DEM 7223 - Event History Analysis - Competing Risks in the Cox Model

Corey S. Sparks, PhD

03 November, 2022

Notes

The nature of competing risks

- So far, all of our models have dealt with single events.
- Meaning: each observation is only at risk experiencing one type of event
- Imagine if there were multiple types of events that a person could experience
- Furthermore, imagine if a person was at risk of experiencing each of these events at the same time
- This is the foundation of the *competing risk model*

Simple example - Death

Q: How many ways can someone die? A: Lots

- How could we keep track in a life table framework of the different probabilities of a person dying from cancer or heart disease or homicide?
- Ideally we would like the number at risk of experiencing each type of event at each time point, and the number experiencing each type of event
- This would give us hazard function estimates
- Competing risks models form a subset of another general form of models
- **Multi-state models**
- We have discussed these in passing several times e.g. multinomial logistic regression

- Multi-state models are a general form, because they not only allow multiple types of transitions, but these transitions can occur multiple times
- Think of the single>married>divorced>dead model

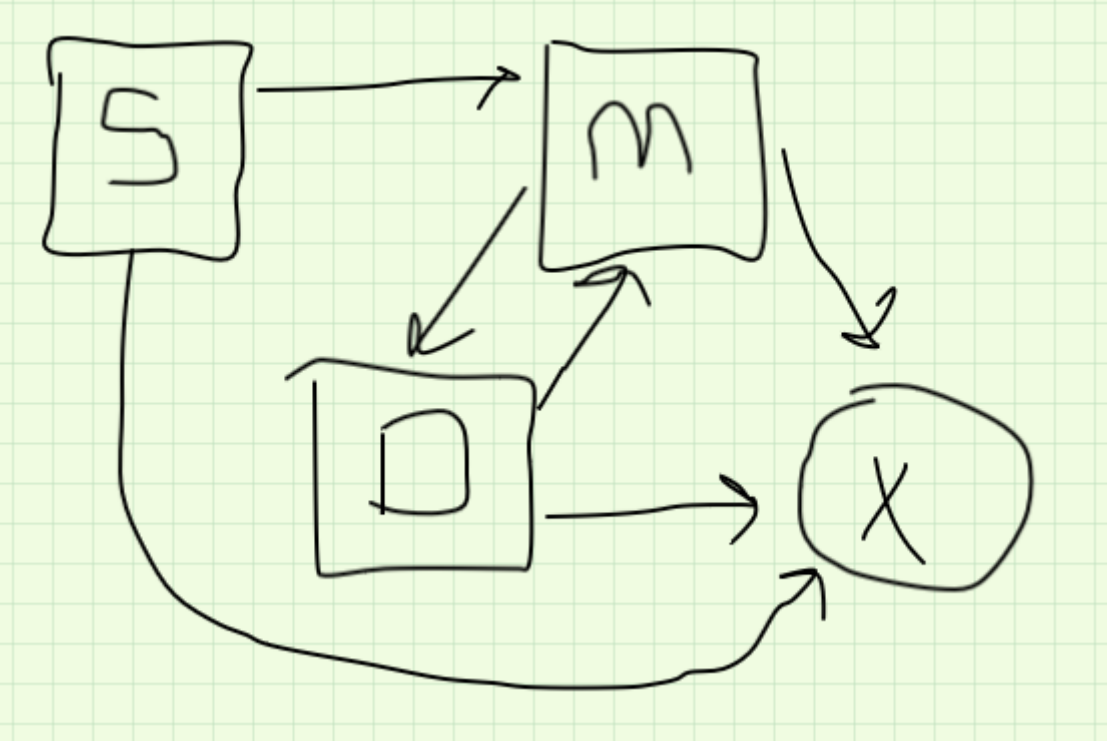


Figure 1: SMD Model

Type - specific hazards

- When we classify an event into types, first we need to have a discrete set of failure types - like causes of death
- Now we want to write a cause-specific hazard function for each event type j

$$h_{ij}(t) = Pr(t < T_i < t + \Delta t, J_i = j | T_i \geqslant t)$$

- which is the hazard of decrementing from cause j at time t.
- The only difference between this hazard and the one we have seen in the past, is the inclusion of the J's.

- This conditional probability is for an event of type j occurring at time T, given the person has not died before time t
- This basically separates the total hazard into the type-specific hazards

$$h_i(t) = \sum_j h_{ij}(t)$$

* This method assumes each of the competing risks are independent of one another

- This can be a weak assumption, if we know that causes are correlated

Regression models for competing risks

- The simplest way to treat competing risk, if we assume causes are independent is to fit the Cox model for each cause, and treat all other causes as censored
- If event = event type 1, then $failure_1$=1, else $failure_1$=0 Repeat this for each failure type
- Allison gives a test for testing the difference in a covariate across different models fit on the same data, in the competing risk setting

Test for $\beta_{1j} = \beta_{1k}$ in two models j and k

$$z = \frac{\beta_{1j} - \beta_{1k}}{\left[s.e.(\beta_{1j})\right]^2 + \left[s.e.(\beta_{1k})\right]^2}$$

- compare the $|z|$ test to a normal distribution for the p value.
- Allison also give a deviance test (Chow test) that asks the question: Is it work fitting separate models?

$$\chi^2 = -2LL_F - \sum_k -2LL_l$$

* Where LL_F is the log likelihood from the model with no competing risk and LL_k are the log likelihoods from the k competing risk models.

Examples

This example uses data from the National Health Interview Survey (NHIS) linked mortality data obtained from the Minnesota Population Center's IHIS program, which links the NHIS survey files from 1986 tp 2009 to mortality data from the National Death Index (NDI). The death follow up in *this* data file used in the current example ends at 2006.

Below, I code a competing risk outcome, using four different causes of death as competing events, and age at death as the outcome variable.

The data are pretty big, so I take a subset of 20,000 people for the example presented below. Using the whole sample may make your computer explode. **You have been warned**

```
library(survey)
```

```
Loading required package: grid
```

```
Loading required package: Matrix
```

```
Loading required package: survival
```

```

Attaching package: 'survey'
```

```
The following object is masked from 'package:graphics':

    dotchart
```

```
library(survival)
library(car)
```

```
Loading required package: carData
```

```
library(cmprsk)
library(haven)
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

```
v ggplot2 3.3.6      v purrr   0.3.5
v tibble  3.1.8      v dplyr   1.0.10
v tidyr   1.2.1      v stringr 1.4.1
v readr   2.1.3      v forcats 0.5.2
-- Conflicts ---------------------------------------- tidyverse_conflicts() --
x tidyr::expand() masks Matrix::expand()
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
x tidyr::pack()   masks Matrix::pack()
x dplyr::recode() masks car::recode()
x purrr::some()   masks car::some()
x tidyr::unpack() masks Matrix::unpack()
```

```r
dat<-haven::read_dta("C:/Users/ozd504/OneDrive - University of Texas at San Antonio/classe

names(dat)<-tolower(names(dat))

dat <- haven::zap_labels(dat)
sub<-subset(dat, dat$mortelig==1&is.na(dat$racea)==F)
samps<-sample(1:length(sub$year), size = 100000, replace = F)
sub<-sub[samps,]

#rm(ihis_mort)
sub$d.age<-ifelse(sub$mortstat==1,sub$mortdody-(sub$year-sub$age) ,
                  ifelse(sub$mortstat==2,2006-(sub$year-sub$age), NA))
sub$d.event<-ifelse(sub$mortstat==1,1,0)
sub$timetodeath<-ifelse(sub$mortstat ==1, sub$mortdody-sub$year  , 2006 - sub$year )
sub$d5yr<-ifelse(sub$timetodeath<=5&sub$mortstat==1, 1,0)
sub$married<-Recode(sub$marstat, recodes="00=NA; 10:13='married'; 20:40='sep'; 50='nm'; 99
sub$male<-ifelse(sub$sex==1,1,0)
sub$mwt<-sub$mortwt/mean(sub$mortwt, na.rm=T)

sub$age5<-cut(sub$age,seq(15,85, 5))

sub$race<-Recode(sub$racea, recodes ="100='wht'; 200 ='blk'; 300:617='other'; 900:990=NA",
sub$college<-Recode(sub$educrec2, recodes="00=NA; 10:42='hs or less'; 50:53='some coll'; 5
sub$black<-ifelse(sub$race=='blk',1,0)
sub$oth<-ifelse(sub$race=='other',1,0)
sub$hs<-ifelse(sub$college=='hs or less',1,0)
sub$col1<-ifelse(sub$college=='some coll',1,0)
sub$sep<-ifelse(sub$married=='sep',1,0)
```

```
sub$nm<-ifelse(sub$married=='nm',1,0)

sub$hisp<-Recode(sub$hispeth, recodes="10=0; 20:70=1; else=NA")

sub$race_eth[sub$hisp == 0 & sub$race=="wht"]<-"NHWhite"
```

```
Warning: Unknown or uninitialised column: `race_eth`.
```

```
sub$race_eth[sub$hisp == 0 & sub$race=="blk"]<-"NHBlack"
sub$race_eth[sub$hisp == 0 & sub$race=="other"]<-"NHother"
sub$race_eth[sub$hisp == 1 ]<-"Hispanic"
sub$race_eth[is.na(sub$hisp) ==T | is.na(sub$race)==T]<-NA
```

Now we want to examine the competing risks of mortality from various causes, we use the mortucod variable create a variable indicating major causes of death lumping other causes together(1=cancers, 2=CVD, 3=infectious, 4=other causes, NA=alive)

```
sub$cod<-Recode(sub$mortucodld, recodes="96=NA")

#Here I generate censoring indicators, one for each type of failure
sub$fail1<-ifelse(sub$cod==1 & sub$d.event==1, 1,0) #heart disease
sub$fail2<-ifelse(sub$cod==2 & sub$d.event==1, 1,0) #cancer
sub$fail3<-ifelse(sub$cod==4 & sub$d.event==1, 1,0) #accident
sub$fail4<-ifelse(sub$cod %in% c(3,5,6,7,8,9,10) & sub$d.event==1, 1,0) #other
#sub$codcens=ifelse(is.na(sub$cod)==T,0,sub$cod)
sub$codcens[sub$fail1==0 & sub$fail2==0 & sub$fail3==0 & sub$fail4==0]<-0
```

```
Warning: Unknown or uninitialised column: `codcens`.
```

```
sub$codcens[sub$fail1==1 | sub$fail2==1 | sub$fail3==1 | sub$fail4==1]<-1
```

```
table(sub$codcens, sub$d.event)
```

```
      0     1
0 81003     0
1     0 18914
```

```r
table(sub$cod, sub$d.event)
```

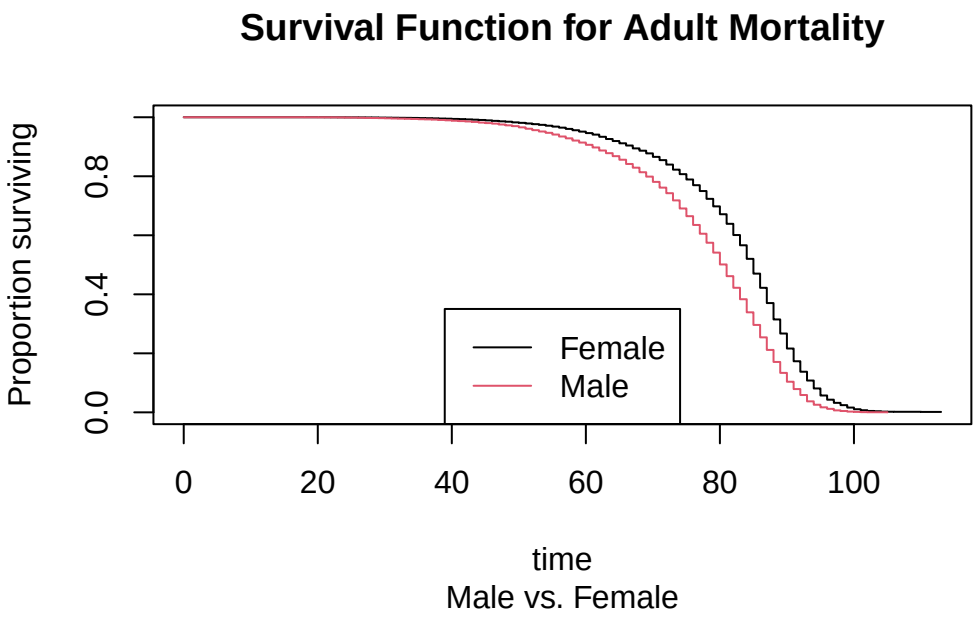
```
      0    1
1     0 3678
2     0 4690
3     0  930
4     0  786
5     0 1063
6     0  492
7     0  599
8     0  420
9     0  355
10    0 5901
```

##Age at death Form a survey design object and examine some basic mortality curves by sex and failure type:

```r
options(survey.lonely.psu="adjust")

des<-svydesign(ids=~psu,
               strata=~strata,
               weights = ~mortwt,
               data=sub[sub$mortwt>0,],
               nest=T)


fit.s<-svykm(Surv(d.age, d.event)~male, design=des, se=F)
fit.s
```

```
Weighted survival curves:
svykm(formula = Surv(d.age, d.event) ~ male, design = des, se = F)
0 : Q1 = 77  median = 85  Q3 = 90
1 : Q1 = 72  median = 81  Q3 = 87
```

```r
plot(fit.s, pars=list(col=c(1,2)) )
title(main="Survival Function for Adult Mortality", sub="Male vs. Female")
legend("bottom", legend = c("Female","Male" ), col=c(1,2), lty=1)
```




```
#test statistic
svylogrank(Surv(d.age, d.event)~male,
           design=des)
```

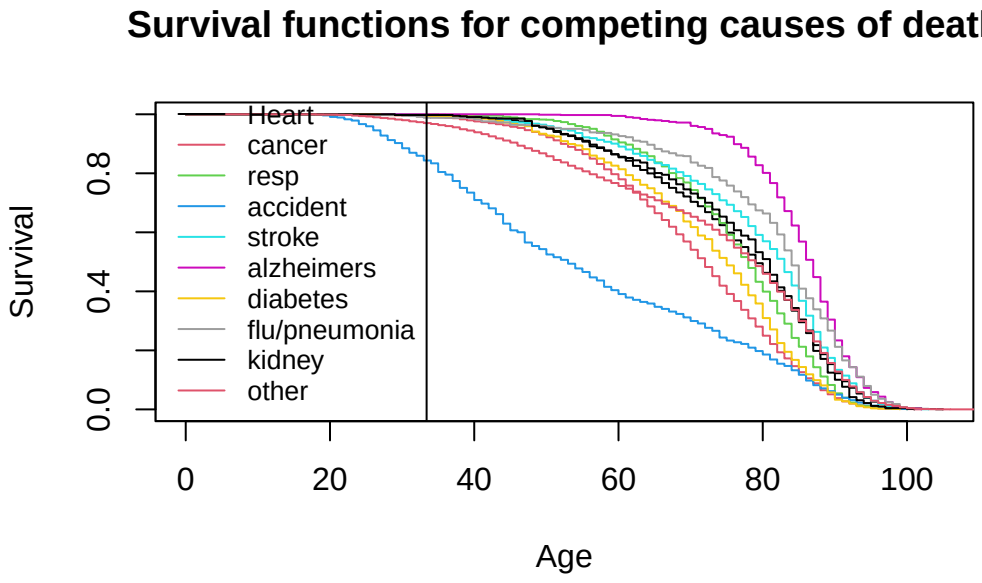
```
[[1]]
      score
[1,] 5867908 208697.8 28.11677 6.110221e-174

[[2]]
       Chisq             p
 7.905528e+02 6.110221e-174

attr(,"class")
[1] "svylogrank"
```

```
fit.s2<-svykm(Surv(d.age, d.event)~strata(cod),
              design=des,
              se=F)
fit.s2
```

```
Weighted survival curves:
```

```
svykm(formula = Surv(d.age, d.event) ~ strata(cod), design = des,
    se = F)
cod=1 : Q1 = 68  median = 79  Q3 = 87
cod=2 : Q1 = 62  median = 72  Q3 = 80
cod=3 : Q1 = 70  median = 78  Q3 = 84
cod=4 : Q1 = 39  median = 53  Q3 = 74
cod=5 : Q1 = 72  median = 83  Q3 = 88
cod=6 : Q1 = 82  median = 87  Q3 = 90
cod=7 : Q1 = 64  median = 75  Q3 = 82
cod=8 : Q1 = 76  median = 84  Q3 = 90
cod=9 : Q1 = 69  median = 81  Q3 = 87
cod=10 : Q1 = 61  median = 79  Q3 = 87
```

```r
plot(fit.s2, pars=list(col=1:10),
     ylab="Survival", xlab="Age",
     main="Survival functions for competing causes of death")
legend("bottomleft",
       legend=c("Heart", "cancer", "resp", "accident", "stroke", "alzheimers", "diabetes",
       lty=1,
       col=1:10,
       cex=.8)
```



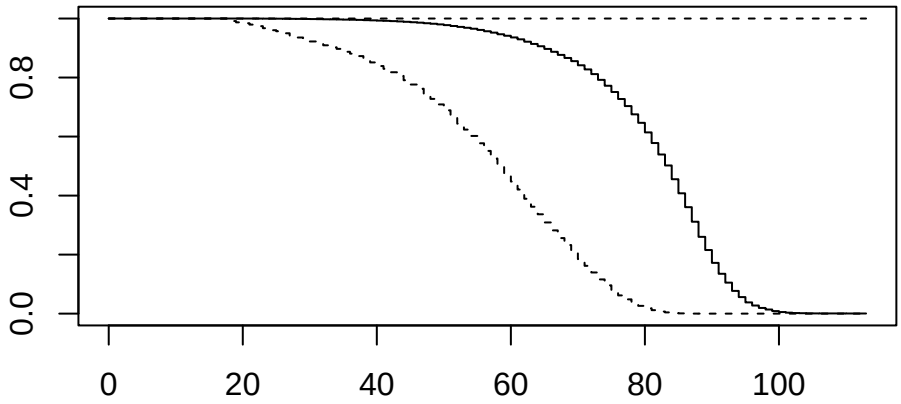
Here is the overall hazard model using the Cox PH model, this model is for all-cause mortality.

```
#all failures
fita<-svycoxph(Surv(d.age,d.event)~male+married+race+college,
               design=des)
summary(fita)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (1918) clusters.
svydesign(ids = ~psu, strata = ~strata, weights = ~mortwt, data = sub[sub$mortwt >
    0, ], nest = T)

Call:
svycoxph(formula = Surv(d.age, d.event) ~ male + married + race +
    college, design = des)

  n= 97731, number of events= 18566
   (2009 observations deleted due to missingness)

                    coef exp(coef) se(coef) robust se       z Pr(>|z|)
male                0.43257   1.54122  0.01665   0.01794  24.115  < 2e-16 ***
marriednm           0.62656   1.87116  0.02854   0.04173  15.015  < 2e-16 ***
marriedsep         -0.18787   0.82873  0.01828   0.01937  -9.697  < 2e-16 ***
raceother          -0.22943   0.79499  0.04703   0.05966  -3.846  0.00012 ***
racewht            -0.33593   0.71467  0.02519   0.02895 -11.603  < 2e-16 ***
collegehs or less   0.22207   1.24866  0.02308   0.02448   9.072  < 2e-16 ***
collegesome coll    0.25738   1.29354  0.02761   0.02866   8.980  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                  exp(coef) exp(-coef) lower .95 upper .95
male                 1.5412     0.6488    1.4880    1.5964
marriednm            1.8712     0.5344    1.7242    2.0306
marriedsep           0.8287     1.2067    0.7978    0.8608
raceother            0.7950     1.2579    0.7073    0.8936
racewht              0.7147     1.3992    0.6752    0.7564
collegehs or less    1.2487     0.8009    1.1902    1.3100
collegesome coll     1.2935     0.7731    1.2229    1.3683

Concordance= 0.641  (se = 0.004 )
Likelihood ratio test= NA  on 7 df,   p=NA
```

```
Wald test            = 1502  on 7 df,   p=<2e-16
Score (logrank) test = NA  on 7 df,   p=NA

  (Note: the likelihood ratio and score tests assume independence of
     observations within a cluster, the Wald and robust score tests do not).
```

```
plot(survfit(fita))
```



Type-specific hazard models

These models take the approach suggested by Allison, where for a given cause of death, any other cause is assumed to be censored.

```
#Cancer
fit1<-svycoxph(Surv(d.age,fail1==1)~male+married+race+college,
               des)
summary(fit1)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (1918) clusters.
```

```
svydesign(ids = ~psu, strata = ~strata, weights = ~mortwt, data = sub[sub$mortwt >
    0, ], nest = T)

Call:
svycoxph(formula = Surv(d.age, fail1 == 1) ~ male + married +
    race + college, design = des)

  n= 97653, number of events= 3591
   (2087 observations deleted due to missingness)

                     coef exp(coef) se(coef) robust se      z Pr(>|z|)
male                0.66295   1.94051  0.03889   0.04187 15.834  < 2e-16 ***
marriednm           0.52326   1.68752  0.07268   0.08218  6.367 1.92e-10 ***
marriedsep         -0.03795   0.96276  0.04145   0.04357 -0.871  0.38376
raceother          -0.30954   0.73379  0.11343   0.11842 -2.614  0.00895 **
racewht            -0.34780   0.70624  0.05814   0.05556 -6.260 3.86e-10 ***
collegehs or less   0.22301   1.24984  0.05334   0.05383  4.143 3.43e-05 ***
collegesome coll    0.16979   1.18505  0.06536   0.06623  2.564  0.01036 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                   exp(coef) exp(-coef) lower .95 upper .95
male                  1.9405     0.5153    1.7876    2.1065
marriednm             1.6875     0.5926    1.4365    1.9824
marriedsep            0.9628     1.0387    0.8839    1.0486
raceother             0.7338     1.3628    0.5818    0.9255
racewht               0.7062     1.4159    0.6334    0.7875
collegehs or less     1.2498     0.8001    1.1247    1.3889
collegesome coll      1.1851     0.8438    1.0408    1.3493

Concordance= 0.651  (se = 0.007 )
Likelihood ratio test= NA  on 7 df,   p=NA
Wald test            = 402.8  on 7 df,   p=<2e-16
Score (logrank) test = NA  on 7 df,   p=NA

  (Note: the likelihood ratio and score tests assume independence of
     observations within a cluster, the Wald and robust score tests do not).
```

```
#CVD
fit2<-svycoxph(Surv(d.age, fail2==1)~male+married+race+college,
               des)
summary(fit2)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (1918) clusters.
svydesign(ids = ~psu, strata = ~strata, weights = ~mortwt, data = sub[sub$mortwt >
    0, ], nest = T)


Call:
svycoxph(formula = Surv(d.age, fail2 == 1) ~ male + married +
    race + college, design = des)

  n= 97653, number of events= 4591
   (2087 observations deleted due to missingness)

                     coef exp(coef) se(coef) robust se      z Pr(>|z|)
male               0.37575   1.45608  0.03308   0.03525 10.660  < 2e-16 ***
marriednm          0.22259   1.24930  0.06717   0.07128  3.123 0.001793 **
marriedsep        -0.31668   0.72856  0.03714   0.04111 -7.703 1.33e-14 ***
raceother         -0.18701   0.82943  0.09158   0.08975 -2.084 0.037194 *
racewht           -0.38575   0.67994  0.05015   0.04838 -7.973 1.55e-15 ***
collegehs or less  0.13289   1.14213  0.04447   0.04767  2.788 0.005311 **
collegesome coll   0.20114   1.22280  0.05345   0.05782  3.479 0.000503 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                  exp(coef) exp(-coef) lower .95 upper .95
male                 1.4561     0.6868    1.3589    1.5602
marriednm            1.2493     0.8004    1.0864    1.4366
marriedsep           0.7286     1.3726    0.6722    0.7897
raceother            0.8294     1.2056    0.6956    0.9890
racewht              0.6799     1.4707    0.6184    0.7476
collegehs or less    1.1421     0.8756    1.0402    1.2540
collegesome coll     1.2228     0.8178    1.0918    1.3695

Concordance= 0.586  (se = 0.006 )
Likelihood ratio test= NA  on 7 df,   p=NA
Wald test            = 350.3  on 7 df,   p=<2e-16
Score (logrank) test = NA  on 7 df,   p=NA

  (Note: the likelihood ratio and score tests assume independence of
     observations within a cluster, the Wald and robust score tests do not).
```

```r
#Infectious
fit3<-svycoxph(Surv(d.age, fail3==1)~male+married+race+college,
               des)
summary(fit3)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (1918) clusters.
svydesign(ids = ~psu, strata = ~strata, weights = ~mortwt, data = sub[sub$mortwt >
    0, ], nest = T)


Call:
svycoxph(formula = Surv(d.age, fail3 == 1) ~ male + married +
    race + college, design = des)

  n= 97653, number of events= 767
   (2087 observations deleted due to missingness)

                     coef exp(coef) se(coef) robust se      z Pr(>|z|)
male               0.80823   2.24393  0.08147   0.08374  9.651  < 2e-16 ***
marriednm          1.40993   4.09566  0.09636   0.09768 14.434  < 2e-16 ***
marriedsep        -0.06833   0.93395  0.09989   0.11170 -0.612    0.541
raceother          0.26961   1.30946  0.20822   0.20109  1.341    0.180
racewht            0.12125   1.12891  0.12915   0.13397  0.905    0.365
collegehs or less  0.63601   1.88892  0.11545   0.13019  4.885 1.03e-06 ***
collegesome coll   0.58754   1.79956  0.13198   0.15054  3.903 9.50e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                  exp(coef) exp(-coef) lower .95 upper .95
male                  2.244     0.4456    1.9043     2.644
marriednm             4.096     0.2442    3.3821     4.960
marriedsep            0.934     1.0707    0.7503     1.163
raceother             1.309     0.7637    0.8829     1.942
racewht               1.129     0.8858    0.8682     1.468
collegehs or less     1.889     0.5294    1.4635     2.438
collegesome coll      1.800     0.5557    1.3398     2.417

Concordance= 0.729  (se = 0.013 )
Likelihood ratio test= NA  on 7 df,   p=NA
Wald test            = 322.2  on 7 df,   p=<2e-16
Score (logrank) test = NA  on 7 df,   p=NA
```

```
  (Note: the likelihood ratio and score tests assume independence of
     observations within a cluster, the Wald and robust score tests do not).
```

```
#Other
fit4<-svycoxph(Surv(d.age, fail4==1)~male*married+race+college,
               des)
summary(fit4)
```

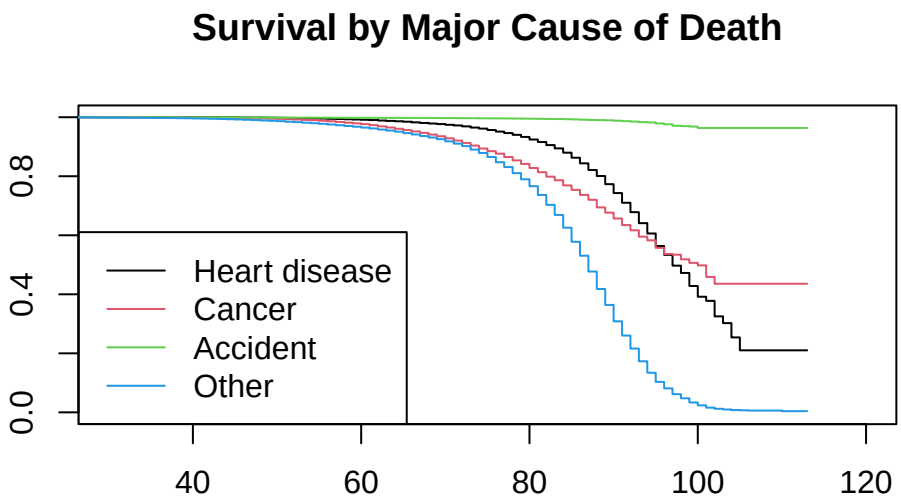
```
Stratified 1 - level Cluster Sampling design (with replacement)
With (1918) clusters.
svydesign(ids = ~psu, strata = ~strata, weights = ~mortwt, data = sub[sub$mortwt >
    0, ], nest = T)

Call:
svycoxph(formula = Surv(d.age, fail4 == 1) ~ male * married +
    race + college, design = des)

  n= 97731, number of events= 9539
   (2009 observations deleted due to missingness)

                     coef exp(coef) se(coef) robust se       z Pr(>|z|)
male               0.14978   1.16158  0.02953   0.02858   5.240 1.60e-07 ***
marriednm          0.29258   1.33988  0.05935   0.07319   3.998 6.40e-05 ***
marriedsep        -0.33673   0.71410  0.03133   0.03042 -11.068  < 2e-16 ***
raceother         -0.29112   0.74742  0.06680   0.07674  -3.794 0.000148 ***
racewht           -0.34978   0.70484  0.03502   0.03951  -8.852  < 2e-16 ***
collegehs or less  0.20504   1.22757  0.03265   0.03594   5.705 1.16e-08 ***
collegesome coll   0.25909   1.29575  0.03885   0.04130   6.273 3.54e-10 ***
male:marriednm     0.77964   2.18068  0.07723   0.10609   7.349 1.99e-13 ***
male:marriedsep    0.31872   1.37537  0.05049   0.05238   6.085 1.16e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                  exp(coef) exp(-coef) lower .95 upper .95
male                 1.1616     0.8609    1.0983    1.2285
marriednm            1.3399     0.7463    1.1608    1.5466
marriedsep           0.7141     1.4004    0.6728    0.7580
raceother            0.7474     1.3379    0.6431    0.8687
racewht              0.7048     1.4188    0.6523    0.7616
collegehs or less    1.2276     0.8146    1.1441    1.3172
```

```
collegesome coll     1.2958     0.7718    1.1950    1.4050
male:marriednm       2.1807     0.4586    1.7713    2.6847
male:marriedsep      1.3754     0.7271    1.2412    1.5241

Concordance= 0.657  (se = 0.005 )
Likelihood ratio test= NA  on 9 df,   p=NA
Wald test            = 802  on 9 df,   p=<2e-16
Score (logrank) test = NA  on 9 df,   p=NA

  (Note: the likelihood ratio and score tests assume independence of
     observations within a cluster, the Wald and robust score tests do not).
```

```
#Plot all
plot(survfit(fit1), conf.int =F,
     col=1,xlim=c(30, 120) ,
     main="Survival by Major Cause of Death")

lines(survfit(fit2),
      col=2,conf.int =F)
lines(survfit(fit3),
      col=3, conf.int =F)
lines(survfit(fit4),
      col=4, conf.int =F)
```

```
Warning in survfit.coxph(fit4): the model contains interactions; the default
curve based on columm means of the X matrix is almost certainly not useful.
Consider adding a newdata argument.
```

```
legend("bottomleft",
       legend=c("Heart disease", "Cancer", "Accident", "Other"),
       col = 1:4,
       lty=rep(1, 4))
```



Construct a test of whether the betas are the same for each failure type using a Chow Test (See Allison p 217 for this). Basically we compare the deviance of the model with all causes of death to the sum of the deviances from each of the competing risk situations. If the test is significant, the it suggests that each cause of death has a different combination of the beta's in the model. I.e. the regression effects are not the same across causes of death.

```
#deviance from total model
d1<--2*fita$ll[2]

#sum of deviances from cause-specific models
otherds<- (-2*fit1$ll[2]+ -2*fit2$ll[2]+ -2*fit3$ll[2]+ -2*fit4$ll[2])

#Chow test
test<- d1-otherds
df<-(length(coef(fit1))*3)-length(coef(fita))
#print the test results
print(list(test=test, df=df,pval= pchisq(test, df=df, lower=F)))
```

```
$test
[1] 747.8513

$df
```

```
[1] 14

$pval
[1] 1.557881e-150
```

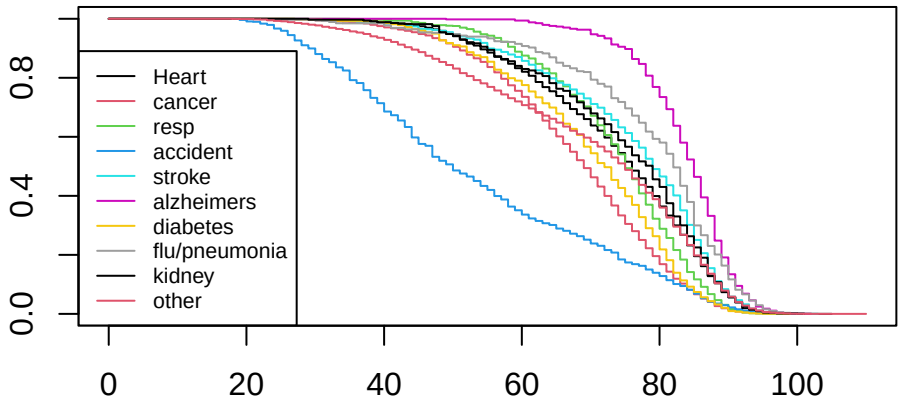
Alternatively, we could simply stratify the baseline hazard by type of failure

```
fits<-svycoxph(Surv(d.age, d.event)~male+married+race+college+strata(cod), des)
summary(fits)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (1918) clusters.
svydesign(ids = ~psu, strata = ~strata, weights = ~mortwt, data = sub[sub$mortwt >
    0, ], nest = T)

Call:
svycoxph(formula = Surv(d.age, d.event) ~ male + married + race +
    college + strata(cod), design = des)

  n= 18488, number of events= 18488
   (81252 observations deleted due to missingness)

                     coef exp(coef) se(coef) robust se       z Pr(>|z|)
male                0.23982   1.27102  0.01696   0.01776  13.501  < 2e-16 ***
marriednm           0.57320   1.77394  0.02874   0.04376  13.098  < 2e-16 ***
marriedsep         -0.32640   0.72151  0.01834   0.01872 -17.437  < 2e-16 ***
raceother           0.01640   1.01653  0.04747   0.07029   0.233    0.816
racewht            -0.37140   0.68977  0.02535   0.02844 -13.061  < 2e-16 ***
collegehs or less  -0.01596   0.98417  0.02302   0.02486  -0.642    0.521
collegesome coll    0.14502   1.15606  0.02767   0.03021   4.800 1.59e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                   exp(coef) exp(-coef) lower .95 upper .95
male                  1.2710     0.7868    1.2275    1.3160
marriednm             1.7739     0.5637    1.6281    1.9328
marriedsep            0.7215     1.3860    0.6955    0.7485
raceother             1.0165     0.9837    0.8857    1.1667
racewht               0.6898     1.4498    0.6524    0.7293
collegehs or less     0.9842     1.0161    0.9374    1.0333
collegesome coll      1.1561     0.8650    1.0896    1.2266
```

```
Concordance= 0.615  (se = 0.003 )
Likelihood ratio test= NA  on 7 df,   p=NA
Wald test            = 1343  on 7 df,   p=<2e-16
Score (logrank) test = NA  on 7 df,   p=NA

  (Note: the likelihood ratio and score tests assume independence of
     observations within a cluster, the Wald and robust score tests do not).
```

```
plot(survfit(fits),
     col=c(1:10))

legend("bottomleft",
       legend=c("Heart", "cancer", "resp", "accident", "stroke", "alzheimers", "diabetes",
       lty=1, col=1:10, cex=.65)
```




```
d2<- -2*fits$ll[2]
df<-(length(coef(fits)))-length(coef(fita))
d1-d2
```

```
[1] 77084.8
```

```
#AIC(fits, fita)
```

Competing Risk Regression

The `crr()` function in the `cmprsk` library uses the methods discussed in Fine and Gray, 1999 for regression modeling for the subdistribution function for a competing risk. This is still a proportional hazards model for the key event of interest, but takes into account failures from other causes.

```
sub$cod2<-ifelse(is.na(sub$cod)==T,0,sub$cod)
#Make a matrix of predictors
covs<-data.frame(sub$male,sub$nm, sub$sep, sub$black, sub$oth, sub$hs, sub$col1)

names(covs)<-c("male", "neverm", "separated", "black", "other", "hsorless", "somecoll")
head(covs)
```

```
  male neverm separated black other hsorless somecoll
1    1      0         0     0     0        0        0
2    1      0         1     0     1        1        0
3    1      0         1     0     0        0        1
4    0      0         0     0     0        0        1
5    0      0         1     0     0        1        0
6    0      0         0     0     0        1        0
```

```
#Fit the cumulative incidence model of Fine and Gray for cancer mortality
fit.crr<-crr(ftime=sub$d.age,
             fstatus=sub$cod2,
             cov1=covs,
             failcode=1,
             cencode=0 )
```

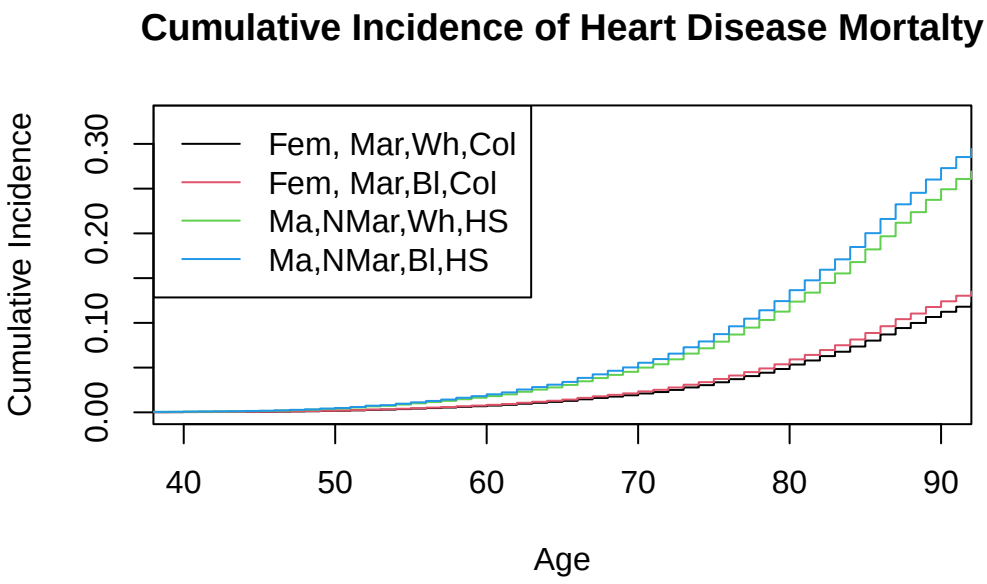
```
2010 cases omitted due to missing values
```

```
summary(fit.crr)
```

```
Competing Risks Regression

Call:
```

```
crr(ftime = sub$d.age, fstatus = sub$cod2, cov1 = covs, failcode = 1,
    cencode = 0)

             coef exp(coef) se(coef)      z p-value
male       0.4083     1.504   0.0348 11.722 0.0e+00
neverm     0.2953     1.343   0.0677  4.360 1.3e-05
separated  0.1828     1.201   0.0371  4.925 8.4e-07
black      0.1057     1.111   0.0498  2.124 3.4e-02
other     -0.0685     0.934   0.0886 -0.773 4.4e-01
hsorless   0.1742     1.190   0.0505  3.452 5.6e-04
somecoll   0.0557     1.057   0.0624  0.894 3.7e-01

           exp(coef) exp(-coef)  2.5% 97.5%
male           1.504      0.665 1.405  1.61
neverm         1.343      0.744 1.176  1.53
separated      1.201      0.833 1.116  1.29
black          1.111      0.900 1.008  1.23
other          0.934      1.071 0.785  1.11
hsorless       1.190      0.840 1.078  1.31
somecoll       1.057      0.946 0.936  1.19

Num. cases = 97990 (2010 cases omitted due to missing values)
Pseudo Log-likelihood = -35020
Pseudo likelihood ratio test = 170  on 7 df,
```

```r
#Plot some interesting cases
z.p<-predict(fit.crr, rbind(c(0,0,0,0,0,0,0),
                            c(0,0,0,1,0,0,0),
                            c(1,1,0,0,0,1,0),
                            c(1,1,0,1,0,1,0)))

plot(z.p, col=1:4, lty=1, xlim=c(40,90),
     ylab="Cumulative Incidence", xlab="Age")

legend("topleft",
       legend=(c("Fem, Mar,Wh,Col",
                 "Fem, Mar,Bl,Col",
                 "Ma,NMar,Wh,HS",
                 "Ma,NMar,Bl,HS")),
       col=1:4, lty=1)

title(main="Cumulative Incidence of Heart Disease Mortalty")
```



competing risks using the multinomial model

```
pp<-survSplit(Surv(d.age, d.event)~., data=sub,
              cut =seq(20, 100, 5))

library(nnet)

pp$cause<-ifelse(pp$fail1==1, "hd",
                 ifelse(pp$fail2==1, "can",
                    ifelse(pp$fail3==1, "acc",
                        ifelse(pp$fail4==1, "other","0alive"))))

fitm<-multinom(cause ~ factor(tstart)+ male+married+race+college,
               weights = pp$mortwt/mean(pp$mortwt, na.rm=T),
               data=pp)
```

```
# weights:  130 (100 variable)
initial  value 1240358.410024
iter  10 value 645247.942814
iter  20 value 636636.432905
iter  30 value 635293.447213
```

```
iter  40 value 633493.646606
iter  50 value 622904.998915
iter  60 value 619755.543885
iter  70 value 618921.315325
iter  80 value 616809.093615
iter  90 value 615180.082447
iter 100 value 614214.173191
final  value 614214.173191
stopped after 100 iterations
```

```
newd<-expand.grid(tstart = seq(20, 100, 5),
                  male =c(0,1),
                  married="married",
                  race=levels(as.factor(pp$race)),
                  college=levels(as.factor(pp$college) ))

est<-predict(fitm, newd, type = "probs")

est<-data.frame(est)

names(est)<-c("pralive","prhd", "prcan", "pracc" , "prother" )

newd<-cbind(newd, est)
head(newd)
```

```
  tstart male married race college   pralive         prhd      prcan      pracc
1     20    0 married  blk    coll 0.9313803 0.0007595152 0.02099846 0.01044579
2     25    0 married  blk    coll 0.9309252 0.0007941170 0.02034403 0.01045095
3     30    0 married  blk    coll 0.9241439 0.0007228005 0.02274431 0.01098910
4     35    0 married  blk    coll 0.9162761 0.0008183342 0.02583923 0.01261936
5     40    0 married  blk    coll 0.9070223 0.0008014473 0.02782800 0.01461415
6     45    0 married  blk    coll 0.8932351 0.0010468788 0.03269026 0.01634720
     prother
1 0.03641594
2 0.03748566
3 0.04139991
4 0.04444700
5 0.04973408
6 0.05668051
```

```
library(data.table)
```

```
Attaching package: 'data.table'

The following objects are masked from 'package:dplyr':

    between, first, last

The following object is masked from 'package:purrr':

    transpose
```

```
library(magrittr)
```

```
Attaching package: 'magrittr'

The following object is masked from 'package:purrr':

    set_names

The following object is masked from 'package:tidyr':

    extract
```

```
out<-melt(setDT(newd), id = c("tstart", "male", "married","race", "college"),
          measure.vars = list(haz=c("pralive", "prhd","prcan","pracc", "prother")))
head(out, n=20)
```

```
    tstart male married race college variable      value
 1:     20    0 married  blk    coll  pralive 0.93138030
 2:     25    0 married  blk    coll  pralive 0.93092525
 3:     30    0 married  blk    coll  pralive 0.92414388
 4:     35    0 married  blk    coll  pralive 0.91627607
 5:     40    0 married  blk    coll  pralive 0.90702233
 6:     45    0 married  blk    coll  pralive 0.89323515
 7:     50    0 married  blk    coll  pralive 0.86559849
 8:     55    0 married  blk    coll  pralive 0.83600209
 9:     60    0 married  blk    coll  pralive 0.78834903
10:     65    0 married  blk    coll  pralive 0.72767335
```

```
11:     70    0 married  blk    coll  pralive 0.67233783
12:     75    0 married  blk    coll  pralive 0.55485442
13:     80    0 married  blk    coll  pralive 0.41147206
14:     85    0 married  blk    coll  pralive 0.22984703
15:     90    0 married  blk    coll  pralive 0.07893715
16:     95    0 married  blk    coll  pralive 0.58848776
17:    100    0 married  blk    coll  pralive 0.07215755
18:     20    1 married  blk    coll  pralive 0.90393812
19:     25    1 married  blk    coll  pralive 0.90359267
20:     30    1 married  blk    coll  pralive 0.89463739
```

```r
library(ggplot2)

out%>%
  dplyr::filter(race =="wht"&college=="coll", tstart <=90)%>%
  ggplot(aes(x=tstart, y=value,group=factor(male), color=factor(male) ))+
  geom_line()+
  facet_wrap(~variable)
```