat{\mu}^{(n)} + D^{(n)^{-1}} U^{(n)}$. Przy granicy oczekuje się wyniku bliskiego $\hat{\mu} = y$, więc zazwyczaj y jest dobrym estymatorem dla $\hat{\mu}^{(0)}$ jako początkowa wartość dla iteracji.[2]

Zauważmy, że gdy nie ma obserwacji cenzurowanych i $\hat{\mu}^{(0)} = y$ to $z^{(0)} = \frac{y - X^T \hat{\beta}^{(0)}}{\sigma} = 0$ oraz dla rozkładów z modą w zerze[3] $U^{(i)} = 0$ $(i = 0, 1, \ldots)$. Z 1.43 mamy $\hat{\delta}^{(i)} = 0$, więc również $D^{(i)} = a = \text{const.}$ $(i = 0, 1, \ldots)$ oraz

$$\hat{\beta} = \left(X^T (aI) X\right)^{-1} X^T (aI) \hat{\mu}^{(0)} = \left(X^T X\right)^{-1} X^T y, \tag{1.46}$$

[2]W przypadku, gdy obserwacja l jest cenzurowana przedziałowo y_l to środek przedziału.

[3]Taką własnością cechuje się większość rozkładów używanych w analizie przeżycia, m. in. rozkład wykładniczy i rozkład Weibulla o $\lambda = 1$.

czyli $\hat{\beta}$ uzyskuje się metodą najmniejszych kwadratów.

Jeśli chodzi o estymację parametru σ, autorzy pakietu *survival* [13] zdecydowali się na liczenie pochodnych po $\log(\sigma)$. Zadanie optymalizacji z ograniczeniami: $\sigma > 0$, zamienia się wtedy na szukanie optimum bez ograniczeń: $\log(\sigma) \in \mathbb{R}$, co upraszcza obliczenia. Pochodne logarytmu funkcji wiarygodności po $\log(\sigma)$, jak widać poniżej, to w większości przeskalowane przez σ pochodne logarytmu funkcji wiarygodności po σ:

$$\frac{\partial l(\beta,\sigma)}{\partial \log(\sigma)} = \sigma \frac{\partial l(\beta,\sigma)}{\partial \sigma}, \tag{1.47}$$

$$\frac{\partial^2 l(\beta,\sigma)}{\partial (\log(\sigma))^2} = \sigma^2 \frac{\partial^2 l(\beta,\sigma)}{\partial \sigma^2} + \sigma \frac{\partial l(\beta,\sigma)}{\partial \sigma}, \tag{1.48}$$

$$\frac{\partial^2 l(\beta,\sigma)}{\partial \mu \partial \log(\sigma)} = \sigma \frac{\partial^2 l(\beta,\sigma)}{\partial \mu \partial \sigma}. \tag{1.49}$$

Wypisane wyżej Hessiany i macierze pierwszych pochodnych służą do iteracyjnego szukania maksimum logarytmu funkcji wiarygodności przy użyciu algorytmu Newtona-Raphsona czy algorytmu *Fisher scoring*.

Estymacja parametrów modelu dla rozkładu wykładniczego

W tej części rozpatrywany jest najprostszy model parametryczny, czyli model z rozkładem wykładniczym i jednakowym parametrem λ dla wszystkich obserwacji. Dopuszczając cenzurowanie prawostronne wyznaczam estymator λ maksymalizujący funkcję wiarygodności, jego wariancję oraz przedziały ufności dla kwantyli rozkładu. Wyprowadzenia oparłam na [16], gdzie można znaleźć podobne wyniki dla bardziej skomplikowanych przypadków, takich jak m. in. modele z rozkładem Weibulla czy model dla dwóch podprób pochodzących z różnych rozkładów wykładniczych. Na mocy 1.39 mamy

$$L = \prod_{i=1}^{N} \left(f(y_i)\right)^{c_i} \left(S(y_i)\right)^{1-c_i}, \tag{1.50}$$

gdzie f, S to odpowiednio gęstość i funkcja przeżycia dla rozkładu wykładniczego, a c_i to zmienna binarna przyjmująca wartość 1, gdy obserwacja i nie była cenzurowana, 0 w.p.p. Mamy zatem

$$L = \prod_{i=1}^{N} \left(\lambda \exp(-\lambda y_i)\right)^{c_i} \left(\exp(-\lambda y_i)\right)^{1-c_i} = \prod_{i=1}^{N} (\lambda)^{c_i} \exp(-\lambda y_i). \tag{1.51}$$

Logarytm funkcji wiarygodności jest więc ostatecznie postaci

$$l = \sum_{i=1}^{N} c_i \log(\lambda) - \lambda \sum_{i=1}^{N} y_i = d \log(\lambda) - \lambda \sum_{i=1}^{N} y_i, \tag{1.52}$$

dla d - liczby obserwacji cenzurowanych (prawostronnie). Pochodna l po λ jest postaci

$$\frac{\partial l}{\partial \lambda} = \frac{d}{\lambda} - \sum_{i=1}^{N} y_i, \tag{1.53}$$

a przyrównana do zera daje estymator największej wiarygodności wyrażony jako

$$\hat{\lambda} = \frac{d}{\sum_{i=1}^{N} y_i}. \tag{1.54}$$

Parametr λ w rozkładzie wykładniczym jest więc szacowany procentem cenzurowanych obserwacji. Asymptotycznie wariacja może być przybliżana jako odwrotność informacji Fishera, czyli przy

$$\frac{\partial^2 l}{\partial \lambda^2} = -\frac{d}{\lambda^2} \tag{1.55}$$

dostajemy

$$V(\hat{\lambda}) = \frac{\hat{\lambda}^2}{d}. \tag{1.56}$$

Mając wariancję $\hat{\lambda}$ możemy znajdować wariancję interesujących na funkcji od λ, takich jak mediana, kwantyle, czy funkcja przeżycia. Na przykład, stosując metodę „delta"[4] uzyskujemy oszacowanie wariancji p-tego kwantyla wyrażoną jako

$$V(\hat{t}_p) \approx \left(\frac{1}{\hat{\lambda}^2}\log(1-p)\right)^2 V(\hat{\lambda}). \tag{1.57}$$

Korzystając z własności $S(t_p) = 1 - p \Rightarrow t_p = -\frac{\log(1-p)}{\lambda}$ oraz równości 1.56 dostajemy

$$V(\hat{t}_p) \approx \frac{(\hat{t}_p)^2}{d}. \tag{1.58}$$

Do konstrukcji przedziałów ufności najlepiej używać logarytmów z kwantyli, stąd ponownie stosując metodę „delta" mamy

$$V\left(\log(\hat{t}_p)\right) \approx \frac{V(\hat{t}_p)}{(t_p)^2} \approx \frac{1}{d}. \tag{1.59}$$

Ostatecznie otrzymujemy przedziały ufności dla t_p przy poziomie ufności $1-\alpha$ postaci

$$\exp\left(\log(t_p) \pm \frac{z_{\alpha/2}}{d}\right) \text{ lub } \hat{t}_p \exp\left(\pm\frac{z_{\alpha/2}}{d}\right). \tag{1.60}$$

1.6. Nieparametryczny model Cox'a

U podstaw modelu Cox'a leży założenie proporcjonalnego hazardu, zadane wzorem 1.31.

Zauważmy, że hazard z definicji to prawdopodobieństwo zdarzenia w chwili t, pod warunkiem, że zdarzenie nie nastąpiło do chwili t

$$h(t) = \lim_{\triangle t \to 0} \frac{\mathbb{P}(t \leqslant T < t + \triangle t \,|\, T \geqslant t)}{\triangle t} = \tag{1.61}$$

$$= \lim_{\triangle t \to 0} \frac{\mathcal{F}(t + \Delta t \,|\, T \geqslant t) - \mathcal{F}(t \,|\, T \geqslant t)}{\Delta t} = \mathbb{P}(T = t \,|\, T \geqslant t).$$

Zatem, gdy dla $h_i(t) = h_0(t)\exp(\beta^T x_i)$, $i = 1, 2, \ldots, N$ (równość 1.31) dany jest estymator parametru β można powiedzieć, o ile bardziej (mniej) prawdopodobne jest badane zdarzenie przy zmianie o Δx zmiennych objaśniających. Dla pełniejszego zrozumienia posłużmy się poniższym przykładem: Modelujemy długość życia pacjentów chorych na raka. Jedną ze zmiennych

[4] Niekoniecznie dla rozkładów normalnych, ale np. przy „małej" wariancji, w metodzie „delta" używa się aproksymacji $V\left(h(\hat{\beta})\right) \approx \sum_i \left(\frac{\partial h}{\partial \hat{\beta}_i}\right)^2 \mathrm{Var}(\hat{\beta}_i) + \sum_i \sum_{i \neq j} \left(\frac{\partial h}{\partial \hat{\beta}_i}\right)\left(\frac{\partial h}{\partial \hat{\beta}_j}\right) \mathrm{Cov}(\hat{\beta}_i, \hat{\beta}_j)$, czyli dla β jednowymiarowego $V\left(h(\hat{\beta})\right) \approx \left(\frac{\partial h}{\partial \hat{\beta}}\right)^2 \mathrm{Var}(\hat{\beta})$. Na podstawie [5].

objaśniających jest zmienna zero-jedynkowa: „rak w wywiadzie rodzinnym", przyjmująca wartość 1, gdy ktoś z najbliższej rodziny badanego był chory na raka, 0 w p.p.. Niech $\beta_i > 0$ będzie parametrem przy tej zmiennej objaśniającej w funkcji hazardu. Wtedy szansa śmierci osób, w których rodzinie ktoś chorował na raka jest $\frac{h_0(t)\exp(\beta^T x_2)}{h_0(t)\exp(\beta^T x_1)} = \exp(\beta_i)$ razy większa ($x_2 = (x^1, x^2, \ldots, x^{i-1}, 1, x^{i+1}, \ldots, x^p)^{\mathrm{T}}$, $x_2 = (x^1, x^2, \ldots, x^{i-1}, 0, x^{i+1}, \ldots, x^p)^{\mathrm{T}}$) niż u pozostałych osób.

Funkcja hazardu jest eksponencjalną funkcją zmiennych objaśniających, nieznana jest natomiast postać funkcji bazowej, co bez dalszych założeń uniemożliwia estymację standardową metodą największej wiarygodności.

Rozwiązaniem Cox'a jest maksymalizacja tylko tego fragmentu funkcji wiarygodności, który zależy od estymowanych parametrów. Niech $Z(t)$ oznacza zbiór indeksów dla tych obserwacji, dla których zdarzenie wystąpiło w chwili t (rozważany model jest modelem z czasem ciągłym, więc $Z(t)$ nie może posiadać więcej niż jednego indeksu - $\mathbb{P}\left(\mathrm{moc}(Z(t)) > 1\right) = 0$), $R(t)$ - zbiór indeksów tych obserwacji dla których możliwe jest wystąpienie zdarzenia w chwili t (zbiór indeksów obserwacji w stanie ryzyka). Rozważmy prawdopodobieństwo zaobserwowania zdarzenia dla zadanego wektora parametrów. Ze wzoru na prawdopodobieństwo warunkowe można je rozbić na dwa człony

$$\mathbb{P}(i \in Z(t)) = \mathbb{P}(\mathrm{moc}(Z(t)) > 0)\mathbb{P}(i \in Z(t) \mid \mathrm{moc}(Z(t)) > 0). \tag{1.62}$$

Mówiąc obrazowo $i \in Z(t)$ oznacza, że w chwili t wystąpiło zdarzenie dla obserwacji i, a warunek $(\mathrm{moc}(Z(t)) > 0)$ jest równoważny temu, że w chwili t wystąpiło jakieś zdarzenie. Pierwszy człon nie zależy od parametrów, więc procedurze maksymalizacji poddawany jest tylko drugi człon. Dalej ze wzoru na prawdopodobieństwo warunkowe mamy

$$\mathbb{P}(i \in Z(t)|\ \mathrm{moc}(Z(t)) > 0) = \frac{\mathbb{P}\left(i \in Z(t) \text{ oraz } \ \mathrm{moc}(Z(t)) > 0\right)}{\mathbb{P}(\ \mathrm{moc}(Z(t)) > 0)} = \tag{1.63}$$

$$\frac{\mathbb{P}(T_i = t|T >= t)}{\sum_{j \in R(t)} \mathbb{P}(T_j = t|T >= t)} = \frac{h_0(t)\exp(\beta^T x_i)}{\sum_{j \in R(t)} h_0(t)\exp(\beta^T x_j)}.$$

Jako, że $h_0(t)$ jest w liczniku i mianowniku ostateczne otrzymujemy wyrażenie

$$\mathbb{P}(i \in Z(t)|\ \mathrm{moc}(Z(t)) > 0) = \frac{\exp(\beta^T x_i)}{\sum_{j \in R(t)} \exp(\beta^T x_j)}, \tag{1.64}$$

które nie zależy od hazardu bazowego.

Przy wyprowadzaniu powyższych wzorów zakłada się, że czas jest ciągły. Implikuje to, iż prawdopodobieństwo, że dla dwóch obiektów zdarzenie nastąpi w tej samej chwili wynosi zero. W praktyce może być jednak inaczej. Często pomiary prowadzi się w ograniczonych - dyskretnych momentach czasowych i dla każdego z czasów $t = 0, 1, \ldots, T$ obserwuje się wiele zdarzeń oraz cenzurowanie.

Jednym ze sposobów radzenia sobie z tym problemem jest aproksymacja Breslowa. Jest to najprostsze podejście i najmniej złożone obliczeniowo [4]. Zakłada się w nim, że jeśli zdarzenie i cenzurowanie wystąpiły w tym samym czasie, to zdarzenie poprzedza cenzurowanie. Załóżmy, że w chwili t wystąpiło d zdarzeń, a w zbiorze ryzyka jest m obserwacji. Niech $\tau_i = I(T_i \geqslant t)\exp(\beta^T x_i)$. Przenumerujmy obserwacje, tak że obserwacje, dla których w chwili t wystąpiło zdarzenie mają numery $1, 2, \ldots, d$. Aproksymacja funkcji częściowej wiarygodności w tym przypadku jest postaci

$$L \propto \prod_{i=1}^{d} \frac{\tau_i}{\sum_{j=1}^{m} \tau_j}. \tag{1.65}$$

Inne rozwiązanie zaproponował Efron. W uproszczeniu zakłada ono, że jeśli dwa zdarzenia - i oraz j - wystąpiły jednocześnie, zdarzenie i znajduje się w stanie ryzyka dla zdarzenia j (i na odwrót) ale z odpowiednio dobraną wagą. Rozwiązanie Efron'a jest tak samo skomplikowane obliczeniowo jak Breslowa [14]. Gdy stosunek liczby zdarzeń do liczby obserwacji w stanie ryzyka nie jest zbyt duży, stanowi za to lepszą aproksymację

$$L \propto \prod_{i=1}^{d} \frac{\tau_i}{\sum_{j=1}^{m} \tau_j - \frac{i-1}{d} \sum_{j=1}^{d} \tau_j}. \tag{1.66}$$

Dla czasu dyskretnego można wykorzystać aproksymację funkcji wiarygodności najbardziej zbliżoną do prawdziwej funkcji wiarygodności

$$L \propto \frac{\tau_1 \tau_2 \dots \tau_d}{\sum_{S(d,m)} \tau_{k_1} \tau_{k_2} \dots \tau_{k_d}}. \tag{1.67}$$

Suma w mianowniku odbywa się po wszystkich różnych kombinacjach d-elementowych z m-elementowego zbioru elementów w stanie ryzyka. Zastosowanie tego wzoru w estymacji wymaga jednak dużo większej mocy obliczeniowej.

W celu lepszego wyjaśnienia rozważmy poniższy przykład: Niech 5 obserwacji znajduje się w stanie ryzyka w chwili t, a dla obserwacji 1 i 2 w chwili t wystąpiło badane zdarzenie. Wtedy aproksymacja Breslowa będzie postaci

$$\frac{\tau_1}{\tau_1 + \tau_2 + \tau_3 + \tau_4 + \tau_5} * \frac{\tau_2}{\tau_1 + \tau_2 + \tau_3 + \tau_4 + \tau_5}, \tag{1.68}$$

czyli obserwacja 2 będzie w stanie ryzyka dla obserwacji 1, jak i obserwacja 1 będzie w stanie ryzyka dla obserwacji 2. Przy użyciu aproksymacji Efon'a sytuacja będzie przedstawiać się następująco:

$$\frac{\tau_1}{\tau_1 + \tau_2 + \tau_3 + \tau_4 + \tau_5} * \frac{\tau_2}{0.5\tau_1 + 0.5\tau_2 + \tau_3 + \tau_4 + \tau_5}. \tag{1.69}$$

Mnożniki w mianowniku przy τ_1 i τ_2 w drugiej części wzoru można interpretować tak: obserwacje 1 i 2 mają 0.5 szansy na znalezienie się w stanie ryzyka dla drugiego mianownika. Przy zastosowaniu właściwej funkcji wiarygodności dostaniemy

$$\frac{\tau_1 \tau_2}{\tau_1\tau_2 + \tau_1\tau_3 + \tau_1\tau_4 + \tau_1\tau_5 + \tau_2\tau_3 + \tau_2\tau_4 + \tau_2\tau_5 + \tau_3\tau_4 + \tau_3\tau_5 + \tau_4\tau_5}. \tag{1.70}$$

W tym przypadku mianownik składa się z 10 członów - wszystkie możliwe podgrupy dwuelementowe zbioru 5-elementowego. Gdyby jednak rozważyć w danej chwili 10 zdarzeń w zbiorze ryzyka składającym się z 1000 elementów, mianownik w powyższym wyrażeniu zawierałby $2.6 * 10^{23}$ składników! Wpływa to znacznie na złożoność obliczeń.

Do estymacji modelu Cox'a stosuje się metodę największej wiarygodności - maksymalizując iteracyjnie funkcję częściowej wiarygodności (lub którąś z jej aproksymacji).

W programie R (pakiet *survival*) zaimplementowana jest funkcja *coxph*() służąca do estymacji modelu Cox'a. Domyślnie przeprowadza ona maksymalizację przy użyciu aproksymacji Breslowa.

1.7. Diagnostyka w modelu Cox'a

Najbardziej naturalnym sposobem liczenia residuów byłoby policzenie różnicy między rzeczywistym czasem do zaobserwowania zdarzenia a tym wynikającym z modelu, osobno dla każdej obserwacji. Pojawia się tu jednak problem obserwacji cenzurowanych, dla których dokładny czas

do zaobserwowania zdarzenia jest nieznany. Zdarza się, że te obserwacje stanowią przeważającą część zbioru danych. W tym podrozdziale omówionych jest kilka najważniejszych rodzajów residuów występujących w analizie przeżycia, które na różne sposoby radzą sobie z brakiem informacji wynikającym z cenzurowania danych.

1.7.1. Residua Cox'a-Snella

Residua Cox'a-Snella dane są wzorem

$$rc_i = \exp(\beta^T x_i)\hat{\Lambda}_0(y_i) = -\log\left(\hat{S}_i(t)\right). \tag{1.71}$$

Załóżmy na początek, że nie występuje cenzurowanie. Niech T będzie zmienną losową oznaczającą czas do zajścia badanego zdarzenia z funkcją gęstości postaci $f_T(t)$. Funkcja gęstości dla zmiennej $\Lambda(T) = -\log(S(T))$ zadana jest przez[5]

$$f_{\Lambda(t)}(\Lambda(T)) = f_T\left(S^{-1}\left(\exp\left(-\Lambda(T)\right)\right)\right) \| \frac{d\left(-\log\left(S(t)\right)\right)}{dt} \|^{-1} \tag{1.72}$$

$$= f_T\left(S^{-1}\left(\exp\left(-\Lambda(T)\right)\right)\right) \frac{S(t)}{f_T(t)}$$

$$= f_T\left(S^{-1}\left(\exp\left(-\Lambda(T)\right)\right)\right) \frac{S\left(S^{-1}\left(\exp\left(-\Lambda(t)\right)\right)\right)}{f_T\left(S^{-1}\left(\exp\left(-\Lambda(T)\right)\right)\right)} = \exp(-\Lambda(t)).$$

Zatem $\Lambda(T)$ ma rozkład wykładniczy z $\lambda = 1$ bez względu na postać $S(t)$.

Jeśli model jest właściwy, $\hat{S}_i(t)$ powinno mieć rozkład zbliżony do $S_i(t)$. Stąd dalej, jeśli spełnione jest $rc_i = -\log\left(\hat{S}_i(t)\right)$, rc_i powinno być zbliżone do rozkładu wykładniczego.

Gdy mamy do czynienia z cenzurowaniem prawostronnym prawdziwy czas do zajścia zdarzenia jest większy niż zaobserwowana wartość y_i. Stąd również residuum w modelu bez cenzurowania powinno być większe niż uzyskane z 1.71 (ponieważ $\Lambda_0(y^1) > \Lambda_0(y^2)$, gdy $y^1 > y^2$). Residua dla właściwego modelu odpowiadają więc cenzurowanej prawostronnie próbie z rozkładu wykładniczego.

By uchwycić efekt cenzurowania, jako residuów można użyć, zaproponowanej przez Cox'a i Snell'a w [1], następującej modyfikacji wzoru 1.71:

$$rmc_i = rc_i + (1 - \delta_i)\Delta, \tag{1.73}$$

gdzie δ_i zmienna binarna oznaczająca brak cenzurowania ($\delta_i = 0$ - i-ta obserwacja była cenzurowana). Pozostaje jednak problem doboru Λ. Korzystając z własności braku pamięci w modelu wykładniczym, Λ, podobnie jak rc_i, pochodzi z rozkładu wykładniczego o $\lambda = 1$, stąd proponowane wartości to $E(\Lambda) = 1$ lub mediana rozkładu Λ równa $\log(2) = 0.693$.

Uzyskane tym sposobem residua przyjmują tylko wartości większe od zera (bo $\Delta(t) > 0$), nie są więc symetrycznie rozłożone wokół zera jak residua standardowych modeli linowych. Ponadto cechują się skośnością rozkładu (własność rozkładu wykładniczego).

[5]Korzystam ze wzoru na zmianę funkcji gęstości: Dla monotonicznej funkcji g, takiej że $g(X) = Y$, rozkład X - f_X - niezależny, zachodzi $f_y(y) = \left(g'(g^{-1}(y))\right)^{-1} f_x(g^{-1}(y))$. W tym przypadku $X = T$ oraz $g(T) = -\log(S(T))$, źródło: [2].

1.7.2. Residua martyngałowe

Residua martyngałowe (ang. *martingale residuals*) są funkcją residuów Cox'a-Snell'a i wyrażają się jako (oznaczenia jak poprzednio)

$$rm_i = \delta_i - rc_i. \tag{1.74}$$

Ponieważ $rc_i > 0$ to $-\infty \leqslant rm_i \leqslant 1$, a obserwacje cenzurowane mają wartości mniejsze od zera. Można pokazać, że residua martyngałowe są nieskorelowane i mają średnia równą zero. Nie są jednak symetryczne wokół zera.

Residua te można interpretować jako różnicę między zaobserwowaną liczbą zdarzeń równą δ_i, a oczekiwaną (wynikającą z modelu) - rc_i.

1.7.3. Residua deviance

W [15] można znaleźć kolejną propozycję residuów zwanych po angielsku *deviance residuals* i zadanych jako

$$rd_i = \text{sign}(rm_i)\sqrt{-2(rm_i + \delta_i \log(\delta_i - rm_i))}. \tag{1.75}$$

Residua te powiązane są ze sposobem badania dopasowania modelu opartym na porównywaniu funkcji wiarygodności. Rozważmy różnicę między logarytmami funkcji wiarygodności dla dwóch modeli, gdzie jeden jest prawdziwym modelem, a drugi się w nim zawiera, zwanej po angielsku *deviance*

$$\text{deviance} = -2(\log L_2 - \log L_1), \tag{1.76}$$

gdzie model 2 (L_2) zawiera się w prawdziwym modelu 1 (L_1). Sformułowanie „model 2 zawiera się w modelu 1" oznacza, że jeśli x_{i_1} jest zmienną objaśniającą w modelu 2 i estymowany parametr $\beta^2_{i_1}$ przy tej zmiennej jest różny od zera to zmienna ta jest też zmienną objaśniającą dla modelu 1 z niezerowym parametrem $\beta^1_{i_1}$. Inaczej wyrażenie 1.76 można zapisać jako

$$\text{deviance} = \sum_{i=1}^{N} rd_i^2. \tag{1.77}$$

W porównaniu z residuami martygałowymi, *deviance residuals* są bardziej symetryczne dookoła zera.

1.7.4. Residua Schoenfeld'a

Residua Schoenfeld'a zostały po raz pierwszy przedstawione przez Schoenfeld'a w [10]. Dla każdej obserwacji można wyznaczyć p residuów, gdzie p to liczba zmiennych objaśniających w modelu. Residuum Schoenfeld'a dla k-tej zmiennej objaśniającej i i-tej obserwacji jest postaci

$$rs_{ik} = \delta_i \left(x_{ik} - \frac{\sum_{l \in R(y_i)} x_{lk} \exp(\hat{\beta}^T x_l)}{\sum_{l \in R(y_i)} \exp(\hat{\beta}^T x_l)} \right). \tag{1.78}$$

Powyższe wyrażenie odpowiada różnicy między rzeczywistą zmienną objaśniającą a oczekiwaną (wynikającą z modelu) wartością tej zmiennej w chwili y_i.

Zauważmy, że ze względu na występowanie zmiennej δ_i we wzorze, tylko obserwacje niecenzurowane mają wartości rs_{ik} różne od zera. Ponadto wyrażenie $\sum_{i=1}^{N} rs_{ik}$ można otrzymać przez różniczkowanie logarytmu funkcji wiarygodności

$$\frac{\partial l(\beta)}{\partial \beta_k} = \sum_{i=1}^{N} \delta_i \left(x_{ik} - \frac{\sum_{l \in R(y_i)} x_{lk} \exp(\hat{\beta}^T x_l)}{\sum_{l \in R(y_i)} \exp(\hat{\beta}^T x_l)} \right). \tag{1.79}$$

Z warunku maksymalizacji funkcji częściowej wiarygodności

$$\frac{\partial l(\beta)}{\partial \beta_k}|_{\hat{\beta}} = 0, \tag{1.80}$$

stąd suma residuów Schoenfeld'a dla każdego parametru β_k $(k = 1, \ldots, p)$ wynosi 0.

Residua Schoenfeld'a służą m. in. do testowania założenia proporcjonalnego hazardu - przykład analizy opartej na residuach tej postaci znajduje się w podrozdziale 3.5.3.

Rozdział 2

Badanie własności estymatorów i testów w analizie przeżycia

Rozdział ten poświęcony jest badaniu własności estymatorów i testów w analizie przeżycia. W kręgu zainteresowania znajdują się: obciążoność estymatorów nieparametrycznych (Kaplana-Meiera i Flemingtona-Harringtona), moc testu log-rank oraz rozkład estymatorów parametrów w modelach parametrycznych. Analiza odbywa się na podstawie symulacji przy użyciu programu R. Kody funkcji zaprogramowanych do tego badania wraz z dokładnym opisem działania znajdują się w dodatku.

2.1. Badanie obciążoności estymatorów funkcji przeżycia

Celem tego podrozdziału jest zbadanie własności statystycznych estymatorów nieparametrycznych funkcji przeżycia. Przeprowadzam symulacje na obserwacjach pochodzących z rozkładów parametrycznych. Dodatkowo generowane są wartości zmiennych z innego rozkładu, który w dalszej części pracy nazywam *rozkładem cenzurującym*. Tymi wartościami cenzurowane są obserwacje, a cenzurowanie odbywa się w następujący sposób: Niech X będzie wektorem wygenerowanych obserwacji z danego rozkładu o długości równej n, zaś Y to tak samo wygenerowany wektor cenzurującym zmienne. Wtedy do modelowania używa się wektora Z: $Z[i] = \min(X[i], Y[i])$ dla $i = 1, 2, ..., n$ oraz informacji czy dana zmienna została ocenzurowana, np. wektor binarny C: $C[i] =$ TRUE, gdy $X[i] > Y[i]$ i $C[i] =$ FALSE w p.p., dla $i = 1, 2, ..., n$. Przy użyciu zaprojektowanej przez mnie funkcji *plot_unbiased_survival*() (dodatek A.1) generowane są wykresy pudełkowe rozkładu estymatorów nieparametrycznych

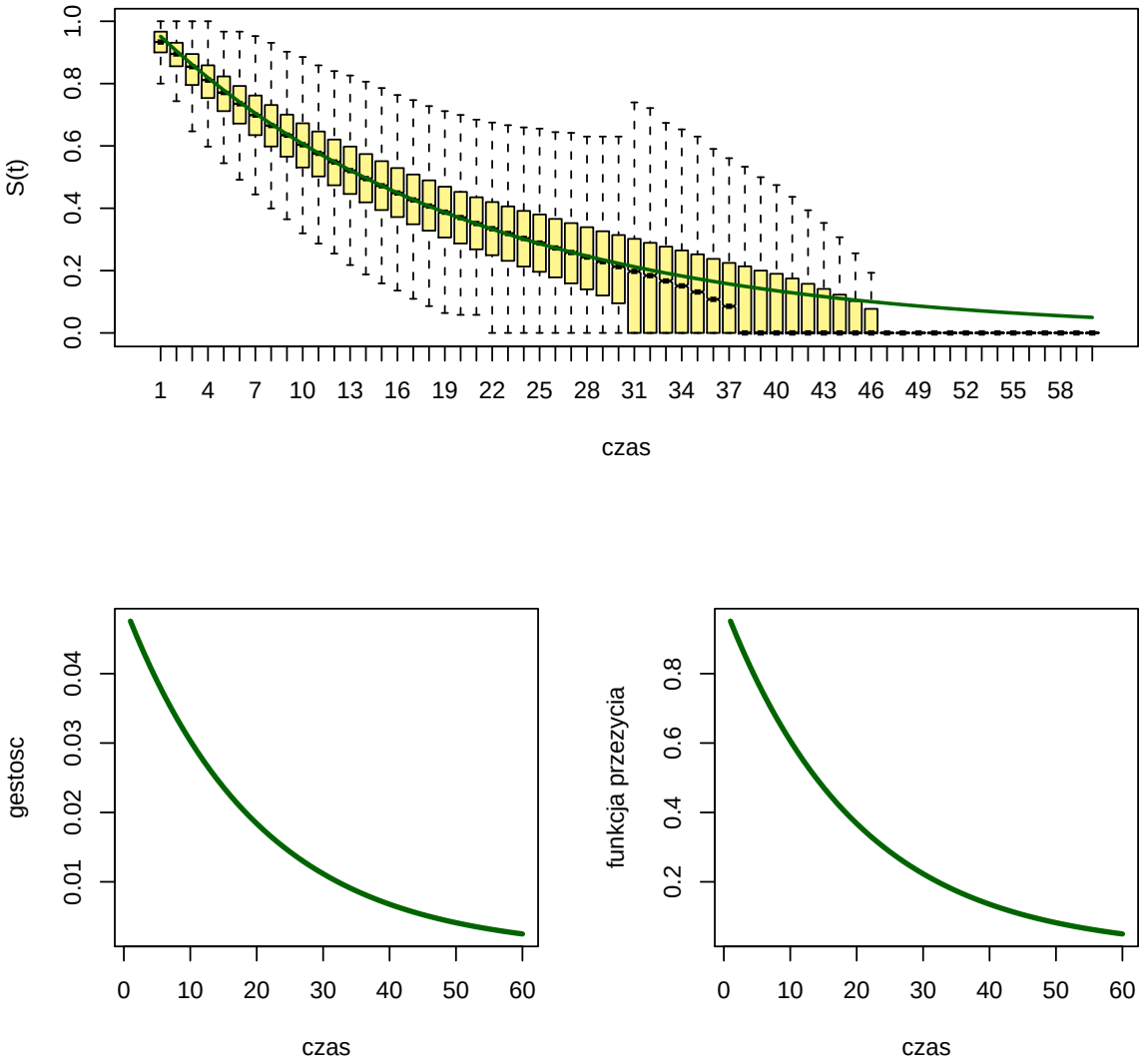
- Kaplana-Meiera,
- oraz Flemingtona-Harringtona.

Wektory obserwacji oraz zmiennych cenzurujących są 30-elementowe i pochodzą z następujących rozkładów:

- wykładniczego - Exp(0.05),
- oraz Weibulla(0.05,0.5).

```
A1=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=60)
A2=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=100, p1=0.5, p2=0.5)
A3=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=60,method="fleming-harrington")
A4=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=100,method="fleming-harrington",p1=0.5, p2=0.5)
A5=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=60, p2=0.5)
```

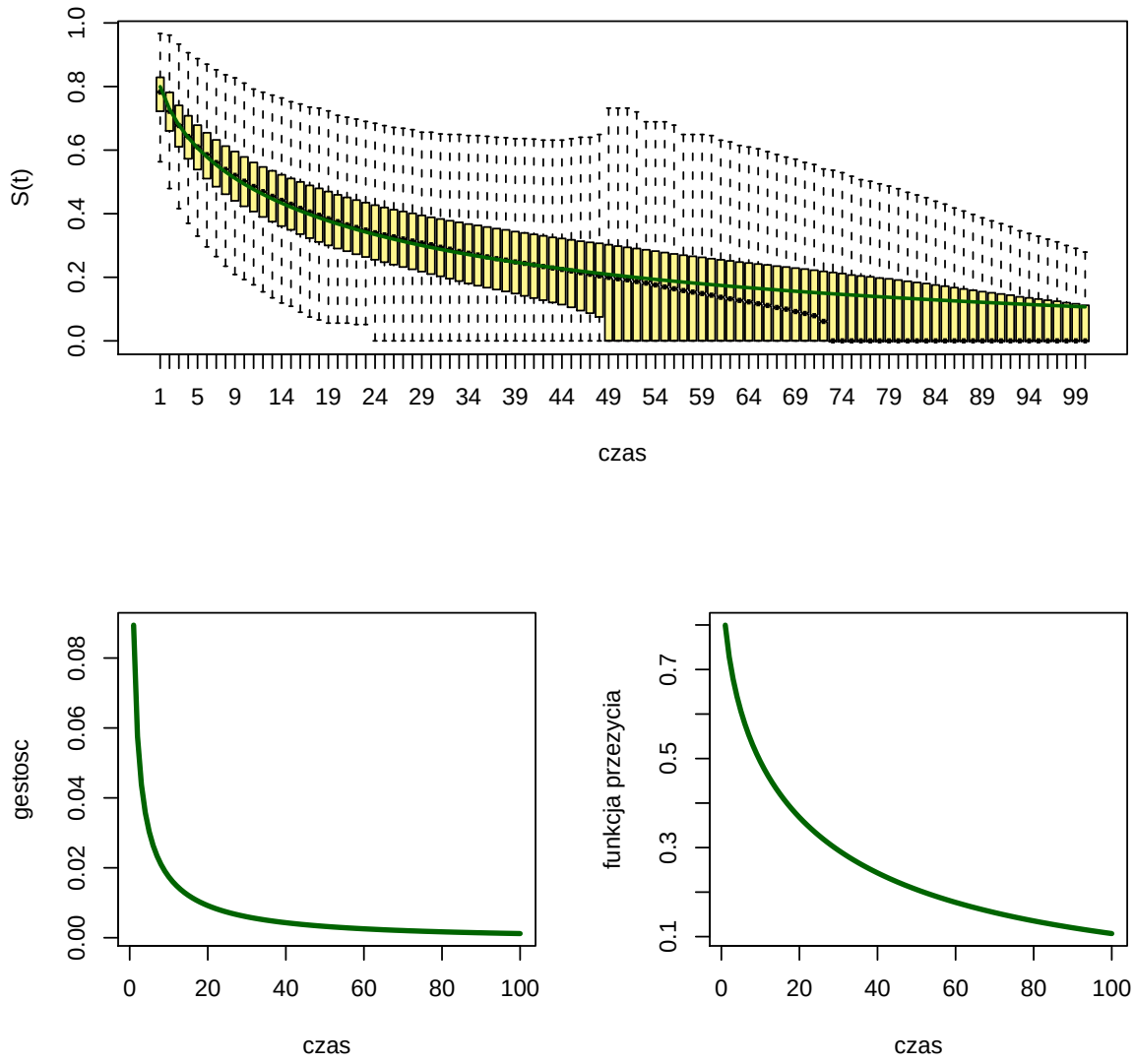
```
A6=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=60, p1=0.5)
A7=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=60, p2=0.5,method="fleming-harrington")
A8=plot_unbiased_survival(30, 0.05, 0.05, N=10000, xlab=60, p1=0.5,method="fleming-harrington")
```



Rysunek 2.1: Na górze: Estymator Kaplana-Meiera i rzeczywista funkcja przeżycia dla zmiennych z rozkładu wykładniczego Exp(0.05) z cenzurowaniem wykładniczym Exp(0.05) na podstawie 10000-krotnej symulacji, przy liczbie generowanych obserwacji równej 30. Na dole: Gęstość i funkcja przeżycia dla rozkładu Exp(0.05).

Rysunki 2.1 - 2.8 prezentują wyniki dla przeprowadzonych symulacji. Podstawowym celem badania była weryfikacja hipotezy o obciążoności estymatorów nieparametrycznych. Przy przyjętych założeniach (odnośnie generowanych rozkładów, sposobu cenzurowania) otrzymałam następujące wyniki (podsumowanie znajduje się również w tabelce 2.1):

1. Estymatory funkcji przeżycia $\hat{S}(t)$ dla dużych wartości t (od t_1) są niedoszacowane. Dla zmiennych z rozkładu wykładniczego X, t_1 w przybliżeniu wynosi $EX + \frac{3}{4}\sqrt{\mathrm{Var}(X)}$



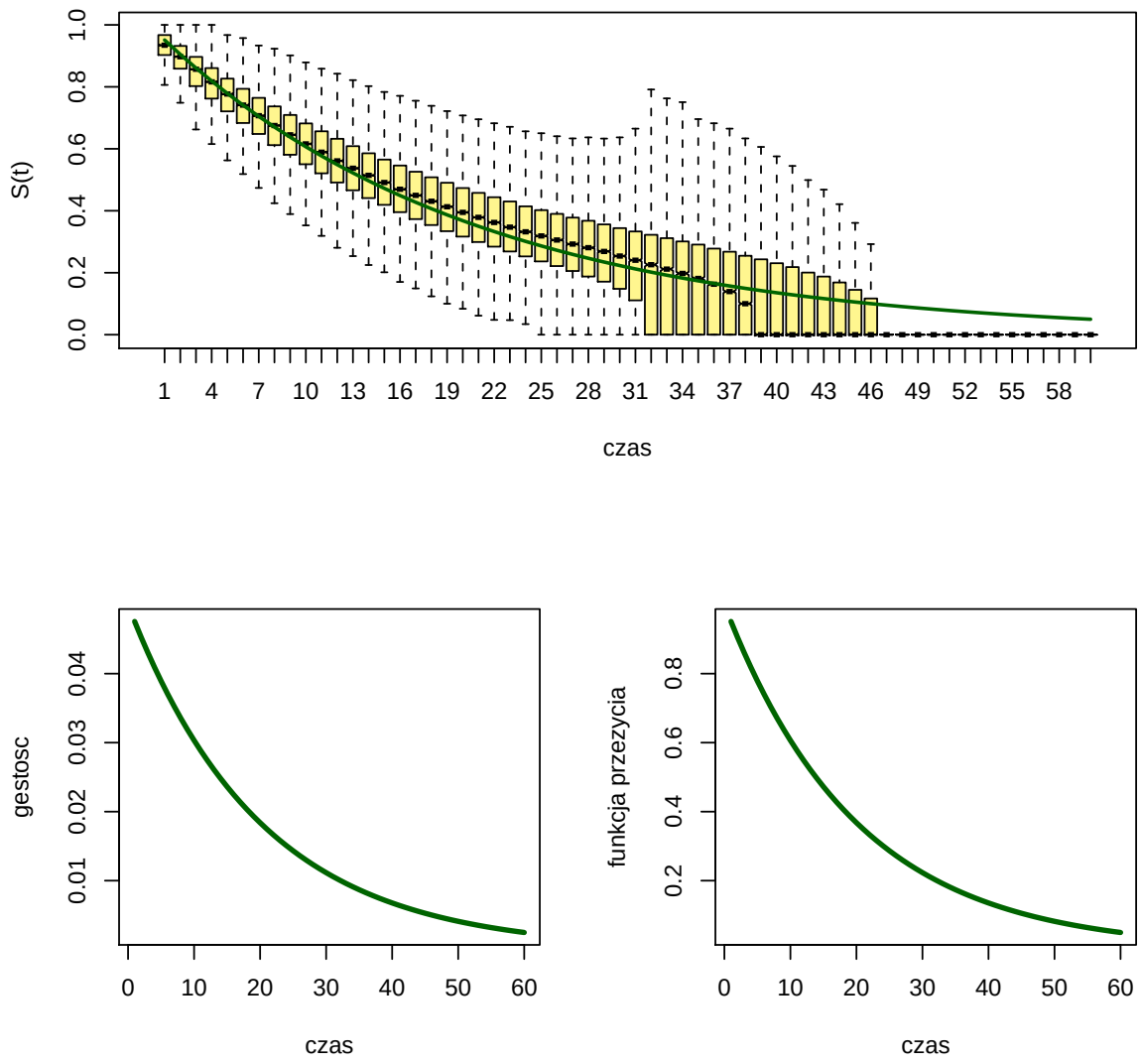
Rysunek 2.2: Na górze: Estymator Kaplana-Meiera a rzeczywista funkcja przeżycia dla zmiennych z rozkładu Weibulla z $\lambda = 0.05$ i $k = 0.5$ oraz cenzurowaniem z tego samego rozkładu, na podstawie 10000-krotnej symulacji, $n = 30$. Na dole: Gęstość i funkcja przeżycia dla rozkładu Weibull(0.05,0.5).

(dla przyjętych λ i k: $E(X) + \frac{3}{4}\sqrt{\mathrm{Var}(X)} = 35$[1]), dla rozkładu Weibulla Y ($EY = 40$, $\sqrt{Var(Y)} \approx 90$) $t_1 = cEY$, gdzie $c \in (-\frac{3}{4}, \frac{3}{2})$. Przedstawione wzory zostały dopasowane do konkretnego przypadku i nie muszą być ogólną własnością.

2. Estymator Flemingtona-Harringtona dodatkowo przeszacowuje watrości $\hat{S}(t)$ mniejsze od t_1 i większe od $t_0 \in (4, 5)$.

3. Estymator Kaplana-Meiera zachowuje się lepiej dla małych wartości t niż estymator

[1]Dla rozkładu wykładniczego $EX = \frac{1}{\lambda}$, $\mathrm{Var}(X) = \frac{1}{\lambda^2}$. Zatem dla $X \sim \mathrm{Exp}(0.05)$, $EX = 20$, $\mathrm{Var}X = 400$. Dla rozkładu Weibulla $EY = \Gamma(1 + \frac{1}{k})\frac{1}{\lambda}$, $\mathrm{Var}(Y) = \Gamma(1 + \frac{2}{k})\frac{1}{\lambda^2} - EY$.



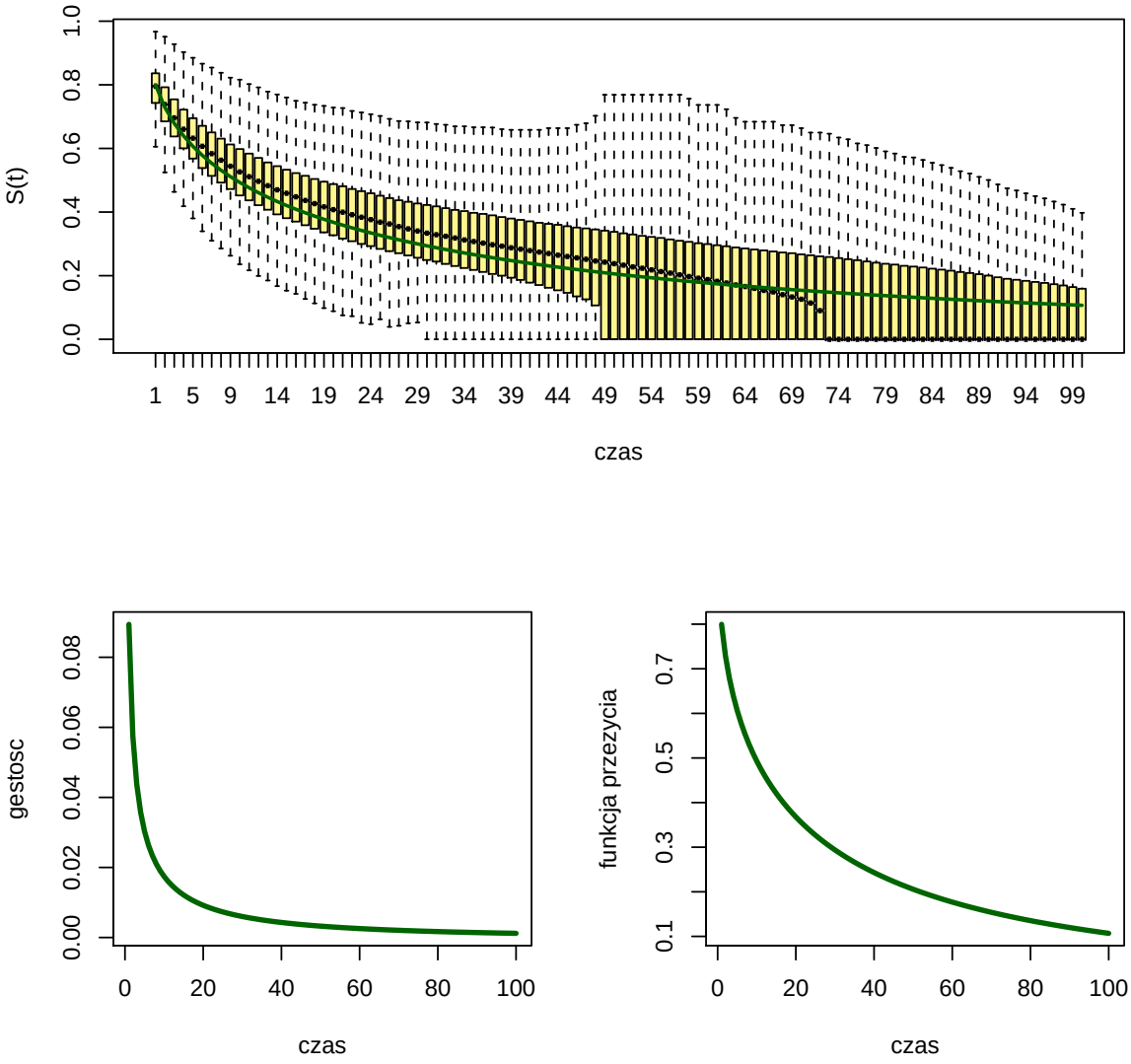
Rysunek 2.3: Na górze: Estymator Flemingtona-Harringtona i rzeczywista funkcja przeżycia dla zmiennych z rozkładu wykładniczego z $\lambda = 0.05$ oraz cenzurowaniem z tego samego rozkładu, na podstawie 10000-krotnej symulacji, $n = 30$. Na dole: Gęstość i funkcja przeżycia dla rozkładu Exp(0.05).

Flemingtona-Harringtona.

2.2. Badanie mocy testu log-rank

W tej części pracy badam moc testu log-rank dla konkretnych rozkładów, długości podprób i stopnia cenzurowania. Moc testu to prawdopodobieństwo nie popełnienia błędu drugiego rodzaju, czyli prawdopodobieństwo odrzucenia hipotezy zerowej, gdy jest ona nieprawdziwa.

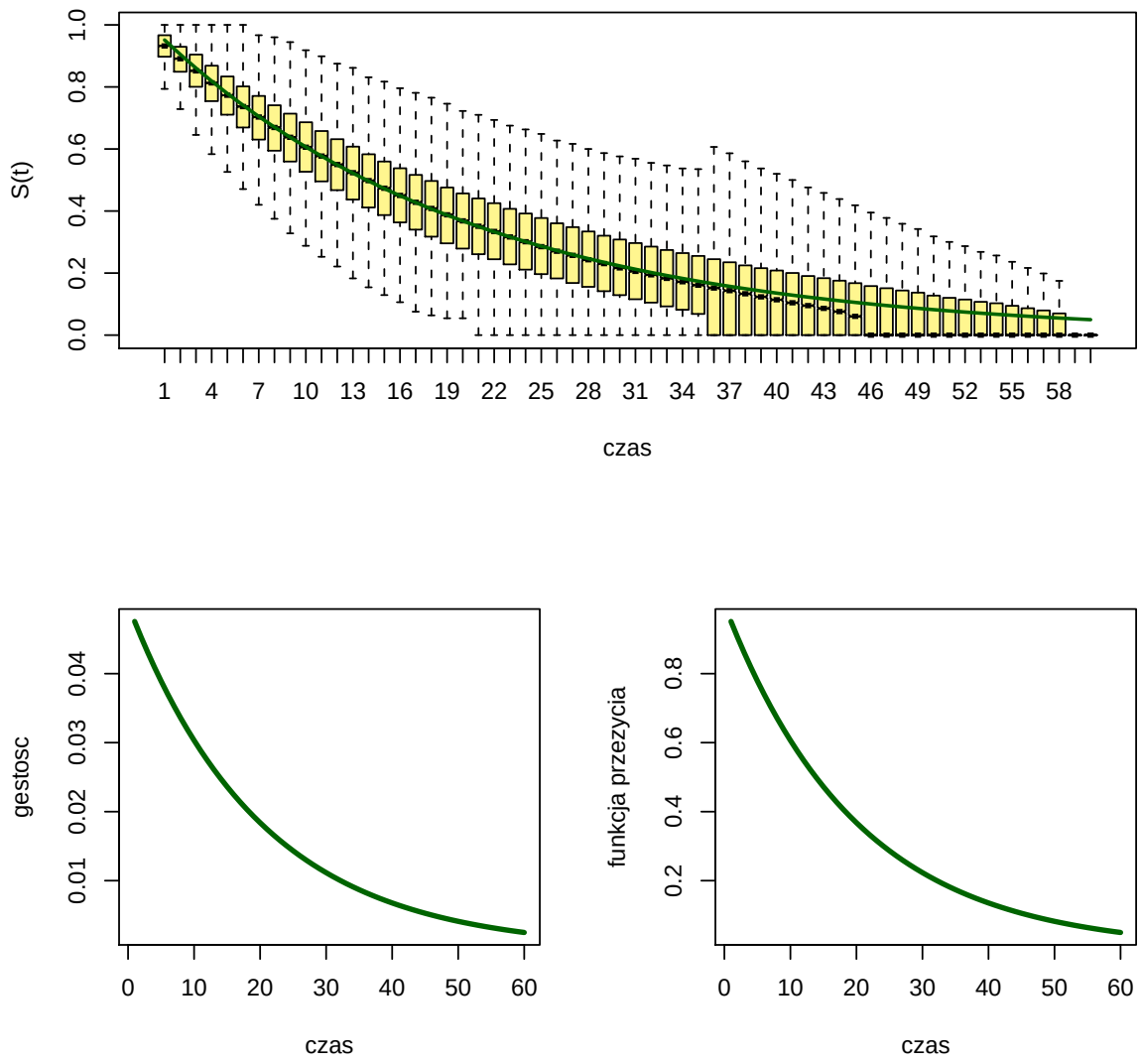
Symulacyjnie sprawdzam jak na moc testu log-rank wpływa cenzurowanie. Do badania używam zaimplementowanej przeze mnie funkcji *logrankplotdep*() (dodatek 2.2). Badanie polega



Rysunek 2.4: Na górze: Estymator Flemingtona-Harringtona i rzeczywista funkcja przeżycia dla zmiennych z rozkładu Weibulla z $\lambda = 0.05$ i $k = 0.5$ oraz cenzurowaniem z tego samego rozkładu, na podstawie 10000-krotnej symulacji, $n = 30$. Na dole: Gęstość i funkcja przeżycia dla rozkładu Weibull(0.05,0.5).

na tym, że N razy (tutaj $N = 10000$), dla każdego poziomu cenzurowania, generuje się dwie podpróby o określonej długości oraz rozkład cenzurujący (który jest zmienną losową z rozkładu Weibulla lub wykładniczego). Ponadto losowo wybiera się zmienne, które poddawane są cenzurowaniu. Wyniki testu log-rank, na poziomie istotności 0.05, dla tak wygenerowanych podprób pozwalają na obliczenie mocy testu w zależności od procentu cenzurowania danych. Symulacje (S1) przeprowadzam dla (szczegóły również w tabeli 2.2)

- podprób, które mają po 30 obserwacji,
- pochodzących z rozkładów Exp(0.03) i Exp(0.08),



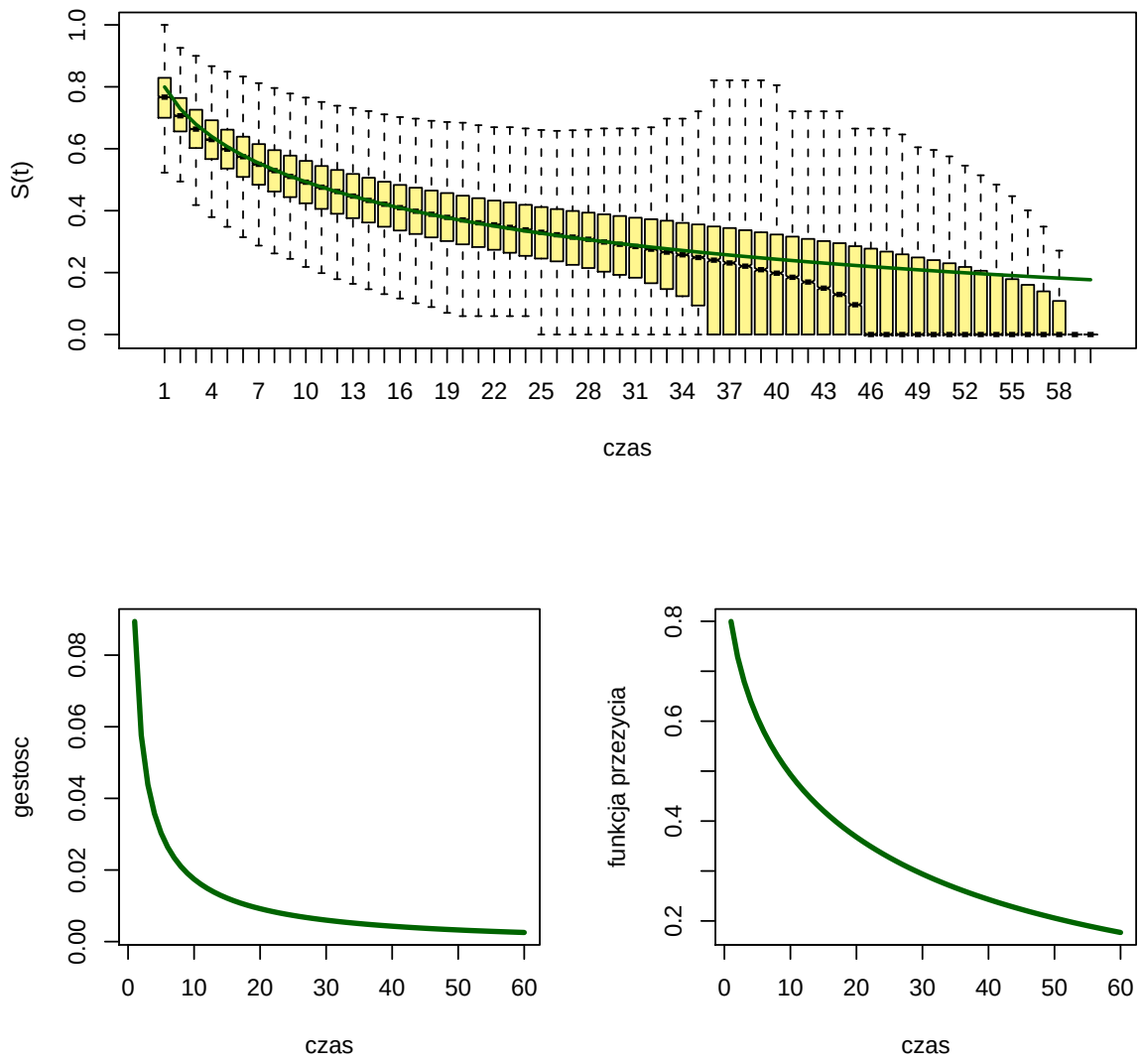
Rysunek 2.5: Na górze: Estymator Kaplana-Meiera i rzeczywista funkcja przeżycia dla zmiennych z rozkładu wykładniczego Exp(0.05) z cenzurowaniem z rozkładu Weibulla $\lambda = 0.05$, $k = 0.5$ na podstawie 10000-krotnej symulacji, przy liczbie generowanych obserwacji równej 30. Na dole: Gęstość i funkcja przeżycia dla rozkładu Exp(0.05).

- oraz rozkładu cenzurującego z Exp(0.1).

```
#S1
Logrankplotdep(30, 30, 0.03, 0.08, 0.1, m=100, NN=10000, var="censoring")
```

Wyniki znajdują się na wykresie 2.9. Wraz ze wzrostem cenzurowania w badanym zakresie parametrów, moc testu maleje w tempie -0.6[2] i dla poziomu cenzurowania $45 - 50\%$ znajduje się między $70 - 75\%$.

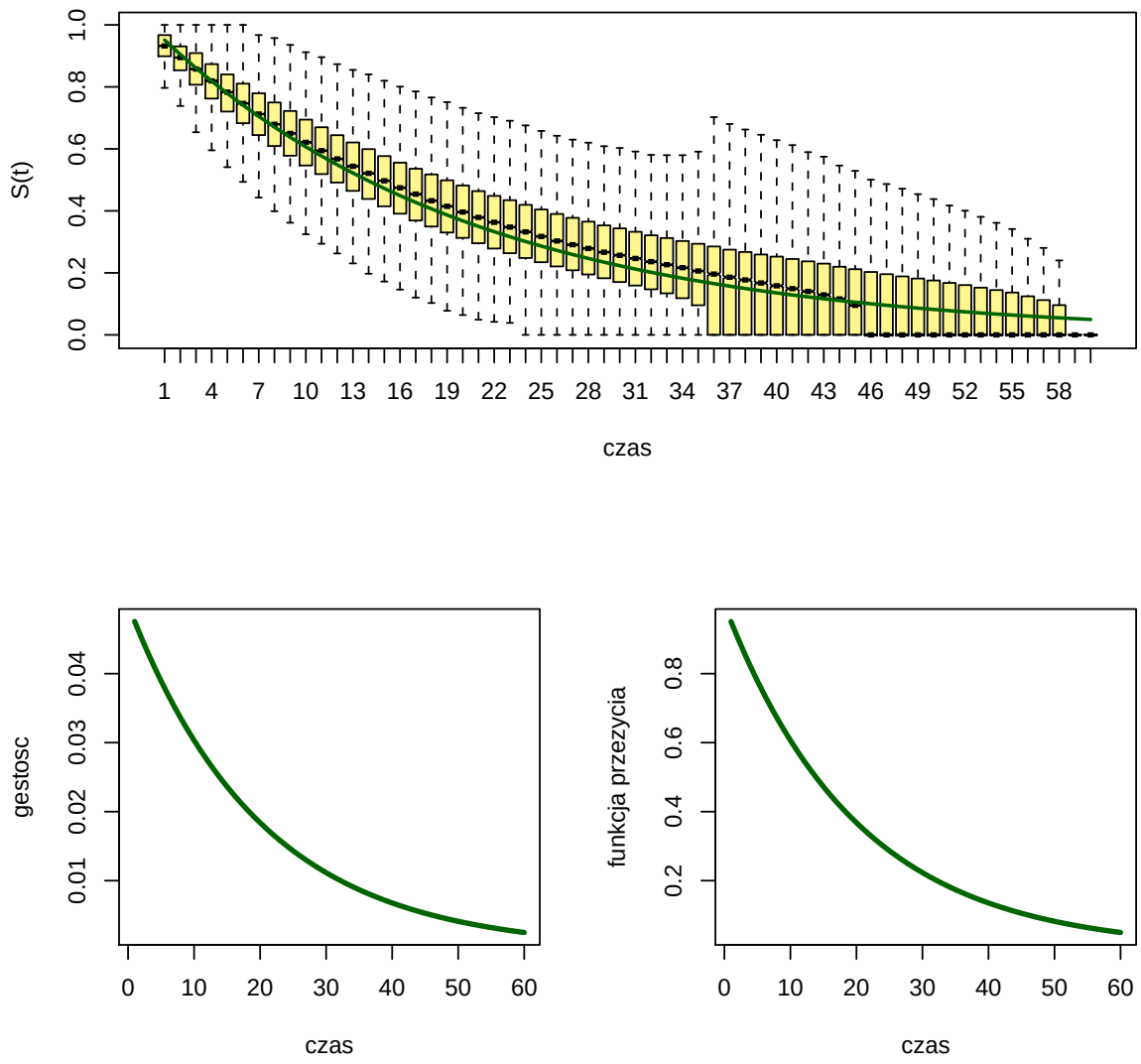
[2]Tempo zmiany (a) zostało uzyskane za pomocą regresji liniowej postaci $y = at + \text{const.}$ dla zmiennej y - moc testu, t - poziom cenzurowania.



Rysunek 2.6: Na górze: Estymator Kaplana-Meiera a rzeczywista funkcja przeżycia dla zmiennych z rozkładu Weibulla z $\lambda = 0.05$ i $k = 0.5$ z cenzurowaniem z rozkładu wykładniczego $\lambda = 0.05$, na podstawie 10000-krotnej symulacji, $n = 30$. Na dole: Gęstość i funkcja przeżycia dla rozkładu Weibull(0.05,0.5).

Interesujące wydaje się również zagadnienie, jak zmienia się moc testu log-rank w zależności od długości podprób. W funkcji *logrankplotdep*() dostępny jest parametr var="observlength", który pozwala na przeprowadzanie tego typu analiz. Generuję więc dane do symulacji S2 (patrz również tabela 2.2) o następujących własnościach:

- podpróby (równej długości) pochodzą z rozkładów Exp(0.03) i Exp(0.06),
- rozkład cenzurujący to zmienna losowa z Exp(0.02),
- długość podprób waha się między $10 - 110$ dla każdej podpróby.



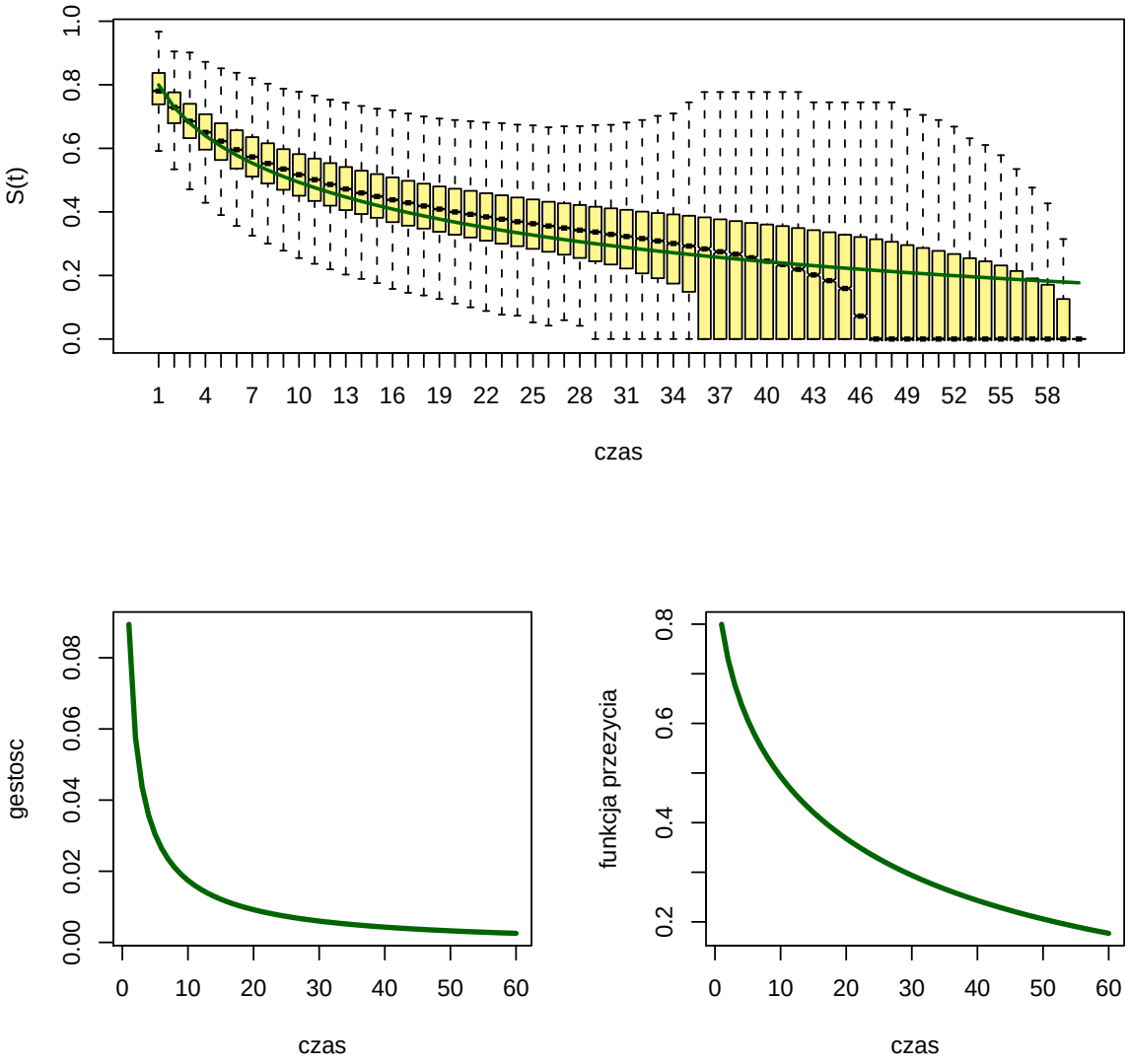
Rysunek 2.7: Na górze: Estymator Fleminga-Harringtona i rzeczywista funkcja przeżycia dla zmiennych z rozkładu wykładniczego Exp(0.05) z cenzurowaniem z rozkładu Weibulla $\lambda = 0.5$, $k = 0.5$ na podstawie 10000-krotnej symulacji, przy liczbie generowanych obserwacji równej 30. Na dole: Gęstość i funkcja przeżycia dla rozkładu Exp(0.05).

```
#S2
Logrankplotdep(10, 110, 0.03, 0.06, 0.02, m=100, NN=10000, var="observlength")
```

Rezultat można obejrzeć na rysunku 2.9. Moc testu rośnie w tempie logarytmicznym (krzywa dopasowania to $y = 35.41 \log(x) - 61.93$) i już dla podprób o długościach po 70 obserwacji osiąga 90%.

Ostatnią zależnością jaką badam jest zależność między mocą testu log-rank a różnicami w parametrze rozkładu podprób (S3). W tym przypadku podpróby pochodzą z rozkładu wykładniczego oraz

- mają po 50 obserwacji,



Rysunek 2.8: Na górze: Estymator Fleminga-Harringtona i rzeczywista funkcja przeżycia dla zmiennych z rozkładu Weibulla z $\lambda = 0.05$ i $k = 0.5$ z cenzurowaniem z rozkładu wykładniczego $\lambda = 0.05$, na podstawie 10000-krotnej symulacji, $n = 30$. Na dole: Gęstość i funkcja przeżycia dla rozkładu Weibull(0.05,0.5).

- rozkład cenzurujący to Exp(0.01),
- rozkład pierwszy to Exp(0.01), a drugi waha się od Exp(0.011) do Exp(0.1).

```
#S3
Logrankplotdep(50, 50, 0.01, 0.1, 0.01, m=100, NN=10000, var="lambdadistance")
```

Wykres dla tej symulacji znajduje się na rysunku 2.9. Dla małych różnic w parametrze λ (mniejszych od 0.02) zależność jest zbliżona do logarytmicznej (dopasowanie: $y = 38.44 \log(x) + 249.33$), a dla różnic 0.02 i większych moc testu jest w zasadzie 100%.

Rozkłady: symulowany-cenzurujący	Kaplan-Meier	Flemington-Harrington
Exp(0.05)-Exp(0.05)	niedoszacowuje dla $t > 30$	niedoszacowuje dla $t > 35$ przeszacowuje dla $t \in (5, 35)$
Exp(0.05)-Weibull(0.05,0.5)	niedoszacowuje dla $t > 35$	niedoszacowuje dla $t > 45$ przeszacowuje dla $t \in (5, 45)$
Weibull(0.05,0.5)-Exp(0.05)	niedoszacowuje dla $t > 35$	niedoszacowuje dla $t > 40$ przeszacowuje dla $t \in (4, 40)$
Weibull(0.05,0.5)-Weibull(0.05,0.5)	niedoszacowuje dla $t > 50$	niedoszacowuje dla $t > 60$ przeszacowuje dla $t < 60$

Tabela 2.1: Wyniki symulacji na obciążoność estymatorów nieparametrycznych.

Nazwa symulacji	S1	S2	S3
liczba obserwacji - podpróba 1	30	$10 - 110$	50
liczba obserwacji - podpróba 2	30	$10 - 110$	50
Parametr λ rozkładu Exp(λ) dla podpróby 1	0.03	0.03	0.01
Parametr λ rozkładu Exp(λ) dla podpróby 2	0.08	0.06	$0.011 - 0.1$
Parametr λ rozkładu Exp(λ) dla rozkładu cenzurującego	0.1	0.02	0.01
Liczba obserwacji porównywanych z rozkładem cenzurującym	$1 - 30$	wszystkie	wszystkie

Tabela 2.2: Charakterystyka danych użytych do symulacji mocy testu log-rank.

2.3. Bootstrapowe badanie modelu parametrycznego

W tej części rozdziału badam zachowanie się estymatorów w modelu parametrycznym. Estymuję model na dwóch próbach pochodzących z różnych rozkładów wykładniczych:

- próba pierwsza z Exp(λ_1),
- oraz próba druga z Exp(λ_2).

Zmienną objaśniającą jest zmienna binarna różnicująca obie próby (próba 2 jest próbą bazową). Dodatkowo w modelu występuje stała. Model jest więc następującej postaci:

$$\log(T) \sim \beta_0 + \beta_1 x + \log(\epsilon), \tag{2.1}$$

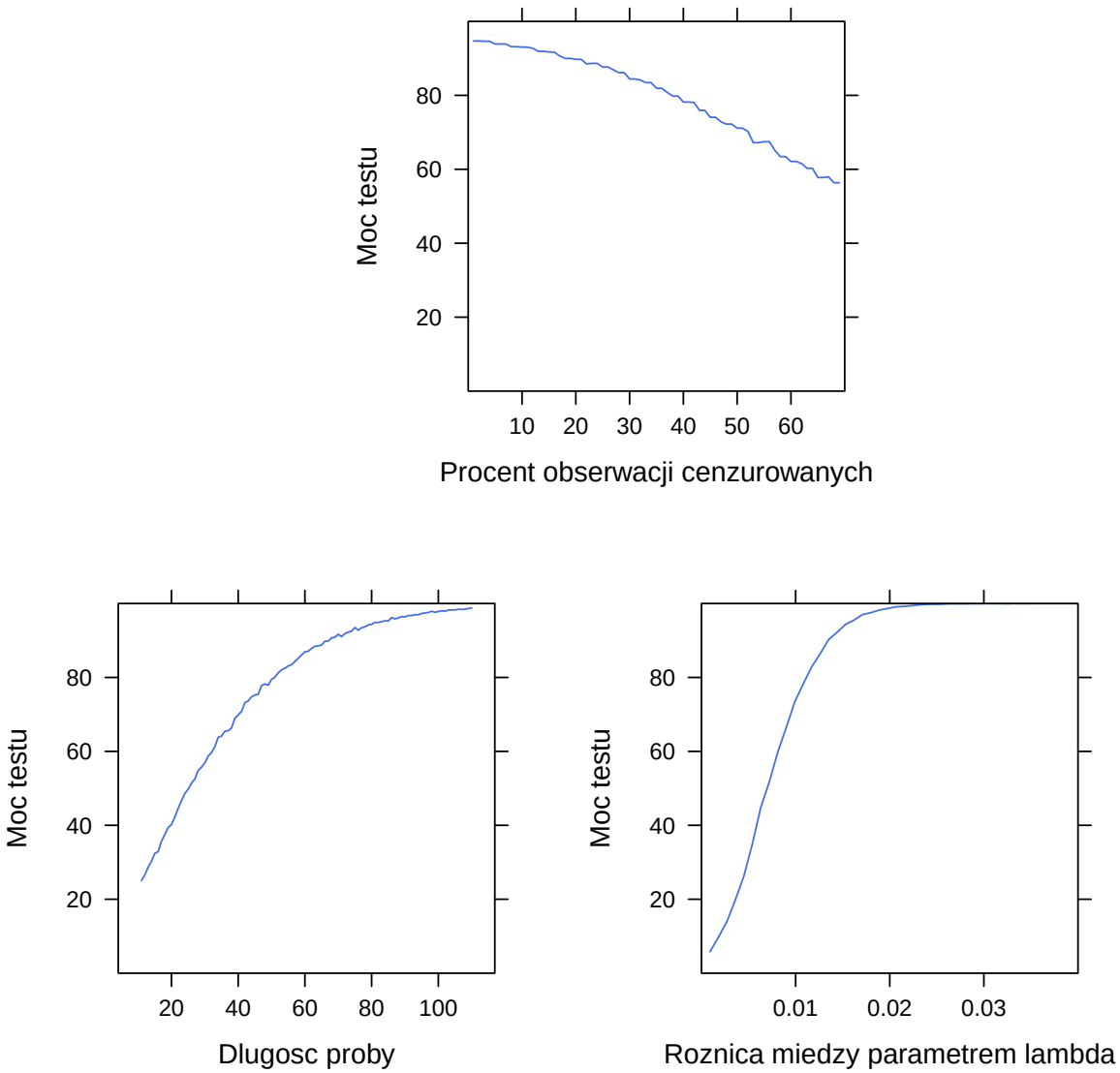
gdzie ϵ pochodzi z rozkładu Exp(1), a prawdziwymi wartościami parametrów są

$$\beta_1 = \log\left(\frac{\lambda_2}{\lambda_1}\right), \ \beta_0 = -\log(\lambda_2). \tag{2.2}$$

Analizę przeprowadzam metodą *bootstrap*. Losuję ze zwracaniem podpróbę ze zbioru danych (tu zbiorem danych są obie próby) i tylko na tej podpróbie estymuję model 2.1. Powtarzając tą czynność N razy (przyjmuję $N = 999$) otrzymuję rozkład estymatorów parametrów modelu. W ten sposób badam zachowanie $\hat{\beta}_1$ w zależności od stopnia cenzurowania i liczby obserwacji w każdej z prób.

Zakładam, że zmienne cenzurujące pochodzą z rozkładu wykładniczego Exp(λ_3). Symulacje przeprowadzam dla różnych wartości λ_3 oraz na następujących danych:

- próby mają po 300 obserwacji,
- próba pierwsza pochodzi z rozkładu Exp(0.08), a druga z Exp(0.04),



Rysunek 2.9: Moc testu w zależności od procentu obserwacji cenzurowanych (S1), długości próby (S2) i różnicy między parametrem lambda w podpróbach (S3) na podstawie symulacji ($N = 10000$ dla każdego poziomu zmiennej). Charakterystyki danych generowanych do wykresów S1, S2, S3 znajdują się w tabeli 2.2.

- możliwe wartości λ_3 to 0.01, 0.05, 0.08 oraz 0.1.

W tym celu używam zaimplementowanej przeze mnie funkcji *bootBeta*() (dodatek A.3).

```
a1=bootBeta(300,0.08,0.04,0.01, p1=1, p3=1)
a2=bootBeta(300,0.08,0.04,0.05)
a3=bootBeta(300,0.08,0.04,0.08)
a4=bootBeta(300,0.08,0.04,0.10)
```

Przeprowadzam cztery identyczne symulacje - ich wynik znajduje się na rysunku 2.10. Przerywaną linią zaznaczony jest prawdziwy parametr β_1, który dla tych symulacji wynosi $\beta_1 = \log\left(\frac{\lambda_2}{\lambda_1}\right) = -0.69$. Wnioski z doświadczenia są następujące:

1. Im większy jest parametr λ_3 (rozkład cenzurujący ma mniejszą średnią), tym rozkład $\hat{\beta}_1$ jest bardziej rozproszony.

2. Rozkład $\hat{\beta}_1$ jest niestabilny - raz estymator jest przeszacowany, raz niedoszacowany. Mody dopasowanych rozkładów zazwyczaj nie pokrywają się z prawdziwą wartością estymatora, bez względu na rozkład cenzurujący.

W drugiej części badania bootstrapowego, sprawdzam, jak na rozkład estymatora $\hat{\beta}_1$ wpływa liczba obserwacji w każdej z prób. Dane generowane są z rozkładów wykładniczych tak, że

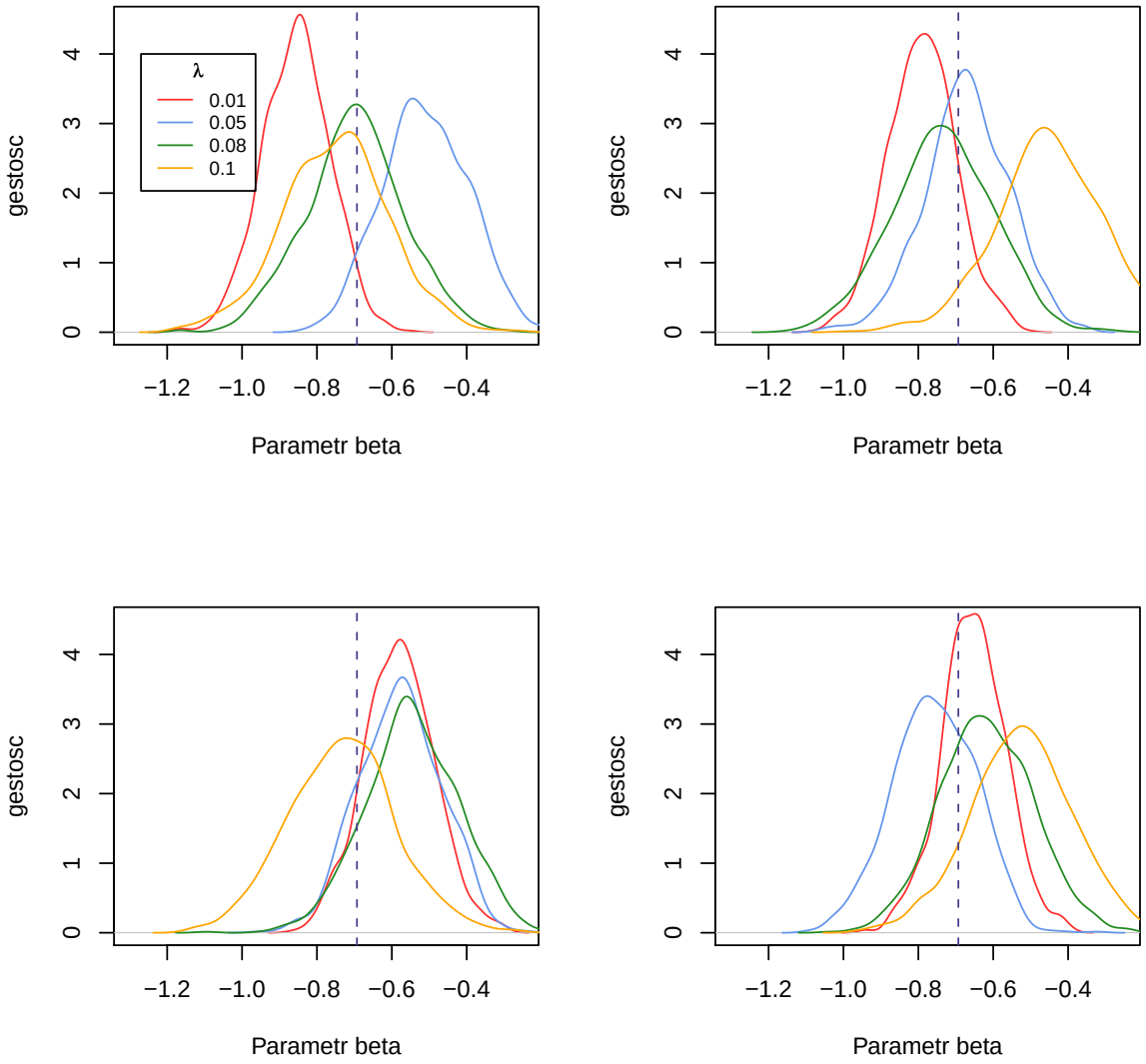
- próba pierwsza jest z Exp(0.08), druga z Exp(0.06),
- rozkład cenzurujący to Exp(0.01),
- obie próby są równej długości i możliwe liczby obserwacji dla każdej z prób to 50, 100, 400 oraz 1000.

```
a1=bootBeta(50,0.08,0.06,0.01)
a2=bootBeta(100,0.08,0.06,0.01)
a3=bootBeta(400,0.08,0.06,0.01)
a4=bootBeta(1000,0.08,0.06,0.01)
```

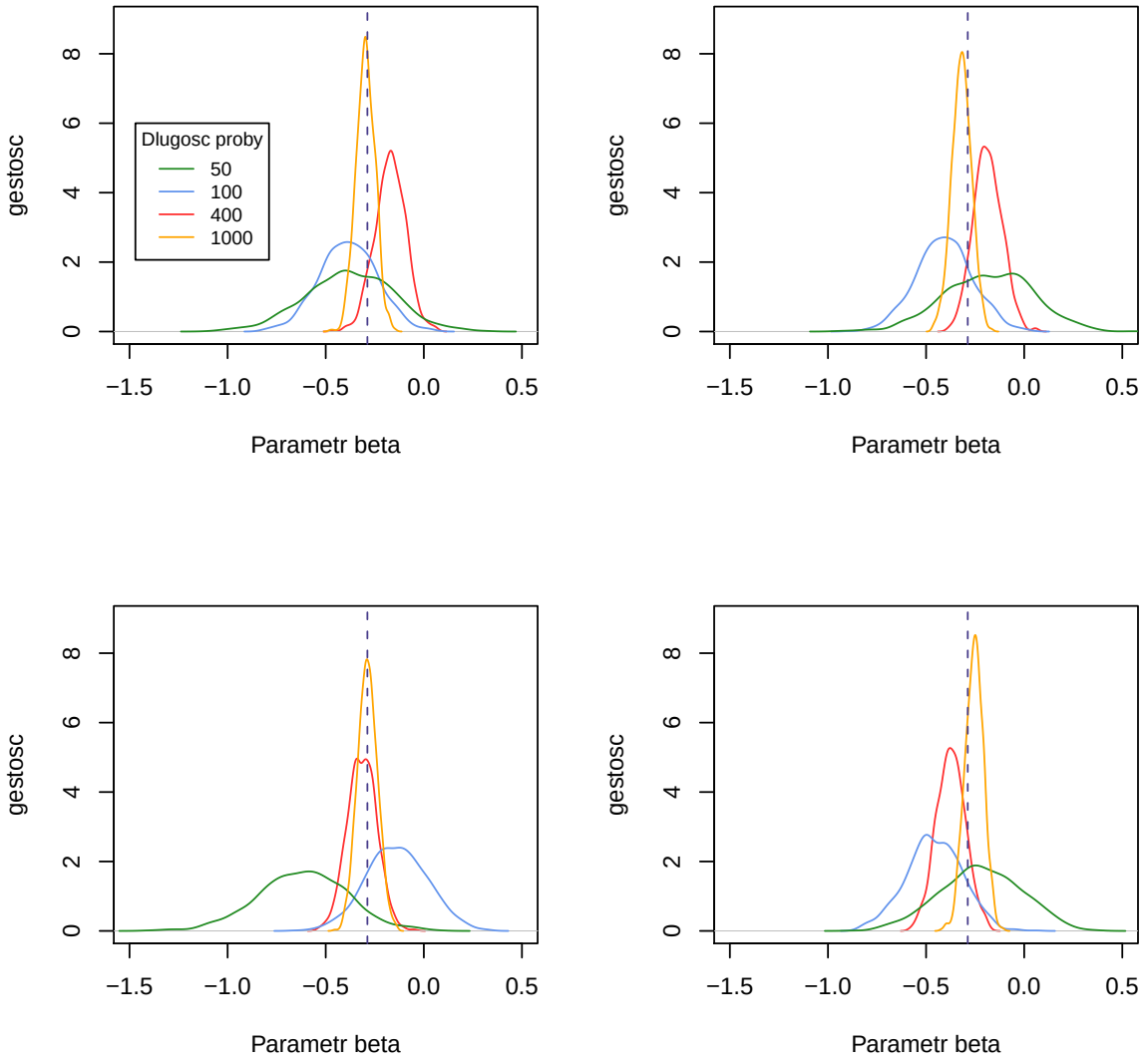
Rysunek 2.11 przedstawia wyniki czterech takich symulacji. Wynikają z nich następujące własności:

1. Im większa liczba obserwacji w każdej z prób, tym rozkład $\hat{\beta}_1$ jest bardziej skupiony wokół prawdziwej wartości β_1.

2. Dla 1000 obserwacji w każdej z prób rozkład ma modę w β_1 oraz jest symetryczny i o stosunkowo małej wariancji. W pozostałych przypadkach rozkłady odznaczają się większą nieregularności, bywają zarówno przeszacowane i niedoszacowane w zależności od symulacji.



Rysunek 2.10: Rozkład parametru $\hat{\beta}_1$ dla różnych poziomów cenzurowania. Cztery identyczne symulacje przeprowadzone metodą bootstrap na danych o następujących charakterystykach: dwie próby 300 elementowe z rozkładów Exp(0.08) oraz Exp(0.04).



Rysunek 2.11: Rozkład parametru $\hat{\beta}_1$ dla różnej liczby obserwacji w próbach. Cztery identyczne symulacje przeprowadzone metodą bootstrap na danych o następujących charakterystykach: dwie próby o równiej długości z rozkładów Exp(0.08) oraz Exp(0.06) cenzurowane rozkładem Exp(0.01).

Rozdział 3

Analiza danych rzeczywistych

3.1. Opis zbioru danych

W tym rozdziale przeprowadziłam analizę przeżycia na danych rzeczywistych zawierających informacje o pacjentkach chorych na raka piersi. Dane pochodzą z Dolnośląskiego Centrum Onkologii i są zbierane po 2000 roku. Każda obserwacja dotyczy jednej pacjentki i zawiera zarówno informacje związane z rozwojem choroby, takie jak np. data rozpoznania choroby, data nawrotu choroby, wielkość guza, data śmierci - jeśli nastąpiła, jak również informacje z wywiadu rodzinnego i charakterystyki jednostki tj. m. in. występowanie raka piersi w wywiadzie rodzinnym czy liczba porodów. Mamy zatem do czynienia zarówno ze zmiennymi binarnymi (występowanie przerzutów w węzłach), jakościowymi (typ raka) oraz ilościowymi (wiek, liczba porodów). Tabela 3.1 zawiera fragment analizowanego zbioru oraz podstawowe statystyki dla każdej zmiennej.

Dane dotyczą pewnego przedziału czasowego. Interesuje nas długość życia chorych, więc modelowanym zdarzeniem jest śmierć pacjentki. Część z badanych pacjentek nie umarła w analizowanym okresie. Dane dla tych pacjentek stanowią więc obserwacje cenzurowane - przyjmuję, że cenzurowanie nastąpiło w dniu ostatniej wizyty kontrolnej. Ponadto jedna pacjentka zginęła z przyczyn niezwiązanych z chorobą, tu również przyjmuję, że cenzurowanie nastąpiło w chwili jej śmierci. Konstruuję nowe zmienne:

- zmienną *czas do zdarzenia*, która jest długością życia od chwili wykrycia raka (dla zmiennych cenzurowanych jest to długość życia od chwili wykrycia do ostatniej zarejestrowanej wizyty) mierzoną w miesiącach,
- oraz zmienną binarną *zdarzenie* zawierającą informacje o cenzurowaniu. *Zdarzenie*=1, gdy obserwacja nie była cenzurowana, 0 w p.p.

Zbiór składa się z 260 obserwacji, z czego tylko 26 pacjentek umarło z powodu raka piersi w okresie badania, 90% wszystkich obserwacji jest więc cenzurowanych prawostronnie! Drugim spostrzeżeniem, jakie nasuwa się podczas wstępnej analizy danych (tabela 3.1) jest fakt, że tylko u 12 pacjentek wystąpiły inne (poza rakiem piersi) nowotwory w wywiadzie rodzinnym i wszystkie te obserwacje były cenzurowane. Wykluczam więc zmienną odpowiadająca tej cesze z dalszych analiz. Zmienna *typ raka* jest zmienną jakościową, zawierającą 9 rodzajów raka piersi (m. in. typ d, d+l, metaplastic, papillare, itp.). Większość tych rodzajów występuje jednak stosunkowo rzadko – typ d odnotowano dla 182 obserwacji, pozostałe wystąpiły łącznie u 78 pacjentek. Dlatego zdecydowałam się zastąpić *typ raka* nową zmienną objaśniającą - *typ d*, przyjmującą wartość jeden, jeśli typ raka to d i zero w przeciwnym przypadku.

Przerzuty w wezłach	Wielkość guza w mm pT	Typ raka wg WHO	Rak piersi w wywiadzie rodzinnym	Inne nowotwory w w.r.	Okres aktywności hormonalnej	Menopauza
0	11	d	0	0	28	0
1	45	d	1	0	37	1
0	30	l	0	0	36	1
1	15	d	0	0	38	1
1	25	d	0	0	31	0
0	15	l	0	1	36	1

Liczba porodów	Liczba poronień	Wiek w momencie rozpoznania	Data rozpoznania	Data nawrotu	Data zgonu	Zdarzenie	Czas do zdarzenia
0	0	42	14.05.2003	bez nawrotu	zyje	0	70
2	0	80	15.04.2005	bez nawrotu	zyje	0	34
3	0	64	09.06.2004	bez nawrotu	zyje	0	44
1	1	74	21.05.2004	01.11.2005	2006-05-11	1	24
2	0	47	16.04.2004	bez nawrotu	zyje	0	59
1	0	51	19.,3.2004	bez nawrotu	zyje	0	60

	Zdarzenie			Zdarzenie			Zdarzenie			Zdarzenie			Zdarzenie	
Przerzuty w węzłach	0	1	Rak piersi w rodzinie	0	1	Inne nowotwory w rodzinie	0	1	Menopauza	0	1	Razem	0	1
0	145	4	0	229	25	0	222	26	0	96	2		234	26
1	89	22	1	5	1	1	12	0	1	138	24			

	Zdarzenie			Zdarzenie			Zdarzenie	
Typ raka	0	1	Liczba porodów	0	1	Liczba poronień	0	1
d	160	22	0	23	1	0	167	16
d+l	3	0	1	56	5	1	52	5
d+muc	2	0	2	106	7	2	12	4
l	46	3	3	31	10	3	2	0
med.	7	0	4	12	3	4	0	0
metaplastic	2	0	5	5	0	5	0	1
muc	7	0	6	1	0	6	0	0
papillare	2	0	7	0	0	7	0	0
tub	5	1	8	0	0	8	1	0

Wielkość guza	Min.	1 Kw.	Mediana	Średnia	3 Kw.	Max.
	2.00	13.00	17.00	20.12	25.00	80.00
Wiek	Min.	1 Kw.	Mediana	Średnia	3 Kw.	Max.
	29.00	49.75	57.00	57.92	68.00	80.00
Okres aktywności hormonalnej	Min.	1 Kw.	Mediana	Średnia	3 Kw.	Max.
	17.00	31.00	35.00	34.25	38.00	50.00

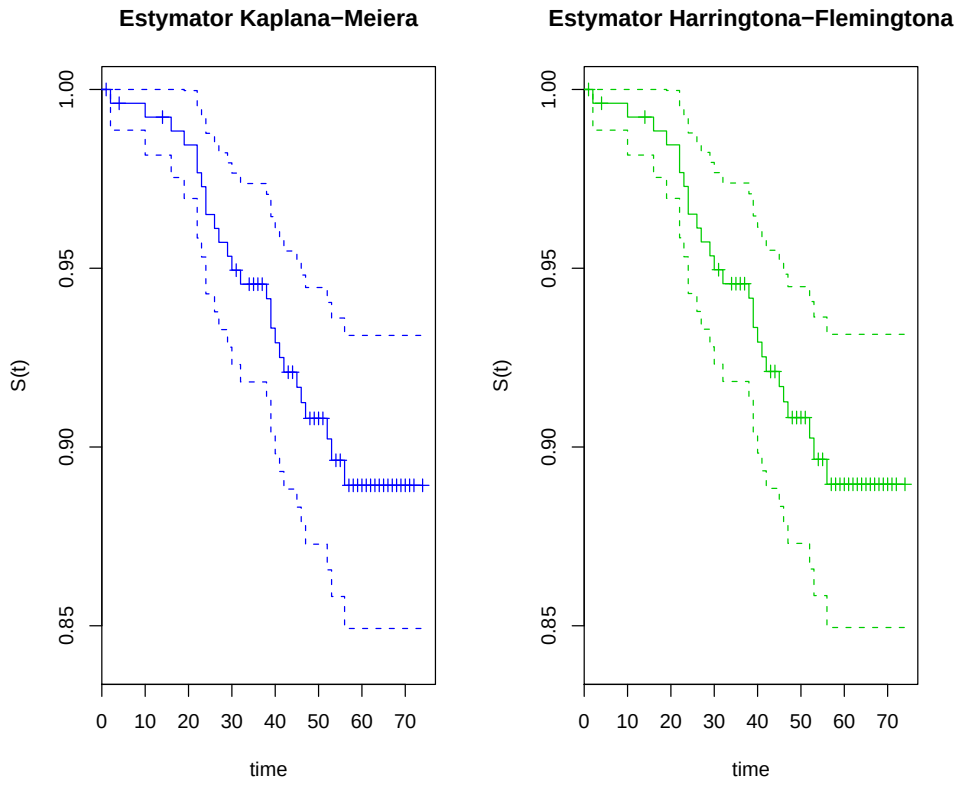
Tabela 3.1: Na górze: Fragment analizowanych danych - informacje o pacjentkach chorych na raka piersi. Na dole: podstawowe statystyki zmiennych z tego zbioru danych - częstości dla zmiennych dyskretnych, średnie i kwantyle dla zmiennych ciągłych.

3.2. Estymatory funkcji przeżycia dla pacjentek z rakiem piersi

Na początek wyznaczam estymatory funkcji przeżycia dla analizowanych danych.

```
library(survival)
par(mfrow=c(1,2))
km=survfit(Surv(A$time_month,A$event)~1, type="kaplan-meier")
fh=survfit(Surv(A$time_month,A$event)~1, type="fleming-harrington")
plot(km,col=4, xlab="time",ylab="S(t)", main="Estymator Kaplana-Meiera", ylim=c(0.84,1))
plot(fh,col=3, xlab="time",ylab="S(t)", main="Estymator Harringtona-Flemingtona", ylim=c(0.84,1))
```

Rysunek 3.1 przedstawia wyniki estymacji metodami Kaplana-Meiera i Harringtona-Flemingtona. Interesuje mnie, które zmienne dobrze różnicują rozkład umieralności na raka. Rysuję więc do-



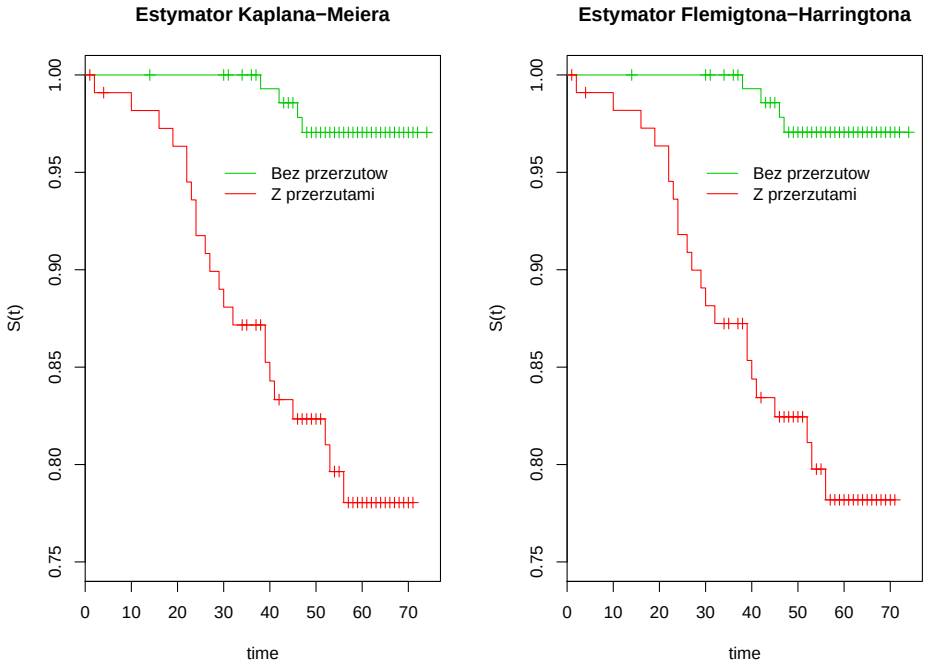
Rysunek 3.1: Estymatory funkcji przeżycia dla pacjentek z rakiem piersi.

datkowo estymatory funkcji przeżycia, osobno dla pacjentek z przerzutami i bez przerzutów - rysunek 3.2.

```
km=survfit(Surv(A$time_month,A$event)~A$Przerzuty, type="kaplan-meier", conf.type="none")
fh=survfit(Surv(A$time_month,A$event)~A$Przerzuty, type="fleming-harrington", conf.type="none")
plot(km, col=c(3,2), xlab="time",ylab="S(t)", main="Estymator Kaplana-Meiera", ylim=c(0.75,1))
legend(27,0.96, c("Bez przerzutow", "Z przerzutami"), col=c(3,2), lty=1, bty="n")
plot(fh, col=c(3,2), xlab="time",ylab="S(t)", main="Estymator Flemigtona-Harringtona", ylim=c(0.75,1))
legend(27,0.96, c("Bez przerzutow", "Z przerzutami"), col=c(3,2), lty=1, bty="n")
```

Widać, że estymator krzywej przeżycia dla obserwacji, u których wystąpiły przerzuty jest bardziej stromy niż dla obserwacji bez przerzutów. Pacjentki z przerzutami odznaczają się więc większą umieralnością.

Intuicyjne wydaje się, że wiek pacjentki może być ważnym czynnikiem różnicującym w kontekście umieralności na raka piersi. W zbiorze danych znajduje się zmienna *wiek w momencie rozpoznania* (w skrócie będę ją nazywać *wiek*), którą można w zasadzie traktować jako zmienną



Rysunek 3.2: Estymatory funkcji przeżycia dla pacjentek z rakiem piersi, osobno dla pacjentek, u których wystąpiły przerzuty i dla pacjentek bez przerzutów.

ciągłą. Chcąc zobaczyć, jak wyglądają estymatory krzywych przeżycia dla kobiet w różnym wieku, wprowadzam następujący podział:

- pacjentki do 50 lat,
- pacjentki między 51-64 rokiem życia,
- pacjentki w wieku 65 lat i starsze.[1]

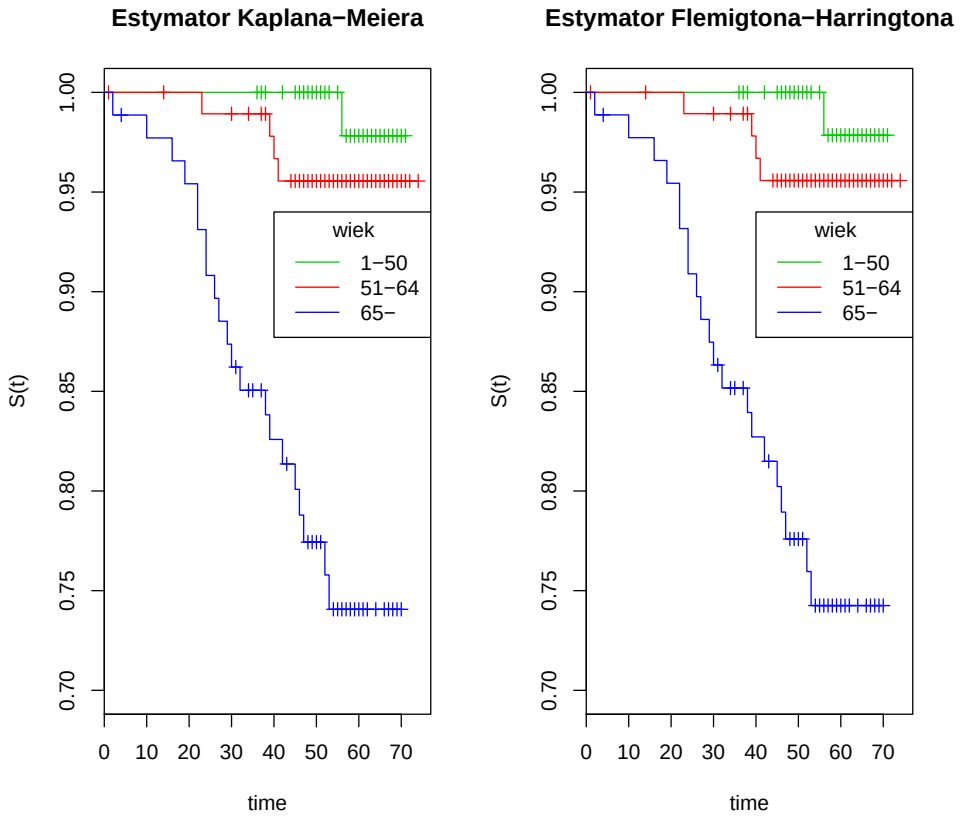
Rysunek 3.3 przedstawia te estymatory w oparciu o powyższy podział. Widać z niego, że osoby chorujące na raka w późniejszym wieku podleją większemu ryzyku niż pacjentki młodsze.

Jako ostatnią cechę różnicującą estymatory funkcji przeżycia wybrałam informację o liczbie poronień u pacjentek. Osobno narysowałam estymatory funkcji przeżycia dla obserwacji, u których wystąpiło chociaż jedno poronienie, osobno dla pacjentek bez poronień. Wyniki znajdują się na rysunku 3.4. W tym przypadku trudniej jest formułować jednoznaczne wnioski dotyczące wpływu poronienia na ryzyko śmierci - estymatory krzywych przeżycia przecinają się w paru miejscach. W porównaniu z wiekiem i występowaniem przerzutów cecha ta, jeżeli jakkolwiek, różnicuje najgorzej.

3.3. Testowanie różnic

We wcześniejszym podrozdziale, na podstawie wykresów estymatorów funkcji przeżycia badałam wpływ niektórych zmiennych na śmiertelność pacjentek. Teraz wykorzystam bardziej formalne podejście i przetestuję moje wcześniejsze spostrzeżenia testami różnic - przede wszystkim testem log-rank.

[1]Zastosowany podział oparty jest na kwantylach rozkładu wieku, 50 odpowiada kwantylowi $\frac{1}{3}$, a 65 - $\frac{2}{3}$.



Rysunek 3.3: Estymatory funkcji przeżycia dla pacjentek z rakiem piersi, w zależności od wieku w momencie rozpoznania.

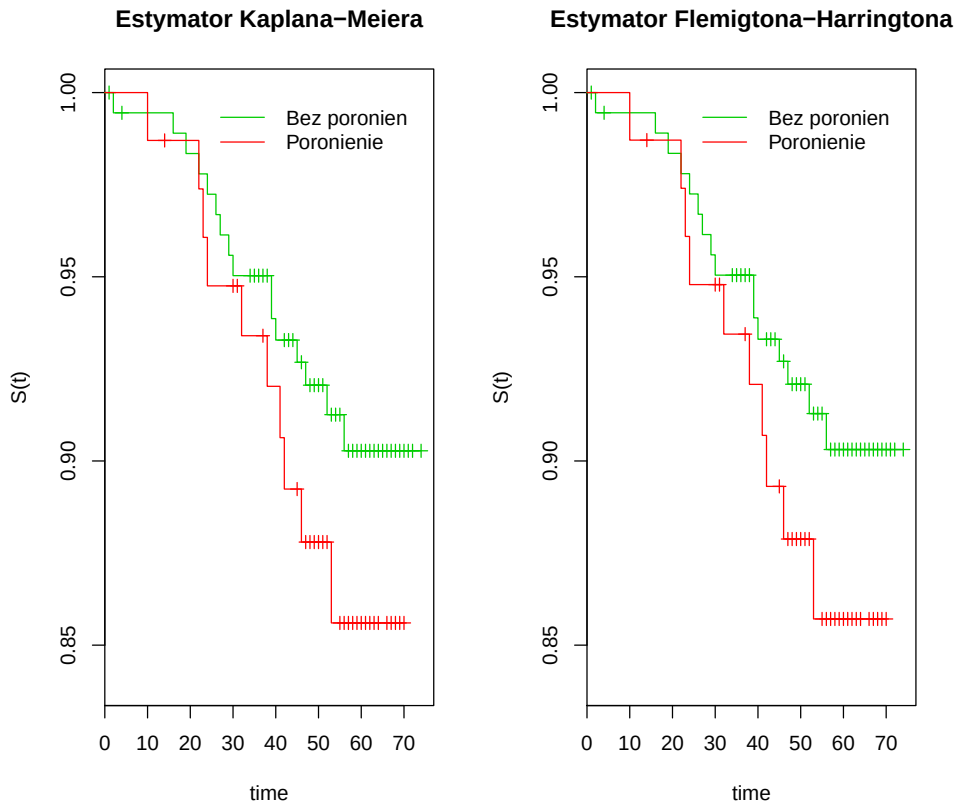
Program R udostępnia funkcję *survdiff*() służącą do testowania różnic między krzywymi przeżycia. Funkcja ta ma parametr rho z przedziału $[0, 1]$, który nadaje wagi dla czasu występowania zdarzeń, w ten sposób, że każda śmierć jest przemnażana przez $S(t)^{\text{rho}}$. Dla rho=0 *survdiff*() jest zwykłym testem log-rank.

Jako pierwszą cechę testuję występowanie przerzutów.

```
survdiff(formula = Surv(A[, 3], A[, 4]) ~ A$Przerzuty, rho = 0)

                N Observed Expected (O-E)^2/E (O-E)^2/V
A$Przerzuty=0 149        4     15.6       8.6      21.5
A$Przerzuty=1 111       22     10.4      12.9      21.5

Chisq= 21.5  on 1 degrees of freedom, p=3.56e-06
```

P-wartość jest w tym przypadku rzędu 10^{-6} zdecydowanie odrzucam więc hipotezę zerową o równości krzywych przeżycia dla pacjentek z przerzutami i bez przerzutów. Zauważam również, że wartość statystyki nie zmienia się znacząco wraz ze zmianą rho (tj. gdy większą wagę przykłada się do zdarzeń wcześniejszych).

```
survdiff(formula = Surv(A[, 3], A[, 4]) ~ A$Przerzuty, rho = 1)

                N Observed Expected (O-E)^2/E (O-E)^2/V
A$Przerzuty=0 149      3.7    14.80      8.33      21.8
A$Przerzuty=1 111     21.0     9.92     12.43      21.8

Chisq= 21.8  on 1 degrees of freedom, p= 2.98e-06
```



Rysunek 3.4: Estymatory funkcji przeżycia dla pacjentek z rakiem piersi, w zależności od tego, czy wystąpiło poronienie.

Stąd wniosek, że umieralność na raka piersi jest większa dla pacjentek z przerzutami niż bez przerzutów.

Podobnie, test log-rank dla trzech poziomów wieku daje następujący wynik:

```
survdiff(formula = Surv(A[, 1], A[, 2]) ~ wiek, rho = 0)

        N Observed Expected (O-E)^2/E (O-E)^2/V
wiek=0 77        1     8.23      6.35      9.31
wiek=1 95        4     9.74      3.39      5.42
wiek=2 88       21     8.03     20.97     30.42

 Chisq= 30.8  on 2 degrees of freedom, p= 2.06e-07
```

Silnie przemawia on za odrzuceniem hipotezy zerowej i większą śmiertelnością pacjentek zapadających na chorobę w późniejszym wieku.

Jeśli chodzi o ostatnią interesującą mnie cechę, czyli wystąpienie poronienia, tu wyniki nie są już takie jednoznaczne.

```
survdiff(formula = Surv(A[, 1], A[, 2]) ~ poronienie, rho = 0)

               N Observed Expected (O-E)^2/E (O-E)^2/V
poronienie=0 183       16    18.48     0.332      1.15
poronienie=1  77       10     7.52     0.814      1.15

Chisq= 1.1  on 1 degrees of freedom, p= 0.284
```

Przy p-wartości na poziomie 0.05 nie ma postaw do odrzucenia hipotezy, że krzywe przeżycia dla pacjentek u których wystąpiło poronienie i bez poronień są takie same.

3.4. Model parametryczny

3.4.1. Wybór modelu

Na początku estymuję model z rozkładem Weibulla postaci 1.30 na wszystkich dostępnych zmiennych.

```
#Model P1

survreg(formula = Surv(A[, 1], A[, 2]) ~ 1 + przerzuty + wielkosc_guza +
    rak_piersi_w_rodzinie + okres_aktywnosci_hormonalnej + liczba_porodow +
    liczba_poronien + wiek + typ_d + menopauza, dist = "weibull")

                              Value  Std.Error         z          p
(Intercept)                 10.7594     1.6303    6.5995   4.13e-11
przerzuty                   -1.1077     0.3878   -2.8563   4.29e-03
wielkosc_guza               -0.0187     0.0120   -1.5579   1.19e-01
rak_piersi_w_rodzinie        0.2697     0.7129    0.3783   7.05e-01
okres_aktywnosci_hormonalnej 0.0177     0.0243    0.7302   4.65e-01
liczba_porodow              -0.1129     0.0861   -1.3118   1.90e-01
liczba_poronien             -0.0842     0.0893   -0.9433   3.46e-01
wiek                        -0.0656     0.0233   -2.8187   4.82e-03
typ_d                       -0.5191     0.3525   -1.4725   1.41e-01
menopauza                    0.0298     0.6514    0.0457   9.64e-01
Log(scale)                  -0.4946     0.1779   -2.7805   5.43e-03

Scale= 0.61

Weibull distribution
Loglik(model)= -157   Loglik(intercept only)= -187.7
    Chisq= 61.35 on 9 degrees of freedom, p= 7.4e-10

Number of Newton-Raphson Iterations: 10
n= 260
```

Oraz taki sam model dla rozkładu wykładniczego o parametryzacji 1.28.

```
#Model P2

survreg(formula = Surv(A[, 1], A[, 2]) ~ 1 + przerzuty + wielkosc_guza +
    rak_piersi_w_rodzinie + okres_aktywnosci_hormonalnej + liczba_porodow +
    liczba_poronien + wiek + typ_d + menopauza, dist = "exponential")

                              Value  Std.Error         z          p
(Intercept)                 14.65418     1.9555   7.49375   6.69e-14
przerzuty                   -1.77552     0.5564  -3.19119   1.42e-03
wielkosc_guza               -0.02826     0.0190  -1.48425   1.38e-01
rak_piersi_w_rodzinie        0.54487     1.1533   0.47244   6.37e-01
okres_aktywnosci_hormonalnej 0.02823     0.0389   0.72513   4.68e-01
liczba_porodow              -0.19941     0.1412  -1.41181   1.58e-01
liczba_poronien             -0.10344     0.1493  -0.69290   4.88e-01
wiek                        -0.10151     0.0333  -3.05006   2.29e-03
typ_d                       -0.79647     0.5638  -1.41279   1.58e-01
menopauza                   -0.00221     1.0542  -0.00209   9.98e-01

Scale fixed at 1

Exponential distribution
Loglik(model)= -160.2   Loglik(intercept only)= -189.3
      Chisq= 58.14 on 9 degrees of freedom, p= 3.1e-09

Number of Newton-Raphson Iterations: 7
n= 260
```

Na podstawie logarytmu funkcji wiarygodności można stwierdzić, że model z rozkładem Weibulla jest lepiej dopasowany do danych niż model z rozkładem wykładniczym. Ponadto parametr skali (w rozkładzie wykładniczym będący stałą równą 1) okazał się istotny, w dalszej części analiz wykorzystuję więc rozkład Weibulla.

Na początku usuwam z modelu część zmiennych nieistotnych, opierając się na teście Walda na poziomie istotności 0.2. Wynikiem jest nowy model postaci

```
#Model P3

survreg(formula = Surv(A[, 1], A[, 2]) ~ 1 + przerzuty + wielkosc_guza +
    liczba_porodow + wiek + typ_d, dist = "weibull")

                Value  Std.Error       z          p
(Intercept)    11.1064     1.5409    7.21   5.69e-13
przerzuty      -1.1324     0.3898   -2.90   3.67e-03
wielkosc_guza  -0.0178     0.0103   -1.72   8.51e-02
liczba_porodow -0.1027     0.0855   -1.20   2.30e-01
wiek           -0.0619     0.0170   -3.64   2.73e-04
typ_d          -0.5127     0.3458   -1.48   1.38e-01
Log(scale)     -0.4823     0.1772   -2.72   6.47e-03

Scale= 0.617

Weibull distribution
Loglik(model)= -157.9   Loglik(intercept only)= -187.7
      Chisq= 59.69 on 5 degrees of freedom, p= 1.4e-11

Number of Newton-Raphson Iterations: 9
n= 260
```

Testem ilorazu wiarygodności testuję hipotezę łącznej istotności zmiennych: *wielkość guza*, *liczba porodów* oraz *typ d*.

```
anova(survreg(formula = Surv(A[, 1], A[, 2]) ~ 1 + wiek + przerzuty,
    dist = "weibull"), survreg(formula = Surv(A[, 1], A[, 2]) ~ 1 +
 przerzuty + wielkosc_guza + liczba_porodow + wiek + typ_d, dist = "weibull"))

  Terms Resid. Df
1                                        1 + przerzuty + wiek       256
2 1 + przerzuty + wielkosc_guza + liczba_porodow + wiek + typ_d       253

     -2*LL                                    Test Df   Deviance   P(>|Chi|)
1 321.7587                                        NA         NA          NA
2 315.7298+wielkosc_guza+liczba_porodow+typ_d      3   6.028893   0.1102129
```

P-wartość wynosi 0.11, nie ma więc podstaw do odrzucenia hipotezy o zerowej wartości współczynników przy zmiennych: *wielkość guza*, *liczba porodów* oraz *typ d*. Dostaję więc ostateczny model postaci

```
#Model P4

survreg(formula = Surv(A[, 1], A[, 2]) ~ 1 + wiek + przerzuty,
    dist = "weibull")

Coefficients:
(Intercept)        wiek   przerzuty
10.63759643 -0.06846531 -1.28940708

Scale= 0.6341682

Loglik(model)= -160.9   Loglik(intercept only)= -187.7
      Chisq= 53.66 on 2 degrees of freedom, p= 2.2e-12
n= 260
```

Zmienne *wiek* oraz *przerzuty* okazały się istotne. Potwierdzają to również wyniki testów log-rank oraz wykresy 3.2 i 3.3.

W funkcji *survreg*(), której używam do estymacji mogę wprowadzić funkcję *strata* na określonym parametrze. Sprawia ona, że parametr skali jest różny dla różnych wartości tej zmiennej. Jeszcze raz estymuję model, tym razem jednak zmienna *przerzuty* służy do różnicowania parametru skali.

```
#Model P5
```

```
survreg(formula = Surv(A[, 1], A[, 2]) ~ 1 + wiek + strata(przerzuty),
    dist = "weibull")

Coefficients:
(Intercept)        wiek
 9.12461526 -0.06275034

Scale:
przerzuty=0 przerzuty=1
  0.2385050   0.7303222

Loglik(model)= -158.6   Loglik(intercept only)= -175.5
      Chisq= 33.81 on 1 degrees of freedom, p= 6.1e-09
n= 260
```

Dwa ostatnie modele (P4 oraz P5) okazują się być najlepiej dopasowane do danych. Model ze zróżnicowanymi parametrami skali (P5) ma najwyższą wartość logarytmu funkcji wiarygodności. Ze względu na łatwiejszą interpretację parametrów w dalszej części podrozdziału będę jednak odwoływać się do modelu P4. Interpretacja wartości estymatorów znajduje się w podrozdziale 3.4.3.

3.4.2. Diagnostyka

Wykres 3.5 zawiera residua martyngałowe (ich charakterystyka znajduje się w podrozdziale 1.7.2). Dla większości obserwacji residua te znajdują się w otoczeniu zera, co świadczy o dobrym dopasowaniu modelu.[2]

3.4.3. Interpretacja parametrów – ryzyko śmierci

Na podstawie wzorów 1.31, 1.37 oraz parametrów uzyskanych w estymacji mogę wyznaczyć wpływ poszczególnych zmiennych na funkcję hazardu.

- *Przerzuty*: $\exp\left(-(-1.28940708/0.6341682)\right) = 7.64$
- *Wiek*: $\exp\left(-(-0.06846531/0.6341682)\right) = 1.11$

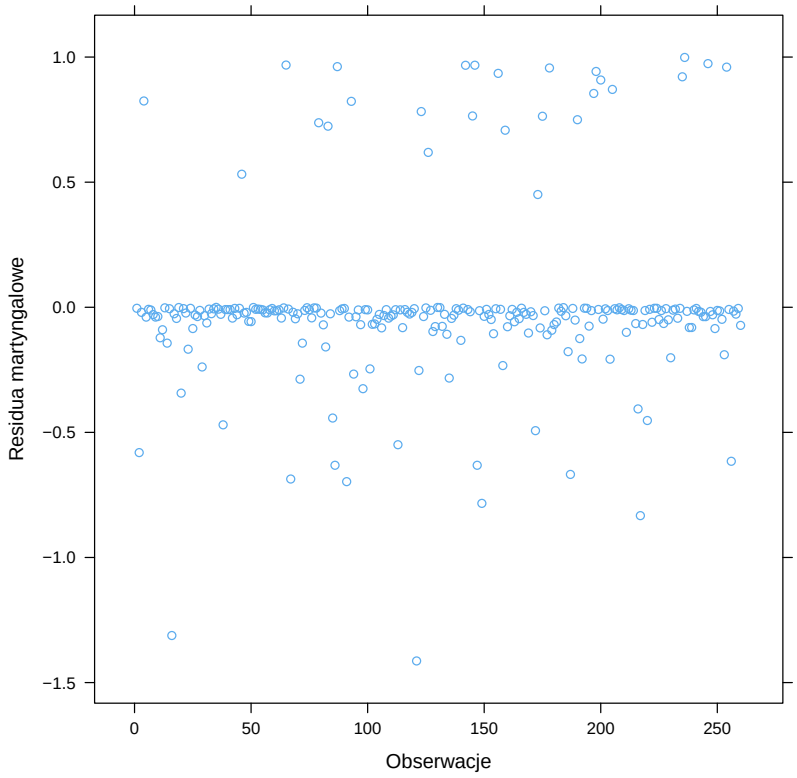
Powyższe wyniki interpretuje się następująco:

- Dla osób z przerzutami ryzyko śmierci (zdefiniowane jako funkcja hazardu) jest ponad 7.6 razy większe, niż dla osób bez przerzutów.
- Gdy porównuje się dwie osoby chore na raka piersi, gdzie jedna zachorowała w wieku m lat, a druga $k+m$, ryzyko śmierci osoby, która zachorowała w późniejszym wieku jest większe ok. 1.11^k razy w porównaniu z drugą osobą. Na przykład pacjentka, która zachorowała w wieku 64 lat ma $1.11^{65-50} = 4.78$ razy większe ryzyko śmierci od osoby, która zachorowała w wieku 50 lat, przy pozostałych charakterystykach identycznych dla obu kobiet.

3.5. Nieparametryczny model Cox'a

W tym podrozdziale dane dotyczące pacjentek chorych na raka piersi estymuję nieparametrycznym modelem proporcjonalnego hazardu (modelem Cox'a). Podczas przeprowadzania analizy wzoruję się na [12].

[2] W podrozdziale 3.5, dotyczącym modelu Cox'a, temat residuów został potraktowany bardziej szczegółowo i analizowanych jest więcej rodzajów residuów: residua deviance, residua Schoenfeld'a. Znajdują się tam również kody programu R użyte do liczenia różnych typów residuów, m. in. zaprezentowanych w tym podrozdziale residuów martyngałowych.



Rysunek 3.5: Residua martyngałowe dla modelu parametrycznego P4.

3.5.1. Wybór modelu

Na początku do objaśnienia zjawiska używam wszystkich dostępnych zmiennych.

```
#Model Coxa 1

coxph(formula = Surv(A[, 1], A[, 2]) ~ przerzuty + wielkosc_guza + rak_piersi_w_rodzinie
+ okres_aktywnosci_hormonalnej + menopauza + liczba_porodow + liczba_poronien + wiek + typ_d, method = "breslow")

                              coef  exp(coef) se(coef)  z       p
przerzuty                   1.8409   6.302   0.5549   3.317   0.00091
wielkosc_guza               0.0319   1.032   0.0191   1.675   0.09400
rak_piersi_w_rodzinie      -0.5969   0.551   1.1715  -0.509   0.61000
okres_aktywnosci_hormonalnej -0.0242   0.976   0.0402  -0.602   0.55000
menopauza                  -0.1752   0.839   1.0656  -0.164   0.87000
liczba_porodow              0.1967   1.217   0.1376   1.430   0.15000
liczba_poronien             0.1103   1.117   0.1501   0.735   0.46000
wiek                        0.1120   1.118   0.0344   3.252   0.00110
typ_d                       0.8105   2.249   0.5683   1.426   0.15000

Likelihood ratio test=61 on 9 df, p=8.63e-10  n= 260
```

Zmienne *rak piersi w rodzinie*, *menopauza*, *okres aktywności hormonalnej* oraz *liczba poronień* nie przechodzą testu Walda na poziomie istotności 0.2. Estymuję więc model bez tych zmiennych.

```
#Model Coxa 2

coxph(formula = Surv(A[, 1], A[, 2]) ~ 1 + przerzuty + wielkosc_guza +
     liczba_porodow + wiek + typ_d, method = "breslow")


                  coef    exp(coef) se(coef)      z        p
```

```
przerzuty         1.879     6.55     0.5467     3.44     5.9e-04
wielkosc_guza     0.029     1.03     0.0163     1.79     7.4e-02
liczba_porodow    0.184     1.20     0.1363     1.35     1.8e-01
wiek              0.103     1.11     0.0230     4.46     8.2e-06
typ_d             0.779     2.18     0.5514     1.41     1.6e-01

Likelihood ratio test=59.7  on 5 df, p=1.43e-11  n= 260
```

Testu ilorazu wiarygodności używam do sprawdzenia łącznej istotności zmiennych *wielkość guza*, *liczba porodów* i *typ d*.

```
x2=2*coxph(Surv(A[,1],A[,2])~1+przerzuty+wiek,method="breslow")$loglik[2]-
2*coxph(Surv(A[,1],A[,2])~1+przerzuty+wielkosc_guza+liczba_porodow+wiek+typ_d, method="breslow")$loglik[2]
1-pchisq(-x2,3)
```

Otrzymuję p-wartość równą 0.11 - nie ma więc podstaw do odrzucenia hipotezy zerowej, że $\beta_i = 0$ dla tych zmiennych. Powyższy rezultat potwierdzają również testy Walda na poziomie istotności 0.05 dla poszczególnych zmiennych. Ostatecznie estymuję więc *model Coxa 3* następującej postaci:

```
#Model Coxa 3

coxph(formula = Surv(A[, 1], A[, 2]) ~ 1 + przerzuty + wiek, method = "breslow")

            coef   exp(coef) se(coef)     z         p
przerzuty   2.06     7.87     0.5451     3.78     1.5e-04
wiek        0.11     1.12     0.0239     4.59     4.4e-06

Likelihood ratio test=53.6  on 2 df, p=2.26e-12  n= 260
```

Podobnie jak w modelu parametrycznym zmiennymi istotnymi okazały się *wiek* oraz *przerzuty*.

Badanie interakcji

Sprawdzam również możliwe interakcje: przerzuty:wiek, przerzuty:wielkość guza, liczba porodów:przerzuty, etc. Wszystkie okazują się być nieistotne i nie zostają uwzględnione w modelu.

3.5.2. Badanie odpowiedniości skali parametrów ciągłych

Następnym krokiem jest zbadanie odpowiedniości skali dla zmiennych ciągłych znajdujących się w modelu. W *modelu Coxa 3* jest jedna zmienna ciągła - *wiek*. Chcę sprawdzić, czy nie wymaga ona transformacji.

Jeśli skala dla zmiennej ciągłej jest dobrze dobrana, zależność między tą zmienną, a logarytmem z funkcji hazardu powinna być liniowa. Odpowiedniość skali można badać na kilka sposobów. Poniżej przedstawione są dwie metody, które zastosowałam w pracy.

Metoda pierwsza jest następująca: Należy zastąpić zmienną ciągłą kilkoma zmiennymi binarnymi dla różnych poziomów danej zmiennej. Tworzy się nowe zmienne z_1, z_2, z_3, dobierając odpowiednio $Q_1 < Q_2 < Q_3 < Q_4 < Q_5$ (Q_1, Q_2, Q_3, Q_4, Q_5 to zazwyczaj kwantyle odpowiednio 0, 0.25, 0.5, 0.75, 1 rozkładu badanej zmiennej) tak, że $z_i = 1$, gdy wartość zmiennej należy do przedziału $[Q_{i+1}, Q_{i+2})$, $z_i = 0$ w pp. Ponownie estymuje się model, zastępując badaną zmienną ciągłą nowopowstałymi zmiennymi binarnymi. Na wykresie przedstawia się zależność między regresorami dla zmiennych z_i (dodatkowo dodając zero, odpowiadające regersorowi dla wartości zmiennej w $[Q_1, Q_2)$) oraz środkami przedziałów $[Q_1, Q_2)$, $[Q_2, Q_3)$, $[Q_3, Q_4)$, $[Q_4, Q_5)$. Jeśli skala jest liniowa, krzywa łącząca punkty na wykresie powinna być linią prostą.

Drugą metodą badania odpowiedniości skali są wykresy uwzględniające zachowanie residuów. Generuje się dwa wykresy. Pierwszy to

- wykres zmiennej w zależności od residuów martyngałowych pochodzących z modelu nieuwzględniającego badanej zmiennej. Oczekiwaną zależnością jest zależność linowa.

Konstrukcja drugiego wykresu odbywa się następująco:

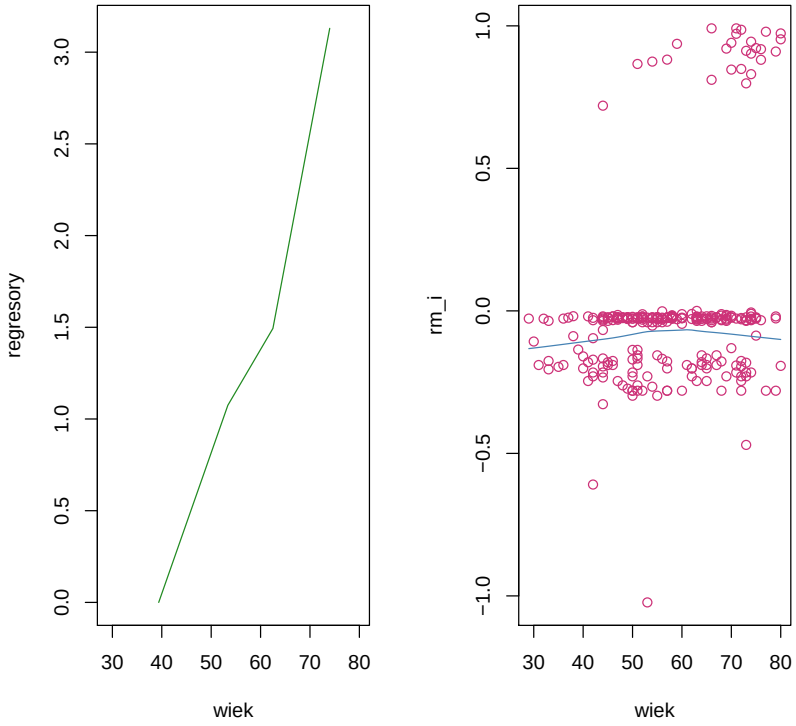
1. Estymuje się model z uwzględnieniem badanej zmiennej.
2. Zapamiętuje się residua martyngałowe (rm_i) dla tego modelu.
3. Na ich podstawie oblicza się residuum Cox'a-Snella: $rc_i = \delta_i - rm_i$, gdzie zmienna binarna $\delta_i = 1$ przy braku cenzurowania dla obserwacji i.
4. Estymuje się wygładzoną zależność między wartościami δ_i i badaną zmienną - c_{LSM} (δ_i na osi y).
5. Estymuje się wygładzoną zależność między rc_i i badaną zmienną - H_{LSM} (rc_i na osi y).
6. Wartości na osi y, uzyskane w dwóch poprzednich krokach, są użyte do obliczenia y_i, jako
$$y_i = \log\left(\frac{c_{LSM}}{H_{LSM}}\right) + \beta_{\text{zmienna}}\text{zmienna}_i \tag{3.1}$$
7. Pary (y_i, zmienna) przedstawia się na wykresie, a zależność miedzy nimi powinna być liniowa.

Poniżej znajduje się implementacja w programie R wyżej opisanych metod do badania zmiennej *wiek*:

```
#Metoda 1
quantile(wiek,probs=c(0,0.25,0.5,0.75,1))
#   0%   25%   50%   75%  100%
#  29.00 49.75 57.00 68.00 80.00
wiek.cat=rep(0,m)
wiek.cat=rep(0,m)
wiek.cat[wiek<=49.75]=1
wiek.cat[wiek>49.75 & wiek<=57.00]=2
wiek.cat[wiek>57.00 & wiek<=68.00]=3
wiek.cat[wiek>68.00]=4
wiek.cat=factor(wiek.cat, labels=c("1","2","3","4"))
contrasts(wiek.cat)=contr.treatment(4, base=1,contrasts=TRUE)
c=coxph(Surv(A[,1],A[,2])~1+przerzuty+wiek.cat, method="breslow")
x=c((49.75+min(wiek))/2,(49.75+57.00)/2,(57.00+68.00)/2,(68+max(wiek))/2)
y=c(0,c$coefficients[2:4])
par(mfrow=c(1,2))
plot(x,y,type="l",xlim=c(29,80), xlab="wiek",ylab="regresory",col="forestgreen")

#Metoda2
c=coxph(Surv(A[,1],A[,2])~przerzuty+wiek, method="breslow")
c.mi=residuals(c,type="martingale")
c.hi=A[,2]-c.mi
c.clsm=lowess(wiek,A[,2])
c.hlsm=lowess(wiek, c.hi)
c.yi=log(c.clsm$y/c.hlsm$y)+(c$coefficients[2]*wiek)
c1=coxph(Surv(A[,1],A[,2])~przerzuty, method="breslow")
c.mg=residuals(c1,type="martingale")
plot(wiek,c.mg, ylab="rm_i", col="violetred3")
lines(lowess(wiek,c.mg), col="steelblue")
#plot(c.yi,wiek, xlab="y_i", col="steelblue")
```

Ostatni z wykresów (z drugiej metody) został pominięty - nie jest on bowiem wiarygodny, ze względu na zbyt duży poziom cenzurowania danych (90%). Pozostałe znajdują się na rysunku 3.6. Wykres z prawej strony (metoda 1) jest zbliżony do liniowego, natomiast wykres residuów martyngałowych (metoda 2) jest mniej regularny. Może to być jednak znów efekt wysokiego poziomu cenzurowania danych. Transformacje zmiennej *wiek* metodą Box'a-Cox'a nie polepszają znacząco wyników.[3] Pozostawiam więc tą zmienną w jej pierwotnej postaci.

[3]Symulacyjnie wyznaczam taką wartość p w przekształceniu Box'a-Cox'a ($x' = \frac{x^p-1}{p}$, dla $p \neq 0$ i $x' = \log(x)$, dla $p = 0$), która maksymalizuje funkcję wiarygodności. Znaleziona transformacja nie poprawia jednak istotnie wykresu residuów matyngałowych vs. wiek, ani logarytmu funkcji wiarygodności.



Rysunek 3.6: Badanie odpowiedniości skali dla zmiennej *wiek*.

3.5.3. Testowanie założenia o proporcjonalnej funkcji hazardu

Następnym krokiem jest przetestowanie podstawowego założenia modelu Cox'a - założenia, że parametr β oraz bazowa funkcja hazardu są stałe w czasie.

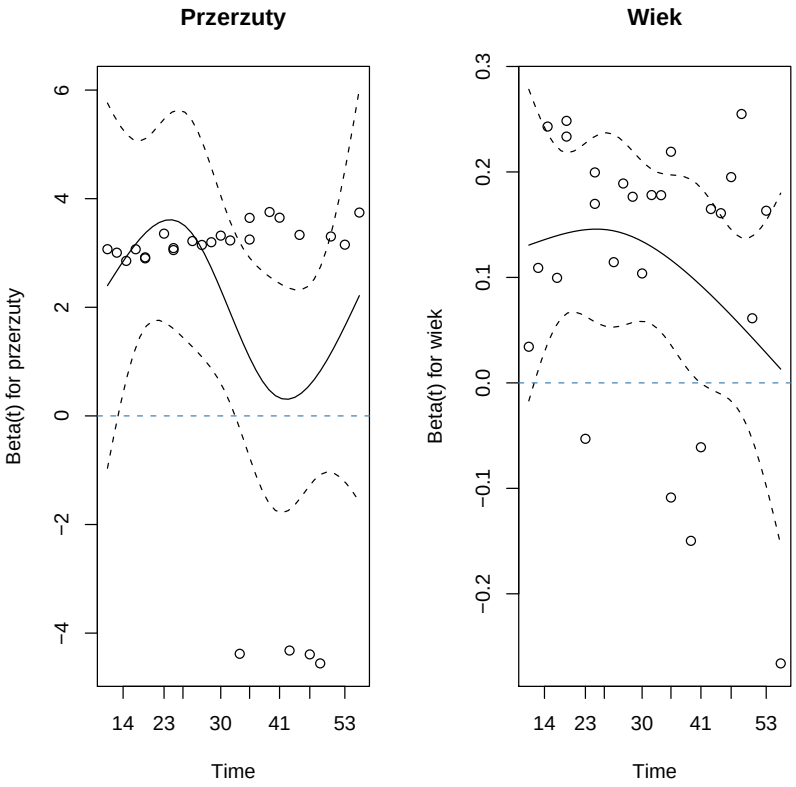
Jednym ze sposobów testowania proporcjonalności hazardu jest badanie wykresu zależności przeskalowanych residuów Schoenfeld'a (są one omówione w podrozdziale 1.7.4) od czasu wraz z dopasowaną wygładzoną funkcją zależności. Funkcja ta powinna być krzywą o nachyleniu zerowym ($y = a,\ a = \text{const.}$), gdzie wartość a to w przybliżeniu parametr β_i stojący przy badanej zmiennej. W programie R (pakiet *survival*) znajduje się funkcja *cox.zph*(), która służy do testowania założenia o proporcjonalnym hazardzie, bazująca właśnie na tym podejściu. Testuję więc hipotezę dla wszystkich zmiennych i całego modelu łącznie.

```
cox.zph(c, global=TRUE)
```

```
             rho      chisq         p
przerzuty  -0.310       2.49    0.1149
wiek       -0.267       2.17    0.1409
GLOBAL       NA         4.92    0.0856
```

P-wartość testu dla całego modelu wynosi 0.086, więc na poziomie istotności 0.05 nie ma podstaw do odrzucenia hipotezy zerowej o jednakowym bazowym hazardzie dla wszystkich obserwacji. Przy użyciu funkcji *cox.zph*() generuję wykresy Schoenfeld residuów dla zmiennych *wiek* i *przerzuty* - rysunek 3.7. Krzywe dopasowania nie są do końca równoległe do osi x.

Dla zmiennych dyskretnych założenie o proporcjonalnym hazardzie można również testować badając zależność $-\log(-\log(S(t)))$ od czasu dla każdego poziomu zmiennej. Krzywe dla różnych poziomów zmiennej powinny być do siebie równoległe. W badanym modelu jedyną zmienną dyskretną są *przerzuty*. Wykres $-\log(-\log(S(t)))$ dla pacjentek z przerzutami i bez przerzutów

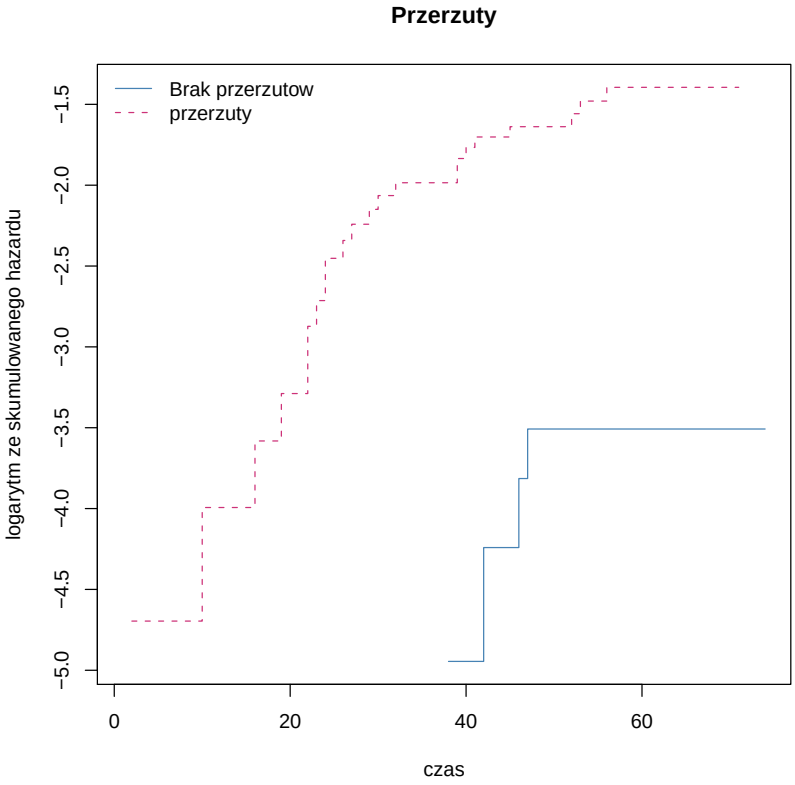


Rysunek 3.7: Testowanie założenia proporcjonalnego hazardu dla zmiennych *wiek* i *przerzuty* za pomocą wykresu residuów Schoenfeld'a.

znajduje się na rysunku 3.8. Badane krzywe wydają się być równoległe - nie ma więc podstaw do odrzucania hipotezy o proporcjonalnym hazardzie. Wynik jest bardziej rozstrzygający niż wnioski z wykresu 3.7.

```
#residua schoenfield'a
c=coxph(Surv(A[,1],A[,2])~przerzuty+wiek, method="breslow")
c.zph=cox.zph(c)
par(mfrow=c(3,2))
plot(c.zph[1], main="Przerzuty"); abline(h=0, lty=2, col="steelblue");
plot(c.zph[2], main="Wiek"); abline(h=0, lty=2, col="steelblue");

#log(-log S(t)) versus czas
#krzywe powinny być równoległe
c.km=survfit(Surv(A[,1],A[,2])~przerzuty, type="kaplan-meier")
c=c(rep(1, times=c.km$strata[1]),rep(2, times=c.km$strata[2]))
c.haz=as.data.frame(cbind(c=c,time=c.km$time, surv=c.km$surv))
c1=log(-log(c.haz$surv[c.haz$c==1]))
c2=log(-log(c.haz$surv[c.haz$c==2]))
a=c.haz$time[c.haz$c==1]
b=c.haz$time[c.haz$c==2]
plot(c(a,b),c(c1,c2),type="n",xlab="czas", ylab="logarytm ze
skumulowanego hazardu", main="Przerzuty")
lines(c.haz$time[c.haz$c==1], c1, type="s", lty=1, col="steelblue")
lines(c.haz$time[c.haz$c==2], c2, type="s", lty=2, col="violetred3")
legend(x="topleft",legend=c("Brak przerzutow","przerzuty"),
lty=c(1,2), bty="n", col=c("steelblue","violetred3"))
```

Bardziej formalnym podejściem jest estymacja modeli (dla każdej zmiennej objaśniającej



Rysunek 3.8: Wykres $-\log(-\log(S(t)))$ osobno dla pacjentek z przerzutami i bez przerzutów - testowanie założenia proporcjonalnego hazardu.

oddzielnie) uwzględniających dodatkowo interakcję zmiennej objaśniającej z czasem. Jeśli parametr przy tej interakcja jest istotny, zmienna objaśniająca nie spełnia badanego założenia.

Jeśli założenie proporcjonalnego hazardu nie jest spełnione dla którejś ze zmiennych należy wprowadzić różne bazowe funkcje hazardu w zależności od poziomów tej zmiennej. Wtedy nie jest jednak możliwe mierzenie wpływu stratyfikującej zmiennej na czas do wystąpienia zdarzenia, gdyż cały jej efekt jest zawarty w bazowych funkcjach hazardu.

W celu dodatkowego upewnienia się o poprawności założenia o proporcjonalności hazardu dla zmiennej *przerzuty*, estymuję model (*model Coxa 4*), gdzie funkcje hazardu bazowego dla pacjentek z przerzutami i bez przerzutów różnią się między sobą.

```
#Model Coxa 4

coxph(formula = Surv(A[, 1], A[, 2]) ~ strata(przerzuty) + wiek,
method = "breslow")


          coef    exp(coef) se(coef)    z           p
wiek     0.108     1.11     0.0236     4.55     5.3e-06

Likelihood ratio test=30.9  on 1 df, p=2.7e-08  n= 260
```

Wartość logarytmu funkcji wiarygodności jest jednak dużo niższa od tej dla modelu z jedną bazową funkcją hazardu (*model Coxa 3*), również wartość testu LR, która jest rzędu 10^{-6}, wskazuje na przewagę modelu z jedną bazową funkcją hazardu.

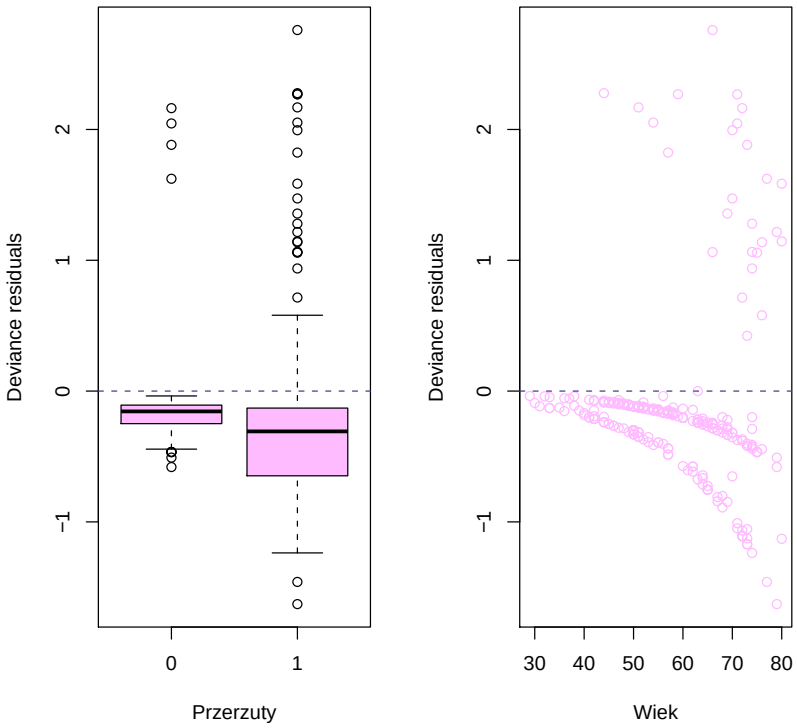
```
l=2*(coxph(Surv(A[,1],A[,2])~wiek+strata(przerzuty),method="breslow")$loglik[2]
-coxph(Surv(A[,1],A[,2])~wiek+przerzuty,method="breslow")$loglik[2])
1-pchisq(l,1)
#4.117351e-06
```

Przyjmuję więc, że zmienne spełniają założenie proporcjonalnego hazardu i ostatecznym modelem jest model z jednakową bazową funkcją hazardu i zmiennymi objaśniającymi: *wiek* oraz *przerzuty* (*model Coxa 3*).

3.5.4. Diagnostyka modelu

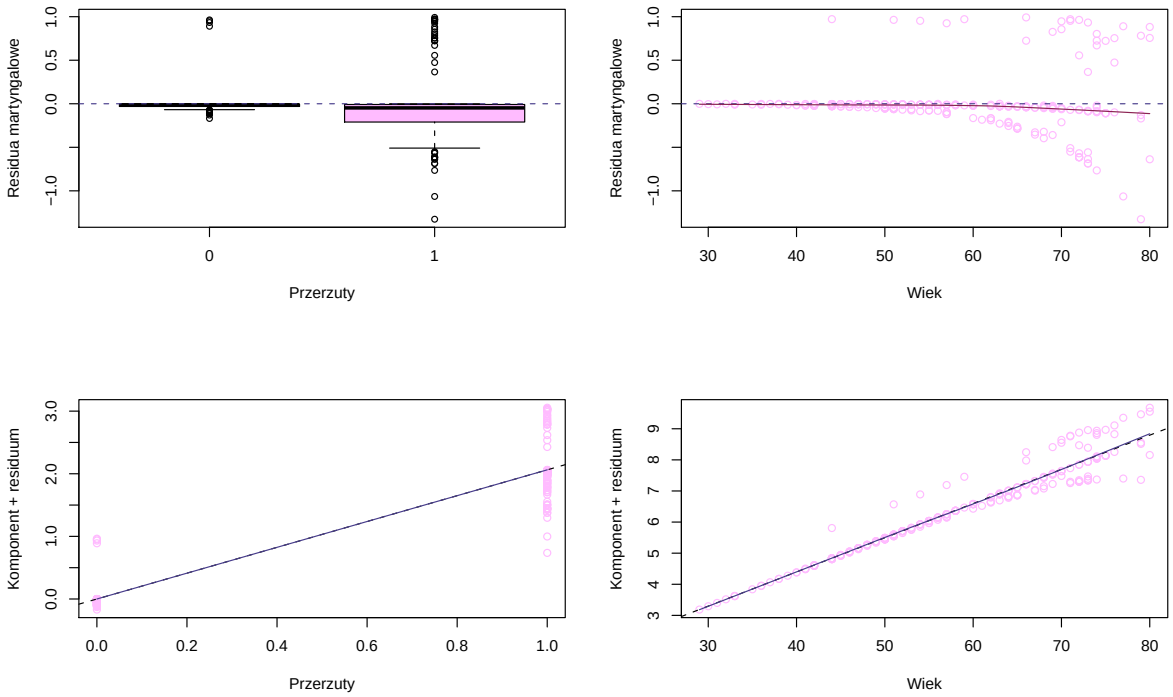
Rysunki 3.9 oraz 3.10 przedstawiają wykresy residuów martyngałowych i residuów deviance w zależności od zmiennych objaśniających: *wiek* i *przerzuty*. Charakterystyka tych residuów znajduje się w podrozdziałach 1.7.2 oraz 1.7.3. Oba rodzaje residuów przyjmują wartości mniejsze od zera dla zmiennych cenzurowanych, czyli w tym wypadku aż dla 90% obserwacji residua mają ujemną wartość. Dodatkowo, na wykresie znajduje się zależność między zmiennymi objaśniającymi a sumą komponenta (równego $\hat{\beta}_k x_{ki}$ dla k-tej zmiennej i i-tej obserwacji) i residuum martyngałowego. W poprawnym modelu powinna być ona liniowa.

Rysunek 3.11 zawiera natomiast wykresy dfbeta residuów. Dla każdej obserwacji i ($i = 1, \ldots, N$) pokazana jest wystandaryzowana zmiana w estymowanym parametrze $\hat{\beta}_k$ (dla k-tej zmiennej objaśniającej), gdy z modelu usunie się tą obserwację. Wykresy pozwalają na identyfikację obserwacji wpływowych (ew. outlierów). W analizowanym modelu parametr przy przerzutach jest stosunkowo odporny na poszczególne obserwacje, natomiast dla parametru zmiennej *wiek* widoczny jest wpływ kilku znaczących obserwacji. Może to wynikać z połączenia małej liczby obserwacji niecenzurowanych i ciągłości zmiennej *wiek*.



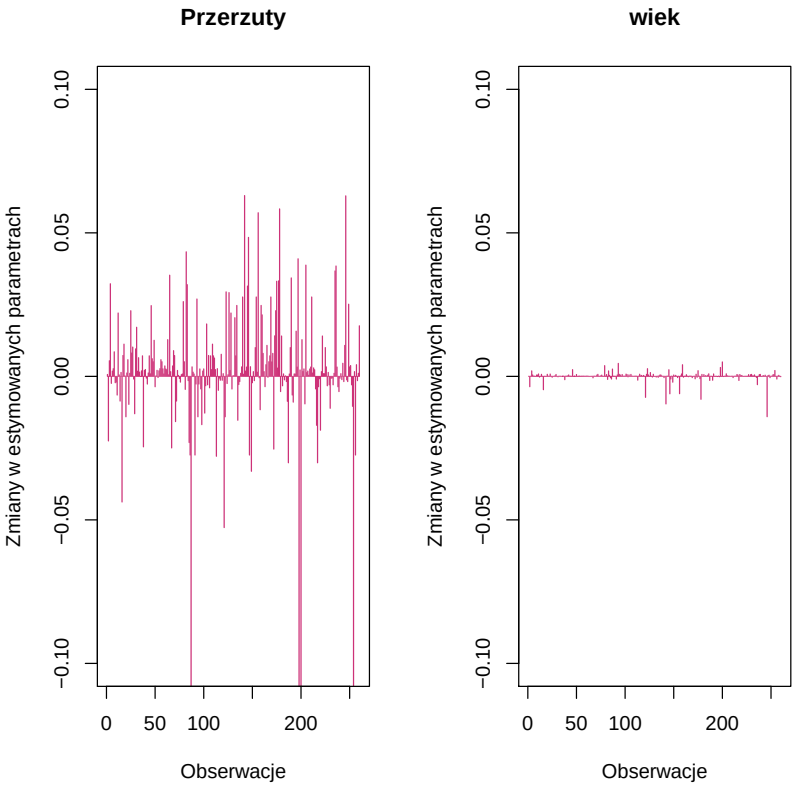
Rysunek 3.9: Wykresy residuów deviance dla zmiennych *wiek* i *przerzuty*.

```
#deviance residuals
c=coxph(Surv(A[,1],A[,2])~przerzuty+wiek, method="breslow")
c.res=residuals(c,type="deviance")
par(mfrow=c(3,2))
```



Rysunek 3.10: Wykresy residuów martyngałowych dla zmiennych *wiek* i *przerzuty*.

```
boxplot(c.res~przerzuty, xlab="Przerzuty", ylab="Deviance residuals",
col="plum1"); abline(h=0, lty=2, col=" slateblue4")
plot(wiek,c.res, xlab="Wiek", ylab="Deviance residuals", col="plum1")
abline(h=0, lty=2, col=" slateblue4")

#residua martyngalowe
res=residuals(c, type="martingale")
X=as.matrix(cbind(przerzuty, wiek))
boxplot(res~X[,1],xlab="Przerzuty", ylab="Residua martyngalowe", col="plum1")
abline(h=0, lty=2, col="slateblue4")
plot(X[,2], res,xlab="Wiek", ylab="Residua martyngalowe", col="plum1")
abline(h=0, lty=2, col="slateblue4")
lines(lowess(X[,2],res),col="slateblue4")

#cz2
b=coef(c[1:2])
for(j in 1:2){
plot(X[,j], b[j]*X[,j]+res, xlab=c("Przerzuty", "Wiek")[j],
ylab="Component + residuum",col="plum1")
abline(lm(b[j]*X[,j]+res~X[,j]),lty=2)
lines(lowess(X[,j],b[j]*X[,j]+res, iter=0),col="slateblue4")
}

#dfbeta residuals
c.res=resid(c, type="dfbeta")
par(mfrow=c(1,2))
main=c("Przerzuty","wiek")
for (i in 1:2){
plot(1:length(c.res[,1]), c.res[,i], type="h", xlab="Obserwacje",
ylim=c(-0.1,0.1),ylab="Zmiany w estymowanych parametrach",
col="violetred3")
title(main[i])
}
```



Rysunek 3.11: Badanie wpływu obserwacji na estymatory zmiennych *wiek* i *przerzuty*.

3.5.5. Zgodność dopasowania

Jednym ze sposobów badania dopasowania modelu jest porównywanie oczekiwanych (wynikających z modelu) czasów do wystąpienia zdarzenia z ich rzeczywistymi wartościami. Możliwe jest to jednak tylko dla zmiennych niecenzurowanych. W przypadku analizowanego modelu - gdzie tylko 26 obserwacji nie podlegało cenzurowaniu - takie podejście zdaje się być niewłaściwe.

Użyteczna byłoby natomiast statystyka pozwalająca oceniać i porównywać nieparametryczne modele Cox'a, jak na przykład R^2 dla modeli regresji liniowej. Schemper i Stare w [9] pokazali jednak, że taka prosta i uniwersalna miara nie istnieje. Hosmer i Lemeshow [6] zalecają stosowanie następującej statystyki:

$$R^2_M = 1 - \exp\left(\frac{2}{N}(L_0 - L_M)\right), \tag{3.2}$$

gdzie N - liczba wszystkich obserwacji, L_M - logarytm częściowej funkcji wiarygodności dla badanego modelu oraz L_0 - logarytm częściowej funkcji wiarygodności dla modelu tylko ze stałą. Dla badanego modelu, policzona bezpośrednio wartość 3.2 wynosi 18.6%, co pozwala na nieformalną interpretację, że model objaśnia prawie 20% badanego zjawiska.

```
c=coxph(Surv(A[,1],A[,2])~przerzuty+wiek, method="breslow")
r.square=1-exp((2/260)*(c$loglik[1]-c$loglik[2]))
r.square
#0.1864015
```

3.5.6. Interpretacja parametrów

Analogicznie jak przy estymacji modelem parametrycznym, wyznaczam zmiany w funkcji hazardu dla poszczególnych zmiennych.

- *Przerzuty*: $\exp(2.06) = 7.87$ (dla modelu parametrycznego wynosi 7.64)
- *Wiek*: $\exp(0.11) = 1.12$ (dla modelu parametrycznego wynosi 1.11)

Jak widać wyniki są bardzo zbliżone do tych uzyskanych modelem parametrycznym (z przeskalowanym czasem życia). Interpretacja pozostaje taka sama jak poprzednio, czyli

- u pacjentek z przerzutami ryzyko śmierci jest 7.87 razy większe niż dla tych bez przerzutów,
- pacjentka w wieku 65 lat ma $1.12^{(65-50)} = 5.47$ większe ryzyko śmierci niż pacjentka w wieku 50 lat (przy pozostałych charakterystykach niezmienionych).

Zakończenie

W pracy zostały przeprowadzone symulacje badające własności statystyczne estymatorów i testów wykorzystywanych w analizie przeżycia.

Okazuje się, że dla badanych rozkładów (pochodzących z rodziny rozkładów wykładniczych i Weibulla) estymator nieparametryczny Kaplana-Meiera jest obciążony - niedoszacowuje on funkcji przeżycia dla dużych wartości jej argumentów. Podobnie estymator Flemingtona-Harringtona, który dodatkowo przeszacowuje funkcję przeżycia dla pozostałych argumentów.

Z badania mocy testu log-rank dla rozkładu wykładniczego wynika, że test dobrze różnicuje próby nawet przy małej różnicy między parametrem λ. Gdy długość porównywanych prób jest większa niż 60, skuteczność testu jest wysoka (większa niż 80%). Procent obserwacji cenzurowanych wpływa negatywnie na moc testu, krzywa zależności ma jednak łagodne nachylenie.

Ostatnia grupa symulacji, badająca rozkład estymatorów parametrów w modelach parametrycznych za pomocą metody bootstrap, wskazuje na dość losowe kształtowanie się funkcji gęstości estymatorów (często moda nie odpowiada prawdziwej wartości parametru), bez względu na stopień cenzurowania i długość próby.

Symulacje zostały przeprowadzone dla konkretnych rozkładów, długości prób czy sposobu cenzurowania. Ciekawe wydaje mi się porównanie otrzymanych wyników z badaniem przeprowadzonym dla innych parametrów. W dodatku znajdują się zaimplementowane przez mnie funkcje przeprowadzające zaprezentowane badania dla zadanych argumentów. Ich wykorzystanie pozwala na automatyczne generowanie interesujących czytelnika analiz.

Rozdział trzeci opiera się na danych rzeczywistych dotyczących pacjentek chorych na raka piersi. Przeprowadzona analiza (składająca się m. in. z testów log-rank, modelu parametrycznego oraz modelu Cox'a) wskazuje na istotny wpływ występowania przerzutów oraz wieku w momencie rozpoznania choroby na długość życia pacjentek. Wartości estymowanych parametrów i uzyskana z nich miara wpływu zmiennych objaśniających na funkcję hazardu w modelu parametrycznym i modelu Cox'a są bardzo zbliżone - różnią się o mniej niż 3%. Okazuje się więc, że nawet przy 90% cenzurowaniu danych, możliwe jest uzyskanie w miarę stabilnych wyników.

Dodatek A

Kody programu R użyte w pracy

A.1. Badanie własności estymatorów funkcji przeżycia

Przedstawiony program bada własności estymatorów funkcji przeżycia na podstawie generowanych losowo danych.

Algorytm rozdzieliłam na trzy etapy. Pierwszy krok to losowanie obserwacji. Służy do tego funkcja *dprep()*, która wymaga następujących argumentów:

- a - określający liczbę obserwacji,
- lambda1 - zadający rozkład dla generowanych zmiennych,
- lambda2 – zadający rozkład cenzurujący obserwacje.

Bazowo funkcja losuje obserwacje z rozkładu wykładniczego Exp(lambda1) z cenzurowaniem wykładniczym Exp(lambda2). Jeśli jednak chcemy, by generowane zmienne lub zmienne cenzurujące pochodziły z rozkładu Weibulla, dodatkowo można zadać parametry

- p1 – parametr kształtu rozkładu generowanych zmiennych – Weibull(lambda1, p1)
- p2 – parametr kształtu rozkładu cenzurującego obserwacje – Weibull(lambda2, p2)

W wyniku otrzymujemy zmienne potrzebne do wyliczenia estymatora. Wektor Z zawiera dane z uwzględnionym cenzurowaniem, a N1 - informacje o cenzurowaniu dla poszczególnej obserwacji.

Funkcja *unbiased_survival_estimators()* generuje N estymatorów funkcji przeżycia, w oparciu o losowane funkcją *dprep()* zmienne. W rezultacie otrzymujemy macierz ERR, której kolumny zawierają prawdopodobieństwa przeżycia do czasu t=1,2,...,xlab dla konkretnego estymatora. Dla przykładu i-ty wiersz i j-ta kolumna macierzy ERR to prawdopodobieństwo przeżycia j-tego estymatora (uzyskanego w j-tej iteracji) dla czasu i. Parametry obowiązkowe są takie same, jak dla funkcji *dprep()*, tj. a, lambda1 i lambda2. Dodatkowo można zadać parametry kształtu (p1, p2), gdy chcemy losować z rozkładu Weibulla oraz następujące argumenty:

- N – określający liczbę iteracji - liczbę generowanych estymatorów, gdzie bazowo jest to 10 000 powtórzeń,
- xlab – czas, do którego estymowane są funkcje przeżycia, wyjściowo ustawiony na 60.
- method – rodzaj generowanych estymatorów. Wyjściowo method= "kaplan-meier", tzn. estymator Kaplana-Meiera, można jednak przeprowadzić symulacje dla estymatora Fleminga-Harringtona zadając: method="fleming-harrington".

Ostatnim etapem jest przedstawienie wyników w formie graficznej. Odbywa się ono za pomocą funkcji *plot_unbiased_survival()*. Wynikiem są dwa wykresy. Na pierwszym znajdują się

- dla czasów t=1,2,..., xlab wykresy pudełkowe (ang *box plot*) wygenerowane na podstawie estymowanych funkcji przeżycia
- prawdziwa funkcja przeżycia - zadana rozkładem, z którego losowane były obserwacje.

W celach pomocniczych zamieszczona jest również gęstość oraz funkcja przeżycia dla rozkładu, z którego pochodzą generowane obserwacje.

Wykres pierwszy pozwala na zbadanie własności (obciążalności, dyspersji) estymatorów dla konkretnych rozkładów i długości obserwacji.

```
library(survival)

dprep <- function(a, lambda1, lambda2, p1=1, p2=1){

 X=rweibull(a, shape=p1, scale = 1/lambda1)
 Y=rweibull(a, shape=p2, scale = 1/lambda2)
 N1=X<Y
 Z=pmin(X,Y)
 Z=round(Z,0.1)
 n=max(Z)
l  ist(Z=Z, N1=N1)
}
unbiased_survival_estimators <- function(a, lambda1, lambda2,
N=10000, xlab=60, p1=1, p2=1, method="kaplan-meier"){

 ERR=array(0, dim=c(N,xlab))
 for(k in 1:N){
     data=dprep(a, lambda1, lambda2, p1=p1, p2=p2)
     Z=data$Z
     N1=data$N1
     n=max(Z)
     if (method=="fleming-harrington")
         SS=survfit(Surv(Z,N1)~1, type="fleming-harrington")
      else
         SS=survfit(Surv(Z,N1)~1, type="kaplan-meier")
      SS1=summary(SS,1:n)$surv
      for(j in 1:min(n,xlab))
         ERR[k,j]=SS1[j]
      }
 list(ERR=ERR)
}

plot_unbiased_survival <- function(a, lambda1, lambda2,
N=10000, xlab=60, p1=1, p2=1, method="kaplan-meier"){

   layout(matrix(c(1,1,2,3), 2, 2, byrow=TRUE),respect=TRUE)
   A=unbiased_survival_estimators(a, lambda1, lambda2,
   N=N, xlab=xlab, p1=p1, p2=p2, method=method)$ERR
   S=rep(0,xlab)
   for(t in 1:xlab)
      S[t]=exp(-(t*lambda1)^p1)
   boxplot(A, outline=FALSE,, xlab="czas",ylab="S(t)",col="khaki1", notch=TRUE)
   lines(S,type="l", col="darkgreen", lwd=2)
   G=rep(0,xlab)
   for(t in 1:xlab)
     G[t]=p1*lambda1*(lambda1*t)^(p1-1)*exp(-(t*lambda1)^p1)
   plot(G,type="l", col="darkgreen", xlab="czas",ylab="gestosc",lwd=3)
   for(t in 1:xlab)
      G[t]=exp(-(t*lambda1)^p1)
   plot(G,type="l", col="darkgreen", xlab="czas",ylab="funkcja przezycia",lwd=3)
   list(A=A)
}
```

A.2. Badanie mocy testu log-rank

Poniższy kod służy do badania własności testu log-rank w zależności od długości próby, rozkładów, z których pochodzą obserwacje oraz poziomu cenzurowania. Test log-rank ma na celu sprawdzić, czy dwie podpróby pochodzą z tych samych rozkładów z hipotezą zerową o równości rozkładów. Dla obserwacji z jednakowych rozkładów test log-rank powinien przyjmować hipotezę zerową, w przeciwnym przypadku odrzucać. Prezentowany program symulacyjnie sprawdza moc testu, czyli procent dobrych decyzji dla konkretnego poziomu istotności.

Symulacje rozłożyłam na kilka etapów. W pierwszym, za pomocą funkcji *dprep*(), generuję obserwacje i zapisuję je w odpowiedniej formie. W tym celu muszę zadać rozkłady dla każdej z podprób, rozkład cenzurujący oraz stopień cenzurowania, gdzie

- a1 - liczba obserwacji w pierwszej podpróbie,
- a2 - liczba obserwacji w drugiej podpróbie,
- lambda1 - parametr zadający rozkład dla pierwszej podpróby. Wyjściowo losuję z rozkładu wykładniczego Exp(lambda1). Deklarując dodatkowo parametr p1 losowanie odbywa się z rozkładu Weibulla(lambda1, p1).
- lambda2 - parametr rozkładu wykładniczego dla drugiej podpróby, argument p2 określam, gdy chcę, by obserwacje pochodziły z rozkładu Weibulla(lambda2, p2).
- lambda3, p3 - tak jak w powyższych przypadkach zadaję rozkład (wykładniczy lub Weibulla) do cenzurowania obserwacji,
- p - liczba obserwacji, które mogą być poddane cenzurowaniu.

Procedura cenzurowania odbywa się następująco: Mając wektor XY, zawierający wygenerowane podpróby o długościach a1 i a2, pochodzące z zadanych rozkładów, losowo wybieranych jest p obserwacji i tylko te porównywane są z rozkładem cenzurującym. Z reguły przyjmuje się, że p=a1+a2, czyli wszystkie obserwacje mogą być cenzurowane. Zmienność p jest potrzebna przy badaniu mocy testu w zależności od poziomu cenzurowania. W tym przypadku logiczne jest również zakładanie jednakowej długości obu podgrup. Warto wspomnieć, że liczba cenzurowanych obserwacji jest zawsze mniejsza lub równa p. Parametr p określa, ile obserwacji może być cenzurowanych. Cenzurowane są natomiast tylko te, dla których wartość zmiennej cenzurującej jest mniejsza od wartości obserwacji.

Na wyjściu otrzymuje się wektory: Z -zawierający obserwacje z uwzględnionym cenzurowaniem, K - określający numer podgrupy oraz N - z informacją, czy zmienna Z była cenzurowana.

Rozważane są dwa przypadki

- gdy podpróby pochodzą z różnych rozkładów - wtedy moc testu to stosunek liczby testów odrzucających hipotezę zerową do wszystkich przeprowadzonych testów,
- gdy podpróby pochodzą z tych samych rozkładów - moc testu to stosunek liczby testów zachowujących hipotezę zerową do wszystkich przeprowadzonych testów.

Badaniu przy próbach pochodzących z różnych rozkładów służy funkcja *symlogrankcen*(), korzystająca między innymi z *dprep*(). Na wejściu należy zatem określić argumenty *dprep*() (długości podprób i parametry rozkładów). Dodatkowo można zadać parametry

- NN - liczbę przeprowadzanych iteracji - bazowo NN=10 000,

- m=100 - parametr określający dokładność badania. Zadaje liczbę poziomów dla zmiennej wpływającej na moc testu. Na przykład dla badania wpływu cenzurowania odpowiada procentom cenzurowania.
- pval - wartość kry,tyczną, przy której odrzuca się hipotezę zerową, pval=0.05
- var - sposób badania.

Argument var decyduje o motywacji przeprowadzanych symulacji. Może przyjmować następujące wartości:

- censoring,
- lambdadistance,
- oraz observlength.

Gdy var nie jest zadeklarowany "censoring" jest domyślnym parametrem. W tym przypadku badana jest zależność między mocą testu a procentem cenzurowania. Dla p (liczby obserwacji podlegającej cenzurowaniu) od 1 do całkowitej liczby próby, generowanych jest NN symulacji i na ich podstawie obliczana jest moc testu. W rezultacie otrzymuje się więc wektory

- alpha -liczbę odrzuceń hipotezy zerowej dla różnych poziomów cenzurowania,
- count - liczbę wygenerowanych obserwacji dla każdego poziomu cenzurowania (jak wiadomo jest ona różna od NN).

Gdy var="lambdadistance" w badaniu interesuje nas zależność między mocą testu a poziomem zróżnicowania rozkładów. Dokładniej, w kręgu zainteresowania znajduje się różnica między parametrem λ dla obu podprób. Z tego powodu parametry lambda1 i lambda2 mają tu trochę inną interpretację niż w pozostałych przypadkach. Pierwsza podpróba losowana jest bowiem z rozkładu o parametrze lambda1, parametr drugiego rozkładu znajduje się w przedziale (lambda1, lambda2]. Dla każdego przypadku określana jest różnica między parametrami rozkładów. Dla każdej różnicy przeprowadzanych jest NN symulacji, a o liczbie różnic decyduje parametr m. Na wyjściu dostajemy wektory alpha i count, czyli liczbę odrzuceń hipotezy zerowej i liczbę symulacji dla różnych poziomów parametru lambda2 (w tym przypadku jest to wektor o wartościach równych NN).

By opisana procedura odbywała się prawidłowo lambda1 musi być mniejsze od lambda2. Gdy tak nie jest przyjmuje się lambda1=min(lambda1,lambda2) oraz lambda2=max(lambda1,lambda2).

Ostatnią możliwą opcją - var="observlength" - jest badanie zależności między długością podprób a mocą testu. W tym przypadku zakłada się, że obie podpróby są równe i liczba obserwacji w każdej z podprób waha się w przedziale (a1,a2]. Wynikiem tej operacji są, analogicznie jak w pozostałych przypadkach, wektory alpha i count.

Ostatecznym wynikiem funkcji *symlogrankcen*() są wektory

- alpha1 - moc testu dla różnych poziomów badanej zmiennej,
- count - liczba symulacji przeprowadzonych w zależności od różnych poziomów badanej zmiennej.

Funkcja *symlogrankequal*() odpowiedzialna jest za symulacje dla podprób pochodzących z tych samych rozkładów. Jej działanie jest analogiczne jak dla *symlogrankcen*() (posiada również te same argumenty), jedyna różnica polega na innej definicji mocy testu. Dostępne są dwie metody badania var="observlength" i var="censoring".

Funkcja *symlogrank*() łączy opisane wyżej przypadki: podprób pochodzących z tych samych i różnych rozkładów. Jej bazą są funkcje *symlogrankcen*() oraz *symlogrankequal*().

Do wizualizacji wyników służy funkcja *logrankplotdep*(). Przedstawia ona moc testu w zależności od analizowanych zmiennych: procentu cenzurowania, różnicy między parametrami rozkładów oraz długości próby. Jej argumenty są tożsame z argumentami zagnieżdżonej w niej funkcji *symlogrank*().

```
library(survival)

dprep <- function(a1, a2, lambda1, lambda2, lambda3, p, p1=1, p2=1, p3=1){
   X=rweibull(a1, scale = 1/lambda1, shape=p1)
   Y=rweibull(a2, scale = 1/lambda2, shape=p2)
   XY=rep(0,a1+a2)
   for(i in 1:a1)
     XY[i]=X[i]
   for(i in (a1+1):(a1+a2))
     XY[i]=Y[i-a1]
   S=c(1:(a1+a2))
   S1=sample(S,p)
   N1=rep(0,a1+a2)
   for(i in 1:(a1+a2))
       for(j in 1:p)
         if(S1[j]==i) N1[i]=1
   N2=rweibull((a1+a2), scale = 1/lambda3, shape=p3)
   Z=XY
   N=rep(1,a1+a2)
   for(i in 1:(a1+a2)) {
     if(N1[i]==1){
       Z[i]=min(XY[i],N2[i])
       if (XY[i]>N2[i]) N[i]=0
       }
     }
   Z=round(Z,0.1)
   K=rep(0,a1+a2)
   for(i in 1:a1)
     K[i]=1
   list(Z=Z, N=N, K=K)
  }


symlogrankcen<- function(a1, a2, lambda1, lambda2, lambda3, m=100,
        NN=10000, p1=1, p2=1, p3=1, pval=0.05, var="censoring"){

   m1=m
   if(var=="censoring"){
     alpha=rep(0,m)
     count=rep(0,m)
     for(j in 1:NN){
       for(p in 1:(a1+a2)){
         dprep1=dprep(a1, a2, lambda1, lambda2,lambda3, p, p1,p2,p3)
         test=survdiff(Surv(dprep1$Z,dprep1$N)~dprep1$K,rho=0)
         pvalue=1-pchisq(test$chisq,1)
         odrzucenie=0
         if(pvalue<pval)
           odrzucenie=1
         x=((a1+a2)-sum(dprep1$N))/(a1+a2)*m
         x=round(x,0.1)
         alpha[x]=alpha[x]+odrzucenie
         count[x]=1+count[x]
         }
       }
     }
   if(var=="lambdadistance"){
     alpha=rep(0,m)
     count=rep(0,m)
   for(j in 1:NN){
     for(p in 1:m){
       dprep1=dprep(a1, a2, min(lambda1, lambda2),
       min(lambda1,lambda2)+max(lambda1-lambda2, lambda2-lambda1)/m*p,
```

```
       lambda3, a1+a2, p1,p2,p3)
       test=survdiff(Surv(dprep1$Z,dprep1$N)~dprep1$K,rho=0)
       pvalue=1-pchisq(test$chisq,1)
       odrzucenie=0
       if(pvalue<pval)
         odrzucenie=1
      alpha[p]=alpha[p]+odrzucenie
      count[p]=1+count[p]
      }
    }
   }
   if(var=="observlength"){
     if(max(a1-a2, a2-a1)<m1) m=max(a1-a2, a2-a1) else
     m=round(max(a1-a2, a2-a1)/round(max(a1-a2, a2-a1)/m1,0.1),0.1)
     alpha=rep(0,m)
     count=rep(0,m)
     for(j in 1:NN){
       for(p in 1:m){
         dprep1=dprep(min(a1,a2)+round(max(a1-a2,
         a2-a1)/m,0.1)*p,min(a1,a2)+round(max(a1-a2, a2-a1)/m,0.1)*p, lambda1,
         lambda2,lambda3, 2*( min(a1,a2)+round(max(a1-a2, a2-a1)/m,0.1)*p),
         p1,p2,p3)
         test=survdiff(Surv(dprep1$Z,dprep1$N)~dprep1$K,rho=0)
         pvalue=1-pchisq(test$chisq,1)
         odrzucenie=0
         if(pvalue<pval)
           odrzucenie=1
         alpha[p]=alpha[p]+odrzucenie
         count[p]=1+count[p]
         }
       }
   }
   alpha1=rep(0,m)
   count1=rep(0,m)
   for(i in 1:m) {
     count1[i]=count[i]
     if (count[i]==0) count1[i]=1
     alpha1[i]=100*alpha[i]/count1[i]
     }
   list(alpha1=alpha1, count=count)
 }


symlogrank<- function(a1, a2, lambda1, lambda2, lambda3, m=100,
        NN=10000, p1=1, p2=1, p3=1, pval=0.05, var="censoring"){

   if(lambda1==lambda2 && p1==p2){
     res=symlogrankequal(a1=a1, a2=a2, lambda1=lambda1,
     lambda2=lambda2, lambda3=lambda3, m=m, NN=NN, p1=p1, p2=p2, p3=p3,
     pval=pval, var=var)
     list(alpha1=res$alpha1, count=res$count)
     }
   else{
     res=symlogrankcen(a1=a1, a2=a2,
     lambda1=lambda1, lambda2=lambda2, lambda3=lambda3, m=m, NN=NN, p1=p1,
     p2=p2, p3=p3, pval=pval, var=var)
     list(alpha1=res$alpha1, count=res$count)
   }
 }


symlogrankequal<- function(a1, a2, lambda1, lambda2, lambda3, m=100,
        NN=10000, p1=1, p2=1, p3=1, pval=0.05, var="censoring"){

   m1=m
   if(var=="censoring"){
     alpha=rep(0,m)
     count=rep(0,m)
     for(j in 1:NN){
       for(p in 1:(a1+a2)){
```

```
        dprep1=dprep(a1, a2, lambda1, lambda2,lambda3, p, p1,p2,p3)
        test=survdiff(Surv(dprep1$Z,dprep1$N)~dprep1$K,rho=0)
        pvalue=1-pchisq(test$chisq,1)
        zachowanie=0
     if(pvalue>pval)
       zachowanie=1
     x=((a1+a2)-sum(dprep1$N))/(a1+a2)*m
     x=round(x,0.1)
     alpha[x]=alpha[x]+zachowanie
     count[x]=1+count[x]
    }
   }
  }
  if(var=="observlength"){
     if(max(a1-a2, a2-a1)<m1) m=max(a1-a2, a2-a1) else
     m=round(max(a1-a2, a2-a1)/round(max(a1-a2, a2-a1)/m1,0.1),0.1)
     alpha=rep(0,m)
     count=rep(0,m)
     for(j in 1:NN){
       for(p in 1:m){
         dprep1=dprep(min(a1,a2)+round(max(a1-a2,a2-a1)/m,0.1)*p,
         min(a1,a2)+round(max(a1-a2, a2-a1)/m,0.1)*p, lambda1, lambda2,
         lambda3, 2*(min(a1,a2)+round(max(a1-a2, a2-a1)/m,0.1)*p), p1, p2, p3)
         test=survdiff(Surv(dprep1$Z,dprep1$N)~dprep1$K,rho=0)
         pvalue=1-pchisq(test$chisq,1)
         zachowanie=0
         if(pvalue>pval)
           zachowanie=1
         alpha[p]=alpha[p]+zachowanie
         count[p]=1+count[p]
       }
     }
    }
   alpha1=rep(0,m)
   count1=rep(0,m)
   for(i in 1:m) {
     count1[i]=count[i]
     if(count[i]==0) count1[i]=1
     alpha1[i]=100*alpha[i]/count1[i]
     }
   list(alpha1=alpha1, count=count)
 }

Logrankplotdep<-function(a1, a2, lambda1, lambda2, lambda3, m=100,
        NN=10000, p1=1, p2=1, p3=1, pval=0.05, var="censoring"){

   symlogrankcen1=symlogrank(a1, a2, lambda1, lambda2, lambda3, m=m,
   NN=NN, p1=p1, p2=p2, p3=p3, pval=pval, var=var)
   if(var=="censoring"){
     labelx=seq(1,m,by=1)
     sym=symlogrankcen1$alpha1
     mincount=NN*0.5
     ending=m
     for(i in 2:m){
       if(symlogrankcen1$count[i]==0){
       sym[i]=max(symlogrankcen1$alpha1[i-1],sym[i-1])
       }
    if(symlogrankcen1$count[1]==0) sym[1]=sym[2]
    }
    b=seq(m,2,by=-1)
    for(i in b){
      if (symlogrankcen1$count[i]<mincount &
      symlogrankcen1$count[i-1]<mincount)ending=i-1
     }
    labelx=labelx*100/m
    plot(labelx[1:ending], sym[1:ending],
    type="l",xlab="Procent cenzurowanych obserwacji",ylab="Moc
    testu",ylim=c(0,100))
    data=sym[1:ending]
    labelx=labelx[1:ending]
```

```
    }
   if(var=="lambdadistance"){
     labelx=seq((lambda2-lambda1)/m,lambda2-lambda1,
     by=(lambda2-lambda1)/m)
     plot(labelx, symlogrankcen1$alpha1,
     type="l",xlab="Roznica miedzy parametrem rozkladu lambda",ylab="Moc
     testu",ylim=c(0,100))
    data=symlogrankcen1$alpha1
    }
   if(var=="observlength"){
     if(max(a1-a2, a2-a1)<m) m=max(a1-a2, a2-a1) else
        m=round(max(a1-a2, a2-a1)/round(max(a1-a2, a2-a1)/m,0.1),0.1)
     labelx=seq(min(a1,a2)+round(max(a1-a2, a2-a1)/m,0.1),a2,by=round(max(a1-a2,
     a2-a1)/m,0.1))
     plot(labelx, symlogrankcen1$alpha1,
     type="l",xlab="dlugosc proby",ylab="Moc testu", ylim=c(0,100))
     data=symlogrankcen1$alpha1
   }
   list(labelx=labelx, data=data)
}
```

A.3. Bootstrapowe badanie rozkładu estymowanych parametrów

Program służy do badania rozkładu parametrów w modelach parametrycznych z przeskalowanym czasem życia.

Na początku funkcją *dprep*() przygotowywane są dane do późniejszej analizy. Losowane są dwie próby, każda o długości a, odpowiednio z rozkładów Weibulla(lambda1,p1) i Weibulla(lambda2, p2) oraz rozkład cenzurujący obserwacje z obu prób, również z rozkładu Weibulla(lambda3, p3). Parametry a, lamda1, lambda2 i lambda3 są obowiązkowe, natomiast parametry kształtu w rozkładach (p1, p2, p3) można zadawać opcjonalnie. W przypadku ich braku przyjmują wartość 1, tzn. wszystkie obserwacje pochodzą z rozkładów wykładniczych. Wskazane jest by obie próby pochodziły z tego samego rodzaju rozkładu, np. obie były zadane jakimiś rozkładami wykładniczymi lub obie pochodziły z rozkładu Weibulla.

Dodatkowo konstruowana jest zmienna binarna x różnicująca dwie próby i przyjmująca wartość 1, gdy zmienna pochodzi z próby pierwszej. Na obu próbach (przy uwzględnieniu cenzurowania) estymowany jest model

$$\log(T) \sim \beta_0 + \beta_1 x + \frac{1}{k}\log(\varepsilon), \tag{A.1}$$

gdzie prawdziwe wartości β_0 i β_1 dla obserwacji z rozkładów wykładniczych (p1=p2=p3=1) są następujące:

$$\beta_1 = \log(\text{lambda2}) - \log(\text{lambda1}),\ \beta_0 = -\log(\text{lambda2}). \tag{A.2}$$

Na wyjściu funkcji *dprep*() dostaje się: Z - wektor wszystkich obserwacji (z obu prób) z uwzględnionym cenzurowaniem, N - wektor zawierający informacje, czy dana obserwacja była cenzurowana i x - wektor rozróżniający próby.

Docelowo badany jest rozkład parametru β_1 odpowiadającego binarnej zmiennej x. Analiza przeprowadzana jest metodą bootstrap. Losowana jest ze zwracaniem pewna podpróba z całego zbioru obserwacji i tylko na tej podpróbie estymowany jest model. Powtarzając czynność N razy - w tym przypadku 999 - dostaje się N różnych estymatorów dla parametrów modelu A.1. Do symulacji bootstapowych wykorzystywana jest znajdująca się w pakiecie *boot* funkcja *boot*(). Funkcja ta jako jeden z argumentów wymaga pewnej statystyki, czyli zmiennej, której rozkład chce się badać. Zadają ją funkcje *TestForBootExp*(), *TestForBootWeibull*() zwracające estymator parametru β_1 z modelu parametrycznego (odpowiednio z rozkładem Weibulla i wykładniczym) przeprowadzonego na wylosowanej wcześniej podpróbie.

Funkcja *bootBeta*() jest docelową funkcją, która zwraca wartości wyestymowanego parametru dla wszystkich symulacji (t) oraz rysuje wykres rozkładu tego parametru otrzymany bootstrapowo.

```
library(survival)
library(boot)

dprep<-function(a,lambda1, lambda2, lambda3, p1=1, p3=1){

   D1=rweibull(a, shape=p1, scale = 1/lambda1)
   D2=rweibull(a, shape=p1, scale = 1/lambda2)
   C=rweibull(2*a, shape=p3, scale = 1/lambda3)
   x1=rep(1,a)
   x2=rep(0,a)
   x=c(x1,x2)
   D=c(D1,D2)
   N=D<C
   Z=pmin(D,C)
   list(Z=Z, N=N, x=x)
}

TestForBootExp<-function(x,w){
   survreg(Surv(x[w,1],x[w,2])~1+x[w,3],dist="exponential")$coefficient[2]
}
TestForBootWeibull<-function(x,w){
   survreg(Surv(x[w,1],x[w,2])~1+x[w,3],dist="weibull",scale=p1)$coefficient[2]
}


bootBeta<-function(a,lambda1, lambda2, lambda3, p1=1, p3=1){

  data=dprep(a=a,lambda1=lambda1, lambda2=lambda2, lambda3=lambda3, p1=p1, p3=p3)
 if(p1==1){
        bootstr=boot(cbind(data$Z,data$N,data$x),TestForBootExp, R=999, stype="i")
         }
     if(p1!=1){
        bootstr=boot(cbind(data$Z,data$N,data$x),TestForBootWeibull, R=999, stype="i")
         }
par(mfrow=c(2,1))
plot(bootstr)
list(t=bootstr$t)
}
```

Bibliografia

[1] DR Cox, EJ Snell. A general definition of residuals. *Journal of the Royal Statistical Society. Series B (Methodological)*, 30(2):248–275, 1968.

[2] Internetowa encyklopedia www.wikipedia.org.

[3] R.D. Etzioni, E.J. Feuer, S.D. Sullivan, D. Lin, C. Hu, S.D. Ramsey. On the use of survival analysis techniques to estimate medical care costs. *Journal of Health Economics*, 18(3):367–382, 1999.

[4] J.P. Klein, M.L. Moeschberger. *Survival analysis: techniques for censored and truncated data*. Springer Verlag, 2003.

[5] L.R. Klein. *A textbook of econometrics*. Prentice Hall, 1974.

[6] S. Lemeshow, DW Hosmer Jr. A review of goodness of fit statistics for use in the development of logistic regression models. *American Journal of Epidemiology*, 115(1):92, 1982.

[7] S. Macran, H. Joshi, S. Dex. Employment after childbearing: a survival analysis. *Work, Employment & Society*, 10(2):273, 1996.

[8] Ekonometria III rok www.ekonometria.wne.uw.edu.pl.

[9] M. Schemper, J. Stare. Explained variation in survival analysis. *Statistics in Medicine*, 15(19):1999–2012, 1996.

[10] D. Schoenfeld. Partial residuals for the proportional hazards regression model. *Biometrika*, strony 239–241, 1982.

[11] GraphPad software www.graphpad.com/welcome.htm.

[12] M. Stevenson, I. EpiCentre. An Introduction to Survival Analysis. 2007.

[13] T.M. Therneau. A package for survival analysis in S. 1999.

[14] T.M. Therneau, P.M. Grambsch. *Modeling survival data: extending the Cox model*. Springer Verlag, 2000.

[15] T.M. Therneau, P.M. Grambsch, T.R. Fleming. Martingale-based residuals for survival models. *Biometrika*, 77(1):147, 1990.

[16] F. Abegaz P. Janssen I. V. Keilegom L. Duchateau Y. Getachew, B. Gashaw. Survival analysis. *Survival Analysis, course*. North-South-South project in Biostatistics Series VLIR-UOS, Belgium, 2009. Academic Year 2009-2010 Master in Biostatistics, South West Ethiopia.