

Using `csm` to estimate cause-specific mortality from left-truncated data

Glen A. Sargeant

USGS Northern Prairie Wildlife Research Center
`gsargeant@usgs.gov`

November 22, 2010

Heisey and Patterson (2006) provided S-Plus code to facilitate estimation of cause-specific mortality from left-truncated survival records ("staggered entry" in wildlife literature). Package `wild1` includes an implementation of their non-parametric cumulative incidence function estimator (function `csm`), which produces objects of a new class ("`csm`"). The package also includes methods for the new class and several functions that facilitate tasks commonly associated with wildlife survival analysis. This vignette demonstrates an application of the package to the coyote survival data used for examples presented by Heisey and Patterson (2006).

Software

Package `wild1` and required supporting packages must be installed prior to use. To install `wild1`, copy the file `wild1.zip` (available at <http://www.npwrc.usgs.gov/staff/sargeant.htm>) to your preferred location, initiate an R session, and use the pull-down "Packages" menu (*Packages>Install package(s) from local zip files*).

See the package DESCRIPTION file for a current list of required supporting packages (available from CRAN, the Comprehensive R Archive Network, via the pull-down "Packages" menu).

Installed packages must be loaded at the start of each R session or they will not be available for use. Either `require()` or `library()` may be used to load `wild1`. Loading `wild1` will automatically load supporting packages.

```
> require(wild1)
```

Loading and summarizing data

This example features coyote survival data originally analyzed by Patterson (1999) and Heisey and Patterson (2006). The data are provided in `wild1`, courtesy of Brent Patterson, Ontario Ministry of Natural Resources.

The data must be loaded prior to use:

```
> data(coyote.mort)
```

Next we will summarize the structure of the data, noting the names, coding, and classes of variables, and print a sample of records. For our purposes (analysis with `csm`), times of entry and exit must be represented by numeric vectors, integer vectors, or vectors of class `"chron."` Events must be indicated by numeric, integer, or logical vectors. Fates may be represented by numeric vectors, character vectors, or factors. This flexibility of coding facilitates the representation of data by classes that are most appropriate for other planned uses.

```
> str(coyote.mort,strict.width="cut")
'data.frame':      97 obs. of  9 variables:
 $ id      : chr  "AF1" "AF1" "AF1" "AF1" ...
 $ gender  : Factor w/ 2 levels "female","male": 1 1 1 1 1 1..
 $ age     : int   2 2 2 2 2 2 2 2 1 2 ...
 $ area    : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 2 2 2..
 $ year    : Ord.factor w/ 5 levels "1992"<"1993"<...: 1 2 3 ..
 $ entry   : int   141 0 0 0 0 198 208 147 173 197 ...
 $ exit    : int   365 365 365 365 365 268 215 192 365 365 ...
 $ event   : int    0 0 0 0 0 1 1 1 0 0 ...
 $ fate    : Factor w/ 6 levels "car","shot","snare",...: NA ..
```

```
> head(coyote.mort)
   id gender age area year entry exit event fate
1  AF1 female  2   1 1992  141 365    0 <NA>
2  AF1 female  2   1 1993   0 365    0 <NA>
3  AF1 female  2   1 1994   0 365    0 <NA>
4  AF1 female  2   1 1995   0 365    0 <NA>
5  AF1 female  2   1 1996   0 365    0 <NA>
6 AF10 female  2   2 1995  198 268    1 snare
```

Survival intervals are open on the left and closed on the right, i.e., `(entry, exit]`.

Recurrent time scales

The sample data exemplify the common practice of pooling years via use of a time scale with a recurrent origin (i.e., dates range from 0 to 365 and individuals contribute 1 record per calendar year). If data are recorded as ranges of calendar dates (i.e., each individual contributes 1 record and dates may span >1 calendar year), they can be translated with `partition.intervals`.

We can use `table` to tally deaths for each fate:

```
> table(coyote.mort$fate)
      car      shot      snare      strife      trap
      5       12       20       2         2
unknown nat
      3
```

Sample sizes (numbers of records in the risk set) can be computed for each time step, t , with `n.risk` (first 6 times shown):

```
> head(with(coyote.mort,n.risk(entry,exit,1)))
  t  n
1 0 48
2 1 49
3 2 49
4 3 49
5 4 49
6 5 50
```

Estimating mortality: 1 cause

We will first estimate mortality due to "harvest," which encompasses several fates named in a vector of causes. This provision for aggregating fates is a nice feature of `csm`.

```
> harvest <- c("snare","shot","trap")
> cause1 <- with(coyote.mort,
  csm(entry=entry, exit=exit, event=event,
      fate=fate, cause=harvest, alpha=0.10))
```

Subscripts can be used to display particularly relevant results, e.g., probabilities of harvest (CIF) by dates shown (`time`), with associated confidence limits (`lcl.90` and `ucl.90`):

```
> vars <- c("time","CIF","lcl.90","ucl.90")
> na.omit(cause1[vars])
```

	time	CIF	lcl.90	ucl.90
3	177	0.012	-0.0075	0.031
4	178	0.024	-0.0035	0.051
5	185	0.035	0.0023	0.068
7	192	0.046	0.0091	0.084
8	195	0.058	0.0166	0.099
9	198	0.069	0.0244	0.114
10	212	0.080	0.0324	0.128
11	215	0.091	0.0407	0.142
12	221	0.113	0.0577	0.169
13	222	0.124	0.0665	0.182
14	244	0.135	0.0754	0.195
15	252	0.146	0.0844	0.208
18	268	0.157	0.0935	0.220
19	273	0.168	0.1026	0.233
20	276	0.178	0.1118	0.245
21	280	0.189	0.1211	0.257
22	290	0.211	0.1400	0.281
23	295	0.221	0.1495	0.293
25	325	0.232	0.1589	0.305
26	348	0.242	0.1682	0.316

Note that the estimates in Table 3 of Heisey and Patterson (2006) are 90%, rather than 95% confidence limits (columns are mislabeled), so our results agree.

Package `wild1` includes a plotting method (`plot.csm`) that produces survival curves from objects of class "csm." See the function documentation for available plotting options.

```
> plot(cause1)
```

Estimating mortality: multiple causes

If vectors of causes are stored in a list, `lapply` can be used to compute several estimates in 1 step and store results in a single list:

```
> other <- c("strife","car","unknown nat")
> cause.lst <- list(harvest=harvest, other=other)
> csm.lst <- lapply(cause.lst,function(cause.vec){
  with(coyote.mort,
    csm(entry=entry, exit=exit, event=event,
      fate=fate, cause=cause.vec, alpha=0.10))})
> str(csm.lst)
```

```
List of 2
```

```
$ harvest:Classes 'csm' and 'data.frame':      26 obs. of  13 variables:
```

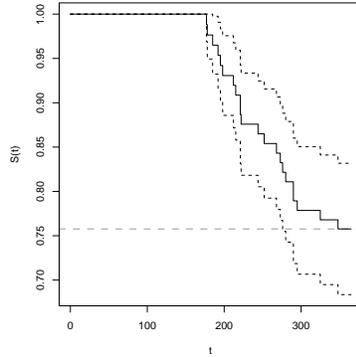


Figure 1: A sample plot constructed with `plot.csm`. Depicts harvest mortality of coyotes, with 90% confidence limits. Data from Heisey and Patterson (2006).

```

..$ time      : num [1:26] 75 123 177 178 185 186 192 195 198 212 ...
..$ n.event.all : num [1:26] 1 1 1 1 1 1 1 1 1 1 ...
..$ n.risk.all  : num [1:26] 56 57 82 81 82 81 80 79 79 80 ...
..$ survival.all: num [1:26] 0.982 0.965 0.953 0.941 0.93 ...
..$ n.event.s   : num [1:26] NA NA 1 1 1 NA 1 1 1 1 ...
..$ n.risk.s    : num [1:26] NA NA 82 81 82 NA 80 79 79 80 ...
..$ survival.s  : num [1:26] NA NA 0.988 0.976 0.964 ...
..$ mort.rate   : num [1:26] NA NA 0.0118 0.0118 0.0115 ...
..$ CIF         : num [1:26] NA NA 0.0118 0.0235 0.035 ...
..$ cumvar      : num [1:26] NA NA 0.000137 0.000271 0.000395 ...
..$ SE         : num [1:26] NA NA 0.0117 0.0164 0.0199 ...
..$ lcl.90     : num [1:26] NA NA -0.00748 -0.00352 0.00234 ...
..$ ucl.90     : num [1:26] NA NA 0.031 0.0506 0.0677 ...
$ other :Classes 'csm' and 'data.frame':      26 obs. of  13 variables:
..$ time      : num [1:26] 75 123 177 178 185 186 192 195 198 212 ...
..$ n.event.all : num [1:26] 1 1 1 1 1 1 1 1 1 1 ...
..$ n.risk.all  : num [1:26] 56 57 82 81 82 81 80 79 79 80 ...
..$ survival.all: num [1:26] 0.982 0.965 0.953 0.941 0.93 ...
..$ n.event.s   : num [1:26] 1 1 NA NA NA 1 NA NA NA NA ...
..$ n.risk.s    : num [1:26] 56 57 NA NA NA 81 NA NA NA NA ...
..$ survival.s  : num [1:26] 0.982 0.965 NA NA NA ...
..$ mort.rate   : num [1:26] 0.0179 0.0172 NA NA NA ...
..$ CIF         : num [1:26] 0.0179 0.0351 NA NA NA ...
..$ cumvar      : num [1:26] 0.000313 0.000594 NA NA NA ...
..$ SE         : num [1:26] 0.0177 0.0244 NA NA NA ...
..$ lcl.90     : num [1:26] -0.0113 -0.005 NA NA NA ...
..$ ucl.90     : num [1:26] 0.047 0.0752 NA NA NA ...

```

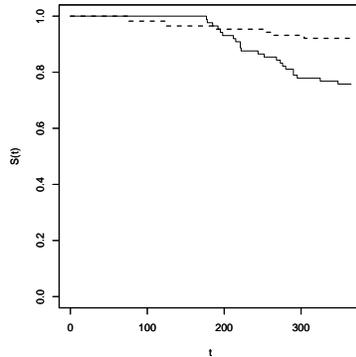


Figure 2: A sample plot constructed with `plot.csm`. Depicts harvest (solid line) and "other" mortality (dashed line) of coyotes. Data from Heisey and Patterson (2006).

The following code generates a compact display of the most relevant results. Again, note the agreement of results with Table 3 of Heisey and Patterson (2006):

```
> lapply(csm.lst,function(output){
  tail(na.omit(output[vars]),1)}
$harvest
  time CIF lcl.90 ucl.90
26 348 0.24  0.17  0.32

$other
  time  CIF lcl.90 ucl.90
24 304 0.079 0.027  0.13
```

`plot.csm` can also be used to plot several survival curves on a single figure. Arguments shown below 1) suppress confidence limits (`alpha=NULL`), 2) suppress horizontal reference lines (`href=FALSE`), 3) specify limits for the y axis (`ylim=c(0,1)`), and 4) cause the second curve to be represented by a dashed line (`lty=2`). Specifying `add=TRUE` prevents R from initiating a new plot or creating a new plotting window, causing the second curve to be plotted along with the first. In this fashion, which is simpler to implement than describe, any number of curves can be added to the same plot.

```
> plot(csm.lst[[1]],alpha=NULL,href=FALSE,ylim=c(0,1))
> plot(csm.lst[[2]],alpha=NULL,add=TRUE,href=FALSE,
  lty=2, lwd=2)
```

References

- Heisey, D. M., and B. R. Patterson. 2006. A review of methods to estimate cause-specific mortality in presence of competing risks. *Journal of Wildlife Management* 70(6):1544-1555.
- Patterson, B. R. 1999. The effects of prey distribution and abundance on eastern coyote life history and predation on white-tailed deer. Dissertation, University of Saskatchewan, Saskatoon, Canada.