# Biostats - Parametric Survival

Rob Leonard (robleonard@tamu.edu (mailto:robleonard@tamu.edu))

# 1) Prednisolone Data

**1a) Fit best model and derive percentile formula.**

```
library(survival)
myv <- c(2,6,12,54,56,68,89,96,96,125,128,131,140,141,143,145,146,148,162,168,173,181,
         2,3,4,7,10,22,28,29,32,37,40,41,54,61,63,71,127,140,146,158,167,182)
mydel <- c(1,1,1,1,0,1,1,1,1,0,0,0,0,0,1,0,1,0,0,1,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,
           0,0,0,0,0,0)
mycov <- c(rep(1,22),rep(0,22))
mydata <- data.frame(myv, mydel, mycov)
names(mydata) <- c("time","mystatus","mycov")
head(mydata)
```

```
##   time mystatus mycov
## 1    2        1     1
## 2    6        1     1
## 3   12        1     1
## 4   54        1     1
## 5   56        0     1
## 6   68        1     1
```

```
out1 <- survreg(Surv(time,mystatus) ~ mycov, data = mydata, dist="lognormal")
out2 <- survreg(Surv(time,mystatus) ~ mycov, data = mydata, dist="exponential")
out3 <- survreg(Surv(time,mystatus) ~ mycov, data = mydata, dist="weibull")
extractAIC(out1)  # lowest AIC at 319
```

```
## [1]   3.0000 319.4332
```

```
extractAIC(out2)
```

```
## [1]   2.0000 320.2051
```

```
extractAIC(out3)
```

```
## [1]   3.0000 320.0339
```

The log-normal model is the best choice, as it has the minimum AIC value (319.43). inf{t: pr(T>t|$X_*$)} = inf(t: S(t) $\leq$ 1-p)

$$1 - \phi(\frac{log(t) - \beta_0 - X_0^T\beta_1}{\sigma}) = (1-p)$$
$$\phi(\frac{log(t) - \beta_0 - X_0^T\beta_1}{\sigma}) = (p)$$
$$(\frac{log(t) - \beta_0 - X_0^T\beta_1}{\sigma}) = \phi^{-1}(p)$$
$$log(t) - \beta_0 - X_0^T\beta_1 = (\sigma)\phi^{-1}(p)$$
$$log(t) = (\sigma)\phi^{-1}(p) + \beta_0 + X_0^T\beta_1$$
$$t = e^{(\sigma)\phi^{-1}(p)+\beta_0+X_0^T\beta_1}$$

**1b) Estimate the above three percentiles and obtain the 95% CI for the percentiles for the two groups separately.**

```
(sm1 <- summary(out1))
```

```
##
## Call:
## survreg(formula = Surv(time, mystatus) ~ mycov, data = mydata,
##     dist = "lognormal")
##             Value Std. Error    z       p
## (Intercept) 3.852      0.396 9.73 < 2e-16
## mycov       1.250      0.580 2.15   0.031
## Log(scale)  0.574      0.146 3.92 8.9e-05
##
## Scale= 1.78
##
## Log Normal distribution
## Loglik(model)= -156.7   Loglik(intercept only)= -159
##  Chisq= 4.56 on 1 degrees of freedom, p= 0.033
## Number of Newton-Raphson Iterations: 3
## n= 44
```

Group 1: Treatment with Pred

```
library(msm)
qvec <- c(qnorm(.25),qnorm(0.5),qnorm(.75))
estimate <- ll <- ul <-  rep(0,3)

for (i in 1:3){
 estimate[i] <- exp(sm1$scale*qvec[i]+sm1$coefficients[1]+sm1$coefficients[2])
 tempest <- estimate[i]
 sestar <- deltamethod(~(log(tempest)-x1-x2)/exp(x3),c(sm1$coefficients[1],sm1$coefficients[2],l
og(sm1$scale)),out1$var)


ll[i] <- estimate[i] - 1.96*sestar
ul[i] <- estimate[i] + 1.96*sestar

}
print(c("Estimate = ", round(estimate,2), "Lower =", round(ll,2), "Upper = ", round(ul,2)))
```

```
##  [1] "Estimate = " "49.62"       "164.31"      "544.1"       "Lower ="
##  [6] "49.15"       "163.83"      "543.53"      "Upper = "    "50.09"
## [11] "164.8"       "544.67"
```

Group 2: Control Group

```
qvecC <- c(qnorm(.25),qnorm(0.5),qnorm(.75))
estimateC <- llC <- ulC <-  rep(0,3)

for (i in 1:3){
 estimateC[i] <- exp(sm1$scale*qvec[i]+sm1$coefficients[1]+sm1$coefficients[2]*0)
 tempest <- estimate[i]
 sestar <- deltamethod(~(log(tempest)-x1-x2*0)/exp(x3),c(sm1$coefficients[1],sm1$coefficients[2
],log(sm1$scale)),out1$var)


llC[i] <- estimateC[i] - 1.96*sestar
ulC[i] <- estimateC[i] + 1.96*sestar

}
print(c("Estimate = ", round(estimateC,2), "Lower =", round(llC,2), "Upper = ", round(ulC,2)))
```

```
##  [1] "Estimate = " "14.22"       "47.1"        "155.96"      "Lower ="
##  [6] "13.78"       "46.59"       "155.33"      "Upper = "    "14.66"
## [11] "47.61"       "156.59"
```

**1c) Redo 1b with non parametric method**

```
datatreat = mydata[mydata$mycov==1,]  # split data set by treatment and control
datacontr = mydata[mydata$mycov==0,]
# treatment group
treatFit <- survfit(Surv(datatreat$time,datatreat$mystatus)~1)
quantile(treatFit, prob=c(0.25, 0.5, 0.75), conf.int=TRUE)
```

```
## $quantile
##  25  50  75
##  89 146  NA
##
## $lower
##  25  50  75
##  12  96 168
##
## $upper
## 25 50 75
## NA NA NA
```

```
# control group
contrFit <- survfit(Surv(datacontr$time,datacontr$mystatus)~1)
quantile(contrFit, prob=c(0.25, 0.5, 0.75), conf.int=TRUE)
```

```
## $quantile
##   25   50   75
## 22.0 40.5   NA
##
## $lower
## 25 50 75
##  4 29 54
##
## $upper
## 25 50 75
## 41 NA NA
```

The non-parametric method struggles here due to the small sample size for each group. Several of the bound values are returned as NA. The parametric methods are more useful for this dataset.

# 2) Colon Data

**2a) Set up colon cancer data. Choose best model using stepwise and AIC then BIC.**

```
data(colon)
colondata <- colon[colon$etype==1,]  # exclude subjects who died (unknown reason)
colondata$differ <- as.factor(colondata$differ) # change variable to factor
colondata$extent <- as.factor(colondata$extent)
colondata <- colondata[complete.cases(colondata),] # only look at complete data
nrowcolon = nrow(colondata)
outf <- survreg(Surv(time,status) ~ sex+age+perfor+adhere+nodes+differ+extent+
        sex*age+sex*perfor+sex*adhere+sex*nodes+sex*differ+sex*extent+
        age*perfor+age*adhere+age*nodes+age*differ+age*extent+
        perfor*adhere+perfor*nodes+perfor*differ+perfor*extent+
        adhere*nodes+adhere*differ+adhere*extent+
        nodes*differ+nodes*extent+differ*extent, data=colondata, dist="weibull")
library(MASS)
out1bothAIC <- step(outf, k=2, trace=F)  # AIC
summary(out1bothAIC)
```

```
##
## Call:
## survreg(formula = Surv(time, status) ~ sex + age + perfor + adhere +
##     nodes + differ + extent + sex:age + sex:perfor + sex:nodes +
##     sex:extent + age:perfor + age:adhere + age:differ + perfor:nodes +
##     adhere:nodes + nodes:extent, data = colondata, dist = "weibull")
##                 Value Std. Error     z       p
## (Intercept)   12.0332     1.8798  6.40 1.5e-10
## sex            2.4207     1.7074  1.42 0.15625
## age           -0.0211     0.0176 -1.20 0.23079
## perfor         3.5815     2.4754  1.45 0.14795
## adhere        -2.2303     0.8713 -2.56 0.01047
## nodes         -0.6547     0.6126 -1.07 0.28515
## differ2       -2.2164     1.1558 -1.92 0.05516
## differ3       -4.4880     1.3013 -3.45 0.00056
## extent2       -1.5397     1.5587 -0.99 0.32324
## extent3       -1.9848     1.5296 -1.30 0.19441
## extent4       -2.0683     1.6462 -1.26 0.20897
## sex:age       -0.0300     0.0111 -2.71 0.00671
## sex:perfor     1.2641     0.7052  1.79 0.07306
## sex:nodes      0.0955     0.0284  3.36 0.00077
## sex:extent2    0.5332     1.6572  0.32 0.74765
## sex:extent3   -0.9545     1.5882 -0.60 0.54785
## sex:extent4   -1.7347     1.7042 -1.02 0.30874
## age:perfor    -0.0570     0.0347 -1.64 0.10007
## age:adhere     0.0275     0.0144  1.91 0.05612
## age:differ2    0.0370     0.0183  2.03 0.04281
## age:differ3    0.0687     0.0210  3.27 0.00107
## perfor:nodes  -0.2933     0.1326 -2.21 0.02694
## adhere:nodes   0.0613     0.0419  1.46 0.14350
## nodes:extent2  0.3518     0.6139  0.57 0.56656
## nodes:extent3  0.4790     0.6127  0.78 0.43430
## nodes:extent4  0.4778     0.6156  0.78 0.43769
## Log(scale)     0.2971     0.0414  7.17 7.4e-13
##
## Scale= 1.35
##
## Weibull distribution
## Loglik(model)= -3859.1   Loglik(intercept only)= -3938.1
##  Chisq= 158.1 on 25 degrees of freedom, p= 2.7e-21
## Number of Newton-Raphson Iterations: 5
## n= 888
```

BIC method

```
out1bothBIC <- step(outf, k=log(nrowcolon), trace=F)  # BIC
summary(out1bothBIC)
```

```
##
## Call:
## survreg(formula = Surv(time, status) ~ sex + age + nodes + sex:age,
##     data = colondata, dist = "weibull")
##               Value Std. Error     z      p
## (Intercept)  7.3887     0.4534 16.30 <2e-16
## sex          1.9942     0.6911  2.89 0.0039
## age          0.0193     0.0076  2.54 0.0112
## nodes       -0.1354     0.0136 -9.97 <2e-16
## sex:age     -0.0305     0.0114 -2.67 0.0076
## Log(scale)   0.3437     0.0418  8.22 <2e-16
##
## Scale= 1.41
##
## Weibull distribution
## Loglik(model)= -3898.3   Loglik(intercept only)= -3938.1
##  Chisq= 79.68 on 4 degrees of freedom, p= 2e-16
## Number of Newton-Raphson Iterations: 5
## n= 888
```

**2b)CI for survival based on given data.** AIC model (skip BIC per discussion board):

```
Xaic <- matrix(c(1, 60, 0,  0,  2,  1,  0,  0,  1,  0,  60, 0,  2,  0,  1,  0,  0,  0,  60, 0,
0,  0,  0,  2,  0,
1,  60, 0,  1,  2,  1,  0,  0,  1,  0,  60, 0,  2,  0,  1,  0,  0,  60, 60, 0,  0,  2,  0,  2,
0,
1,  60, 1,  0,  2,  1,  0,  0,  1,  0,  60, 1,  2,  0,  1,  0,  60, 0,  60, 0,  2,  0,  0,  2,
0,
1,  60, 1,  1,  2,  1,  0,  0,  1,  0,  60, 1,  2,  0,  1,  0,  60, 60, 60, 0,  2,  2,  0,  2,
0,
0,  60, 0,  0,  2,  1,  0,  0,  1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  60, 0,  0,  0,  0,  2,
0,
0,  60, 0,  1,  2,  1,  0,  0,  1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  60, 60, 0,  0,  2,  0,  2,
0,
0,  60, 1,  0,  2,  1,  0,  0,  1,  0,  0,  0,  0,  0,  0,  0,  0,  60, 0,  60, 0,  2,  0,  0,  2,
0,
0,  60, 1,  1,  2,  1,  0,  0,  1,  0,  0,  0,  0,  0,  0,  0,  0,  60, 60, 60, 0,  2,  2,  0,  2,
0),nrow=8, ncol=25, byrow = TRUE)
colnames(Xaic) <- c('sex',  'age',  'perfor',   'adhere',   'nodes',    'differ2',  'differ3',
'extent2',  'extent3',  'extent4',  'sex:age',  'sex:perfor',   'sex:nodes',    'sex:extent2',
'sex:extent3',  'sex:extent4',  'age:perfor',   'age:adhere',   'age:differ2',  'age:differ3',
'perfor:nodes', 'adhere:nodes', 'nodes:extent2',    'nodes:extent3',    'nodes:extent4')
# now get survival probability estimates
timevec <- c(365, 730, 1095, 1460, 1825)
Baic <- as.matrix(out1bothAIC$coefficients)
Baic <- Baic[-1]   # remove intercept
XBaic <- rep(0,8)
for (i in 1:8) {
XBaic[i] <- t(Xaic[i,])%*%(Baic)
}
resultsAIC <- matrix(rep(0,40), nrow=8, ncol=5)
intaic <- as.numeric(out1bothAIC$coef[1])
alphaaic <- as.numeric(out1bothAIC$scale)
for (i in 1:5){
  for (j in 1:8){
   resultsAIC[j,i] <- exp(-(timevec[i]*exp(-XBaic[j]-intaic))^(1/alphaaic) )
  }
}
#resultsAIC  # survival probabilities for times 1 to 5 in the time list given

# delta method to get std error
sebicAIC <- matrix(rep(0,40),nrow=8,ncol=5)
coefAIC <- c(as.numeric(out1bothAIC$coefficients),0.2971)

for (t in 1:5){
  for (r in 1:8){
    l01 = Xaic[r,1]; l02 = Xaic[r,2]; l03 = Xaic[r,3]; l04 = Xaic[r,4]; l05 = Xaic[r,5]; l06 = X
aic[r,6]; l07 = Xaic[r,7];
    l08 = Xaic[r,8]; l09 = Xaic[r,9]; l10 = Xaic[r,10]; l11 = Xaic[r,11]; l12 = Xaic[r,12]; l13
 = Xaic[r,13]; l14 = Xaic[r,14];
    l15 = Xaic[r,15]; l16 = Xaic[r,16]; l17 = Xaic[r,17]; l18 = Xaic[r,18]; l19 = Xaic[r,19]; l2
0 = Xaic[r,20];
    l21 = Xaic[r,21]; l22 = Xaic[r,22]; l23 = Xaic[r,23]; l24 = Xaic[r,24]; l25 = Xaic[r,25];
  sebicAIC[r,t] <- deltamethod(~(-l01*x2-l02*x3-l03*x4-l04*x5-l05*x6-l06*x7-l07*x8-l08*x9-l09*x1
0-l10*x11-l11*x12-l12*x13-
```

```
                                    l13*x14-l14*x15-l15*x16-l16*x17-l17*x18-l18*x19-l19*x20-l20*x2
1-l21*x22-l22*x23-l23*x24-
                                    l24*x25-l25*x26-x1+log(365))*exp(-x27),coefAIC, out1bothAIC$va
r)
}
}
#sebicAIC # estimated survival probabilities given time period 1 to 5 listed in problem

ulbicAIC <- matrix(rep(0,40),nrow=8,ncol=5)
llbicAIC <- matrix(rep(0,40),nrow=8,ncol=5)
for (t in 1:5){
  for (r in 1:8){
    ulbicAIC[r,t] <- exp( -exp((-XBaic[r]-out1bothAIC$coef[1]+log(timevec[t]))*(1/out1bothAIC$sc
ale)-1.96*sebicAIC[r,t]))
    llbicAIC[r,t] <- exp( -exp((-XBaic[r]-out1bothAIC$coef[1]+log(timevec[t]))*(1/out1bothAIC$sc
ale)+1.96*sebicAIC[r,t]))
  }
}
```

**Estimated Survival Probability for each of the 5 time periods and for all 8 observations:**

```
resultsAIC
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.8440539 0.7529627 0.6814831 0.6219688 0.5709278
## [2,] 0.7878112 0.6708919 0.5830623 0.5127274 0.4545415
## [3,] 0.9129425 0.8586146 0.8138157 0.7748276 0.7399914
## [4,] 0.8797398 0.8069963 0.7483998 0.6984585 0.6546907
## [5,] 0.8588060 0.7751167 0.7087218 0.6528978 0.6045863
## [6,] 0.8072493 0.6988251 0.6161105 0.5489538 0.4926848
## [7,] 0.8112670 0.7046557 0.6230682 0.5566407 0.5008380
## [8,] 0.7451047 0.6111438 0.5140028 0.4386222 0.3780531
```

**Upper Bound of the 95% Confidence Interval for each of the 5 time periods and 8 observations:**

```
ulbicAIC  # upper bound for 95% CI by observations in rows and time period in columns
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.8697598 0.7917334 0.7293331 0.6764913 0.6304567
## [2,] 0.8396468 0.7463946 0.6734611 0.6129155 0.5611318
## [3,] 0.9621853 0.9375228 0.9165001 0.8976539 0.8803464
## [4,] 0.9471356 0.9131109 0.8843948 0.8588797 0.8356397
## [5,] 0.8841607 0.8137947 0.7569335 0.7083339 0.6656304
## [6,] 0.8560123 0.7709014 0.7035176 0.6469664 0.5981087
## [7,] 0.9126844 0.8582085 0.8132955 0.7742143 0.7393001
## [8,] 0.8774692 0.8035134 0.7440377 0.6934209 0.6491210
```

**Lower Bound of the 95% Confidence Interval for each of the 5 time periods and 8 observations:**

```
llbicAIC  # lower bound for 95% CI by observations in rows and time period in columns
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.8138415 0.7084021 0.6275495 0.5616024 0.5061116
## [2,] 0.7222000 0.5800291 0.4789545 0.4018932 0.3409764
## [3,] 0.8063713 0.6975535 0.6145958 0.5472831 0.4909155
## [4,] 0.7391402 0.6029785 0.5047430 0.4288585 0.3681403
## [5,] 0.8284622 0.7298297 0.6533399 0.5903209 0.5367981
## [6,] 0.7446047 0.6104577 0.5132230 0.4377984 0.3772151
## [7,] 0.6195164 0.4487252 0.3385631 0.2615508 0.2053669
## [8,] 0.5156635 0.3300788 0.2235605 0.1564446 0.1119666
```

**2c) Log Likelihood tests.**

```
outfull2c <- survreg(Surv(time,status) ~ age+sex+rx+nodes+age*sex+age*rx+age*nodes+
                       sex*rx+sex*nodes+rx*nodes, data=colondata, dist="weibull")
outred2c <- survreg(Surv(time,status) ~ sex+rx+nodes+sex*rx+sex*nodes+rx*nodes, data=colondata,
 dist="weibull")
mytest <- as.numeric(-2*(logLik(outred2c)-logLik(outfull2c)))
print(mytest)
```

```
## [1] 12.47309
```

```
df1 <- df.residual(outred2c)-df.residual(outfull2c)
1-pchisq(mytest,df1)
```

```
## [1] 0.02884994
```

$H_0 : \beta_{age} = \beta_{age*sex} = \beta_{age*treat} = \beta_{age*nodes} = 0$ vs. $H_a$ : at least one is nonzero. The test statistic is 12.473 and the p-value is 0.0288. The test statistic has a chi-squared distribution with degrees of freedom = 5 (the # of coefficients removed from the full model). The high test statistic and low p-value are sufficient evidence to reject the null hypothesis that age does not have a statistically significant effect on the time-to-event.