

Sesja komputerowa 1: R

1. Do wczytywania danych przydatne będą funkcje z pakietu *foreign*. Inicjalizujemy go wykonując polecenie:

```
library(foreign)
```

Dane dotyczące choroby lokomocyjnej znajdują się w pliku **seasick_eng_data.dta** w formacie STATA. By je wczytać, używamy funkcji **read.dta**:

```
seasick <-  
read.dta("t:/burzykowski/biostat/datasets/seasick_eng_data.dta")
```

(Uwaga: nazwa katalogu może wymagać modyfikacji.) Zawartość obiektu (dataframe) **seasick** możemy sprawdzić używając funkcji **head**:

```
head(seasick)
```

Polecenie powoduje wydrukowanie pierwszych pięciu rekordów z obiektu. W celu uzyskania większej liczby, np. 10, rekordów, możemy użyć polecenia:

```
head(seasick, 10)
```

Wszystkie rekordy uzyskamy podając po prostu nazwę obiektu:

```
seasick
```

Obiekt zawiera trzy zmienne: *intens* wskazuje badanie (1, 2); *time* zawiera czas do torsji; *vomit* jest wskaźnikiem zdarzenia (0 = cenzurowanie, 1 = zdarzenie).

2. Dla celów analizy przeżycia możemy użyć funkcji z pakietu *survival*. Inicjalizujemy go wykonując polecenie:

```
library(survival)
```

3. Oszacowanie funkcji przeżycia można uzyskać przy pomocy funkcji **survfit()**. Opis funkcji otrzymujemy wydając polecenie **help(survfit)**.

Podstawowe argumenty funkcji to **formula** i **data**. Dodatkowe użyteczne argumenty to m.in. **subset**, **type**, **error**, **conf.type** i **conf.int**.

Oszacowanie funkcji przeżycia dla pierwszego badania choroby lokomocyjnej otrzymujemy przy pomocy polecenia:

```
seal.KM <- survfit(Surv(time, vomit) ~ 1, data=seasick, subset=intens==1,  
conf.type="none")
```

Obiekt **seal.KM** jest obiektem klasy *survfit*, zawierającym oszacowaną krzywą przeżycia oraz dodatkowe informacje i statystyki. **Surv(time, vomit)** definiuje obiekt klasy *Surv*, z czasem do wystąpienia zdarzenia (torsji) i wskaźnikiem zdarzeń zdefiniowanymi zmiennymi **time** i **vomit**, których wartości pobierane są z obiektu **seasick**. Formuła **Surv(time, vomit) ~ 1** definiuje jedną krzywą przeżycia, dla rekordów z obiektu **seasick** zdefiniowanych warunkiem **intens==1** (tj, dla obserwacji z pierwszego badania) podanym w argumencie **subset**. Argument **conf.type** używany jest do definicji metody wyznaczania przedziału ufności, który ma zostać dodany do wynikowego obiektu. Użycie **conf.type="none"** oznacza, że przedział ufności nie jest obliczany.

Wydruk rezultatów oszacowania uzyskujemy poprzez zastosowanie funkcji `summary()` do obiektu `sea1.KM`:

```
summary(sea1.KM)
```

Wynikiem zastosowania funkcji jest obiekt klasy `summary.survfit`, będący listą z wieloma składnikami zawierającymi oszacowanie funkcji przeżycia, błąd oszacowania, przedział ufności i inne informacje. Dokładny opis składników można uzyskać przy pomocy polecenia `help(survfit)`.

Wykres krzywej przeżycia otrzymujemy poprzez użycie funkcji `plot`:

```
plot(sea1.KM)
```

Opis osi współrzędnych, kolor i rodzaj linii można zmienić używając odpowiednich argumentów:

```
plot(sea1.KM,col=c("red"),lty=c(2),xlab="minutes",ylab="survival  
probability")
```

Listę argumentów i opis działania funkcji można uzyskać przy pomocy polecenia `help(plot.survfit)`.

Oszacowania funkcji przeżycia dla obu badań choroby lokomocyjnej otrzymujemy poprzez modyfikację argumentu `formula` funkcji `survfit()` i usunięcie argumentu `subset`:

```
sea2.KM <- survfit(Surv(time, vomit) ~ intens, data=seasick,  
conf.type="none")
```

Zastosowanie funkcji `plot()` daje wykresy krzywych przeżycia:

```
plot(sea2.KM,col=c("red","blue"),lty=c(1,2),xlab="minutes",ylab="survival  
probability")
```

4. Porównywanie krzywych przeżycia umożliwia funkcja `survdiff()`. Opis funkcji uzyskujemy wydając polecenie `help(survdiff)`. Jej syntaks jest podobny do syntaksu funkcji `survfit()` (zob. punkt 3). Użyjemy jej do porównania krzywych przeżycia uzyskanych dla badań choroby lokomocyjnej. Odpowiednie polecenie ma następującą postać:

```
sea.test <- survdiff(Surv(time, vomit) ~ intens, data=seasick, rho=0)
```

Funkcja używa rodziny testów zaproponowanej przez Fleminga i Harringtona z $q=0$. Domyślnie używany jest test logrank, odpowiadający wartości $p=0$ (w syntaksie funkcji odpowiada to argumentowi `rho=0`). Rezultat testu zapisany jest w obiekcie `sea.test`. Wynik testu otrzymujemy przez zastosowanie do obiektu funkcji `print()`:

```
print(sea.test)
```

Rezultat odpowiada informacji podanej na wykładzie.

5. Dane dotyczące ekspresji białka p53 znajdują się w pliku `nsclc_eng.dta` w formacie STATA. By je wczytać, używamy funkcji `read.dta`:

```
nsclc <- read.dta("t:/burzykowski/biostat/datasets/nsclc_eng.dta")
```

(Uwaga: nazwa katalogu może wymagać modyfikacji.) Krzywe przeżycia dla grup zdefiniowanych ekspresją białka otrzymujemy przy pomocy poleceń

```
nsclcl.KM <- survfit(Surv(survtime, survind) ~ expression, data=nsclcl,
conf.type="none")
```

```
plot(nsclcl.KM,col=c("red","blue"),lty=c(1,2))
```

Test logrank dla ekspresji białka uzyskujemy poprzez wydanie poleceń

```
nsclcl.logrank <- survdiff(Surv(survtime, survind) ~ expression, data=nsclcl)
print(nsclcl.logrank)
```

Warstwowy test ze względu na TNM otrzymujemy używając poleceń

```
nsclcl.strat <- survdiff(Surv(survtime, survind) ~ expression + strata(tnm),
data=nsclcl)
print(nsclcl.strat)
```

W celu uzyskania testu logrank na trend dla TNM musimy użyć funkcji z pakietu *survMisc*:

```
library(survMisc)
```

Pakiet zawiera funkcje rozszerzające zakres metod dostępnych w pakiecie *survival*. W szczególności zawiera funkcję **ten**, która przygotowuje dane (time, events, number at risk) do analizy. Syntaks jest bardzo podobny do tego używanego dla funkcji z pakietu *survival*:

```
nsclcl.trend <- ten(Surv(survtime, survind) ~ tnm, data=nsclcl)
```

number at risk) do analizy. Syntaks jest bardzo podobny do tego używanego dla funkcji z pakietu *survival*:

```
nsclcl.trend <- ten(Surv(survtime, survind) ~ tnm, data=nsclcl)
```

Obiekt **nsclcl.trend** (klasy *data.table* i *data.frame*) zawiera pięć kolumn: t (time), e (number of events), n (number at risk), cg (covariate group) oraz ncg (number at risk by covariate group). Obiektu możemy użyć jako argumentu dla funkcji **comp**, która dodaje do obiektu dodatkowe atrybuty zawierające wyniki testów porównujących krzywe przeżycia. Domyślnie dodawane są dwa atrybuty: *lrt* (logrank family of tests, czyli wyniki testu logrank i jego alternatyw) oraz *lrw* (logrank weights, czyli wagi użyte do konstrukcji testów). Jeśli liczba krzywych przeżycia (odpowiadająca liczbie kombinacji poziomów zmiennych użytych w definicji obiektu) jest większa niż 2, dodatkowym atrybutem jest *tft* (tests for trend, czyli testy na trend). Użycie polecenia:

```
comp(nsclcl.trend)
```

powoduje wydruk rezultatów (wartości statystyk testowych, stopni swobody oraz poziomów krytycznych) dla testów: logrank, Wilcoxon-Gehana, Tarone'a-Ware'a, Peto-Peto-Prentice'a, modyfikacji testu PPP, oraz testu Fleminga-Harringtona dla $q=p=1$. Dodatkowo (w części zaznaczonej jako „tft”) podawane są wyniki testu na trend (licznik Q i mianownik Var statystyki testowej, $Z=\sqrt{Q^2/\text{Var}}$ oraz poziom krytyczny testu χ^2 odpowiadającego Z^2) dla rodzin testów wymienionych powyżej. Wydruk zawiera również wagi użyte do wyznaczenia statystyki testowej testu na trend (w naszym przypadku to wartości odpowiadające wartościom zmiennej TNM). Jeśli interesuje nas tylko test na trend, możemy zażądać wydruku wyników tylko dla atrybutu *tft* obiektu **nsclcl.trend**:

```
attr(nsclcl.trend,"tft")
```

Ćwiczenia dodatkowe (samodzielnie)

1. Użyj poleceń `help(survfit.formula)`, `help(summary.survfit)` i `help(plot.survfit)` i przeczytaj uzyskane w ten sposób opisy funkcji.

2. Wykonaj następujące polecenia:

```
seal.KM.1 <- survfit(Surv(time, vomit) ~ 1, data=seasick, subset=intens==1)
summary(seal.KM.1)
plot(seal.KM.1)
```

Wynik oszacowania funkcji przeżycia został uzupełniony o 95% przedziały ufności. Dlaczego?

Jakiej metody użyto do skonstruowania przedziałów ufności?

W jaki sposób można uzyskać 99% przedziały ufności?

W jaki sposób można uzyskać 95% przedziały ufności oparte na transformacji log(-log) funkcji przeżycia?

Wykres krzywej przeżycia został uzupełniony o przedziały ufności. Dlaczego?

3. Użyj polecenia `help(survdiff)` i przeczytaj uzyskane w ten sposób opis funkcji `survdiff()`.

4. Obiekt `nsclc.strat` utworzony w punkcie 5 zawiera wynik warstwowego testu logrank dla ekspresji białka ze względu na TNM.

W jaki sposób, używając tego obiektu, można uzyskać informację o zaobserwowanych i oczekiwanych liczbach zgonów w poszczególnych warstwach?

Jakim poleceniem można uzyskać warstwowy, ze względu na TNM, test Peto-Peto-Prentice'a dla ekspresji białka?

Sesja komputerowa 1: SAS

1. Dane dotyczące choroby lokomocyjnej znajdują się w pliku `seasick_eng_data.sas7bdat`. Aby uzyskać do nich dostęp, musimy najpierw wskazać katalog, w którym znajduje się plik. W tym celu używamy komendy `libname`:

```
libname pw " t:/burzykowski/biostat/datasets";
```

(Uwaga: nazwa katalogu może wymagać modyfikacji.)

2. Podstawową procedurą dla potrzeb szacowania i testowania funkcji przeżycia jest PROC LIFETEST.

Aby uzyskać oszacowanie funkcji przeżycia dla pierwszego badania choroby lokomocyjnej, używamy następującego syntaksu:

```
proc lifetest data=pw.seasick_eng_data;
    where (intens=1);
    time time*vomit(0);
run;
```

Polecenie `where (intens=1);` ogranicza zakres procedury do dancyh dla pierwszego badania. Polecenie `time time*vomit(0);` wskazuje zmienną zawierającą czas obserwacji i wskaźnik cenzurowania. W nawiasie podawane są wartości, które identyfikują obserwacje cenzurowane.

Wyniki pojawiają się w oknie *Output*. Błąd standardowy oszacowania wyznaczany jest metodą Greenwooda.

Aby uzyskać wykres funkcji przeżycia, używamy opcji `plots`:

```
proc lifetest data= pw.seasick_eng_data plots=(s);
    where (intens=1);
    time time*vomit(0);
run;
```

Wykres pojawia się w oknie *Graph* (lub w pliku HTML, jeśli ta opcja jest użyta).

Przedział ufności otrzymujemy przy pomocy opcji `outsurv`:

```
proc lifetest data= pw.seasick_eng_data outsurv=surv_ci;
    where (intens=1);
    time time*vomit(0);
run;
```

Użycie opcji powoduje utworzenie (roboczego) zbioru `surv_ci` zawierającego oszacowanie funkcji przeżycia i 95% przedział ufności obliczony przy użyciu transformacji $\log(-\log)$.

Alternatywnym rozwiązaniem jest użycie polecenia `survival`:

```
proc lifetest data= pw.seasick_eng_data;
```

```

        where (intens=1);
        time time*vomit(0);
        survival out=surv_ci;
run;

```

Użycie tego polecenia również powoduje utworzenie (roboczego) zbioru `surv_ci` zawierającego oszacowanie funkcji przeżycia i 95% przedział ufności obliczony przy użyciu transformacji $\log(-\log)$. Polecenie to daje jednak więcej możliwości m.in. dotyczących rodzajów przedziałów ufności. Do tego celu służy opcja `conftype`. Np. 95% przedział ufności oparty o błąd standardowy wyznaczony metodą Greenwooda uzyskujemy następująco:

```

proc lifetest data= pw.seasick_eng_data;
    where (intens=1);
    time time*vomit(0);
    survival out=surv_ci conftype=linear;
run;

```

Polecenie `survival` pozwala również na obliczenie obszaru ufności dla funkcji przeżycia przy pomocy opcji `confband`.

Oszacowanie i wykres funkcji przeżycia dla obu badań choroby lokomocyjnej otrzymujemy przez użycie następującego syntaksu:

```

proc lifetest data= pw.seasick_eng_data plots=(s);
    time time*vomit(0);
    strata intens /notest;
run;

```

Polecenie `strata intens` wskazuje zmienną `intens` jako definiującą warstwę. Funkcje przeżycia są szacowane osobno dla warstw. Opcja `notest` wyłącza obliczenia testów porównujących funkcje przeżycia. W celu uzyskania testu logrank, syntaks modyfikujemy następująco:

```

proc lifetest data= pw.seasick_eng_data plots=(s);
    time time*vomit(0);
    strata intens /test=(logrank);
run;

```

Opcja `test` daje możliwość obliczenia różnych testów (również jednocześnie; zob. dokumentacja procedury LIFETEST). Użycie polecenia `strata intens;` przez domniemanie jest równoważne użyciu polecenia `strata intens/ test=(logrank wilcoxon lr);` tzn. zakłada jednocześnie obliczanie wartości testów logrank, Wilcoxona-Gehana i testu ilorazu funkcji wiarygodności.

3. Dane dotyczące ekspresji białka znajdują się w pliku `nsclc_eng.sas7bdat`.

Aby uzyskać oszacowanie funkcji przeżycia, wykres, i test logrank dla grup zdefiniowanych ekspresją białka, używamy następującego syntaksu:

```

proc lifetest data= pw.nsclc_eng plots=(s);
    time survtime*survind(0);
    strata expression / test=(logrank);

```

```
run;
```

Warstwowy test logrank ze względu na TNM uzyskujemy poprzez następującą modyfikację polecenia strata:

```
proc lifetest data= pw.nsc1c_eng plots=(s);  
    time survtime*survind(0);  
    strata tnm / group=expression test=(logrank);  
run;
```

Opcja group wskazuje zmienną expression, dla której test(y), wskazane w opcji test, mają być warstwowane ze względu na zmienną tnm. Warto zwrócić uwagę, że wykresy funkcji przeżycia są tworzone dla zmiennej tnm, a nie expression.

Test logrank na trend dla TNM uzyskujemy poprzez następującą modyfikację polecenia strata:

```
proc lifetest data= pw.nsc1c_eng;  
    time survtime*survind(0);  
    strata tnm / test=(logrank) trend;  
run;
```

Opcja trend wskazuje, że obliczona ma być wersja testu logrank na trend. Ponieważ zmienna tnm jest numeryczna, jej wartości użyte są jako wagi (zob. dokumentacja procedury LIFETEST).

Ćwiczenia dodatkowe (samodzielnie)

1. Przeczytaj dokumentację polecenia proc lifetest.

Jaka opcja umożliwia uzyskanie przedziału ufności na poziomie 99%?

2. Przeczytaj dokumentację polecenia strata.

Jakiego syntaksu należałoby użyć aby uzyskać test Peto-Peto-Prentice na trend?

3. Przeczytaj dokumentację polecenia survival.

Ile różnych rodzajów przedziałów ufności można uzyskać przy pomocy opcji conftype?

Ile różnych rodzajów obszarów ufności można uzyskać przy pomocy opcji confband?

Dla danych z drugiego badania choroby lokomocyjnej oblicz 99% przedział ufności przy użyciu transformacji log(-log) oraz 99% obszar ufności o równej precyzji dla przedziału czasu [10 min, 60 min]. Porównaj wyniki.