

# Analiza Przeżycia Raport 4

Paweł Matłowski  
album 249732

25 lutego 2021

## Spis treści

<b>1</b>	<b>Lista nr 1</b>	<b>2</b>
1.1	Zadanie nr 1 . . . . .	2
1.2	Zadanie nr 2 . . . . .	5
1.3	Zadanie nr 3 . . . . .	7
1.4	Zadanie nr 4 . . . . .	8
1.5	Zadanie nr 5 . . . . .	8
1.6	Zadanie nr 6 . . . . .	15

# 1 Lista nr 1

```
library(survival)
library(ggplot2)
library(survminer)

## Loading required package: ggpubr
## Loading required package: magrittr

library(timereg)
library(nltm)
df <- lung
df$status <- as.factor(df$status)
df$sex <- as.factor(df$sex)
df$ph.ecog <- as.factor(df$ph.ecog)
df$ph.karno <- as.factor(df$ph.karno)
df$pat.karno <- as.factor(df$pat.karno)
df <- subset(df, (meal.cal!="NA") & (wt.loss!="NA"))
attach(df)
```

## 1.1 Zadanie nr 1

Dopasujemy model proporcjonalnych szans, uwzględniając wszystkie zmienne objaśniające za wyjątkiem zmiennej *inst*.

```
ageCEN <- age - mean(age)
meal.calCEN <- meal.cal - mean(meal.cal)
wt.lossCEN <- wt.loss - mean(wt.loss)
modelPSa <- prop.odds(Event(time, status==2)~ageCEN+sex+ph.ecog+ph.karno+pat.karno+meal.

summary(modelPSa)

## Proportional Odds model
##
## Did not converge, allow more iterations
##
## Test for baseline
## Test for nonparametric terms
##
## Test for non-significant effects
##          Supremum-test of significance p-value H_0: B(t)=0
## Baseline                6.27                0
##
## Test for time invariant effects
##          Kolmogorov-Smirnov test p-value H_0:constant effect
## Baseline                1.72                0.108
##
```

```
## Covariate effects
##          Coef. SE Robust SE D2log(L)^-1      z P-val lower2.5% upper97.5%
## ageCEN      5.87e-03  0          0          0 Inf      0  5.87e-03  5.87e-03
## sex2       -6.08e-01  0          0          0 -Inf     0 -6.08e-01 -6.08e-01
## ph.ecog1     6.40e-01  0          0          0 Inf      0  6.40e-01  6.40e-01
## ph.ecog2     1.32e+00  0          0          0 Inf      0  1.32e+00  1.32e+00
## ph.ecog3     2.55e+00  0          0          0 Inf      0  2.55e+00  2.55e+00
## ph.karno60    1.03e+00  0          0          0 Inf      0  1.03e+00  1.03e+00
## ph.karno70    1.00e+00  0          0          0 Inf      0  1.00e+00  1.00e+00
## ph.karno80    1.17e+00  0          0          0 Inf      0  1.17e+00  1.17e+00
## ph.karno90    1.31e+00  0          0          0 Inf      0  1.31e+00  1.31e+00
## ph.karno100   1.46e+00  0          0          0 Inf      0  1.46e+00  1.46e+00
## pat.karno40  -3.52e-01  0          0          0 -Inf     0 -3.52e-01 -3.52e-01
## pat.karno50    7.53e-01  0          0          0 Inf      0  7.53e-01  7.53e-01
## pat.karno60    1.23e-01  0          0          0 Inf      0  1.23e-01  1.23e-01
## pat.karno70  -1.74e-01  0          0          0 -Inf     0 -1.74e-01 -1.74e-01
## pat.karno80  -2.81e-01  0          0          0 -Inf     0 -2.81e-01 -2.81e-01
## pat.karno90  -6.89e-02  0          0          0 -Inf     0 -6.89e-02 -6.89e-02
## pat.karno100 -5.68e-01  0          0          0 -Inf     0 -5.68e-01 -5.68e-01
## meal.calCEN  -4.43e-05  0          0          0 -Inf     0 -4.43e-05 -4.43e-05
## wt.lossCEN   -1.39e-02  0          0          0 -Inf     0 -1.39e-02 -1.39e-02
## Test of Goodness-of-fit
##          sup|   hat U(t) | p-value H_0
## ageCEN                86.400      0.266
## sex2                   3.870      0.420
## ph.ecog1                4.780      0.300
## ph.ecog2                4.880      0.190
## ph.ecog3                0.531      0.308
## ph.karno60              1.970      0.628
## ph.karno70              3.350      0.338
## ph.karno80              2.870      0.596
## ph.karno90              2.150      0.880
## ph.karno100             3.170      0.340
## pat.karno40             0.777      0.560
## pat.karno50             0.956      0.614
## pat.karno60             4.250      0.156
## pat.karno70             1.810      0.848
## pat.karno80             2.370      0.732
## pat.karno90             3.070      0.524
## pat.karno100            2.520      0.528
## meal.calCEN            3570.000      0.376
## wt.lossCEN             78.600      0.740
```

Jak wynika z podsumowania modelu, metoda iteracji nie jest zbieżna (does not converge), zatem sprawdzimy czy możemy rozwiązać ten problem zwiększając ilość iteracji dwukrotnie.

```
modelPSb <- prop.odds(Event(time, status==2)~ageCEN+sex+ph.ecog+ph.karno+pat.karno+meal.
```

```
summary(modelPSb)

## Proportional Odds model
##
## Did not converge, allow more iterations
##
## Test for baseline
## Test for nonparametric terms
##
## Test for non-significant effects
##          Supremum-test of significance p-value H_0: B(t)=0
## Baseline                6.27                0
##
## Test for time invariant effects
##          Kolmogorov-Smirnov test p-value H_0:constant effect
## Baseline                1.72                0.113
##
## Covariate effects
##          Coef. SE Robust SE D2log(L)^-1      z P-val lower2.5% upper97.5%
## ageCEN          5.87e-03  0          0          0 Inf      0  5.87e-03  5.87e-03
## sex2            -6.08e-01  0          0          0 -Inf     0 -6.08e-01 -6.08e-01
## ph.ecog1        6.40e-01  0          0          0 Inf      0  6.40e-01  6.40e-01
## ph.ecog2        1.32e+00  0          0          0 Inf      0  1.32e+00  1.32e+00
## ph.ecog3        2.55e+00  0          0          0 Inf      0  2.55e+00  2.55e+00
## ph.karno60       1.03e+00  0          0          0 Inf      0  1.03e+00  1.03e+00
## ph.karno70       1.00e+00  0          0          0 Inf      0  1.00e+00  1.00e+00
## ph.karno80       1.17e+00  0          0          0 Inf      0  1.17e+00  1.17e+00
## ph.karno90       1.31e+00  0          0          0 Inf      0  1.31e+00  1.31e+00
## ph.karno100      1.46e+00  0          0          0 Inf      0  1.46e+00  1.46e+00
## pat.karno40      -3.52e-01  0          0          0 -Inf     0 -3.52e-01 -3.52e-01
## pat.karno50       7.53e-01  0          0          0 Inf      0  7.53e-01  7.53e-01
## pat.karno60       1.23e-01  0          0          0 Inf      0  1.23e-01  1.23e-01
## pat.karno70      -1.74e-01  0          0          0 -Inf     0 -1.74e-01 -1.74e-01
## pat.karno80      -2.81e-01  0          0          0 -Inf     0 -2.81e-01 -2.81e-01
## pat.karno90      -6.89e-02  0          0          0 -Inf     0 -6.89e-02 -6.89e-02
## pat.karno100     -5.68e-01  0          0          0 -Inf     0 -5.68e-01 -5.68e-01
## meal.calcEN      -4.43e-05  0          0          0 -Inf     0 -4.43e-05 -4.43e-05
## wt.lossCEN       -1.39e-02  0          0          0 -Inf     0 -1.39e-02 -1.39e-02
## Test of Goodness-of-fit
##          sup|  hat U(t) | p-value H_0
## ageCEN                86.400        0.284
## sex2                   3.870        0.404
## ph.ecog1               4.780        0.294
## ph.ecog2               4.890        0.199
## ph.ecog3               0.531        0.306
## ph.karno60             1.970        0.618
## ph.karno70             3.370        0.328
## ph.karno80             2.870        0.610
```

## ph.karno90	2.160	0.852
## ph.karno100	3.170	0.311
## pat.karno40	0.777	0.554
## pat.karno50	0.956	0.571
## pat.karno60	4.250	0.165
## pat.karno70	1.830	0.837
## pat.karno80	2.520	0.688
## pat.karno90	3.070	0.538
## pat.karno100	2.520	0.506
## meal.calCEN	3580.000	0.390
## wt.lossCEN	78.400	0.734

Widzimy, że nasza metoda dalej "nie radzi" sobie, z naszym zbiorem danych.

## 1.2 Zadanie nr 2

Powtórzmy zadanie 1, tym razem nie uwzględniając zmiennych objaśniających *meal.cal* oraz *age*.

```
modelPS2 <- prop.odds(Event(time, status==2)~sex+ph.ecog+ph.karno+pat.karno+wt.lossCEN, data=ps2)

summary(modelPS2)
```

```
## Proportional Odds model
##
## Test for baseline
## Test for nonparametric terms
##
## Test for non-significant effects
##      Supremum-test of significance p-value H_0: B(t)=0
## Baseline                0.397                0.891
##
## Test for time invariant effects
##      Kolmogorov-Smirnov test p-value H_0:constant effect
## Baseline                1                0.71
##
## Covariate effects
##      Coef.      SE Robust SE D2log(L)^-1      z      P-val lower2.5%
## sex2      -1.0400 0.3460      0.3560      0.3250 -2.910 0.00367      -1.7200
## ph.ecog1      0.6060 0.5090      0.5180      0.5300  1.170 0.24200      -0.3920
## ph.ecog2      1.7300 0.8540      0.8910      0.8330  1.940 0.05200       0.0562
## ph.ecog3      2.7100 1.4700      1.1500      1.6900  2.350 0.01880      -0.1710
## ph.karno60     1.3100 1.6800      2.3200      1.3200  0.563 0.57400     -1.9800
## ph.karno70     1.3800 1.6300      2.3200      1.3000  0.596 0.55100     -1.8100
## ph.karno80     1.4700 1.6400      2.4100      1.3600  0.612 0.54100     -1.7400
## ph.karno90     1.6300 1.6500      2.4200      1.3800  0.674 0.50000     -1.6000
## ph.karno100    1.5400 1.7300      2.4700      1.4800  0.623 0.53400     -1.8500
## pat.karno40     0.6390 2.5200      3.3300      2.2200  0.192 0.84800     -4.3000
```

```

## pat.karno50    1.6100 1.4200    1.1900    1.6000 1.350 0.17600 -1.1700
## pat.karno60    0.9550 1.0400    0.7590    1.3100 1.260 0.20800 -1.0800
## pat.karno70    0.1140 1.1100    0.7840    1.3300 0.145 0.88500 -2.0600
## pat.karno80    0.1120 1.1100    0.8010    1.3300 0.140 0.88800 -2.0600
## pat.karno90    0.4520 1.1300    0.8330    1.3400 0.542 0.58800 -1.7600
## pat.karno100 -0.3470 1.2300    0.9580    1.3800 -0.362 0.71700 -2.7600
## wt.lossCEN    -0.0113 0.0123    0.0121    0.0115 -0.935 0.35000 -0.0354
##               upper97.5%
## sex2          -0.3620
## ph.ecog1       1.6000
## ph.ecog2       3.4000
## ph.ecog3       5.5900
## ph.karno60     4.6000
## ph.karno70     4.5700
## ph.karno80     4.6800
## ph.karno90     4.8600
## ph.karno100    4.9300
## pat.karno40    5.5800
## pat.karno50    4.3900
## pat.karno60    2.9900
## pat.karno70    2.2900
## pat.karno80    2.2900
## pat.karno90    2.6700
## pat.karno100   2.0600
## wt.lossCEN     0.0128
## Test of Goodness-of-fit
##               sup|   hat U(t) | p-value H_0
## sex2                2.230      0.640
## ph.ecog1            3.960      0.105
## ph.ecog2            1.460      0.939
## ph.ecog3            0.508      0.329
## ph.karno60          1.470      0.554
## ph.karno70          2.150      0.389
## ph.karno80          2.490      0.378
## ph.karno90          1.010      0.996
## ph.karno100         1.710      0.494
## pat.karno40          0.651      0.467
## pat.karno50          0.583      0.567
## pat.karno60          2.350      0.341
## pat.karno70          2.290      0.370
## pat.karno80          1.570      0.855
## pat.karno90          2.000      0.650
## pat.karno100         2.350      0.203
## wt.lossCEN          76.500      0.454

```

Okazuje się, że przy redukcji zmiennych w modelu, metoda proporcjonalnych szans zaczyna działać. Zinterpretujemy teraz współczynniki przy zmiennej *sex* i *ph.ecog*.

$\lambda_2^{sex} = -1.0400$ . Należy, to rozumieć w ten sposób, że stosunek szansy śmierci przez mężczyznę

do śmierci przez kobietę wynosi  $\frac{\exp(0)}{\exp(\lambda_2^{sex})} = \frac{1}{\exp(-1.0400)}$ , który wynosi około 2.83.

$\lambda_3^{ph.ecog} = 2.7100$ . Należy, to rozumieć w ten sposób, że stosunek szansy śmierci pacjenta o sprawności według lekarza 3 do śmierci pacjenta o sprawności prawidłowej wynosi  $\frac{\exp(\lambda_3^{ph.ecog})}{\exp(0)} = \frac{\exp(2.7100)}{1}$ , który wynosi około 15.03. Analogicznie dla innych stopni sprawności.

### 1.3 Zadanie nr 3

Przyjmując model proporcjonalnych szans jak w zadaniu nr 1, zweryfikujemy hipotezę o nieistotności zmiennej *meal.cal*. Przyjmijmy poziom istotności  $\alpha = 0.05$ .

- $H_0 : \beta_m = 0$
- $H_1 : \beta_m \neq 0$

Jeśli p-value będzie większe od poziomu istotności, nie mamy podstawy do odrzucenia hipotezy, zatem zmienna *meal.cal* jest nieistotna. W przeciwnym razie, zmienna jest istotna w przyjętym modelu.

```
modelPS_mealcal <- prop.odds(Event(time, status==2)~ageCEN+sex+ph.ecog+ph.karno+pat.karno)
summary(modelPS_mealcal)
```

```
## Proportional Odds model
##
## Did not converge, allow more iterations
##
## Test for baseline
## Test for nonparametric terms
##
## Test for non-significant effects
##      Supremum-test of significance p-value H_0: B(t)=0
## Baseline                6.27                0
##
## Test for time invariant effects
##      Kolmogorov-Smirnov test p-value H_0:constant effect
## Baseline                1.72                0.098
##
## Covariate effects
##      Coef. SE Robust SE D2log(L)^-1      z P-val lower2.5% upper97.5%
## ageCEN      5.87e-03  0          0          0 Inf      0  5.87e-03  5.87e-03
## sex2      -6.08e-01  0          0          0 -Inf      0 -6.08e-01 -6.08e-01
## ph.ecog1    6.40e-01  0          0          0 Inf      0  6.40e-01  6.40e-01
## ph.ecog2    1.32e+00  0          0          0 Inf      0  1.32e+00  1.32e+00
## ph.ecog3    2.55e+00  0          0          0 Inf      0  2.55e+00  2.55e+00
## ph.karno60  1.03e+00  0          0          0 Inf      0  1.03e+00  1.03e+00
## ph.karno70  1.00e+00  0          0          0 Inf      0  1.00e+00  1.00e+00
## ph.karno80  1.17e+00  0          0          0 Inf      0  1.17e+00  1.17e+00
## ph.karno90  1.31e+00  0          0          0 Inf      0  1.31e+00  1.31e+00
## ph.karno100 1.46e+00  0          0          0 Inf      0  1.46e+00  1.46e+00
## pat.karno40 -3.52e-01  0          0          0 -Inf      0 -3.52e-01 -3.52e-01
```

```
## pat.karno50 7.53e-01 0 0 0 Inf 0 7.53e-01 7.53e-01
## pat.karno60 1.23e-01 0 0 0 Inf 0 1.23e-01 1.23e-01
## pat.karno70 -1.74e-01 0 0 0 -Inf 0 -1.74e-01 -1.74e-01
## pat.karno80 -2.81e-01 0 0 0 -Inf 0 -2.81e-01 -2.81e-01
## pat.karno90 -6.89e-02 0 0 0 -Inf 0 -6.89e-02 -6.89e-02
## pat.karno100 -5.68e-01 0 0 0 -Inf 0 -5.68e-01 -5.68e-01
## meal.calCEN -4.43e-05 0 0 0 -Inf 0 -4.43e-05 -4.43e-05
## wt.lossCEN -1.39e-02 0 0 0 -Inf 0 -1.39e-02 -1.39e-02
## Test of Goodness-of-fit
## sup| hat U(t) | p-value H_0
## ageCEN 86.400 0.278
## sex2 3.870 0.434
## ph.ecog1 4.780 0.298
## ph.ecog2 4.890 0.178
## ph.ecog3 0.531 0.328
## ph.karno60 1.970 0.624
## ph.karno70 3.370 0.372
## ph.karno80 2.870 0.614
## ph.karno90 2.160 0.838
## ph.karno100 3.170 0.336
## pat.karno40 0.777 0.542
## pat.karno50 0.956 0.600
## pat.karno60 4.250 0.168
## pat.karno70 1.830 0.874
## pat.karno80 2.520 0.682
## pat.karno90 3.070 0.564
## pat.karno100 2.520 0.486
## meal.calCEN 3580.000 0.370
## wt.lossCEN 78.400 0.708
```

Niestety przez to, że zaimplementowana metoda iteracyjna nie jest zbieżna dla naszego modelu (taki sam przypadek jak w zadaniu nr 1), nie możemy zbadać naszej hipotezy.

## 1.4 Zadanie nr 4

Na bazie tego samego modelu co w zadaniu nr 3, mamy zbadać hipotezę o nieistotności zmiennej *pat.karno*. Niestety, analogicznie jak w zadaniach nr 1 i 3, nie możemy zbadać tej hipotezy za pomocą modelu proporcjonalnych szans.

## 1.5 Zadanie nr 5

Chcemy dokonać odpowiedniego dopasowania zmiennych objaśniających do modelu proporcjonalnych szans korzystając z kryteriów odpowiednio:

- Akaike’a (AIC)
- BIC



```

model1 <- prop.odds(Event(time, status==2)~ageCEN,data = df, n.sim = 1000, profile = 1)
model2 <- prop.odds(Event(time, status==2)~sex,data = df, n.sim = 1000, profile = 1)
model3 <- prop.odds(Event(time, status==2)~ph.ecog,data = df, n.sim = 1000, profile = 1)
model4 <- prop.odds(Event(time, status==2)~ph.karno,data = df, n.sim = 1000, profile = 1)
model5 <- prop.odds(Event(time, status==2)~pat.karno,data = df, n.sim = 1000, profile = 1)
model6 <- prop.odds(Event(time, status==2)~meal.calCEN,data = df, n.sim = 1000, profile = 1)
model7 <- prop.odds(Event(time, status==2)~wt.lossCEN,data = df, n.sim = 1000, profile = 1)

AIC1 <- -2*model1$loglike[1] + 2*1
AIC2 <- -2*model2$loglike[1] + 2*1
AIC3 <- -2*model3$loglike[1] + 2*3
AIC4 <- -2*model4$loglike[1] + 2*6
AIC5 <- -2*model5$loglike[1] + 2*7
AIC6 <- -2*model6$loglike[1] + 2*1
AIC7 <- -2*model7$loglike[1] + 2*1

AIC1

## [1] 1053.508

AIC2

## [1] 1050.954

AIC3

## [1] 1036.118

AIC4

## [1] 1055.103

AIC5

## [1] 1030.999

AIC6

## [1] 1057.067

AIC7

## [1] 1058.472

```

Najmniejszą wartość AIC posiada model uzależniony od zmiennej *pat.karno*. Zatem wybieramy ten model i badamy, którą następną zmienną objaśniającą dodać do modelu.

```

model_pk <- prop.odds(Event(time, status==2)~pat.karno,data = df, n.sim = 1000, profile = 1)

```

```

model_pk1 <- prop.odds(Event(time, status==2)~pat.karno+ageCEN,data = df, n.sim = 1000,
model_pk2 <- prop.odds(Event(time, status==2)~pat.karno+sex,data = df, n.sim = 1000, pro
model_pk3 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog,data = df, n.sim = 1000,
model_pk4 <- prop.odds(Event(time, status==2)~pat.karno+ph.karno,data = df, n.sim = 1000
model_pk5 <- prop.odds(Event(time, status==2)~pat.karno+meal.calCEN,data = df, n.sim = 1
model_pk6 <- prop.odds(Event(time, status==2)~pat.karno+wt.lossCEN,data = df, n.sim = 10
AIC_pk <- AIC5
AIC_pk1 <- -2*model_pk1$loglike[1] + 2*8
AIC_pk2 <- -2*model_pk2$loglike[1] + 2*8
AIC_pk3 <- -2*model_pk3$loglike[1] + 2*10
AIC_pk4 <- -2*model_pk4$loglike[1] + 2*13
AIC_pk5 <- -2*model_pk5$loglike[1] + 2*8
AIC_pk6 <- -2*model_pk6$loglike[1] + 2*8

AIC_pk
## [1] 1030.999

AIC_pk1
## [1] 1031.786

AIC_pk2
## [1] 1025.641

AIC_pk3
## [1] 1020.591

AIC_pk4
## [1] 1038.814

AIC_pk5
## [1] 1036.565

AIC_pk6
## [1] 1032.963

```

Najmniejszą wartość AIC posiada model uzależniony od zmiennej *pat.karno* i zmiennej *ph.ecog*. Zatem wybieramy ten model i badamy, którą następną zmienną objaśniającą dodać do modelu.

```

model_2 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog,data = df, n.sim = 1000, p
model_21 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+ageCEN,data = df, n.sim =

```

```

model_22 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+sex,data = df, n.sim = 1000)
model_23 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+ph.karno,data = df, n.sim = 1000)
model_24 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+meal.calCEN,data = df, n.sim = 1000)
model_25 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+wt.lossCEN,data = df, n.sim = 1000)

AIC_2 <- AIC_pk3
AIC_21 <- -2*model_21$loglike[1] + 2*11
AIC_22 <- -2*model_22$loglike[1] + 2*11
AIC_23 <- -2*model_23$loglike[1] + 2*16
AIC_24 <- -2*model_24$loglike[1] + 2*11
AIC_25 <- -2*model_25$loglike[1] + 2*11

AIC_2
## [1] 1020.591

AIC_21
## [1] 1022.216

AIC_22
## [1] 1012.609

AIC_23
## [1] 1031.034

AIC_24
## [1] 1027.709

AIC_25
## [1] 1022.444

```

Najmniejszą wartość AIC posiada model uzależniony od zmiennej *pat.karno*, *ph.ecog* i *sex*. Zatem wybieramy ten model i badamy, którą następną zmienną objaśniającą dodać do modelu.

```

model_3 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+sex,data = df, n.sim = 1000)

model_31 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+sex+ageCEN,data = df, n.sim = 1000)
model_32 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+sex+ph.karno,data = df, n.sim = 1000)
model_33 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+sex+meal.calCEN,data = df, n.sim = 1000)
model_34 <- prop.odds(Event(time, status==2)~pat.karno+ph.ecog+sex+wt.lossCEN,data = df, n.sim = 1000)

AIC_3 <- AIC_22
AIC_31 <- -2*model_31$loglike[1] + 2*12

```

```

AIC_32 <- -2*model_32$loglike[1] + 2*17
AIC_33 <- -2*model_33$loglike[1] + 2*12
AIC_34 <- -2*model_34$loglike[1] + 2*12

AIC_3

## [1] 1012.609

AIC_31

## [1] 1014.522

AIC_32

## [1] 1023.4

AIC_33

## [1] 1022.316

AIC_34

## [1] 1013.804

```

Widzimy, że na podstawie kryterium AIC, żaden z rozpatrywanych modeli 4 zmiennych nie jest lepszy od wcześniej przyjętego modelu uzależnionego od zmiennych: *pat.karno*, *ph.ecog*, *sex*.

```

modelB1 <- prop.odds(Event(time, status==2)~ageCEN,data = df, n.sim = 1000, profile = 1)
modelB2 <- prop.odds(Event(time, status==2)~sex,data = df, n.sim = 1000, profile = 1)
modelB3 <- prop.odds(Event(time, status==2)~ph.ecog,data = df, n.sim = 1000, profile = 1)
modelB4 <- prop.odds(Event(time, status==2)~ph.karno,data = df, n.sim = 1000, profile = 1)
modelB5 <- prop.odds(Event(time, status==2)~pat.karno,data = df, n.sim = 1000, profile = 1)
modelB6 <- prop.odds(Event(time, status==2)~meal.calCEN,data = df, n.sim = 1000, profile = 1)
modelB7 <- prop.odds(Event(time, status==2)~wt.lossCEN,data = df, n.sim = 1000, profile = 1)

n= nrow(df)
BIC1 <- -2*modelB1$loglike[1] + 1*log(n)
BIC2 <- -2*modelB2$loglike[1] + 1*log(n)
BIC3 <- -2*modelB3$loglike[1] + 3*log(n)
BIC4 <- -2*modelB4$loglike[1] + 6*log(n)
BIC5 <- -2*modelB5$loglike[1] + 7*log(n)
BIC6 <- -2*modelB6$loglike[1] + 1*log(n)
BIC7 <- -2*modelB7$loglike[1] + 1*log(n)

BIC1

## [1] 1056.649

BIC2

```

```
## [1] 1054.095

BIC3

## [1] 1045.543

BIC4

## [1] 1073.924

BIC5

## [1] 1052.991

BIC6

## [1] 1060.208

BIC7

## [1] 1061.614
```

Najmniejszą wartość BIC posiada model uzależniony od zmiennej *ph.ecog*. Zatem wybieramy ten model i badamy, którą następną zmienną objaśniającą dodać do modelu.

```
modelB11 <- prop.odds(Event(time, status==2)~ph.ecog+ageCEN,data = df, n.sim = 1000, profile=1)
modelB12 <- prop.odds(Event(time, status==2)~ph.ecog+sex,data = df, n.sim = 1000, profile=1)
modelB13 <- prop.odds(Event(time, status==2)~ph.ecog+ph.karno,data = df, n.sim = 1000, profile=1)
modelB14 <- prop.odds(Event(time, status==2)~ph.ecog+pat.karno,data = df, n.sim = 1000, profile=1)
modelB15 <- prop.odds(Event(time, status==2)~ph.ecog+meal.calCEN,data = df, n.sim = 1000, profile=1)
modelB16 <- prop.odds(Event(time, status==2)~ph.ecog+wt.lossCEN,data = df, n.sim = 1000, profile=1)

BIC11 <- -2*modelB11$loglike[1] + 4*log(n)
BIC12 <- -2*modelB12$loglike[1] + 4*log(n)
BIC13 <- -2*modelB13$loglike[1] + 9*log(n)
BIC14 <- -2*modelB14$loglike[1] + 10*log(n)
BIC15 <- -2*modelB15$loglike[1] + 4*log(n)
BIC16 <- -2*modelB16$loglike[1] + 4*log(n)

BIC3

## [1] 1045.543

BIC11

## [1] 1049.488

BIC12

## [1] 1040.341
```

```

BIC13

## [1] 1075.092

BIC14

## [1] 1052.007

BIC15

## [1] 1053.422

BIC16

## [1] 1050.459

```

Najmniejszą wartość BIC posiada model uzależniony od zmiennych: *ph.ecog* i *sex*. Zatem wybieramy ten model i badamy, którą następną zmienną objaśniającą dodać do modelu.

```

modelB21 <- prop.odds(Event(time, status==2)~ph.ecog+sex+ageCEN,data = df, n.sim = 1000,
modelB22 <- prop.odds(Event(time, status==2)~ph.ecog+sex+ph.karno,data = df, n.sim = 1000,
modelB23 <- prop.odds(Event(time, status==2)~ph.ecog+sex+pat.karno,data = df, n.sim = 1000,
modelB24 <- prop.odds(Event(time, status==2)~ph.ecog+sex+meal.calCEN,data = df, n.sim = 1000,
modelB25 <- prop.odds(Event(time, status==2)~ph.ecog+sex+wt.lossCEN,data = df, n.sim = 1000,

BIC21 <- -2*modelB21$loglike[1] + 5*log(n)
BIC22 <- -2*modelB22$loglike[1] + 10*log(n)
BIC23 <- -2*modelB23$loglike[1] + 11*log(n)
BIC24 <- -2*modelB24$loglike[1] + 5*log(n)
BIC25 <- -2*modelB25$loglike[1] + 5*log(n)

BIC12

## [1] 1040.341

BIC21

## [1] 1045.049

BIC22

## [1] 1069.645

BIC23

## [1] 1047.138

BIC24

## [1] 1050.854

BIC25

## [1] 1044.421

```

Widzimy, że na podstawie kryterium BIC, żaden z rozpatrywanych modeli 3 zmiennych nie jest lepszy od wcześniej przyjętego modelu uzależnionego od zmiennych: *ph.ecog*, *sex*.

## 1.6 Zadanie nr 6

Naszukujemy funkcję przeżycia, odpowiadającą mężczyźnie o charakterystykach: *pat.karno*=70 i *ph.ecog*=2.

```
df2 <- lung
df2$status <- as.factor(df2$status)
df2$sex <- as.factor(df2$sex)
df2$ph.ecog <- as.factor(df2$ph.ecog)
df2$ph.karno <- as.factor(df2$ph.karno)
df2$pat.karno <- as.factor(df2$pat.karno)
df2$age <- df2$age - mean(df2$age)
df2$meal.cal <- df2$meal.cal - mean(df2$meal.cal)
df2$wt.loss <- df2$wt.loss - mean(df2$wt.loss)
modelSF <- nltm(Surv(time, status==2)~df2$sex+df2$ph.ecog+df2$pat.karno, data = df2, nlt
summary(modelSF)
```

## Call:

```
## nltm(formula1 = Surv(time, status == 2) ~ df2$sex + df2$ph.ecog + df2$pat.karno,
data = df2, nlt.model = "PO")
##
## Non Linear Transformation Model: PO, fit by maximum likelihood
```

##		coef	exp(coef)	se(coef)	z	p
##	df2\$sex2	0	1	0.253	0	1
##	df2\$ph.ecog1	0	1	0.295	0	1
##	df2\$ph.ecog2	0	1	0.431	0	1
##	df2\$ph.ecog3	0	1	1.994	0	1
##	df2\$pat.karno40	0	1	2.043	0	1
##	df2\$pat.karno50	0	1	1.688	0	1
##	df2\$pat.karno60	0	1	1.386	0	1
##	df2\$pat.karno70	0	1	1.394	0	1
##	df2\$pat.karno80	0	1	1.399	0	1
##	df2\$pat.karno90	0	1	1.409	0	1
##	df2\$pat.karno100	0	1	1.422	0	1

##

##		exp(coef)	exp(-coef)	lower .95	upper .95
##	df2\$sex2	1	1	0.6095	1.64
##	df2\$ph.ecog1	1	1	0.5604	1.78
##	df2\$ph.ecog2	1	1	0.4296	2.33
##	df2\$ph.ecog3	1	1	0.0201	49.81
##	df2\$pat.karno40	1	1	0.0183	54.77
##	df2\$pat.karno50	1	1	0.0366	27.32
##	df2\$pat.karno60	1	1	0.0661	15.14
##	df2\$pat.karno70	1	1	0.0651	15.36

```
## df2$pat.karno80      1      1      0.0644      15.52
## df2$pat.karno90      1      1      0.0632      15.81
## df2$pat.karno100     1      1      0.0616      16.24

##
## Likelihood ratio test=0 on 11 df, p=1
##
## n=224 (4 observations deleted due to missingness)

theta0 = (1-modelSF$surv)/modelSF$surv
theta = theta0*exp(sum(modelSF$coefficients*c(0,0,1,0,0,0,0,1,0,0,0)))
plot(1/(theta+1), xlab="Days", ylab="Survival Function")
```

