# 140.623.01 - Statistical Methods in Public Health III

Assignment 3: Survival in Diffuse Histiocytic Lymphoma

*Martin Skarzynski*

*March 1, 2018*

Learning Objectives: Students who successfully complete this section will be able to: - Translate survival time data into groups that allow calculation of crude incidence rates. - Analyze the grouped survival time data using log-linear Poisson regression models. - Analyze the survival time data (without grouping) by the Kaplan-Meier estimate of the survival function, the log- rank statistic and Cox proportional hazards model. - Check the estimated model for its consistency with the observed data; in particular, check the proportional hazards assumption using the complementary log-log plot of the estimated survival function. - Summarize the findings for public health readers and document and archive the steps of the statistical analysis by creating an R script file.

Data Set: Below find the survival times in days for two groups of patients with diffuse histiocytic lymphoma. The data are stored in the csv data set lymphoma.csv, which may be downloaded from the course website. One group has Stage-3 cancer (stage = 0); the second group (stage =1) has Stage-4 cancer. The question of interest is whether stage at diagnosis predicts survival time.

Times to Death(days) Stage 3 (stage=0)

6, 19, 32, 42, 42, 43, *94, 126, 169*, 207, 211, *227*, 253, 255, *270*, 310, *316*, 335, *346*

Stage 4 (stage=1) 4, 10, 11, 13, 31, 40, 50, 56, 68, 82, 85, 93, 175, 247, *291*, 345 = censored (alive at the end of follow-up)

Methods: a. An alternative to calculating Kaplan-Meier estimates of the survival curve is to calculate lifetable estimates when the time intervals are grouped or binned. Using the lymphoma.csv data set, we could divide the total time of exposure into roughly ten bins and determine the numbers of deaths and person-days experienced for each of the two groups in each bin. For example, (0-7] is the bin from 0 up to but not including 7 days.

Assume the following bins: (0-7], (7-15], (15-30], (30-60], (60-90], (90-120], (120-150], (150-180], (180-270], (270-360]

b. Download the csv data set binlymph.csv from CoursePlus. Verify that the calculations of total time of exposure and person-days experienced appears to be correct by reviewing the contents of this dataset. Also, using R create a plot of S(t) –vs.- mid_days for each group.

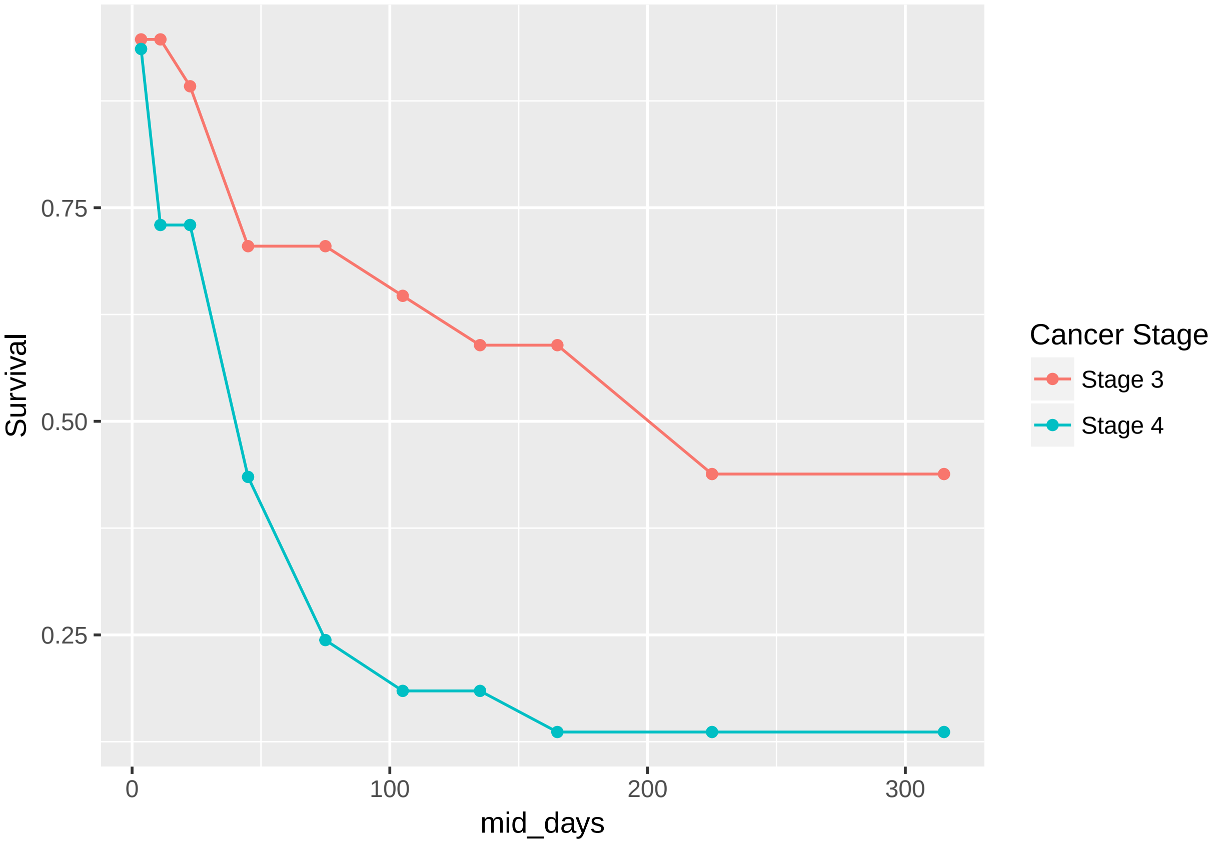
```r
setwd("~/github/140-623_Statistical-Methods-in-Public-Health3")
library(readr)
binData = read_csv("binlymph.csv")
```

```
## Parsed with column specification:
## cols(
##   stage = col_integer(),
##   bin = col_integer(),
##   D = col_integer(),
##   P_Days = col_integer(),
##   I_Rate = col_double(),
##   L = col_integer(),
##   mid_days = col_double(),
##   P = col_double(),
##   Survival = col_double(),
##   Stage3 = col_double(),
##   Stage4 = col_double()
## )
```

```
print(binData)
```

```
## # A tibble: 20 x 11
##    stage   bin     D P_Days  I_Rate     L mid_days     P Survival Stage3
##    <int> <int> <int>  <int>   <dbl> <int>    <dbl> <dbl>    <dbl>  <dbl>
##  1     0     0     1    132 0.00758     7     3.50 0.947     0.947  1.00
##  2     0     7     0    144 0           8    11.0  1.00      0.947  0.947
##  3     0    15     1    259 0.00386    15    22.5  0.942     0.892  0.947
##  4     0    30     3    429 0.00699    30    45.0  0.790     0.705  0.892
##  5     0    60     0    390 0          30    75.0  1.00      0.705  0.705
##  6     0    90     1    364 0.00275    30   105    0.918     0.647  0.705
##  7     0   120     1    336 0.00298    30   135    0.911     0.589  0.647
##  8     0   150     0    319 0          30   165    1.00      0.589  0.589
##  9     0   180     2    703 0.00284    90   225    0.744     0.438  0.589
## 10     0   270     0    227 0          NA   315    1.00      0.438  0.438
## 11     1     0     1    109 0.00917     7     3.50 0.936     0.936 NA
## 12     1     7     3    109 0.0275      8    11.0  0.780     0.730 NA
## 13     1    15     0    180 0          15    22.5  1.00      0.730 NA
## 14     1    30     4    297 0.0135     30    45.0  0.596     0.435 NA
## 15     1    60     3    205 0.0146     30    75.0  0.561     0.244 NA
## 16     1    90     1    123 0.00813    30   105    0.756     0.184 NA
## 17     1   120     0    120 0          30   135    1.00      0.184 NA
## 18     1   150     1    115 0.00870    30   165    0.739     0.136 NA
## 19     1   180     0    247 0          90   225    1.00      0.136 NA
## 20     1   270     0     96 0          NA   315    1.00      0.136 NA
## # ... with 1 more variable: Stage4 <dbl>
```

```
library(ggplot2, help)
qplot(x=mid_days, y=Survival, col=factor(stage, labels=c("Stage 3", "Stage 4")), data=binData) + geom_li
```



c. Recall that D is the number of deaths, P_Days is the person-days accumulated in the bin and mid_days is the midpoint of time bin. Rename variables for simplicity:

```
library(dplyr, help)
binData = binData %>%
mutate(t = mid_days) %>%
mutate(N = P_Days)
```

d. Fit the following four log-linear Poisson regression models to the grouped survival data

The four models are in part g below.

e. Generate time terms, centered and spline:

```
binData = binData %>%
mutate(t60 = t-60) %>%
mutate(t60sp = ifelse(t > 60, t-60, 0))
```

f. Generate interaction terms: We don't need to do this in R, since we can include the interaction directly in our model.

g. Fit the models:

```
# Model A: stage
modelA = glm(D ~ stage, offset=log(N), family=poisson(link="log"), data=binData)
summary(modelA)
```

```
##
## Call:
## glm(formula = D ~ stage, family = poisson(link = "log"), data = binData,
##     offset = log(N))
```

```
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.00281  -1.26608   0.03439   0.46564   1.75907
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.9054     0.3333 -17.716   <2e-16 ***
## stage         1.0919     0.4336   2.518   0.0118 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 30.505  on 19  degrees of freedom
## Residual deviance: 24.053  on 18  degrees of freedom
## AIC: 56.908
##
## Number of Fisher Scoring iterations: 5
```

```r
modelA$coefficients; confint.default(modelA) ## coefficients
```

```
## (Intercept)       stage
##   -5.905362    1.091927
```

```
##                  2.5 %    97.5 %
## (Intercept) -6.5586828 -5.252041
## stage        0.2420319  1.941823
```

```r
exp(modelA$coefficients); exp(confint.default(modelA)) ## IRR
```

```
## (Intercept)       stage
## 0.002724796 2.980012492
```

```
##                   2.5 %     97.5 %
## (Intercept) 0.001417752 0.00523682
## stage       1.273834818 6.97144899
```

```r
# Model B: stage + t-60
modelB = glm(D ~ stage + t60, offset=log(N), family=poisson(link="log"), data=binData)
summary(modelB)
```

```
##
## Call:
## glm(formula = D ~ stage + t60, family = poisson(link = "log"),
##     data = binData, offset = log(N))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1036  -1.0874  -0.1017   0.5040   1.1219
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.603146   0.340600 -16.451   <2e-16 ***
## stage        0.942699   0.437674   2.154   0.0312 *
## t60         -0.006977   0.003166  -2.204   0.0275 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 30.505  on 19  degrees of freedom
## Residual deviance: 18.019  on 17  degrees of freedom
## AIC: 52.874
##
## Number of Fisher Scoring iterations: 5
```

```r
modelB$coefficients; confint.default(modelB) ## coefficients
```

```
##  (Intercept)        stage          t60
## -5.603146449  0.942699007 -0.006976953
```

```
##                   2.5 %        97.5 %
## (Intercept) -6.27070994 -4.9355829625
## stage        0.08487278  1.8005252297
## t60         -0.01318209 -0.0007718107
```

```r
exp(modelB$coefficients); exp(confint.default(modelB)) ## IRR
```

```
## (Intercept)       stage         t60
## 0.003686247 2.566900159 0.993047330
```

```
##                   2.5 %     97.5 %
## (Intercept) 0.001890886 0.00718627
## stage       1.088578574 6.05282575
## t60         0.986904409 0.99922849
```

```r
# Model C: stage + t-60 + (t-60)^+
modelC = glm(D ~ stage + t60 + t60sp, offset=log(N), family=poisson(link="log"), data=binData)
summary(modelC)
```

```
##
## Call:
## glm(formula = D ~ stage + t60 + t60sp, family = poisson(link = "log"),
##     data = binData, offset = log(N))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0458  -1.0725  -0.1111   0.5022   1.2952
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.4322926  0.4621789 -11.754   <2e-16 ***
## stage        0.9514927  0.4378395   2.173   0.0298 *
## t60         -0.0006996  0.0123224  -0.057   0.9547
## t60sp       -0.0081450  0.0154147  -0.528   0.5972
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 30.505  on 19  degrees of freedom
## Residual deviance: 17.736  on 16  degrees of freedom
## AIC: 54.591
```

```
##
## Number of Fisher Scoring iterations: 5
```

```r
modelC$coefficients; confint.default(modelC) ## coefficients
```

```
##   (Intercept)         stage           t60         t60sp
## -5.4322926309  0.9514926947 -0.0006995638 -0.0081450320
```

```
##                   2.5 %      97.5 %
## (Intercept) -6.33814667 -4.52643860
## stage        0.09334297  1.80964242
## t60         -0.02485107  0.02345194
## t60sp       -0.03835733  0.02206726
```

```r
#package Model D: stage + t-60 + (t-60)^+ + stage*(t•60) + stage*(t•60)^+
modelD = glm(D ~ stage + t60 + t60sp + stage:t60 + stage:t60sp, offset=log(N), family=poisson(link="log
summary(modelD)
```

```
##
## Call:
## glm(formula = D ~ stage + t60 + t60sp + stage:t60 + stage:t60sp,
##     family = poisson(link = "log"), data = binData, offset = log(N))
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.06305  -0.81064  -0.08793   0.34044   1.38848
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.792139   0.632165  -9.162   <2e-16 ***
## stage        1.614121   0.804147   2.007   0.0447 *
## t60         -0.009640   0.019928  -0.484   0.6286
## t60sp        0.005976   0.023829   0.251   0.8020
## stage:t60    0.016566   0.025548   0.648   0.5167
## stage:t60sp -0.029494   0.032583  -0.905   0.3654
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 30.505  on 19  degrees of freedom
## Residual deviance: 16.233  on 14  degrees of freedom
## AIC: 57.088
##
## Number of Fisher Scoring iterations: 5
```

```r
modelD$coefficients; confint.default(modelD) ## coefficients
```

```
##  (Intercept)        stage          t60        t60sp    stage:t60
## -5.792139416  1.614121315 -0.009639959  0.005975502  0.016565567
##  stage:t60sp
## -0.029494184
```

```
##                   2.5 %      97.5 %
## (Intercept) -7.03115989 -4.55311894
## stage        0.03802288  3.19021975
## t60         -0.04869908  0.02941916
## t60sp       -0.04072942  0.05268042
## stage:t60   -0.03350662  0.06663775
## stage:t60sp -0.09335523  0.03436687
```

h. Use the AIC = -2 log likelihood + 2(# of parameters) to identify the “best” prediction model from among A-D. Interpret the model results in a few sentences, as if for a journal article.

```
AIC(modelA, modelB, modelC, modelD)
```

```
##        df      AIC
## modelA  2 56.90787
## modelB  3 52.87416
## modelC  4 54.59051
## modelD  6 57.08824
```

Based on the Akaike information criterion (AIC), the best model is modelB. ModelB has two variables, `stage` and `t60` in addition the the intercept, all of which are statistically significant according to the model summary. The models that include additional variables have higher AIC values, indicating that the contribution of these additional variables is not worth the burden of a larger model. i. Now use the csv data set lymphoma.csv. Calculate Kaplan-Meierpackage (K-M) estimates of the survival curve with 95% CI separately for each group. Plot the K-M curves against time.

```
lymphData = read_csv("lymphoma.csv")
```

```
## Parsed with column specification:
## cols(
##   id = col_integer(),
##   stage = col_integer(),
##   days = col_integer(),
##   died = col_integer()
## )
```

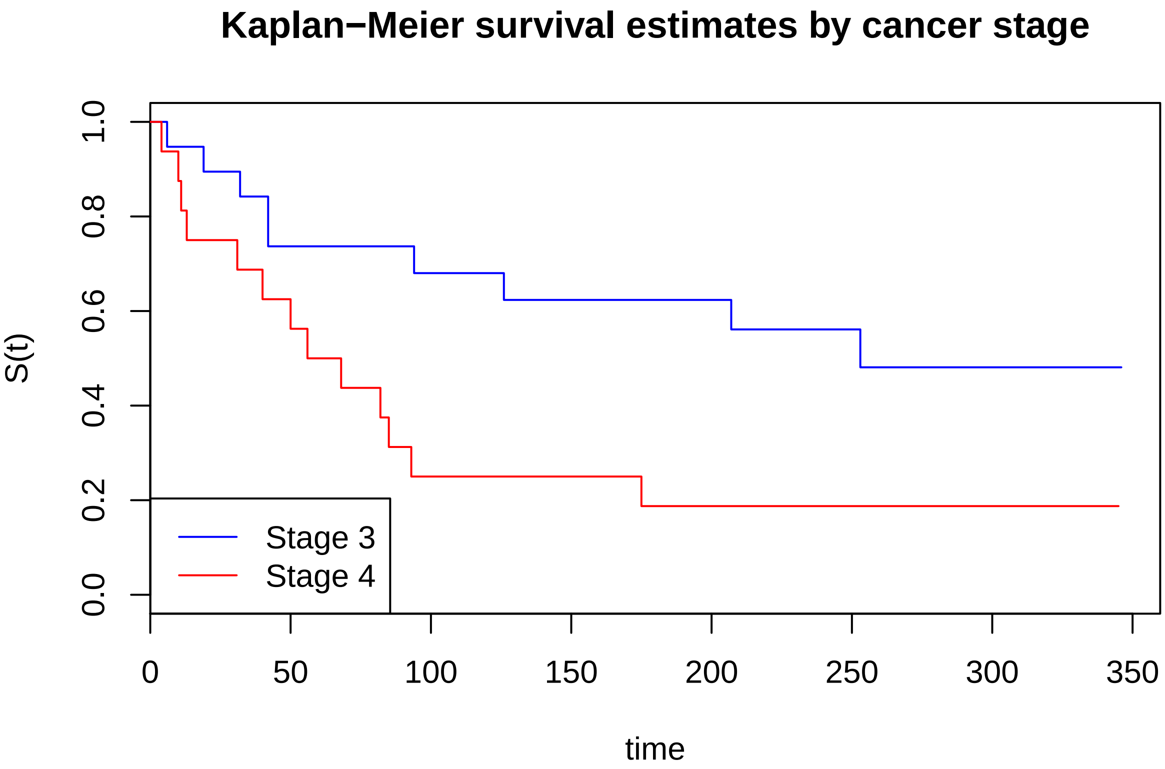
```
head(lymphData)
```

```
## # A tibble: 6 x 4
##      id stage  days  died
##   <int> <int> <int> <int>
## 1     1     0     6     1
## 2     2     0    19     1
## 3     3     0    32     1
## 4     4     0    42     1
## 5     5     0    42     1
## 6     6     0    43     0
```

```
library(survival)
lymphData$SurvObj = with(lymphData, Surv(days, died == 1))
km.stage = survfit(SurvObj ~ stage, data = lymphData,
type="kaplan-meier", conf.type="log-log")
summary(km.stage)
```

```
## Call: survfit(formula = SurvObj ~ stage, data = lymphData, type = "kaplan-meier",
##     conf.type = "log-log")
##
##                 stage=0
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     6     19       1    0.947  0.0512        0.681        0.992
##    19     18       1    0.895  0.0704        0.641        0.973
##    32     17       1    0.842  0.0837        0.587        0.946
##    42      16       2    0.737  0.1010        0.479        0.881
##    94      13       1    0.680  0.1080        0.421        0.842
##   126      12       1    0.623  0.1129        0.367        0.800
##   207      10       1    0.561  0.1176        0.308        0.753
##   253       7       1    0.481  0.1251        0.230        0.694
##
##                 stage=1
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     4      16       1    0.938  0.0605       0.6323        0.991
##    10      15       1    0.875  0.0827       0.5860        0.967
##    11      14       1    0.812  0.0976       0.5246        0.935
##    13      13       1    0.750  0.1083       0.4634        0.898
##    31      12       1    0.688  0.1159       0.4046        0.856
##    40      11       1    0.625  0.1210       0.3486        0.811
##    50      10       1    0.562  0.1240       0.2954        0.762
##    56       9       1    0.500  0.1250       0.2452        0.710
##    68       8       1    0.438  0.1240       0.1981        0.656
##    82       7       1    0.375  0.1210       0.1542        0.598
##    85       6       1    0.312  0.1159       0.1139        0.536
##    93       5       1    0.250  0.1083       0.0775        0.472
##   175       4       1    0.188  0.0976       0.0460        0.402
```

```
plot(km.stage, col=c("blue","red"),
main="Kaplan-Meier survival estimates by cancer stage",
ylab="S(t)", xlab="time" )
legend("bottomleft", c("Stage 3", "Stage 4"),
col=c("blue", "red"), lty=1)
```



j. Compare the K-M curves versus the display of S(t) – vs- mid_days that you produced in step a.

The curves are similar in that stage 4 cancer survival is clearly much less likely than stage 3 cancer survival.

k. Carry out a log-rank test and determine a p-value for the null hypothesis that the two population survival curves are the same for Stage 4 -vs- Stage 3 patients. What do you conclude?

```r
survdiff(SurvObj ~ stage, data=lymphData)
```

```
## Call:
## survdiff(formula = SurvObj ~ stage, data = lymphData)
##
##           N Observed Expected (O-E)^2/E (O-E)^2/V
## stage=0 19        9    14.01      1.79      5.09
## stage=1 16       13     7.99      3.15      5.09
##
##  Chisq= 5.1  on 1 degrees of freedom, p= 0.024
```

The p-value (0.024) indicates that there is enough evidence to reject the null hypothesis that the two population survival curves are the same for Stage 4 -vs- Stage 3 patients.

l. Fit a Cox proportional hazards model with an arbitrary baseline hazard and a group effect for stage

```r
model1 = coxph(SurvObj ~ stage, data = lymphData, ties="breslow")
summary(model1)
```

```
## Call:
## coxph(formula = SurvObj ~ stage, data = lymphData, ties = "breslow")
##
##   n= 35, number of events= 22
##
##         coef exp(coef) se(coef)     z Pr(>|z|)
## stage 0.9576    2.6054   0.4402 2.175   0.0296 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##       exp(coef) exp(-coef) lower .95 upper .95
## stage     2.605     0.3838     1.099     6.175
##
## Concordance= 0.623  (se = 0.057 )
## Rsquare= 0.129   (max possible= 0.98 )
## Likelihood ratio test= 4.83  on 1 df,   p=0.02791
## Wald test            = 4.73  on 1 df,   p=0.02963
## Score (logrank) test = 5.07  on 1 df,   p=0.02431
```

m. Compare the results of the log-rank test from part k. with the corresponding test for the Cox model in part l. Do they differ enough to change interpretation?

The results for parts k and l are similar enough that I would not change my interpretation. The p-value from the Cox model in part k is 0.024, which is close to the Likelihood ratio test, Wald test and logrank test p-values (all < 0.03).

n. Create an R script file that documents and archives the steps of your statistical analysis. This file will make your analysis "reproducible."

This PDF was made using an R script thanks to the `rmarkdown` package, as per the package information on the RStudio website: https://rmarkdown.rstudio.com/articles_report_from_r_script.html.