Proportional Hazards Regression with R*

```
> rm(list=ls()); options(scipen=999)
> # install.packages("survival",dependencies=TRUE) # Only need to do this once
> library(survival) # Do this every time
> # install.packages("asaur",dependencies=TRUE) # Only need to do this once
> library(asaur)
> # summary(pharmacoSmoking)
> attach(pharmacoSmoking)
> # Make patch only the reference category
> contrasts(grp) = contr.treatment(2,base=2)
> colnames(contrasts(grp)) = c('Combo') # Names of dummy vars -- just one
> DayOfRelapse = Surv(ttr+1,relapse) # Day of relapse starts with one.
> # Collapse race categories
> Race = as.character(race) # Small r race is a factor. This is easier to modify.
> Race[Race!='white'] = 'blackOther'; Race=factor(Race)
>
>
> w_All = survreg(DayOfRelapse ~ grp + age + gender + Race + employment +
yearsSmoking + levelSmoking + priorAttempts, dist='weibull'); summary(w_All)

Call:
survreg(formula = DayOfRelapse ~ grp + age + gender + Race +
    employment + yearsSmoking + levelSmoking + priorAttempts,
    dist = "weibull")
                    Value Std. Error       z             p
(Intercept)       1.12177     0.9773  1.1479 0.25102045958
grpCombo          1.09225     0.3819  2.8603 0.00423234508
age               0.08432     0.0341  2.4722 0.01342778276
genderMale        0.03631     0.4142  0.0877 0.93014517788
Racewhite         0.25145     0.3914  0.6424 0.52061740468
employmentother  -1.28799     0.4672 -2.7569 0.00583496922
employmentpt     -1.28482     0.5863 -2.1914 0.02842409501
yearsSmoking     -0.02351     0.0325 -0.7232 0.46955818306
levelSmokinglight -0.07347     0.4315 -0.1703 0.86480382316
priorAttempts    -0.00105     0.0020 -0.5244 0.59996899163
Log(scale)        0.54194     0.0892  6.0774 0.00000000122

Scale= 1.72

Weibull distribution
Loglik(model)= -463.8   Loglik(intercept only)= -476.5
      Chisq= 25.41 on 9 degrees of freedom, p= 0.0025
Number of Newton-Raphson Iterations: 5
n= 125
```

* Copyright information is on the last page.

```
>
> ph_All = coxph(DayOfRelapse ~ grp + age + gender + Race + employment +
yearsSmoking + levelSmoking + priorAttempts); summary(ph_All)

Call:
coxph(formula = DayOfRelapse ~ grp + age + gender + Race + employment +
    yearsSmoking + levelSmoking + priorAttempts)

  n= 125, number of events= 89

                       coef  exp(coef)   se(coef)      z Pr(>|z|)
grpCombo          -0.5994057  0.5491379  0.2203690 -2.720  0.00653 **
age               -0.0479631  0.9531689  0.0198258 -2.419  0.01555 *
genderMale         0.0069130  1.0069369  0.2409092  0.029  0.97711
Racewhite         -0.1394286  0.8698551  0.2279991 -0.612  0.54085
employmentother    0.7086315  2.0312096  0.2727885  2.598  0.00938 **
employmentpt       0.7005798  2.0149207  0.3418680  2.049  0.04044 *
yearsSmoking       0.0144207  1.0145252  0.0188155  0.766  0.44342
levelSmokinglight  0.0329273  1.0334754  0.2495691  0.132  0.89503
priorAttempts      0.0004572  1.0004573  0.0011500  0.398  0.69095
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                  exp(coef) exp(-coef) lower .95 upper .95
grpCombo             0.5491     1.8210    0.3565    0.8458
age                  0.9532     1.0491    0.9168    0.9909
genderMale           1.0069     0.9931    0.6280    1.6146
Racewhite            0.8699     1.1496    0.5564    1.3599
employmentother      2.0312     0.4923    1.1900    3.4670
employmentpt         2.0149     0.4963    1.0310    3.9378
yearsSmoking         1.0145     0.9857    0.9778    1.0526
levelSmokinglight    1.0335     0.9676    0.6337    1.6855
priorAttempts        1.0005     0.9995    0.9982    1.0027

Concordance= 0.647  (se = 0.034 )
Rsquare= 0.168   (max possible= 0.998 )
Likelihood ratio test= 23.04  on 9 df,   p=0.006114
Wald test            = 22.73  on 9 df,   p=0.00684
Score (logrank) test = 23.23  on 9 df,   p=0.005693
```

```
> wfull =  survreg(DayOfRelapse ~ grp + age + employment , dist='weibull')
> summary(wfull)

Call:
survreg(formula = DayOfRelapse ~ grp + age + employment, dist = "weibull")
                  Value Std. Error     z             p
(Intercept)      1.4957     0.8414  1.78 0.07545324261
grpCombo         1.1023     0.3793  2.91 0.00365915983
age              0.0643     0.0186  3.45 0.00055474131
employmentother -1.2880     0.4617 -2.79 0.00527676297
employmentpt    -1.2123     0.5616 -2.16 0.03088499029
Log(scale)       0.5454     0.0894  6.10 0.00000000105

Scale= 1.73

Weibull distribution
Loglik(model)= -464.3   Loglik(intercept only)= -476.5
      Chisq= 24.31 on 4 degrees of freedom, p= 0.000069
Number of Newton-Raphson Iterations: 5
n= 125

>
> phfull =  coxph(DayOfRelapse ~ grp + age + employment); summary(phfull)

Call:
coxph(formula = DayOfRelapse ~ grp + age + employment)

  n= 125, number of events= 89

                    coef exp(coef) se(coef)      z Pr(>|z|)
grpCombo        -0.60788   0.54450  0.21837 -2.784  0.00537 **
age             -0.03529   0.96533  0.01075 -3.282  0.00103 **
employmentother  0.70348   2.02077  0.26929  2.612  0.00899 **
employmentpt     0.65369   1.92262  0.32732  1.997  0.04581 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                exp(coef) exp(-coef) lower .95 upper .95
grpCombo           0.5445     1.8365    0.3549    0.8354
age                0.9653     1.0359    0.9452    0.9859
employmentother    2.0208     0.4949    1.1920    3.4256
employmentpt       1.9226     0.5201    1.0122    3.6518

Concordance= 0.638  (se = 0.034 )
Rsquare= 0.162   (max possible= 0.998 )
Likelihood ratio test= 22.03  on 4 df,   p=0.0001979
Wald test            = 21.91  on 4 df,   p=0.0002084
Score (logrank) test = 22.48  on 4 df,   p=0.0001608

> # How are they getting the confidence intervals for those hazard ratios?
> L = -0.60788 -1.96*0.21837; L
[1] -1.035885
> exp(L)
[1] 0.3549121
```

```
> # Try Partial Likelihood and Wald tests for employment, controlling for age and
> # experimental treatment.
>
> # Partial Likelihood Ratio test
> nojob = coxph(DayOfRelapse ~ grp + age)
> anova(nojob,phfull) # LR test

Analysis of Deviance Table
 Cox model: response is  DayOfRelapse
 Model 1: ~ grp + age
 Model 2: ~ grp + age + employment
   loglik  Chisq Df P(>|Chi|)
1 -379.24
2 -375.14 8.2037  2   0.01654 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

>
> # Wald test: function(L,Tn,Vn,h=0) # H0: L theta = h
> source("http://www.utstat.toronto.edu/~brunner/Rfunctions/Wtest.txt")
> betahat = phfull$coefficients; Vn_hat = vcov(phfull)
> LL = rbind(c(0,0,1,0),
+            c(0,0,0,1) )
> Wtest(LL,betahat,Vn_hat)
         W         df    p-value
8.38888814 2.00000000 0.01507912

> # Estimating the survival function
> # help(survfit.coxph)
>
```

To make this work properly, I had to make my own dummy variable for treatment group. I was forced to do this because when the survival function was estimated, it somehow went back to the original dummy variable coding for grp. Everything was backwards. Luckily there was a warning (though I did not understand the warning for a while). Beware of a message like this: "Warning message: contrasts dropped from factor grp." If it happens, make your own dummy variables for the factor.

```
>
> n = length(grp); combo = numeric(n)
> combo[grp=='combination'] = 1
> phfull = coxph(DayOfRelapse ~ combo + age + employment); summary(phfull)
Call:
coxph(formula = DayOfRelapse ~ combo + age + employment)

  n= 125, number of events= 89

                    coef exp(coef) se(coef)      z Pr(>|z|)
combo           -0.60788   0.54450  0.21837 -2.784  0.00537 **
age             -0.03529   0.96533  0.01075 -3.282  0.00103 **
employmentother  0.70348   2.02077  0.26929  2.612  0.00899 **
employmentpt     0.65369   1.92262  0.32732  1.997  0.04581 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

                exp(coef) exp(-coef) lower .95 upper .95
combo              0.5445     1.8365    0.3549    0.8354
age                0.9653     1.0359    0.9452    0.9859
employmentother    2.0208     0.4949    1.1920    3.4256
employmentpt       1.9226     0.5201    1.0122    3.6518

Concordance= 0.638  (se = 0.034 )
Rsquare= 0.162   (max possible= 0.998 )
Likelihood ratio test= 22.03  on 4 df,   p=0.0001979
```

```
Wald test            = 21.91  on 4 df,   p=0.0002084
Score (logrank) test = 22.48  on 4 df,   p=0.0001608
> # Estimate S(t) for an average aged patient in the patch-only condition,
> # who is employed full-time.
> mean(age)
[1] 48.84
> patchonly = data.frame(combo=0,age=48.8,employment='ft')
> S1 = survfit(phfull,newdata=patchonly,conf.type='plain'); S1

Call: survfit(formula = phfull, newdata = patchonly, conf.type = "plain")

      n  events  median 0.95LCL 0.95UCL
    125      89      51      21      85
> # Plain is just the estimate plus or minus 1.96 * se
> summary(S1)
Call: survfit(formula = phfull, newdata = patchonly, conf.type = "plain")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
    1    125      12    0.916  0.0267        0.864        0.968
    2    113       5    0.879  0.0330        0.814        0.943
    3    108       6    0.833  0.0399        0.755        0.911
    4    102       1    0.825  0.0409        0.745        0.905
    5    101       3    0.802  0.0440        0.715        0.888
    6     98       2    0.786  0.0460        0.696        0.876
    7     96       1    0.778  0.0469        0.686        0.870
    8     95       1    0.771  0.0478        0.677        0.864
    9     94       3    0.747  0.0504        0.648        0.845
   11     91       1    0.739  0.0512        0.638        0.839
   13     90       2    0.723  0.0528        0.619        0.826
   15     88       7    0.666  0.0579        0.553        0.780
   16     81       4    0.633  0.0604        0.515        0.752
   17     77       1    0.625  0.0610        0.506        0.745
   21     76       1    0.617  0.0616        0.496        0.738
   22     75       2    0.601  0.0626        0.478        0.723
   26     73       1    0.592  0.0631        0.469        0.716
   29     72       3    0.567  0.0645        0.440        0.693
   31     69       3    0.541  0.0657        0.412        0.669
   41     66       1    0.532  0.0660        0.403        0.661
   43     65       1    0.523  0.0664        0.393        0.653
   46     64       1    0.515  0.0667        0.384        0.645
   50     63       1    0.506  0.0670        0.375        0.637
   51     62       1    0.497  0.0673        0.366        0.629
   57     61       5    0.453  0.0684        0.319        0.587
   61     56       2    0.436  0.0686        0.301        0.570
   64     54       2    0.418  0.0687        0.283        0.552
   66     52       1    0.409  0.0686        0.274        0.543
   76     51       1    0.400  0.0686        0.266        0.534
   78     50       2    0.382  0.0683        0.249        0.516
   81     48       1    0.374  0.0682        0.240        0.507
   85     47       1    0.365  0.0680        0.232        0.498
  101     46       1    0.356  0.0679        0.223        0.489
  106     45       1    0.347  0.0677        0.215        0.480
  111     44       1    0.338  0.0675        0.206        0.470
  141     43       4    0.302  0.0660        0.172        0.431
  156     39       1    0.293  0.0656        0.164        0.421
  171     38       2    0.274  0.0645        0.148        0.401
```

```
> # Estimate S(t) for an average aged patient in the combination condition,
> # employed full-time.
> combination = data.frame(combo=1,age=48.8,employment='ft')
> S2 = survfit(phfull,newdata=combination,conf.type='plain'); S2
Call: survfit(formula = phfull, newdata = combination, conf.type = "plain")

      n  events  median 0.95LCL 0.95UCL
    125      89     171      64      NA
> summary(S2)
Call: survfit(formula = phfull, newdata = combination, conf.type = "plain")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
    1    125      12    0.953  0.0164        0.921        0.985
    2    113       5    0.932  0.0210        0.891        0.973
    3    108       6    0.905  0.0264        0.853        0.957
    4    102       1    0.901  0.0273        0.847        0.954
    5    101       3    0.887  0.0299        0.828        0.945
    6     98       2    0.877  0.0316        0.815        0.939
    7     96       1    0.872  0.0324        0.809        0.936
    8     95       1    0.868  0.0333        0.802        0.933
    9     94       3    0.853  0.0358        0.783        0.923
   11     91       1    0.848  0.0366        0.776        0.920
   13     90       2    0.838  0.0382        0.763        0.913
   15     88       7    0.802  0.0438        0.716        0.887
   16     81       4    0.780  0.0469        0.688        0.872
   17     77       1    0.774  0.0477        0.681        0.868
   21     76       1    0.769  0.0484        0.674        0.864
   22     75       2    0.758  0.0499        0.660        0.856
   26     73       1    0.752  0.0507        0.653        0.851
   29     72       3    0.734  0.0528        0.630        0.837
   31     69       3    0.715  0.0549        0.608        0.823
   41     66       1    0.709  0.0556        0.600        0.818
   43     65       1    0.703  0.0563        0.593        0.813
   46     64       1    0.697  0.0569        0.585        0.808
   50     63       1    0.690  0.0576        0.577        0.803
   51     62       1    0.684  0.0582        0.570        0.798
   57     61       5    0.650  0.0614        0.530        0.770
   61     56       2    0.636  0.0626        0.513        0.759
   64     54       2    0.622  0.0638        0.497        0.747
   66     52       1    0.614  0.0644        0.488        0.741
   76     51       1    0.607  0.0650        0.480        0.734
   78     50       2    0.593  0.0661        0.463        0.722
   81     48       1    0.585  0.0666        0.455        0.716
   85     47       1    0.578  0.0672        0.446        0.709
  101     46       1    0.570  0.0677        0.437        0.703
  106     45       1    0.562  0.0682        0.428        0.696
  111     44       1    0.554  0.0686        0.420        0.689
  141     43       4    0.521  0.0703        0.383        0.659
  156     39       1    0.512  0.0706        0.374        0.651
  171     38       2    0.494  0.0712        0.355        0.634
```

```
> plot(S1,lwd=2,xlab='Day',ylab='Probability'); lines(S2,col='red',lwd=2)
> legend(x=125,y=1.0, col=c(2,1), lwd=2, legend=c('Combination','Patch Only'))
> title('Probability of lasting beyond Day',sub='Dotted lines are 95% confidence
bands')
```

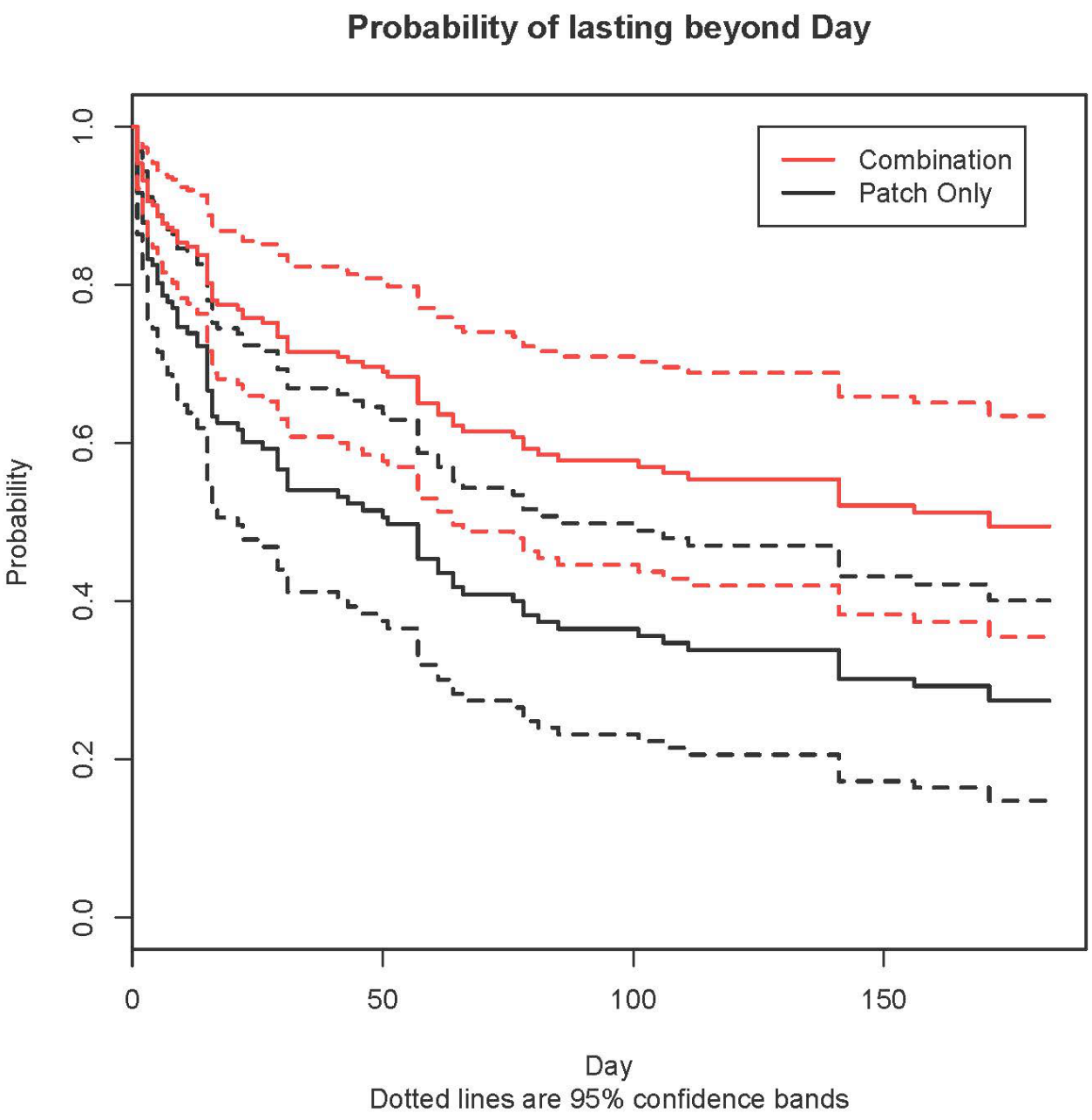
Probability of lasting beyond Day
Combination
Patch Only
Probability
0.0
0.2
0.4
0.6
0.8
1.0
0
50
100
150
Day
Dotted lines are 95% confidence bands

Illustration with Simulated Data

```
> rm(list=ls());  options(scipen=999)
> library(survival)
>
> ##################### beta0 = -8 ####################
> Ex = 5; SDx = 1 # Parameters of (normal) explanatory variable X
> beta0 = -8; beta1 = 2  # Regression parameters.
>
> # Simulate
> set.seed(9999)
> n = 10000; delta = numeric(n) # Indicator for uncensored, initially zero
> x = round(rnorm(n,Ex,SDx),1)
> mu = beta0 + beta1*x
> epsilon = rexp(n)
> lifetime = exp(mu)*epsilon
> # sort(lifetime)
> censortime = 1/runif(n) - 1 # Shifted Pareto censoring time
> # If censoring time is greater than lifetime, then it's NOT censored.
> delta[censortime>lifetime] = 1; # table(delta)
> # Minumum of censortime and lifetime is what we can observe.
> Time = pmin(censortime,lifetime) # pmin is parallel minimum.
> # round(cbind(x,lifetime,censortime,Time,delta)[1:10,],3) # Take a look
> exdata1 = cbind(x,Time,delta); # wdata # This is all you can see in practice.
> # head(exdata1)
> max(Time); table(delta)
[1] 321.6378
delta
   0    1
6919 3081
> minus8 = coxph(Surv(Time,delta)~x); minus8$coefficients
        x
-1.949555
> typical = data.frame(x=5)
> Sminus8 = survfit(minus8,newdata=typical,se.fit=FALSE); Sminus8
Call: survfit(formula = minus8, newdata = typical, se.fit = FALSE)

      n  events  median
10000.0  3081.0     4.8
>
> # Estimate S(t) at x=E(X). True S(t) = exp(-lambda*t),
> # with lambda = exp(-(beta0+beta1*x))
> lambdam8 = exp(-(beta0+beta1*Ex))
> log(2)/lambdam8 # True median
[1] 5.121703
```

```
>
>
> ##################### beta0 = -7 ####################
> Ex = 5; SDx = 1 # Parameters of (normal) explanatory variable X
> beta0 = -7; beta1 = 2 # Regression parameters.
>
> # Simulate
> set.seed(7777)
> n = 10000; delta = numeric(n) # Indicator for uncensored, initially zero
> x = round(rnorm(n,Ex,SDx),1)
> mu = beta0 + beta1*x
> epsilon = rexp(n)
> lifetime = exp(mu)*epsilon
> # sort(lifetime)
> censortime = 1/runif(n) - 1 # Shifted Pareto censoring time
> # If censoring time is greater than lifetime, then it's NOT censored.
> delta[censortime>lifetime] = 1; # table(delta)
> # Minumum of censortime and lifetime is what we can observe.
> Time = pmin(censortime,lifetime) # pmin is parallel minimum.
> # round(cbind(x,lifetime,censortime,Time,delta)[1:10,],3) # Take a look
> exdata2 = cbind(x,Time,delta); # wdata # This is all you can see in practice.
> # head(exdata2)
> max(Time); table(delta)
[1] 542.5331
delta
   0    1
8022 1978
> minus7 = coxph(Surv(Time,delta)~x); minus7$coefficients
        x
-2.005341
>
> typical = data.frame(x=5)
> Sminus7 = survfit(minus7,newdata=typical,se.fit=FALSE); Sminus7
Call: survfit(formula = minus7, newdata = typical, se.fit = FALSE)

      n  events  median
10000.0  1978.0    14.5
>
> # Estimate S(t) at x=E(X). True S(t) = exp(-lambda*t),
> # with lambda = exp(-(beta0+beta1*x))
> lambdam7 = exp(-(beta0+beta1*Ex))
> log(2)/lambdam7 # True median
[1] 13.92223
```

```
> plot(Sminus8,xlim = c(0,125)); lines(Sminus7,xlim = c(0,125),col='red')
>
> # Now add lines showing the two true survival curves
> tt = 0:125
> trueSm8 = exp(-lambdam8*tt); trueSm7 = exp(-lambdam7*tt)
> lines(tt,trueSm8)
> lines(tt,trueSm7,col='red')
>
> key = c(expression(paste(beta[0],' = -8')),expression(paste(beta[0],' = -7')))
> legend("topright", lwd=1, col=1:2,  legend=key)
>
> title("Estimated and True Survival Curves")
>
```