L3: Cox models

2022-01-27

Readings

- Bradburn MJ, Clark TG, Love SB, Altman DG. Survival analysis part II: multivariate data analysis–an introduction to concepts and methods. British journal of cancer. 2003 Aug;89(3):431-6

Introduction

So far, we have focused on estimating risk functions in a single sample. But sometimes we want to compare survival endpoints between groups. In the weeks to come, we will discuss how to compare risk between groups, including how to account for confounding when comparing risk functions. Today, we will discuss the Cox model, a semiparametric regression model that allows us to compare hazard functions.

Comparing hazards using Cox models

Recall, the hazard function

The hazard is the "instantaneous" rate of having the event at a specific time t, among participants who have not had the event prior to that time. In notation, the hazard is typically expressed as the limit of the probability of having the event in an interval of width Δt divided by Δt, as Δt goes to 0.

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t \mid t \leq T)}{\Delta t}$$

We can apply a nonparametric estimator of the hazard at each event time R_k in our data as follows:

$$\hat{h}(k) = y(k)/[n(k)(R_k - R_{k-1})]$$

where $y(k) = \sum_{i=1}^{n} \delta_i I(T_i = R_k)$, $n(k)$ is the size of the risk set at time R_k, and $R_k - R_{k-1}$ is the time between event k and the previous event, where R_0 is set to 0.

Recall, components of survival analysis

1. Origin/timescale: t measures time since the origin
2. Target population
3. Event definition and timing
4. Group membership or actions

Hazard functions, illustrated

First, let's read in some data from the paper below and take a look.

> Cole SR, Hudgens MG. Survival analysis in infectious disease research: describing events in time: Aids. 2010;24(16):2423–2431.

```
library(tidyverse)
dat <- read_fwf("../data/data78.dat",  show_col_types = F)
colnames(dat) <- c("id", "aidsy", "w", "t", "d", "nw", "age", "vl")
head(dat)
```

```
## # A tibble: 6 x 8
##      id aidsy     w     t     d    nw   age     vl
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1     1 1990.  4.58  7.58     0     0  36.2   7932
## 2     2 1991.  3.75  6.75     0     1  36.5  36706
## 3     3 1992.  2.99  5.99     0     0  34.0  56418
## 4     4 1992.  2.97  5.97     0     1  35.1 123078
## 5     5 1992.  2.93  5.93     0     0  31.8   3775
## 6     6 1992.  2.78  4.69     1     0  42.8 108087
```

Now, let's compute the hazard function "by hand" using our nonparametric estimator (and some R code).

```
# first use "Surv()" to count up number at risk and number of events at each time
temp <- survfit(Surv(w, t, d) ~ 1, data = dat)

dat2 <- data.frame(t = temp$time, n = temp$n.risk, events = temp$n.event) %>%
  filter(events > 0) %>%
  mutate(h = events/(n * (t - lag(t, default = 0))))

dat2
```

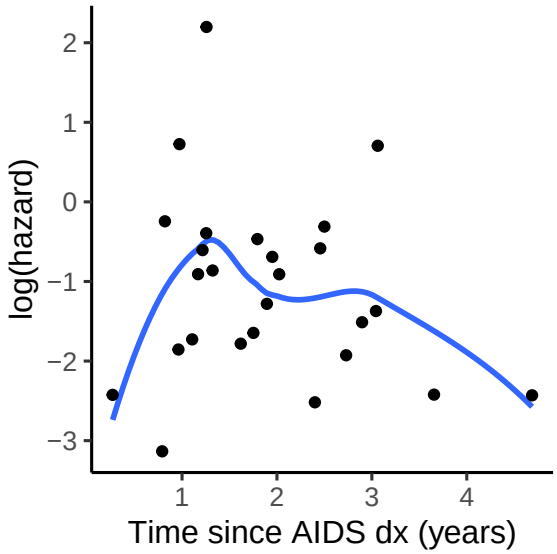
```
##        t  n events          h
## 1  0.269 42      1 0.08851124
## 2  0.791 44      1 0.04353884
## 3  0.820 44      1 0.78369906
## 4  0.962 45      1 0.15649452
## 5  0.973 44      1 2.06611570
## 6  1.107 42      1 0.17768301
## 7  1.169 40      1 0.40322581
## 8  1.216 39      1 0.54555374
## 9  1.255 38      1 0.67476383
## 10 1.258 37      1 9.00900901
## 11 1.322 37      1 0.42229730
## 12 1.619 40      2 0.16835017
## 13 1.752 39      1 0.19278967
## 14 1.794 38      1 0.62656642
## 15 1.894 36      1 0.27777778
## 16 1.951 35      1 0.50125313
## 17 2.024 34      1 0.40290089
## 18 2.400 33      1 0.08059317
## 19 2.456 32      1 0.55803571
## 20 2.500 31      1 0.73313783
## 21 2.729 30      1 0.14556041
## 22 2.897 27      1 0.22045855
## 23 3.043 27      1 0.25367834
```

```
## 24 3.062 26      1 2.02429150
## 25 3.655 19      1 0.08875477
## 26 4.688 11      1 0.08800493
```

... and plot our results.

```
haz_plot <- ggplot() +
  geom_smooth(aes(x = t, y = log(h)), data = dat2, se = F) +
  geom_point(aes(x = t, y = log(h)), data = dat2) +
  ylab("log(hazard)") +
  xlab("Time since AIDS dx (years)") +
  theme_classic(base_size = 12)

haz_plot
```



Now, let's compute our estimator of the hazard function separately for those with low vs high VL

```
# define high vl
dat3 <- dat %>%
  mutate(hivl = as.numeric(vl>1e5))

# first use "Surv()" to count up number at risk and number of events at each time in each group
temp <- survfit(Surv(w, t, d) ~ hivl, data = dat3)

dat4 <- data.frame(t = temp$time, n = temp$n.risk, events = temp$n.event,
                   hivl = c(rep(0, temp$strata[1]), rep(1, temp$strata[2]))) %>%
  filter(events > 0) %>%
  group_by(hivl) %>%
  arrange(hivl, t) %>%
  mutate(h = events/(n * (t - lag(t, default = 0))))

dat4
```

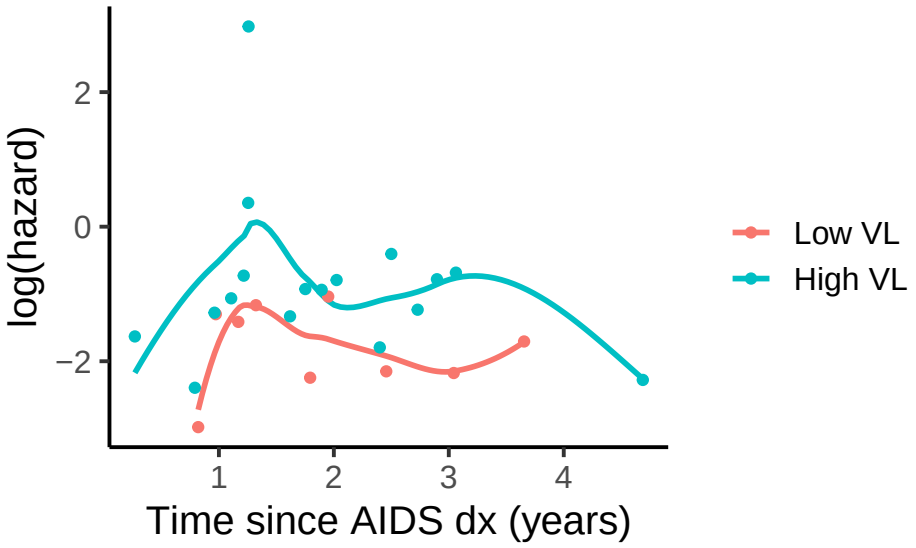
```
## # A tibble: 26 x 5
## # Groups:   hivl [2]
##       t     n events  hivl      h
##   <dbl> <dbl>  <dbl> <dbl>  <dbl>
##  1 0.82     24      1     0 0.0508
```

```
##  2 0.973    24      1     0 0.272
##  3 1.17     21      1     0 0.243
##  4 1.32     21      1     0 0.311
##  5 1.79     20      1     0 0.106
##  6 1.95     18      1     0 0.354
##  7 2.46     17      1     0 0.116
##  8 3.04     15      1     0 0.114
##  9 3.66      9      1     0 0.182
## 10 0.269    19      1     1 0.196
## # ... with 16 more rows
```

... and plot our results.

```
haz_plot <- ggplot() +
  geom_smooth(aes(x = t, y = log(h), color = factor(hivl)), data = dat4, se = F) +
  geom_point(aes(x = t, y = log(h), color = factor(hivl)), data = dat4) +
  ylab("log(hazard)") +
  xlab("Time since AIDS dx (years)") +
  scale_color_discrete(labels = c("Low VL", "High VL"), name = "")+
  theme_classic(base_size = 15)

haz_plot
```



Comparing hazard functions

Above, we computed the hazard function for each group "by hand" using our nonparametric estimator. Now let's think back to the hazard function (the parameter) and consider it once again separately for each group. For example:

Low VL ($X = 0$): $h_{X=0}(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t+\Delta t | t \leq T, X=0)}{\Delta t}$

High VL ($X = 0$): $h_{X=1}(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t+\Delta t | t \leq T, X=1)}{\Delta t}$

We can compare these functions using the *hazard ratio function*, $HR(t) = h_{X=1}(t)/h_{X=0}(t)$.

We often assume that that hazard ratio is constant over time, and collapse $HR(t)$ to HR.

The Cox model

Our parameter of interest is

$$HR = \frac{h_1(t)}{h_0(t)}$$

.

We encode this parameter into a regression model by setting $HR = \exp(\beta) = \frac{h_1(t)}{h_0(t)}$.

Then, we rearrange to write $h_1(t)$ as a function of everything else: $h_1(t) = h_0(t)\exp(\beta)$.

Finally, we generalize to accommodate >2 values of X: $h_x(t) = h_0(t)\exp(\beta X)$, which we often write simply as

$$h(t) = h_0(t)\exp(\beta X)$$

Why is the Cox model called a "proportional hazards model"? Because the hazard for any group is the hazard for the reference group scaled by a fixed value $\exp(\beta X)$.

Cox model intuition

What is β in the Cox model?

β is the average log(HR)
β is the average $\log(h_1/h_0)$
β is the average $\log(h_1) - \log(h_0)$
β is the average difference in log hazards.

HR = $\exp(\beta)$

Pencil and paper example

Please see the slides for this week to work a pencil and paper example in lecture. Feel free to copy that example to this page.

Cox model and the baseline hazard function

> The *baseline hazard function* is the hazard function among participants with the 0 level of all variables included in the model.

An advantage of the Cox model is that we need not specify the baseline hazard function $h_0(t)$. This means that we impose no parametric assumption on the shape of $h_0(t)$, leaving it free to be completely arbitrary.

The fact that $h_0(t)$ remains nonparametric while $\exp(\beta X)$ is parametric makes the cox model a **semiparametric** model.

Partial likelihood

The cox model is estimated using *partial likelihood*

- the likelihood function factors into 2 parts: 1 that depends on $h_0(t)$ and β and one that depends on β alone.
- the Cox model discards the first part and maximizes only the second part.

Consequences:
1. Robustness to shape of baseline hazard function
2. Less precise than if we had used full likelihood (though loss of precision is small)

Form of the partial likelihood:

$$PL = \prod_{i=1}^{n} \left[\frac{\exp(\beta X_i)}{\sum_{j=1}^{n} Y_j(R_i) \exp(\beta X_j)} \right]^{\delta_i}$$

where $Y_j(R_i)$ is an indicator that individual j is in the risk set at event time R_i

Ties

`Ties` refer to event times at which > 1 events occur.

When there are no tied event times, the partial likelihood above is valid. Otherwise, we have several options:

1. Breslow's method (SAS default)
2. Efron's approximation (better)
3. Exact method (best, but computationally intense)
4. Discrete method (best if times really are discrete)

Chapter 5 of the Allison book cover these in detail.

Proportional hazards

Standard Cox model assumes log hazard functions for $X = 1$ and $X = 0$ groups are equal (vertical) distance over time.

Or, that we are interested in the (information-weighted) average difference in log hazard functions over the study period of length τ.

We can check the proportional hazards assumption by plotting the cumulative hazard functions for each group and looking to see if they are parallel (see example).

Alternatives: product term with time, Schoenfeld residuals, others?

Relaxing the proportional hazards assumption

We can fit a model with product term between X and $g(t)$, where $g()$ is a user-specified function (say log), and t is a time-updated covariate (rather than T)

If HR is not constant, a refined choice of $g(t)$ allows the HR to vary over time. But. . . ,

If HR is not constant (and we do not wish to average), we must report >1 HR. (see example).

Accounting for confounding in Cox models

This week, we will cover

- multivariable Cox models
- stratified Cox models

Next week you will cover inverse probability weights, which can also be applied to Cox models.

Say we wish to estimate the association between X and the hazard function, accounting for confounding by Z.

Multivariable Cox models

Recall that the standard Cox model is $h(t) = h_0(t)\exp(\beta X)$.

We can extend this model to include covariates: $h(t) = h_0(t)\exp(\beta_1 X + \beta_2 Z)$.

Interpretation:

- $\exp(\beta_1)$ is the HR for a unit difference in X holding Z constant at any level (calculated at each level, and information-weighted averaged over levels)
- $\exp(\beta_2)$ is the HR for unit difference in Z, likewise holding X constant at any level.

Stratified Cox models

Recall that the standard Cox model is $h(t) = h_0(t)\exp(\beta X)$.

We can extend this model to stratify by covariates: $h(t) = h_{0z}(t)\exp(\alpha X)$.

h_{0z} represents the hazard function among the unexposed in stratum z of discrete nuisance variable Z.

This approach is more flexible than including Z in the model. It does not constrain $h_{0,z=1}$ and $h_{0,z=0}$ to be proportional.

Stratified Cox models work by constructing separate partial likelihood functions for each stratum of Z, multiplying functions together, and choosing the value of α that maximizes combined function

Model form assumptions

While we can ignore the baseline hazard function, we must still model the parametric part of the model correctly

- Assess functional form between regressors and outcome
- Account for nonmultiplicativity using interaction terms
- Careful not to include too many regressors in small samples (e.g., rule of thumb: 10 events/binary regressor)

Example

Our example is based on the Cole and Hudgens 2010 Survival analysis paper referenced above. Details include:

- Interested in describing survival after AIDS in MACS
- Origin = AIDS diagnosis, Event = all-cause mortality, Time scale = AIDS duration
- Enroll 42 men alive on 1 January 1995 with a prior clinical AIDS diagnosis and enroll 36 additional men with a clinical AIDS diagnosis between 1 January 1995 and 1 January 1998
- Follow all 78 (= 42 + 36) men for all-cause mortality through 1 January 1998, the date of study completion

Crude Cox model

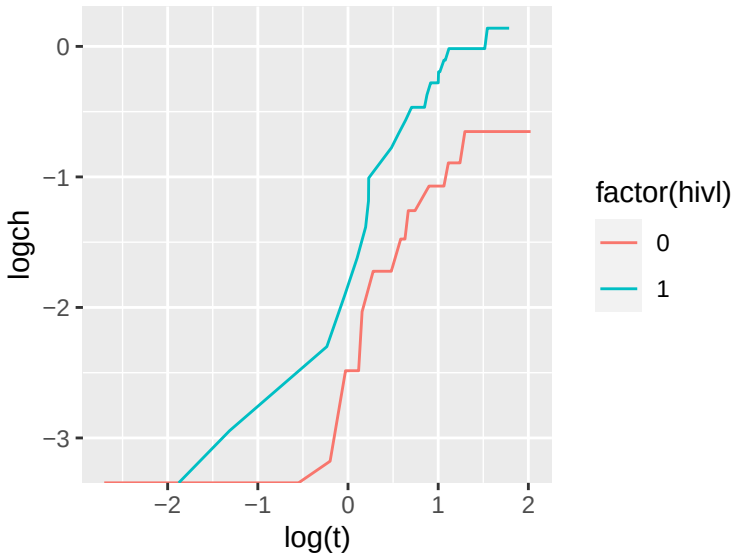
```
crude <- coxph(Surv(w, t, d)~hivl, data = dat3)
summary(crude)
```

```
## Call:
## coxph(formula = Surv(w, t, d) ~ hivl, data = dat3)
##
##   n= 78, number of events= 27
##
##        coef exp(coef) se(coef)     z Pr(>|z|)
## hivl 0.7706    2.1611   0.4087 1.885   0.0594 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## hivl     2.161     0.4627      0.97     4.815
##
## Concordance= 0.597  (se = 0.048 )
## Likelihood ratio test= 3.79  on 1 df,   p=0.05
## Wald test            = 3.55  on 1 df,   p=0.06
## Score (logrank) test = 3.73  on 1 df,   p=0.05
```

Check proportional hazards assumption

First, we will examine the log(H) (or log(-log(S))) plots:

```
mod <- survfit(Surv(w,t, d)~hivl, data = dat3)
chdat <- data.frame(t = mod$time, logch = log(mod$cumhaz),
                hivl = c(rep(0, mod$strata[1]), rep(1, mod$strata[2])))
ggplot()+
  geom_line(aes(x = log(t), y = logch,
                color = factor(hivl)), data = chdat)
```



Alternatively, we can assess interactions with time

```
# #continuous time
# coxph(Surv(t,d) ~ hivl + tt(hivl), data = dat3, ties = "efron",
#   tt = list(function(hivl, t, ...){hivl*t}))
# coxph(Surv(t,d) ~ hivl + tt(hivl), data = dat3, ties = "efron",
#   tt = list(function(hivl, t, ...){hivl*log(t)}))

#before/after 2 years
```

```
coxph(Surv(w,t,d) ~ hivl + tt(hivl), data = dat3, ties = "efron",
      tt = list(function(hivl, t, ...){hivl*t>2.5}))
```

```
## Call:
## coxph(formula = Surv(w, t, d) ~ hivl + tt(hivl), data = dat3,
##     ties = "efron", tt = list(function(hivl, t, ...) {
##         hivl * t > 2.5
##     }))
##
##                   coef exp(coef) se(coef)      z      p
## hivl          0.78382   2.18982  0.46327  1.692 0.0907
## tt(hivl)TRUE -0.06006   0.94170  0.98382 -0.061 0.9513
##
## Likelihood ratio test=3.79  on 2 df, p=0.1503
## n= 78, number of events= 27
```

Multivariable model

Account for confounding due to age and race using a multivariable model

```
library(splines)

# loglinear age and race
adj <- coxph(Surv(w, t, d)~hivl + age + nw, ties = "efron", data = dat3)
summary(adj)
```

```
## Call:
## coxph(formula = Surv(w, t, d) ~ hivl + age + nw, data = dat3,
##     ties = "efron")
##
##   n= 78, number of events= 27
##
##           coef exp(coef)  se(coef)     z Pr(>|z|)
## hivl 1.0234236 2.7827054 0.4304848 2.377  0.01744 *
## age  0.0002042 1.0002042 0.0286822 0.007  0.99432
## nw   1.2170798 3.3773110 0.4584447 2.655  0.00794 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## hivl     2.783     0.3594    1.1968     6.470
## age      1.000     0.9998    0.9455     1.058
## nw       3.377     0.2961    1.3751     8.295
##
## Concordance= 0.654  (se = 0.058 )
## Likelihood ratio test= 10.16  on 3 df,   p=0.02
## Wald test            = 10.28  on 3 df,   p=0.02
## Score (logrank) test = 10.79  on 3 df,   p=0.01
```

```
#curviloglinear age and race
adj_2 <- coxph(Surv(w, t, d)~hivl + ns(age, 3) + nw, ties = "efron", data = dat3)
summary(adj_2)
```

```
## Call:
## coxph(formula = Surv(w, t, d) ~ hivl + ns(age, 3) + nw, data = dat3,
```

```
##     ties = "efron")
##
##   n= 78, number of events= 27
##
##                  coef exp(coef) se(coef)      z Pr(>|z|)
## hivl         1.09013   2.97465  0.44486  2.450  0.01427 *
## ns(age, 3)1  1.25913   3.52237  0.91037  1.383  0.16664
## ns(age, 3)2 -3.55417   0.02861  1.54175 -2.305  0.02115 *
## ns(age, 3)3 -0.59268   0.55284  1.12564 -0.527  0.59852
## nw           1.29840   3.66344  0.47792  2.717  0.00659 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##              exp(coef) exp(-coef) lower .95 upper .95
## hivl           2.97465     0.3362  1.243859    7.1138
## ns(age, 3)1    3.52237     0.2839  0.591458   20.9771
## ns(age, 3)2    0.02861    34.9586  0.001393    0.5872
## ns(age, 3)3    0.55284     1.8088  0.060878    5.0205
## nw             3.66344     0.2730  1.435763    9.3475
##
## Concordance= 0.721  (se = 0.048 )
## Likelihood ratio test= 17.27  on 5 df,   p=0.004
## Wald test            = 18.49  on 5 df,   p=0.002
## Score (logrank) test = 19.69  on 5 df,   p=0.001
```

```
#race only
adj_3 <- coxph(Surv(w, t, d)~hivl + nw, ties = "efron", data = dat3)
summary(adj_3)
```

```
## Call:
## coxph(formula = Surv(w, t, d) ~ hivl + nw, data = dat3, ties = "efron")
##
##   n= 78, number of events= 27
##
##        coef exp(coef) se(coef)     z Pr(>|z|)
## hivl 1.0231    2.7819   0.4283 2.389  0.01691 *
## nw   1.2163    3.3746   0.4445 2.736  0.00621 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## hivl     2.782     0.3595     1.202     6.440
## nw       3.375     0.2963     1.412     8.064
##
## Concordance= 0.643  (se = 0.055 )
## Likelihood ratio test= 10.16  on 2 df,   p=0.006
## Wald test            = 10.28  on 2 df,   p=0.006
## Score (logrank) test = 10.77  on 2 df,   p=0.005
```

Stratified model

Account for confounding due to race using a stratified model

```
#race only
adj_4 <- coxph(Surv(w, t, d)~hivl + strata(nw), ties = "efron", data = dat3)
```

```
summary(adj_4)
```

```
## Call:
## coxph(formula = Surv(w, t, d) ~ hivl + strata(nw), data = dat3,
##     ties = "efron")
##
##   n= 78, number of events= 27
##
##        coef exp(coef) se(coef)     z Pr(>|z|)
## hivl 0.9506    2.5871   0.4268 2.227   0.0259 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## hivl     2.587     0.3865     1.121     5.972
##
## Concordance= 0.59  (se = 0.055 )
## Likelihood ratio test= 5.34  on 1 df,   p=0.02
## Wald test            = 4.96  on 1 df,   p=0.03
## Score (logrank) test = 5.27  on 1 df,   p=0.02
```