Poisson GLM, Cox PH, & degrees of freedom

Michael C. Donohue
Division of Biostatistics and Bioinformatics
University of California, San Diego

December 4, 2010

1 Introduction

We discuss connections between the Cox proportional hazards model and Poisson generalized linear models as described in Whitehead (1980). We fit a sample dataset using coxph() and glm() and show that the model degrees of freedom differ by the number of events.

2 A simple Cox PH example

2.1 Generate data

We generate proportional hazards mixed model data.

```
> options(width = 75)
> library(phmm)
> n <- 50
> nclust <- 5
> clusters <- rep(1:nclust, each = n/nclust)
> beta0 <- c(1, 2)
> set.seed(13)
> Z <- cbind(Z1 = sample(0:1, n, replace = TRUE), Z2 = sample(0:1,
+     n, replace = TRUE), Z3 = sample(0:1, n, replace = TRUE))
> b <- cbind(rep(rnorm(nclust), each = n/nclust), rep(rnorm(nclust),
+     each = n/nclust))
> Wb <- matrix(0, n, 2)
> for (j in 1:2) Wb[, j] <- Z[, j] * b[, j]
> Wb <- apply(Wb, 1, sum)
> T <- -log(runif(n, 0, 1)) * exp(-Z[, c("Z1", "Z2")] %*% beta0 -
```

```
+     Wb)
> C <- runif(n, 0, 1)
> time <- ifelse(T < C, T, C)
> event <- ifelse(T <= C, 1, 0)
> sum(event)

[1] 31

> phmmd <- data.frame(Z)
> phmmd$cluster <- clusters
> phmmd$time <- time
> phmmd$event <- event
```

2.2 Fit the Cox PH model

```
> fit.ph <- coxph(Surv(time, event) ~ Z1 + Z2, phmmd, method = "breslow",
+     x = TRUE, y = TRUE)
> summary(fit.ph)

Call:
coxph(formula = Surv(time, event) ~ Z1 + Z2, data = phmmd, method = "breslow",
    x = TRUE, y = TRUE)

  n= 50

     coef exp(coef) se(coef)     z Pr(>|z|)
Z1 0.8549    2.3513   0.3918 2.182  0.02909 *
Z2 1.0888    2.9708   0.3684 2.955  0.00312 **
---
Signif. codes:  0 `***' 0.001 `**' 0.01 `*' 0.05 `.' 0.1 ` ' 1

   exp(coef) exp(-coef) lower .95 upper .95
Z1     2.351     0.4253     1.091     5.067
Z2     2.971     0.3366     1.443     6.116

Rsquare= 0.237   (max possible= 0.984 )
Likelihood ratio test= 13.55  on 2 df,   p=0.001141
Wald test            = 13.52  on 2 df,   p=0.001158
Score (logrank) test = 14.63  on 2 df,   p=0.0006671

> fit.ph$loglik[2]

[1] -95.97131
```

Next we create data to fit an auxilary Poisson model as described in Whitehead (1980) using the pseudoPoisPHMM() function provided in the phmm package. This function also extracts the linear predictors as estimated from the Cox PH model so that we can calculate likelihoods and degrees of freedom.

2.3 Likelihood and degrees of freedom for Poisson GLM from Cox PH parameters

```
> ppd <- as.data.frame(as.matrix(pseudoPoisPHMM(fit.ph)))
> poisl <- c()
> eventtimes <- sort(phmmd$time[phmmd$event == 1])
> for (h in 1:length(eventtimes)) {
+     js <- ppd$time == eventtimes[h] & ppd$m >= 1
+     j <- ppd$time == eventtimes[h]
+     if (sum(js) > 1)
+         stop("tied event times")
+     poisl <- c(poisl, ppd[js, "N"] * exp(-1) * exp(ppd[js,
+         "linear.predictors"])/sum(ppd[j, "N"] * exp(ppd[j,
+         "linear.predictors"])))
+ }
```

Poisson likelihood:

```
> sum(log(poisl))

[1] -66.5633

> sum(log(poisl)) - fit.ph$loglik[2]

[1] 29.40801
```

Poisson degrees of freedom

```
> length(fit.ph$coef) + sum(phmmd$event)

[1] 33
```

2.4 Fit auxiliary Poisson GLM

We fit an auxiliary Poisson GLM and note that the parameter estimates for z1 and z2 are identical to the coxph() fit, and the likelihood and degrees of freedom are as expected.

```
> ppd$t <- as.factor(ppd$time)
> fit.glm <- glm(m ~ -1 + t + z1 + z2 + offset(log(N)), ppd,
+     family = poisson)
> summary(fit.glm)

Call:
glm(formula = m ~ -1 + t + z1 + z2 + offset(log(N)), family = poisson,
    data = ppd)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
```

```
-0.9685  -0.7531  -0.5553   0.4293   1.6823

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)    
t0.000277233256778163  -5.0494     1.0704  -4.717 2.39e-06 ***
t0.000285092717793308  -5.0035     1.0679  -4.685 2.79e-06 ***
t0.000382448373472765  -4.9876     1.0683  -4.669 3.03e-06 ***
t0.00559427171447325   -4.9388     1.0655  -4.635 3.57e-06 ***
t0.00764335258097282   -4.8875     1.0625  -4.600 4.22e-06 ***
t0.00808285780728387   -4.8648     1.0635  -4.574 4.78e-06 ***
t0.0216256697018544    -4.8013     1.0609  -4.526 6.02e-06 ***
t0.0219649983261458    -4.7930     1.0622  -4.512 6.41e-06 ***
t0.0233956453029104    -4.7681     1.0634  -4.484 7.34e-06 ***
t0.0235837855332384    -4.7069     1.0598  -4.441 8.95e-06 ***
t0.0237625311885084    -4.6797     1.0612  -4.410 1.03e-05 ***
t0.027482795605763     -4.6127     1.0572  -4.363 1.28e-05 ***
t0.0278642961804028    -4.5890     1.0573  -4.340 1.42e-05 ***
t0.0316525538364514    -4.5401     1.0576  -4.293 1.76e-05 ***
t0.0357745779481545    -4.5147     1.0578  -4.268 1.97e-05 ***
t0.0366185731334857    -4.4351     1.0529  -4.212 2.53e-05 ***
t0.066999301944422     -4.3869     1.0556  -4.156 3.24e-05 ***
t0.0742904888064418    -4.3572     1.0557  -4.127 3.67e-05 ***
t0.09491415021304      -4.2493     1.0513  -4.042 5.30e-05 ***
t0.125132209250348     -4.2151     1.0513  -4.010 6.08e-05 ***
t0.132722661166308     -4.1798     1.0513  -3.976 7.01e-05 ***
t0.140357744467437     -4.0667     1.0439  -3.896 9.79e-05 ***
t0.163527928343998     -3.9258     1.0448  -3.757 0.000172 ***
t0.193971448733795     -3.7760     1.0443  -3.616 0.000299 ***
t0.204887967162952     -3.7054     1.0458  -3.543 0.000396 ***
t0.227852125295401     -3.6459     1.0457  -3.486 0.000490 ***
t0.266238317485871     -3.5253     1.0513  -3.353 0.000799 ***
t0.276177426334698     -3.2951     1.0356  -3.182 0.001464 ** 
t0.360993505812205     -3.2039     1.0353  -3.095 0.001970 ** 
t0.426697507683412     -2.7934     1.0367  -2.694 0.007051 ** 
t0.511995413073629     -1.8487     1.0105  -1.830 0.067323 .  
z1                      0.8549     0.3918   2.182 0.029092 *  
z2                      1.0888     0.3684   2.955 0.003123 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 1743.184  on 121  degrees of freedom
Residual deviance:   71.127  on  88  degrees of freedom
AIC: 199.13
```

```
Number of Fisher Scoring iterations: 6

> fit.ph$coef

       Z1        Z2
0.8549497 1.0888337

> logLik(fit.glm)

'log Lik.' -66.5633 (df=33)

> logLik(fit.glm)[1] - sum(log(poisl))

[1] -1.421085e-14
```

The additional parameter estimates correspond to the estimated log baseline hazard, which we verify using the basehaz() function.

```
> bh <- basehaz(fit.ph, centered = FALSE)
> log(bh$hazard - c(0, bh$hazard[1:(length(bh$hazard) - 1)]))[1:10]

 [1] -5.049378 -5.003546 -4.987633 -4.938810 -4.887479 -4.864823      -Inf
 [8] -4.801254 -4.793001 -4.768072
```

3 Extending to PHMM

3.1 Fit PHMM

```
> fit.phmm <- phmm(Surv(time, event) ~ Z1 + Z2 + (Z1 + Z2 |
+     cluster), phmmd, Gbs = 100, Gbsvar = 1000, VARSTART = 1,
+     NINIT = 10, MAXSTEP = 100, CONVERG = 90)
> summary(fit.phmm)

Proportional Hazards Mixed-Effects Model fit by MCMC-EM
  Model: Surv(time, event) ~ Z1 + Z2 + (Z1 + Z2 | cluster)
  Data: phmmd
  Log-likelihood:
Conditional     Laplace         RIS
     -83.44     -122.59     -122.58

Fixed effects: Surv(time, event) ~ Z1 + Z2
   Estimate Std.Error
Z1   0.8113    0.5643
Z2   1.5018    0.5202

Random effects: (Z1 + Z2 | cluster)
Estimated variance-covariance matrix:
```

```
             (Intercept)     Z1     Z2
(Intercept)       0.1461 0.0000 0.0000
Z1                0.0000 0.5193 0.0000
Z2                0.0000 0.0000 0.3705

Number of Observations: 50
Number of Groups:  5
```

3.2 Likelihood and degrees of freedom for Poisson GLMM from PHMM parameters

```
> ppd <- as.data.frame(as.matrix(pseudoPoisPHMM(fit.phmm)))
> poisl <- c()
> eventtimes <- sort(phmmd$time[phmmd$event == 1])
> for (h in 1:length(eventtimes)) {
+     js <- ppd$time == eventtimes[h] & ppd$m >= 1
+     j <- ppd$time == eventtimes[h]
+     if (sum(js) > 1)
+         stop("tied event times")
+     poisl <- c(poisl, ppd[js, "N"] * exp(-1) * exp(ppd[js,
+         "linear.predictors"])/sum(ppd[j, "N"] * exp(ppd[j,
+         "linear.predictors"])))
+ }
```

Poisson likelihood:

```
> sum(log(poisl))

[1] -93.47353

> sum(log(poisl)) - fit.phmm$loglik[1]

Conditional
  -10.03456
```

Poisson degrees of freedom

```
> traceHat(fit.phmm, "pseudoPois")

[1] 6.493713
```

3.3 Fit auxiliary Poisson GLMM

We fit an auxiliary Poisson GLMM, although with a general variance-covariance matrix for the random effects (phmm() only fits models with diagonal variance-covariance matrix).

```
> library(lme4)
> ppd$t <- as.factor(ppd$time)
> fit.lmer <- lmer(m ~ -1 + t + z1 + z2 + (z1 + z2 | cluster) +
+     offset(log(N)), data = ppd, family = poisson)
> summary(fit.lmer)@coefs
```

```
                        Estimate Std. Error   z value     Pr(>|z|)
t0.000277233256778163 -5.9512950  1.1621175 -5.121079 3.037929e-07
t0.000285092717793308 -5.8056783  1.1490919 -5.052406 4.362801e-07
t0.000382448373472765 -5.7868612  1.1507629 -5.028717 4.937722e-07
t0.00559427171447325  -5.6896197  1.1452691 -4.967932 6.767055e-07
t0.00764335258097282  -5.5818941  1.1399125 -4.896774 9.742267e-07
t0.00808285780728387  -5.5730675  1.1412587 -4.883264 1.043440e-06
t0.0216256697018544   -5.3492807  1.1177603 -4.785714 1.703809e-06
t0.0219649983261458   -5.3458771  1.1185332 -4.779364 1.758508e-06
t0.0233956453029104   -5.3007331  1.1214043 -4.726871 2.280062e-06
t0.0235837855332384   -5.0000135  1.0919889 -4.578814 4.676208e-06
t0.0237625311885084   -4.9356030  1.0944301 -4.509747 6.490492e-06
t0.027482795605763    -4.9049582  1.0929065 -4.487994 7.189685e-06
t0.0278642961804028   -4.8723491  1.0932428 -4.456786 8.319764e-06
t0.0316525538364514   -4.8144717  1.0953386 -4.395419 1.105594e-05
t0.0357745779481545   -4.7629377  1.0974199 -4.340124 1.424027e-05
t0.0366185731334857   -4.4645524  1.0678209 -4.180994 2.902377e-05
t0.066999301944422    -4.3400451  1.0743885 -4.039549 5.355409e-05
t0.0742904888064418   -4.3164355  1.0737793 -4.019853 5.823435e-05
t0.09491415021304     -4.2590330  1.0717760 -3.973809 7.073233e-05
t0.125132209250348    -4.1958733  1.0721793 -3.913407 9.100313e-05
t0.132722661166308    -4.1804215  1.0723708 -3.898298 9.687100e-05
t0.140357744467437    -4.0478306  1.0566013 -3.830992 1.276278e-04
t0.163527928343998    -3.8492359  1.0551029 -3.648209 2.640750e-04
t0.193971448733795    -3.5671867  1.0538877 -3.384788 7.123320e-04
t0.204887967162952    -3.4476930  1.0549748 -3.268033 1.082976e-03
t0.227852125295401    -3.3890557  1.0542696 -3.214601 1.306261e-03
t0.266238317485871    -3.2575659  1.0623147 -3.066479 2.165958e-03
t0.276177426334698    -3.0812874  1.0453374 -2.947649 3.202008e-03
t0.360993505812205    -2.8597193  1.0454778 -2.735323 6.231912e-03
t0.426697507683412    -2.3978080  1.0400191 -2.305542 2.113622e-02
t0.511995413073629    -1.6399385  1.0154905 -1.614922 1.063275e-01
z1                     0.7503827  0.4575505  1.640000 1.010052e-01
z2                     1.5467416  0.4543444  3.404337 6.632481e-04
```

```
> fit.phmm$coef
```

```
       Z1        Z2
0.8113196 1.5018102
```

```
> logLik(fit.lmer)
```

```
'log Lik.' -69.97819 (df=39)
```

```
> sum(log(poisl)) - logLik(fit.lmer)[1]
```

```
[1] -23.49534
```

```
> log(fit.phmm$lambda)[1:10]
```

```
 [1] -5.914610 -5.789140 -5.766056 -5.669644 -5.562935 -5.553536      -Inf
 [8] -5.363577 -5.358958 -5.307990
```