Introduction to `rstpm2`

Mark Clements

July 7, 2015

1 Mean survival

This has a useful interpretation for causal inference.

$$E_Z(S(t|Z, X = 1)) - E_Z(S(t|Z, X = 0))$$

```
fit <- rstpm()
predict(fit,type="meansurvdiff",newdata=data)
```

2 Cure models

For cure, we use the melanoma dataset used by Andersson and colleagues for cure models with Stata's stpm2 (see http://www.pauldickman.com/survival/).

Initially, we merge the patient data with the all cause mortality rates.

```
> require(rstpm2)
> data(popmort)
> data(colon)
> popmort2 <- transform(popmort,exitage=age,exityear=year,age=NULL,year=NULL)
> colon2 <- within(colon, {
+    status <- ifelse(surv_mm>120.5,1,status)
+    tm <- pmin(surv_mm,120.5)/12
+    exit <- dx+tm*365.25
+    sex <- as.numeric(sex)
+    exitage <- pmin(floor(age+tm),99)
+    exityear <- floor(yydx+tm)
+    ##year8594 <- (year8594=="Diagnosed 85-94")
+ })
> colon2 <- merge(colon2,popmort2)
```

For comparisons, we fit the relative survival model without and with cure.

```
> fit0 <- stpm2(Surv(tm,status %in% 2:3)~I(year8594=="Diagnosed 85-94"),
+                     data=colon2,
+                     bhazard=colon2$rate, df=5)

> summary(fit <- stpm2(Surv(tm,status %in% 2:3)~I(year8594=="Diagnosed 85-94"),
+                     data=colon2,
+                     bhazard=colon2$rate,
+                     df=5,cure=TRUE))
```

```
Maximum likelihood estimation

Call:
mle2(minuslogl = negll, start = coef, eval.only = TRUE, vecpar = TRUE,
    gr = function (beta, kappa = 1)
    {
        eta <- as.vector(pars$X %*% beta)
        etaD <- as.vector(pars$XD %*% beta)
        h <- link$h(eta, etaD)
        gradh <- link$gradh(eta, etaD, pars)
        gradH <- link$gradH(eta, etaD, pars)
        gconstraint <- colSums((kappa * h * pars$wt * gradh)[h <
            0, ])
        h[h < 0] <- 1e-16
        g <- colSums(pars$wt * (-gradH + ifelse(pars$event, 1/h,
            0) * gradh)) - gconstraint
        if (delayed) {
            eta <- as.vector(pars0$X %*% beta)
            etaD <- as.vector(pars0$XD %*% beta)
            gradH <- link$gradH(eta, etaD, pars0)
            g <- g + colSums(gradH * pars0$wt)
        }
        return(-g)
    }, control = list(parscale = c(0.1, 0.1, 0.1, 0.1, 0.1, 0.1,
    0.1), maxit = 300), lower = -Inf, upper = Inf)

Coefficients:
                                      Estimate Std. Error  z value      Pr(z)
(Intercept)                          -3.977326   0.054778 -72.6078 < 2.2e-16
I(year8594 == "Diagnosed 85-94")TRUE -0.155613   0.025088  -6.2027 5.551e-10
nsx(log(tm), df = 5, cure = TRUE)1    3.323188   0.053166  62.5062 < 2.2e-16
nsx(log(tm), df = 5, cure = TRUE)2    3.628637   0.053160  68.2592 < 2.2e-16
nsx(log(tm), df = 5, cure = TRUE)3    1.634854   0.022465  72.7742 < 2.2e-16
nsx(log(tm), df = 5, cure = TRUE)4    6.592018   0.111504  59.1190 < 2.2e-16
nsx(log(tm), df = 5, cure = TRUE)5    3.371806   0.042788  78.8025 < 2.2e-16

(Intercept)                          ***
I(year8594 == "Diagnosed 85-94")TRUE ***
nsx(log(tm), df = 5, cure = TRUE)1   ***
nsx(log(tm), df = 5, cure = TRUE)2   ***
nsx(log(tm), df = 5, cure = TRUE)3   ***
nsx(log(tm), df = 5, cure = TRUE)4   ***
nsx(log(tm), df = 5, cure = TRUE)5   ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

-2 log L: 42190.77

> predict(fit,head(colon2),se.fit=TRUE)

   Estimate     lower      upper
```

```
1 0.8610835 0.8542905 0.8675849
2 0.7934666 0.7850118 0.8016323
3 0.6967422 0.6863213 0.7068947
4 0.8610835 0.8542905 0.8675849
5 0.8221253 0.8143237 0.8296344
6 0.8610835 0.8542905 0.8675849
```

The estimate for the year parameter from the model without cure is within three significant figures with that in Stata. For the predictions, the Stata model gives:

```
     +--------------------------------+
     |       surv    surv_lci   surv_uci |
     |--------------------------------|
  1. | .86108264    .8542898    .8675839 |
  2. | .79346526    .7850106    .8016309 |
  3. | .69674037    .6863196    .7068927 |
  4. | .86108264    .8542898    .8675839 |
  5. | .82212425    .8143227    .8296332 |
     |--------------------------------|
  6. | .86108264    .8542898    .8675839 |
     +--------------------------------+
```

We can estimate the proportion of failures at the end of follow-up using:

```
> newdata.eof <- data.frame(year8594 = unique(colon2$year8594),
+                           tm=max(colon2$tm))
> 1-predict(fit0, newdata.eof, type="surv", se.fit=TRUE)

   Estimate     lower     upper
1 0.6063672 0.6211721 0.5916026
2 0.5515138 0.5661327 0.5370117

> 1-predict(fit, newdata.eof, type="surv", se.fit=TRUE)

   Estimate     lower     upper
1 0.5913307 0.6055015 0.5772173
2 0.5350813 0.5485372 0.5217434
```
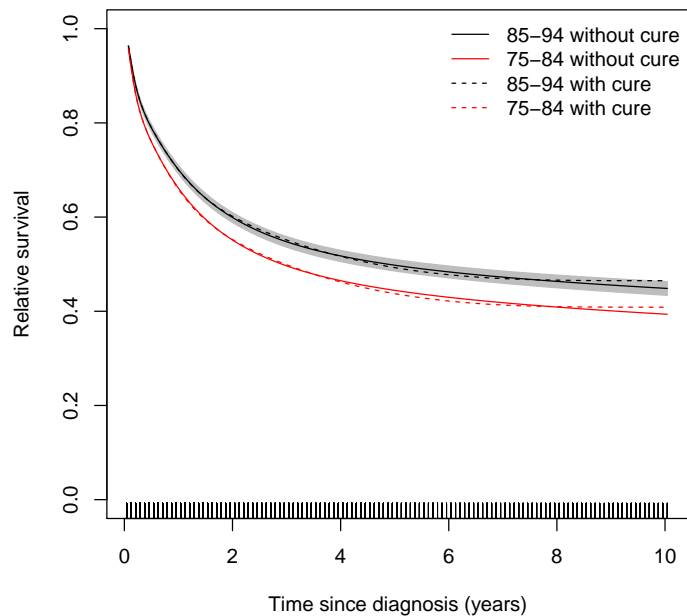
We can plot the predicted survival estimates:

```
> plot(fit0,newdata=data.frame(year8594 = "Diagnosed 85-94"), ylim=0:1,
+      xlab="Time since diagnosis (years)", ylab="Relative survival")
> plot(fit0,newdata=data.frame(year8594 = "Diagnosed 75-84"),
+      add=TRUE,line.col="red",rug=FALSE)
> plot(fit,newdata=data.frame(year8594 = "Diagnosed 85-94"),
+      add=TRUE,ci=FALSE,lty=2,rug=FALSE)
> plot(fit,newdata=data.frame(year8594="Diagnosed 75-84"),
+      add=TRUE,rug=FALSE,line.col="red",ci=FALSE,lty=2)
> legend("topright",c("85-94 without cure","75-84 without cure",
+                 "85-94 with cure","75-84 with cure"),
+      col=c(1,2,1,2), lty=c(1,1,2,2), bty="n")
```

And the hazard curves:

```
> plot(fit0,newdata=data.frame(year8594 = "Diagnosed 85-94"),
+      ylim=c(0,0.5), type="hazard",
+      xlab="Time since diagnosis (years)",ylab="Excess hazard")
> plot(fit0,newdata=data.frame(year8594 = "Diagnosed 75-84"),
+      type="hazard",
+      add=TRUE,line.col="red",rug=FALSE)
> plot(fit,newdata=data.frame(year8594 = "Diagnosed 85-94"),
+      type="hazard",
+      add=TRUE,ci=FALSE,lty=2,rug=FALSE)
> plot(fit,newdata=data.frame(year8594="Diagnosed 75-84"),
+      type="hazard",
+      add=TRUE,rug=FALSE,line.col="red",ci=FALSE,lty=2)
> legend("topright",c("85-94 without cure","75-84 without cure",
+                     "85-94 with cure","75-84 with cure"),
+       col=c(1,2,1,2), lty=c(1,1,2,2), bty="n")
```