

Poisson GLM, Cox PH, & degrees of freedom

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

September 14, 2013

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          # total sample size
> nclust <- 5      # number of clusters
> 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, x = TRUE,
      y = TRUE, method = "breslow")

```

n= 50, number of events= 31

	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

Concordance= 0.71 (se = 0.055)

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 auxiliary 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)))
> # pois likelihood
> poisl <- c()
> eventtimes <- sort(phmmd$time[phmmd$event == 1])
> for(h in 1:length(eventtimes)){
+   js <- ppd$time == eventtimes[h] & ppd$m >= 1 # j star
+   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 z_1 and z_2 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(pois1))
[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
-83.4	-122.7	-122.5

Fixed effects: `Surv(time, event) ~ Z1 + Z2`

	Estimate	Std.Error
Z1	0.808	0.5399
Z2	1.490	0.5710

Random effects: `(Z1 + Z2 | cluster)`

Estimated variance-covariance matrix:

	(Intercept)	Z1	Z2
(Intercept)	0.2125	0.0000	0.00
Z1	0.0000	0.4065	0.00
Z2	0.0000	0.0000	0.35

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)))
> pois1 <- c()
> eventtimes <- sort(phmmd$time[phmmd$event == 1])
> for(h in 1:length(eventtimes)){
+   js <- ppd$time == eventtimes[h] & ppd$m >= 1 # j star
+   j <- ppd$time == eventtimes[h]
+   if(sum(js) > 1) stop("tied event times")
+   pois1 <- c(pois1,
+     ppd[js, "N"]*exp(-1)*exp(ppd[js, "linear.predictors"])/
+     sum(ppd[j, "N"]*exp(ppd[j, "linear.predictors"])))
+ }
```

Poisson likelihood:

```
> sum(log(pois1))

[1] -93.43185

> sum(log(pois1)) - fit.phmm$loglik[1]
```

Conditional
-10.03456

Poisson degrees of freedom

```
> # Poisson GLMM degrees of freedom length(unique(x$cluster)) * x$nrandom + x$nfixed
> traceHat(fit.phmm, "pseudoPois") # + 2*sum(phmmd$event)

[1] 6.480863
```

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)$coef
```

	Estimate	Std. Error	z value	Pr(> z)
t0.000277233256778163	-5.9512281	1.1615657	-5.123454	2.999891e-07
t0.000285092717793308	-5.8056880	1.1486051	-5.054555	4.313950e-07
t0.000382448373472765	-5.7879584	1.1507489	-5.029732	4.911661e-07
t0.00559427171447325	-5.6897184	1.1448282	-4.969932	6.697627e-07
t0.00764335258097282	-5.5823529	1.1396489	-4.898310	9.666464e-07
t0.00808285780728387	-5.5735517	1.1410056	-4.884772	1.035488e-06
t0.0216256697018544	-5.3488322	1.1171433	-4.787955	1.684891e-06
t0.0219649983261458	-5.3472295	1.1187280	-4.779740	1.755217e-06
t0.0233956453029104	-5.3014895	1.1213157	-4.727919	2.268328e-06
t0.0235837855332384	-5.0011204	1.0921693	-4.579071	4.670468e-06
t0.0237625311885084	-4.9345145	1.0935852	-4.512236	6.414773e-06
t0.027482795605763	-4.9068617	1.0934597	-4.487465	7.207572e-06
t0.0278642961804028	-4.8724548	1.0929589	-4.458040	8.271241e-06
t0.0316525538364514	-4.8148580	1.0951772	-4.396419	1.100513e-05
t0.0357745779481545	-4.7639638	1.0975445	-4.340566	1.421164e-05
t0.0366185731334857	-4.4649876	1.0677586	-4.181645	2.894075e-05
t0.066999301944422	-4.3399389	1.0740497	-4.040725	5.328631e-05
t0.0742904888064418	-4.3164130	1.0734800	-4.020953	5.796306e-05
t0.09491415021304	-4.2594803	1.0717029	-3.974497	7.052803e-05
t0.125132209250348	-4.1966683	1.0722568	-3.913865	9.083039e-05
t0.132722661166308	-4.1798589	1.0718076	-3.899822	9.626341e-05
t0.140357744467437	-4.0487452	1.0568006	-3.831135	1.275537e-04
t0.163527928343998	-3.8490278	1.0547774	-3.649138	2.631221e-04
t0.193971448733795	-3.5667256	1.0534138	-3.385873	7.095218e-04
t0.204887967162952	-3.4477845	1.0548029	-3.268653	1.080608e-03
t0.227852125295401	-3.3893374	1.0541947	-3.215096	1.304008e-03
t0.266238317485871	-3.2582771	1.0623812	-3.066957	2.162501e-03
t0.276177426334698	-3.0818402	1.0454457	-2.947872	3.199695e-03
t0.360993505812205	-2.8612940	1.0460557	-2.735317	6.232019e-03
t0.426697507683412	-2.3977423	1.0399353	-2.305665	2.112936e-02
t0.511995413073629	-1.6411486	1.0162063	-1.614976	1.063159e-01
z1	0.7520095	0.4576641	1.643147	1.003525e-01
z2	1.5486597	0.4543010	3.408884	6.522915e-04

```

> fit.phmm$coef

      Z1      Z2
0.8080001 1.4904493

> logLik(fit.lmer)

'log Lik.' -100.9788 (df=39)

> sum(log(pois1)) - logLik(fit.lmer)[1]

[1] 7.546904

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

[1] -5.889349 -5.767659 -5.744810 -5.648380 -5.541649 -5.532449      -Inf
[8] -5.349149 -5.344617 -5.289886

```