DEM 7223 - Event History Analysis - Cox Proportional Hazards Model Part 1

Corey S. Sparks, PhD

22 September, 2022

Notes

Parametric model specifications

When considering a parametric hazard model, we saw that the choice of the specified distribution function is key * If we expect the hazard (or pdf) to take an exponential form, we use that model, same for the Weibull of log-normal, etc.

- So by saying this, we force our data to correspond to the distribution we specify.
- What if, however, the distribution of the durations do not necessarily follow one of the parametric families? We are then left with the most heinous of statistical quandaries: model mis-specification :(
- So in considering the use of hazard models, we need to also consider the case where we cannot (or adequately) specify an appropriate parametric model, this is the reasoning behind the use of Cox's (1972) semi-parametric modeling approach.

The Cox Proportional Hazards Model

- Cox (1972) suggested a more widely applicable model to be used in situations where a suitable parametric distribution is unavailable
- Also, it allows the analyst the freedom to explore the theoretical connections between the covariates and the hazard rate, free of the parametric assumption.
- We are still modeling the effect of individual characteristics on the hazard of an event outcome, in the same way as with the parametric proportional hazards model. We just leave the baseline hazard rate **unspecified in terms of structural parameters**

Model form

The Cox model has the familiar form:

$$h(t) = h_0(t)exp^{(x'\beta)}$$

- Which is the same form as the parametric proportional hazards models we saw last week.
- The key difference is in the value of h_0
- In the Cox model the baseline hazard rate, h_0 is the observed empirical hazard rate for individuals in the baseline, or reference group of the sample.
- For any two individuals with different values of a covariate, x, the *hazard ratio* is:

$$\frac{h_i(t)}{h_0(t)} = exp(\beta(x_i - x_0))$$

so when $x_i = 0, x_0 = 0$, the hazard ratio is just $exp(\beta)$

This is called a *semi-parametric model* because, while the baseline hazard rate h_0 does not have any structural parameters, the regression effects, β are estimated.

Partial likelihood for the Cox Model

- Unlike the parametric models, where we expected the hazard to be a function of time, and where the time between events actually contributed some information to the estimation, the Cox model handles things differently
- The Cox model works, much like the Kaplan-Meier estimator, in terms of *ordered event times.*
- So we actually only get an estimate of the function at the observed failure times, vs the parametric models where we get an estimate of the risk at all possible failure times (which is one of the main reasons for using them!)

Partial likelihood estimation

- First, we sort all observed failure times, in ascending order:

$$t_1 < t_2 < ... < t_n$$

for all individuals in the data. Now we assume all events have unique durations. In actuality, ties exist and there are a variety of ways to handle these.

- Each observation has its censoring indicator δ_i, which tells if the individual is observed or censored at each time.
- These observations are then modeled in terms of their relative hazards.
- The partial likelihood is constructed by taking the cumulative product of the hazard, for the *Risk set* at time t. The probability that a case j will fail at time t is :

$$Pr(t_j = T_i | R(t_i)) = \frac{h(t_i j)}{\sum_{j \in R(t_i)} h(t_i j)}$$

- Where the denominator in this equation is summing over all individuals at risk at time t_i
- The partial likelihood function in terms of the regression parameters is:

$$L_p = \prod_{\delta=1} \frac{h(t_{ij}) exp(x'\beta)}{\sum_{j \in R(t_i)} h(t_{ij}) exp(x'\beta)}$$

* Since both the numerator and denominator contain the overall hazard, it cancels, giving:

$$L_p = \prod_{\delta=1} \frac{exp(x'\beta)}{\sum_{j \in R(t_i)} exp(x'\beta)}$$

- Which says nothing about the baseline hazard function or its shape.
- By maximizing this partial likelihood, estimates of the β's are found
- This is called *Maximum Partial likelihood estimation*, and is not a true likelihood.
- This is because we have not directly included the survival times of the censored cases, instead these are handled by modifying the risk set, $R(t_{ij})$, but not explicitly in the numerator
- Much in the way Kaplan-Meier treats censored cases
- Cox in later papers demonstrated that the same properties (efficiency, asymptotic normality) of the estimates still hold.
- This allows us to use our standard likelihood ratio tests, and Wald parameter tests in interpreting and comparing models.
- Ties in the data may be handled by modification of the partial likelihood function
- Ties are simply events that have the same event time and are very common in demographic work
- We have seen this repeatedly in our child mortality and birth interval analyses.

- The likelihood must be modified to incorporate these "simultaneously" occurring event times
- The ability to modify the likelihood function for the Cox model also exemplifies the flexibility of the model over the parametric forms, which are not specified to handle tied event times.
- The big issue with tied observations is related to determining the risk set at a particular time, but also in determining the future risk sets

Handling ties

- Breslow's Method
  - Assumes the same risk set for all tied events
  - This is weakest if there are a large number of ties at a particular time point
- Efron's Method
  - Uses a different risk set at any time point for tied observations
  - This is done by considering all possible orderings of failure times for the tied observations
- Other methods exist, but Efron's method is the most widely used.

Interpreting the Cox model

- We have already seen how the proportional hazards model is generally interpreted from the Weibull and Exponential cases
- This is done via the $exp(\beta)$, or the hazards ratio
- If the regression coefficient, β, is positive (the hazard is increasing), $\exp(\beta)$ will be >1 and this indicates that an individual with a value of x=1 will have a $1 - exp(\beta)$ higher hazard rate, compared with an individual with x=0
- If the regression coefficient, β, is negative (the hazard is decreasing), $exp(\beta)$ will be <1 and this indicates that an individual with a value of x=1 will have a $1 - exp(\beta)$ lower hazard rate, compared with an individual with x=0

Good variable construction habits

- In order to facilitate the interpretability of the model hazard ratios, in demography, we typically create binary dummy variables for things like age via recoding
- i.e. construct a set of dummy variables for 5 year age intervals between 15 and 50 with the reference group being 30-35.

if age>=15 & age<20 age1=1, else age1=0

if age>=20 & age<25 age2=1, else age2=0

if age>=25 & age<30 age3=1, else age3=0

30-35 is reference group without a covariate constructed

if age>=35 & age<40 age4=1, else age4=0

if age>=40 & age<45 age5=1, else age5=0

if age>=45 & age<50 age6=1, else age6=0

if age>=50 age7=1, else age7=0

- You could also use a factor variable with the appropriate level as the reference category.
- This approach is also useful for coding incomes, but is done in terms of the income distribution's quantiles

Good variable construction: continuous case

- Often if our covariate is continuous (like weight, height, maybe income if we're treating it that way) we construct a z-score for the variable
- The z-score is called the standard score, and centers the covariate around it's mean, so the new mean is 0 and each individual's value represents their departure from the mean
- i.e. if a person's weight z-score was -5, then they are 5 pounds below the average weight
- In R, the `scale()` function does this.

Confidence intervals for hazard ratios

- Because the partial likelihood estimates of the β's have the same asymptotic properties as mle's of β, we can construct $1-\alpha\%$ confidence intervals for both β and the $\exp(\beta)$, or hazard ratio.
- These are often reported in output in tables.
- To find the lower 95% ci for $\exp(\beta)$, we do

$$\text{Lower } 1-\alpha \text{ CI} = exp(\beta - z * s.e.(\beta))$$

$$\text{Upper } 1-\alpha \text{ CI} = exp(\beta + z * s.e.(\beta))$$

- For 95% confidence intervals, z would be 1.96

Risk Scores

- Often we are interested in how "at risk" a certain individual is or at least someone with a particular set of covariates
- *Risk scores* represent the linear combination of all the individual's covariates on their hazard
- If none of our covariates vary with time (which we assume for the present), the "risk score" would be:

$$h_i = h_0 * exp(\beta_1 * x_1 + \beta_2 * x_2 + ... + \beta_k * x_k)$$

Since our "baseline hazard" is h_0, the risk score for an individual with a particular set of covariates is just:

$$h_i = exp(\beta_1 * x_1 + \beta_2 * x_2 + ... + \beta_k * x_k)$$

- Which represents their personal risk relative to the baseline.
- The "baseline" is just the hazard when all covariate values are zero!

Visualizing the Cox model

- We can recover the hazard and survival functions from the Cox model
- If we fit the Cox model with no predictors, the estimates of $h(t)$ and $S(t)$ are EXACTLY the same as the Kaplan-Meier estimates
- The baseline hazards and survival functions are just the Kaplan-Meier estimates, for individuals with the reference level for all predictors
  - i.e. all 0's for all x's
- This is because their risk score is:

$$\text{Risk Score}_i = exp(\beta_1 * x_1 + \beta_2 * x_2 + ... + \beta_k * x_k)$$

- If all x's are 0, then the risk score is 1: $exp(0) = 1$
- By turning "off" and "on" different x's, we ca build different risk scores for different prototypical individuals
  - Remember, you're only as unique as your covariate vector!
- A hazard for an individual with a risk score, y is:

$$\hat{H}(t_{ij}) = \hat{H}_0(t_j) * y_i$$

- which is just a multiplicative effect on the cumulative hazard function.
- To show this in terms of survival, we do:

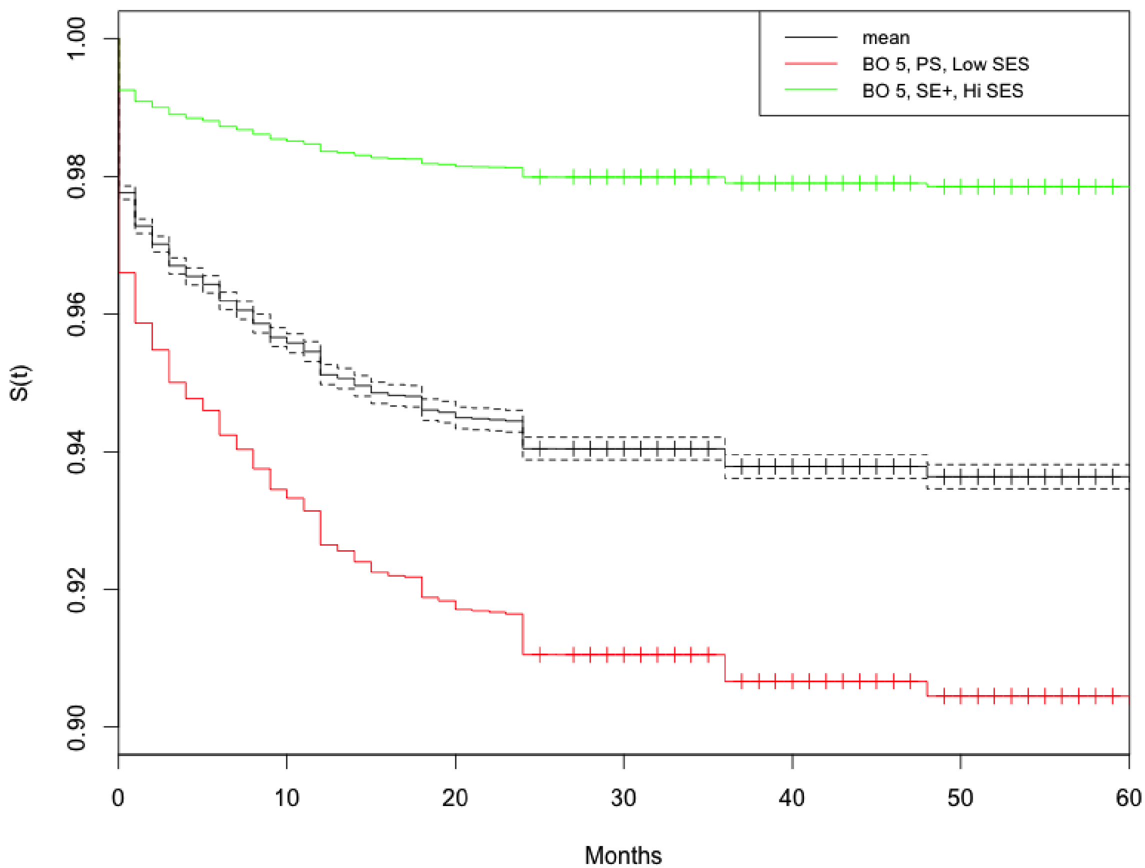
$$\hat{H} = -logS(t_{ij}) = -logS_0(t_j) * y_i$$

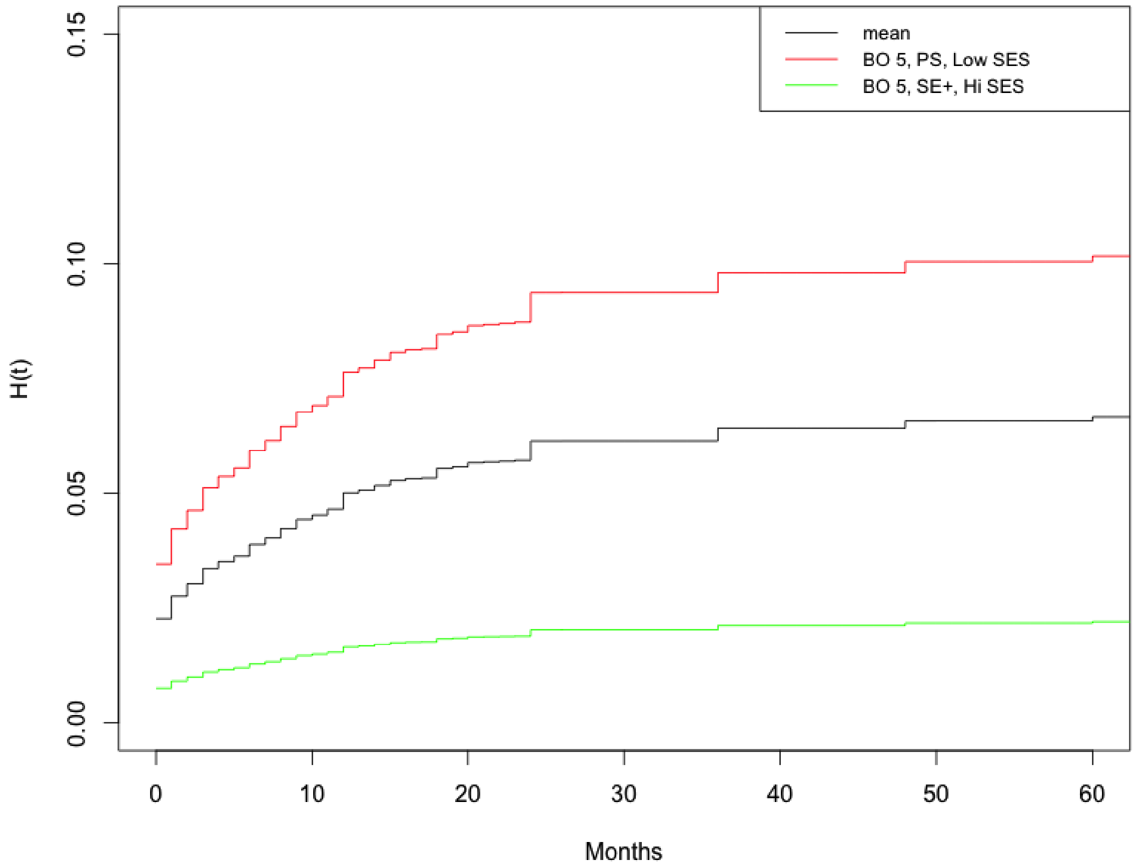
or

$$\hat{S}(t_{ij}) = \hat{S}(t_j)^{y_i}$$

Which says that the survival function for an individual with risk score y is a power of the baseline survival rate.

- So all we need to do is estimate the K-M functions and multiply them by prototypical risk scores, or prototypical "people", and we can recover a hazard or survival function estimate for those kinds of people
- This lets us visualize the results very effectively.


mean
BO 5, PS, Low SES
BO 5, SE+, Hi SES
S(t)
1.00
0.98
0.96
0.94
0.92
0.90
0
10
20
30
40
50
60
Months



Data Example

This example will illustrate how to fit the Cox Proportional hazards model to continuous duration data (i.e. person-level data) and a discrete-time (longitudinal) data set.

The first example uses longitudinal data from the ECLS-K. Specifically, we will examine the transition into poverty between kindergarten and third grade.

In the second example, I use the *time between the first and second birth* for women in the data as the *outcome variable*. The data for this example come from the DHS Model data file Demographic and Health Survey for 2012 individual recode file. This file contains information for all women sampled in the survey between the ages of 15 and 49.

Using Longitudinal Data

As in the other examples, I illustrate fitting these models to data that are longitudinal, instead of person-duration. In this example, we will examine how to fit the Cox model to a longitudinally collected data set.

First we load our data

```
#Load required libraries
library(foreign)
library(survival)
library(car)
library(survey)
library(eha)
library(tidyverse)
options(survey.lonely.psu = "adjust")
```

```
eclskk5<-readRDS("C:/Users/ozd504/OneDrive - University of Texas at San Antonio/classes/de
names(eclskk5)<-tolower(names(eclskk5))
#get out only the variables I'm going to use for this example
myvars<-c( "childid","x_chsex_r", "x_raceth_r", "x1kage_r","x4age",
           "x5age", "x6age", "x7age", "x2povty","x4povty_i", "x6povty_i",
           "x8povty_i","x12par1ed_i", "s2_id","w6c6p_6psu",
           "w6c6p_6str", "w6c6p_20")
eclskk5<-eclskk5[,myvars]
```

Recode variables:

```
# time varying variables
eclskk5$age_1<-ifelse(eclskk5$x1kage_r==-9, NA, eclskk5$x1kage_r/12)
eclskk5$age_2<-ifelse(eclskk5$x4age==-9, NA, eclskk5$x4age/12)
#for the later waves, the NCES group the ages into ranges of months,
#so 1= <105 months, 2=105 to 108 months.
#So, I fix the age at the midpoint of the interval they give,
#and make it into years by dividing by 12

eclskk5$age_3<-ifelse(eclskk5$x5age==-9, NA, eclskk5$x5age/12)

eclskk5$pov_1<-ifelse(eclskk5$x2povty==1,1,0)
eclskk5$pov_2<-ifelse(eclskk5$x4povty_i==1,1,0)
eclskk5$pov_3<-ifelse(eclskk5$x6povty_i==1,1,0)
```

```
#Time constant variables
#Recode race with white, non Hispanic as reference using dummy vars
eclskk5$race_rec<-Recode (eclskk5$x_raceth_r,
                          recodes="1 = 'nhwhite';2='nhblack';3:4='hispanic';5='nhasian'; 6
                          as.factor = T)
eclskk5$male<-Recode(eclskk5$x_chsex_r, recodes="1=1; 2=0; -9=NA")
eclskk5$mlths<-Recode(eclskk5$x12par1ed_i, recodes = "1:2=1; 3:9=0; else = NA")
eclskk5$mgths<-Recode(eclskk5$x12par1ed_i, recodes = "1:3=0; 4:9=1; else =NA")
```

NOTE I need to remove any children who are missing any of the necessary variables, and who are already in poverty in wave 1, because they are not at risk of experiencing **this particular** transition.

Again, this is called forming the *risk set*

```
eclskk5<-eclskk5 %>% filter(is.na(pov_1)==F &
                  is.na(pov_2)==F &
                  is.na(pov_3)==F &
                  is.na(age_1)==F &
                  is.na(age_2)==F &
                  is.na(age_3)==F &
                    pov_1!=1)
```

Now, I need to form the transition variable, this is my event variable, and in this case it will be 1 if a child enters poverty between the first wave of the data and the third grade wave, and 0 otherwise.

Now we do the entire data set. To analyze data longitudinally, we need to reshape the data from the current "wide" format (repeated measures in columns) to a "long" format (repeated observations in rows). The `reshape()` function allows us to do this easily. It allows us to specify our repeated measures, time varying covariates as well as time-constant covariates.

```
e.long1 <- eclskk5 %>%
  #rename(wt = w4c4p_40,strata= w4c4p_4str, psu = w4c4p_4psu)%>%
  select(childid,male, race_rec, mlths, mgths,   #time constant
        age_1, age_2, age_3, #t-varying variables
         pov_1, pov_2, pov_3,
        w6c6p_6psu, w6c6p_6str, w6c6p_20)%>%
   pivot_longer(cols = c(-childid, -male, -race_rec, -mlths, -mgths,
                         -w6c6p_6psu, -w6c6p_6str, -w6c6p_20), #time constant variables go
               names_to  = c(".value", "wave"), #make wave variable and put t-v vars into
               names_sep = "_") %>% #all t-v variables have _ between name and time, like
```

```
  group_by(childid)%>%
  mutate(age_enter = age,
         age_exit = lead(age, 1, order_by=childid))%>%
  mutate(nexpov = dplyr::lead(pov,n=1, order_by = childid))%>%
  mutate(povtran = ifelse(nexpov == 1 & pov == 0, 1, 0))%>%
  filter(is.na(age_exit)==F)%>%
  ungroup()
```

Cox regression model

Compared to the parametric models we saw last week, the Cox model See also the partial likelihood paper, does not specify a parametric form for the baseline hazard rate. The model still looks the same as the other proportional hazards models:

$\mathrm{h}(t) = h_0 \exp\left(x\text{`}\beta\right)$

but h_0 is not a parametric function. Instead, the baseline hazard rate is the empirically observed hazard rate, and the covariates shift it up or down, proportionally.

Using age as the time variable:

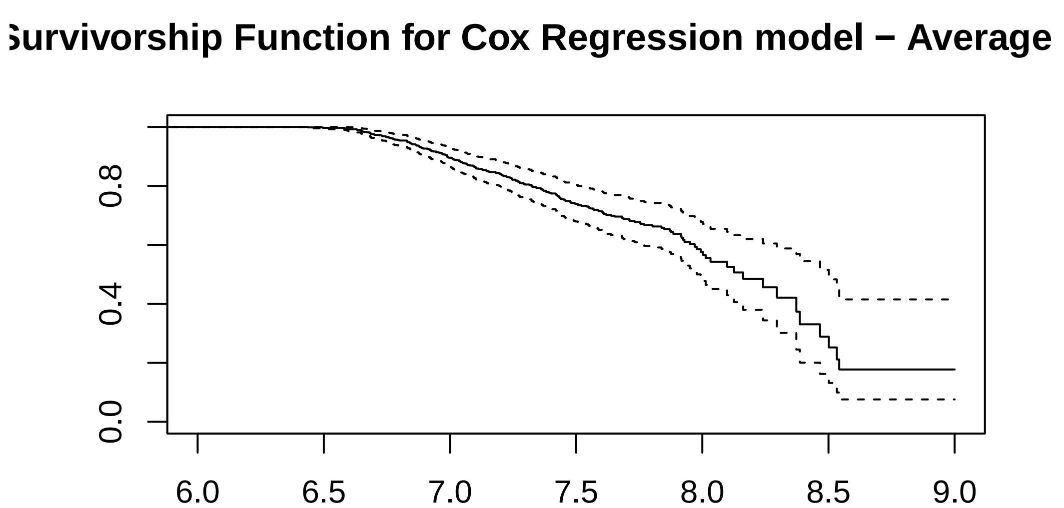
Here I use age of the child as the time variable, this should show how children experience poverty during school.

```
#Cox Model
#interval censored
e.long1<-e.long1%>%
  filter(complete.cases(age_enter, age_exit, povtran,
                        mlths, mgths, race_rec))


fitl1<-coxreg(Surv(time = age_enter,time2=age_exit, event = povtran)~mlths+mgths+race_rec,

              data=e.long1)
summary(fitl1)
```

```
Covariate             Mean       Coef     Rel.Risk   S.E.    LR p
mlths                 0.059     0.479     1.615     0.188    0.012
mgths                 0.783    -1.091     0.336     0.160    0.000
race_rec                                                     0.000
        hispanic      0.225     0         1 (reference)
```

```
         nhasian      0.084     -0.623      0.536      0.289
         nhblack      0.065     -0.058      0.943      0.242
         nhwhite      0.554     -1.180      0.307      0.177
           other      0.071     -0.446      0.640      0.285

Events                    223
Total time at risk        4064.3
Max. log. likelihood      -1480.9
LR test statistic         204.45
Degrees of freedom        6
Overall p-value           0
```

```r
plot(survfit(fitl1),  xlim=c(6, 9),
     main="Survivorship Function for Cox Regression model - Average Child")
```




```r
gtsummary::tbl_regression(fitl1, exponentiate = TRUE)
```

```
Table printed with `knitr::kable()`, not {gt}. Learn why at
https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
To suppress this message, include `message = FALSE` in code chunk header.
```

Characteristic	**HR**	**95% CI**	**p-value**
X12 PARENT 1 EDUCATION LEVEL (IMPUTED)	1.61	1.12, 2.33	0.011
X12 PARENT 1 EDUCATION LEVEL (IMPUTED)	0.34	0.25, 0.46	<0.001
race_rec			
hispanic	—	—	
nhasian	0.54	0.30, 0.95	0.031
nhblack	0.94	0.59, 1.52	0.8
nhwhite	0.31	0.22, 0.44	<0.001
other	0.64	0.37, 1.12	0.12

The model results (`Rel.Risk`) show that children with mom's who have less than a high school education have 1.61 times higher risk of going into poverty during this period, while children whose mother have more than a high school education are 0.66 % less likely to enter poverty, compared to children whose mothers had a high school education. Likewise, Hispanic, Non-Hispanic black, Native American children all face higher risk of entering poverty, compared to Non-Hispanic whites.

Risk scores

The Cox model generates a *"risk score"* for each individual. This is just $\exp(x`\beta)$, or the exponent of the linear predictor. These are not absolute values that have any real direct interpretation, but they are interpretable in a **relative** sense.

Risk scores >1 indicate that a person has higher risk than the baseline category, while risk scores <1 have lower relative risk, compared to the baseline. If you want to interpret these, it's necessary to have the baseline category be a meaningful "type" of person. In our example above, the baseline group would be Hispanic children, with a mother who had a high school education. i.e. all x's are 0.

```
e.long1$risk<-exp(fitl1$linear.predictors)

#highest risk child
e.long1[which.max(e.long1$risk),
        c("childid", "age_enter", "mlths", "mgths","race_rec", "risk")]
```

```
# A tibble: 1 x 6
  childid  age_enter mlths mgths race_rec risk[,1]
  <chr>        <dbl> <dbl> <dbl> <fct>       <dbl>
1 10000744      5.26     1     0 hispanic     1.61
```

```r
#lowest risk child
e.long1[which.min(e.long1$risk),
        c("childid", "age_enter", "mlths", "mgths","race_rec", "risk")]
```

```
# A tibble: 1 x 6
  childid  age_enter mlths mgths race_rec risk[,1]
  <chr>        <dbl> <dbl> <dbl> <fct>       <dbl>
1 10000046      5.92     0     1 nhwhite     0.103
```

```r
e.long1<-e.long1%>%
   mutate(rrisk= round(risk, 4))%>%
   arrange(risk)

e.u<-unique(e.long1$rrisk)
e.u[order(e.u)]
```

```
 [1] 0.1032 0.1801 0.2150 0.3073 0.3168 0.3358 0.4962 0.5362 0.6401 0.8659
[11] 0.9433 1.0000 1.0337 1.5234 1.6149
```

```r
head(e.long1[which.max(e.long1$rrisk),
             c("childid", "age_enter", "mlths", "mgths","race_rec", "risk")],
     n=20)
```

```
# A tibble: 1 x 6
  childid  age_enter mlths mgths race_rec risk[,1]
  <chr>        <dbl> <dbl> <dbl> <fct>       <dbl>
1 10000744      5.26     1     0 hispanic     1.61
```

```r
tail(e.long1[which.min(e.long1$rrisk),
             c("childid", "age_enter", "mlths", "mgths","race_rec", "risk")],
     n=20)
```

```
# A tibble: 1 x 6
  childid  age_enter mlths mgths race_rec risk[,1]
  <chr>        <dbl> <dbl> <dbl> <fct>       <dbl>
1 10000046      5.92     0     1 nhwhite     0.103
```

So, the first of these children has a risk score of `1.614` which means their risk was 61% higher that of the baseline child. Likewise, the lowest risk child had a risk score of `0.103` that means their score is almost 90% less than the baseline category.

Fitting the Cox model with survey design information

Now we fit the Cox model using full survey design. In the ECLS-K, I use the longitudinal weight for waves 1-5, as well as the associated psu and strata id's for the longitudinal data from these waves from the parents of the child, since no data from the child themselves are used in the outcome.

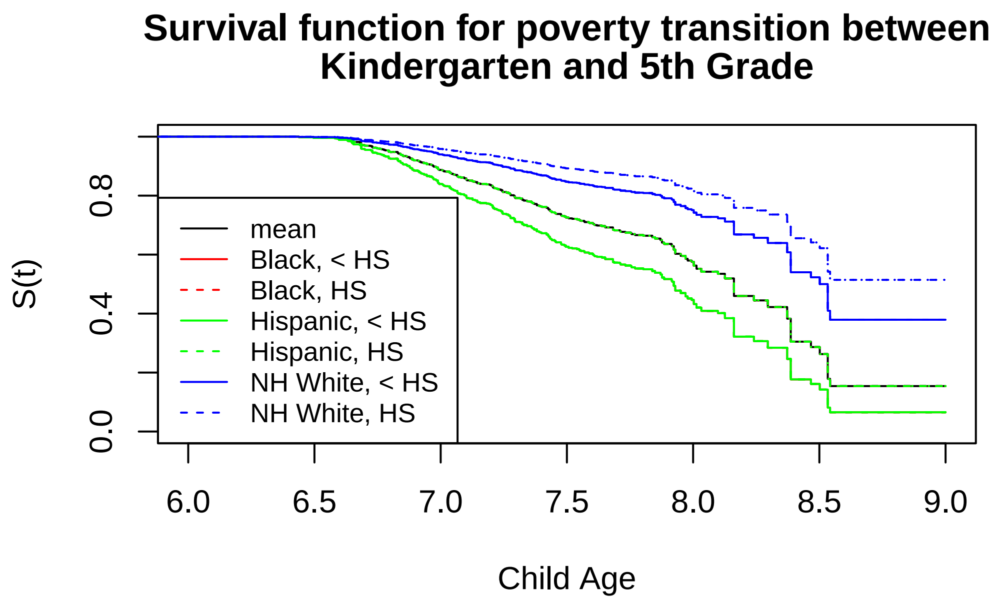
```r
e.long1 <- e.long1 %>%
  filter(complete.cases(w6c6p_6psu,w6c6p_6str, w6c6p_20))

des2<-svydesign(ids = ~w6c6p_6psu,
                strata = ~w6c6p_6str, weights=~w6c6p_20,
                data=e.long1, nest=T)


#Fit the model
fitl1<-svycoxph(Surv(time =age_enter,
                     time2 = age_exit,
                     event = povtran)~mlths+mgths+race_rec,
                design=des2)
summary(fitl1)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (123) clusters.
svydesign(ids = ~w6c6p_6psu, strata = ~w6c6p_6str, weights = ~w6c6p_20,
    data = e.long1, nest = T)

Call:
svycoxph(formula = Surv(time = age_enter, time2 = age_exit, event = povtran) ~
    mlths + mgths + race_rec, design = des2)

  n= 4084, number of events= 221

                      coef exp(coef)   se(coef)  robust se      z Pr(>|z|)
mlths            0.3773932  1.4584777  0.1823545  0.1724230  2.189  0.02861 *
mgths           -1.1042691  0.3314530  0.1511639  0.2024565 -5.454 4.92e-08 ***
race_recnhasian -0.6368146  0.5289747  0.3748378  0.3507131 -1.816  0.06941 .
race_recnhblack  0.0006955  1.0006957  0.2200090  0.2438301  0.003  0.99772
race_recnhwhite -1.0342483  0.3554935  0.1624512  0.1931174 -5.356 8.53e-08 ***
race_recother   -0.5845613  0.5573503  0.2771338  0.1838370 -3.180  0.00147 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
                  exp(coef) exp(-coef) lower .95 upper .95
mlths                1.4585     0.6856    1.0402    2.0449
mgths                0.3315     3.0170    0.2229    0.4929
race_recnhasian      0.5290     1.8904    0.2660    1.0519
race_recnhblack      1.0007     0.9993    0.6205    1.6138
race_recnhwhite      0.3555     2.8130    0.2435    0.5191
race_recother        0.5574     1.7942    0.3887    0.7991

Concordance= 0.738  (se = 0.023 )
Likelihood ratio test= NA  on 6 df,   p=NA
Wald test            = 117.1  on 6 df,   p=<2e-16
Score (logrank) test = NA  on 6 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
library(ggsurvfit)


plot(survfit(fitl1, conf.int = F),
     ylab="S(t)",
     xlab="Child Age",
     xlim=c(6, 9))

lines(survfit(fitl1, newdata = data.frame(mlths=1, mgths=0, race_rec="nhblack"),
              conf.int=F) ,col="red", lty=1)
lines(survfit(fitl1, newdata = data.frame(mlths=0, mgths=0, race_rec="nhblack"),
              conf.int=F) ,col="red", lty=2)

lines(survfit(fitl1, newdata = data.frame(mlths=1, mgths=0, race_rec="hispanic"),
              conf.int=F) ,col="green", lty=1)
lines(survfit(fitl1, newdata = data.frame(mlths=0, mgths=0, race_rec="hispanic"),
              conf.int=F) ,col="green", lty=2)

lines(survfit(fitl1, newdata = data.frame(mlths=1, mgths=0, race_rec="nhwhite"),
              conf.int=F) ,col="blue", lty=1)
lines(survfit(fitl1, newdata = data.frame(mlths=0, mgths=0, race_rec="nhwhite"),
              conf.int=F) ,col="blue", lty=2)
```

```r
title(main=c("Survival function for poverty transition between",
             "Kindergarten and 5th Grade"))
legend("bottomleft",
       legend=c("mean", "Black, < HS ", "Black, HS","Hispanic, < HS ",
                "Hispanic, HS","NH White, < HS ", "NH White, HS"),
       col=c(1,"red","red", "green","green", "blue", "blue"),
       lty=c(1,1,2,1,2,1,2), cex=.8)
```



Use time instead of age

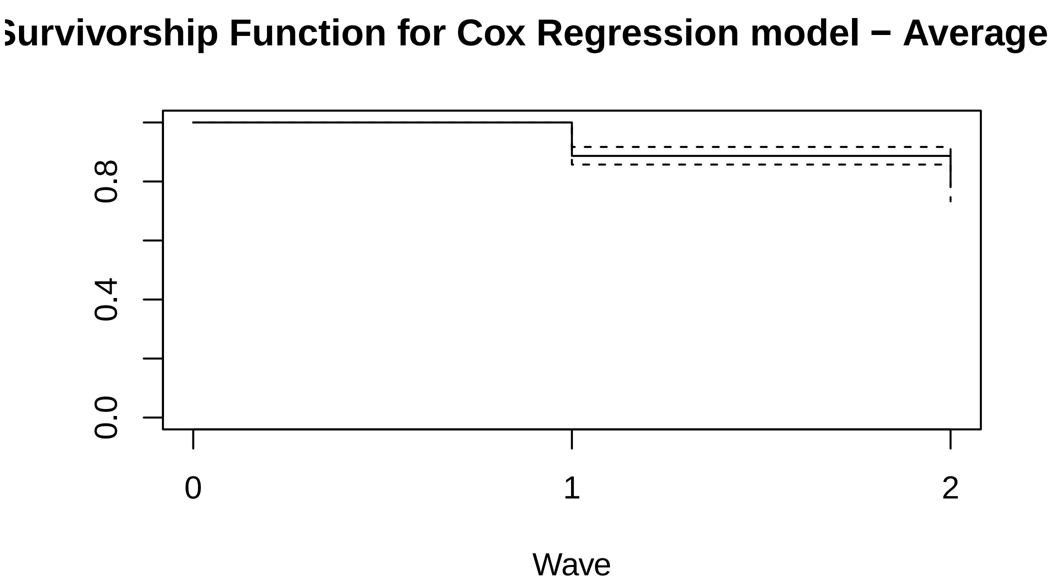
Next, I will use the `time` variable we created in `e.long` as the time axis. This model will not focus on the age of the children, but on the probability of experiencing the transition between waves.

```r
#Cox Model
#interval censored
fitl2<-coxreg(Surv(time = as.numeric(wave), event = povtran)~mlths+mgths+race_rec,
              data=e.long1)
summary(fitl2)
```

```
Covariate                Mean        Coef      Rel.Risk    S.E.     LR p
```

```
mlths                     0.060     0.364     1.440     0.188     0.055
mgths                     0.783    -1.163     0.312     0.160     0.000
race_rec                                                          0.000
        hispanic          0.227     0         1 (reference)
         nhasian          0.082    -0.707     0.493     0.299
         nhblack          0.066    -0.026     0.974     0.242
         nhwhite          0.555    -1.102     0.332     0.179
           other          0.071    -0.335     0.715     0.285

Events                        221
Total time at risk              6126
Max. log. likelihood          -1683.6
LR test statistic             196.05
Degrees of freedom            6
Overall p-value               0
```

```
plot(survfit(fitl2),
     xlab="Wave",xaxt="n",
     main="Survivorship Function for Cox Regression model - Average Child")
axis(1, at=c(0,1,2))
```



The model results (`Rel.Risk`) show that children with mom's who have less than a high school education have 2.1 times higher risk of going into poverty during this period, while children whose mother have more than a high school education are 67% less likely to enter poverty, compared to children whose mothers had a high school education. Likewise, Hispanic, Non-Hispanic black, Native American and Asian children all face higher risk of entering poverty, compared to Non-Hispanic whites.

Now we fit the Cox model using full survey design. In the ECLS-K, I use the longitudinal weight for waves 1-5, as well as the associated psu and strata id's for the longitudinal data from these waves from the parents of the child, since no data from the child themselves are used in the outcome.

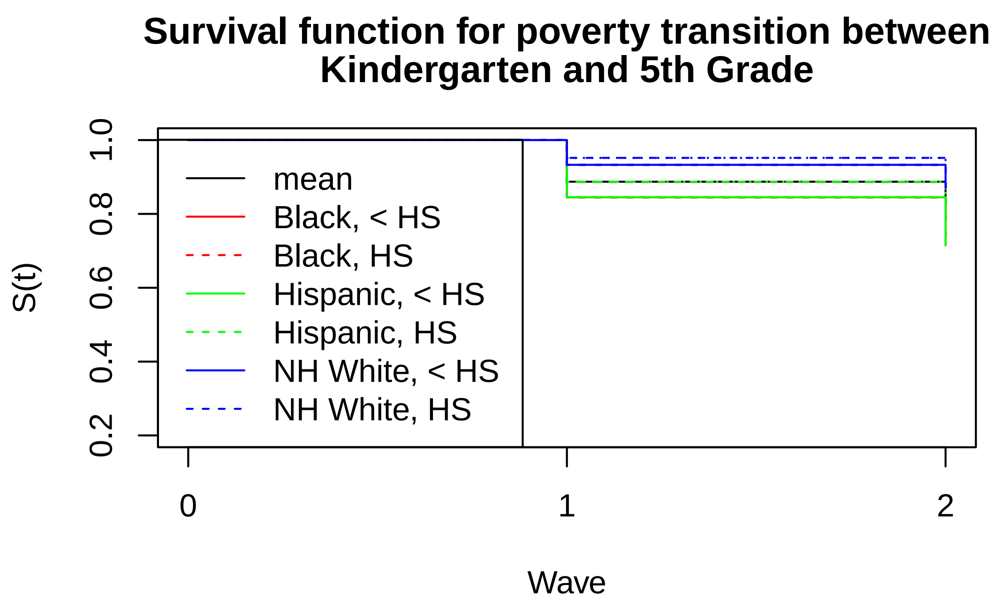
```
#Fit the model
fitl2s<-svycoxph(Surv(time = as.numeric(wave), event = povtran)~mlths+mgths+race_rec,
                 design=des2)
summary(fitl2s)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (123) clusters.
svydesign(ids = ~w6c6p_6psu, strata = ~w6c6p_6str, weights = ~w6c6p_20,
    data = e.long1, nest = T)


Call:
svycoxph(formula = Surv(time = as.numeric(wave), event = povtran) ~
    mlths + mgths + race_rec, design = des2)

  n= 4084, number of events= 221

                     coef exp(coef)  se(coef) robust se      z Pr(>|z|)
mlths            0.339348  1.404033  0.183717  0.184182  1.842 0.065407 .
mgths           -1.161198  0.313111  0.151212  0.202789 -5.726 1.03e-08 ***
race_recnhasian -0.641177  0.526672  0.375516  0.357763 -1.792 0.073104 .
race_recnhblack  0.002279  1.002282  0.220662  0.271874  0.008 0.993311
race_recnhwhite -0.884291  0.413007  0.164992  0.262730 -3.366 0.000763 ***
race_recother   -0.436814  0.646091  0.276626  0.225833 -1.934 0.053084 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                exp(coef) exp(-coef) lower .95 upper .95
mlths              1.4040     0.7122    0.9786    2.0144
mgths              0.3131     3.1938    0.2104    0.4659
race_recnhasian    0.5267     1.8987    0.2612    1.0619
race_recnhblack    1.0023     0.9977    0.5883    1.7077
```

```
race_recnhwhite    0.4130     2.4213    0.2468    0.6912
race_recother      0.6461     1.5478    0.4150    1.0058

Concordance= 0.729  (se = 0.022 )
Likelihood ratio test= NA  on 6 df,   p=NA
Wald test            = 90.13  on 6 df,   p=<2e-16
Score (logrank) test = NA  on 6 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(fitl2s, conf.int = F),
     ylab="S(t)", xlab="Wave", xaxt="n",
     ylim=c(.2,1))
axis(1, at=c(0,1,2))

lines(survfit(fitl2s,
              newdata = data.frame(mlths=1, mgths=0, race_rec="nhblack"),
              conf.int=F) ,
      col="red", lty=1)
lines(survfit(fitl2s,
              newdata = data.frame(mlths=0, mgths=0, race_rec="nhblack"),
              conf.int=F) ,
      col="red", lty=2)

lines(survfit(fitl2s,
              newdata = data.frame(mlths=1, mgths=0, race_rec="hispanic"),
              conf.int=F) ,
      col="green", lty=1)

lines(survfit(fitl2s,
              newdata = data.frame(mlths=0, mgths=0, race_rec="hispanic"),
              conf.int=F),
      col="green", lty=2)

lines(survfit(fitl2s,
              newdata = data.frame(mlths=1, mgths=0, race_rec="nhwhite"),
              conf.int=F) ,
      col="blue", lty=1)
lines(survfit(fitl2s,
              newdata = data.frame(mlths=0, mgths=0, race_rec="nhwhite"),
              conf.int=F) ,
```

```
       col="blue", lty=2)


title(main=c("Survival function for poverty transition between",
             "Kindergarten and 5th Grade"))
legend("bottomleft",
       legend=c("mean", "Black, < HS ", "Black, HS","Hispanic, < HS ",
                "Hispanic, HS","NH White, < HS ", "NH White, HS"),
       col=c(1,"red","red", "green","green", "blue", "blue"),
       lty=c(1,1,2,1,2,1,2))
```



We see similar results as we did in the age based analysis, but now, we are treating time discretely, and separating it from the child's age entirely.

DHS data example

```
library(haven)
#load the data
dat<-read_dta("../data/ZAIR71FL.DTA")
dat<-zap_labels(dat)
```

In the DHS individual recode file, information on every live birth is collected using a retrospective birth history survey mechanism.

Since our outcome is time between first and second birth, we must select as our risk set, only women who have had a first birth.

The `bidx` variable indexes the birth history and if `bidx_01` is not missing, then the woman should be at risk of having a second birth (i.e. she has had a first birth, i.e. `bidx_01==1`).

I also select only non-twin births (`b0 == 0`).

The DHS provides the dates of when each child was born in Century Month Codes.

To get the interval for women who *actually had* a second birth, that is the difference between the CMC for the first birth `b3_01` and the second birth `b3_02`, but for women who had not had a second birth by the time of the interview, the censored time between births is the difference between `b3_01` and `v008`, the date of the interview.

We have 6124 women who are at risk of a second birth.

```
sub<-dat %>%
  filter(bidx_01==1&b0_01==0)%>%
  transmute(CASEID=caseid,
                 int.cmc=v008,
                 fbir.cmc=b3_01,
                 sbir.cmc=b3_02,
                 marr.cmc=v509,
                 rural=v025,
                 educ=v106,
                 age = v012,
                 agec=cut(v012, breaks = seq(15,50,5), include.lowest=T),
                 partneredu=v701,
                 partnerage=v730,
                 weight=v005/1000000,
                 psu=v021,
                 strata=v022)%>%
  select(CASEID, int.cmc, fbir.cmc, sbir.cmc, marr.cmc, rural, educ,
         age, agec, partneredu, partnerage, weight, psu, strata)%>%
  mutate(agefb = (age - (int.cmc - fbir.cmc)/12))
```

Now I need to calculate the birth intervals, both observed and censored, and the event indicator (i.e. did the women *have* the second birth?)

```r
sub2<-sub%>%
  mutate(secbi = ifelse(is.na(sbir.cmc)==T,
                    int.cmc - fbir.cmc,
                    fbir.cmc - sbir.cmc),
         b2event = ifelse(is.na(sbir.cmc)==T,0,1))
```

Create covariates

Here, we create some predictor variables: Woman's education (secondary +, vs < secondary), Woman's age^2, Partner's education (> secondary school)

```r
sub2$educ.high<-ifelse(sub2$educ %in% c(2,3), 1, 0)
sub2$age2<-(sub2$age)^2
sub2$partnerhiedu<-ifelse(sub2$partneredu<3,0,
                          ifelse(sub2$partneredu%in%c(8,9),NA,1 ))

options(survey.lonely.psu = "adjust")
des<-svydesign(ids=~psu, strata=~strata,
               data=sub2[sub2$secbi>0,], weight=~weight )
```

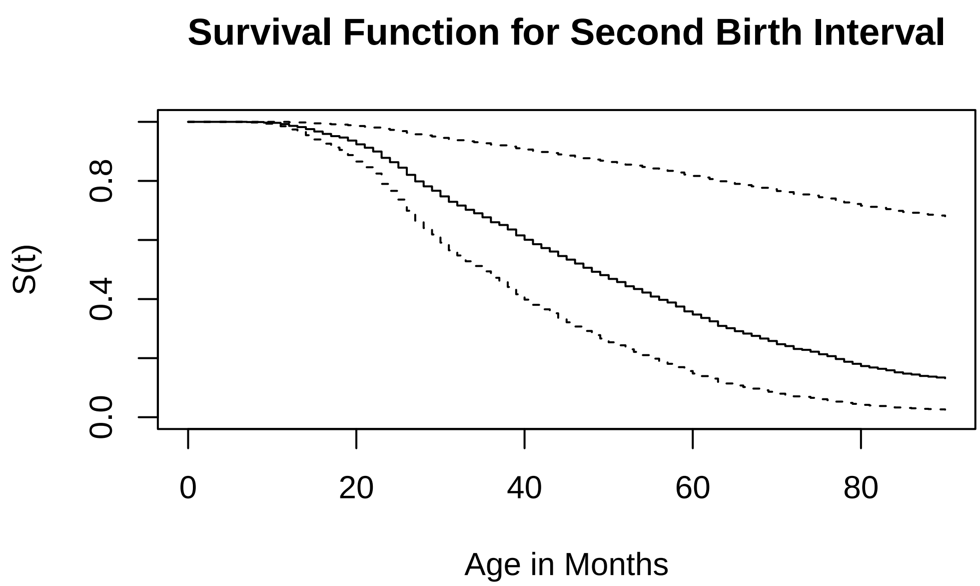
Fit the model

```r
#using coxph in survival library
fit.cox2<-coxph(Surv(secbi,b2event)~educ.high+partnerhiedu+agec ,
                data=sub2)
summary(fit.cox2)
```

```
Call:
coxph(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu + 
    agec, data = sub2)

  n= 2492, number of events= 1980 
   (3547 observations deleted due to missingness)

                 coef exp(coef) se(coef)      z Pr(>|z|)   
educ.high    -0.18901   0.82778  0.06246 -3.026  0.00248 **
partnerhiedu  0.01658   1.01672  0.07039  0.236  0.81377   
agec(20,25]  -0.16894   0.84456  0.42163 -0.401  0.68865   
agec(25,30]  -0.14794   0.86249  0.41318 -0.358  0.72031   
```

```
agec(30,35]  -0.38007   0.68382  0.41221 -0.922  0.35652
agec(35,40]  -0.45227   0.63618  0.41278 -1.096  0.27323
agec(40,45]  -0.48704   0.61444  0.41305 -1.179  0.23835
agec(45,50]  -0.53934   0.58314  0.41395 -1.303  0.19261
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

             exp(coef) exp(-coef) lower .95 upper .95
educ.high       0.8278     1.2081    0.7324    0.9356
partnerhiedu    1.0167     0.9836    0.8857    1.1671
agec(20,25]     0.8446     1.1840    0.3696    1.9298
agec(25,30]     0.8625     1.1594    0.3838    1.9384
agec(30,35]     0.6838     1.4624    0.3048    1.5340
agec(35,40]     0.6362     1.5719    0.2833    1.4287
agec(40,45]     0.6144     1.6275    0.2735    1.3806
agec(45,50]     0.5831     1.7149    0.2591    1.3126

Concordance= 0.542  (se = 0.007 )
Likelihood ratio test= 36.7  on 8 df,   p=1e-05
Wald test            = 37.91  on 8 df,   p=8e-06
Score (logrank) test = 38.13  on 8 df,   p=7e-06
```

```r
plot(survfit(fit.cox2),   xlim=c(0,90),
     ylab="S(t)", xlab="Age in Months")
title(main="Survival Function for Second Birth Interval")
```



Use survey design

```
des<-svydesign(ids=~psu,
               strata = ~strata ,
               weights=~weight,
               data=sub2)

cox.s<-svycoxph(Surv(secbi,b2event)~educ.high+partnerhiedu+agec,
                design=des)
summary(cox.s)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (714) clusters.
svydesign(ids = ~psu, strata = ~strata, weights = ~weight, data = sub2)

Call:
svycoxph(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
    agec, design = des)

  n= 2492, number of events= 1980
   (3547 observations deleted due to missingness)
```

```
                  coef exp(coef)  se(coef) robust se      z Pr(>|z|)  
educ.high    -0.195916  0.822081  0.065699  0.094899 -2.064    0.039 *
partnerhiedu  0.056592  1.058224  0.063986  0.086605  0.653    0.513  
agec(20,25]   0.346577  1.414219  0.493885  0.513147  0.675    0.499  
agec(25,30]   0.273502  1.314560  0.487772  0.513153  0.533    0.594  
agec(30,35]   0.062153  1.064125  0.486865  0.500126  0.124    0.901  
agec(35,40]  -0.002668  0.997336  0.487335  0.501970 -0.005    0.996  
agec(40,45]  -0.033562  0.966995  0.487757  0.506548 -0.066    0.947  
agec(45,50]  -0.082227  0.921063  0.488567  0.508425 -0.162    0.872  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

             exp(coef) exp(-coef) lower .95 upper .95
educ.high       0.8221     1.2164    0.6826    0.9901
partnerhiedu    1.0582     0.9450    0.8930    1.2540
agec(20,25]     1.4142     0.7071    0.5173    3.8664
agec(25,30]     1.3146     0.7607    0.4808    3.5940
agec(30,35]     1.0641     0.9397    0.3993    2.8360
agec(35,40]     0.9973     1.0027    0.3729    2.6676
agec(40,45]     0.9670     1.0341    0.3583    2.6097
agec(45,50]     0.9211     1.0857    0.3400    2.4949

Concordance= 0.538  (se = 0.009 )
Likelihood ratio test= NA  on 8 df,   p=NA
Wald test            = 24.83  on 8 df,   p=0.002
Score (logrank) test = NA  on 8 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).
```

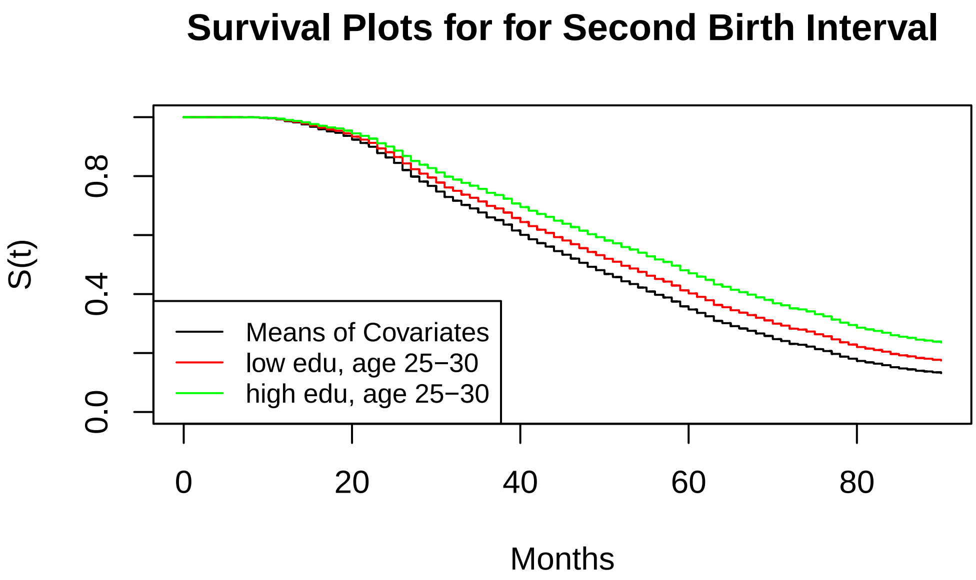
```r
dat<-expand.grid(educ.high=c(0,1),
                 partnerhiedu=c(0,1),
                 agec=c("(25,30]", "(40,45]"),
                 b2event=1)
head(dat)
```

```
  educ.high partnerhiedu    agec b2event
1         0            0 (25,30]       1
2         1            0 (25,30]       1
3         0            1 (25,30]       1
```

```
4          1            1 (25,30]       1
5          0            0 (40,45]       1
6          1            0 (40,45]       1
```

Plot some survival function estimates for various types of children

```
plot(survfit(fit.cox2, conf.int = F),
     xlim=c(0, 90),
     ylab="S(t)",
     xlab="Months")
title (main = "Survival Plots for for Second Birth Interval")

lines(survfit(fit.cox2,
              newdata=data.frame(educ.high=0, partnerhiedu=0, agec="(25,30]"),
              conf.int = F), col="red")
lines(survfit(fit.cox2,
              newdata=data.frame(educ.high=1, partnerhiedu=0, agec="(25,30]"),
              conf.int = F), col="green")

legend("bottomleft",
       legend=c("Means of Covariates",
                "low edu, age 25-30",
                "high edu, age 25-30"),
       lty=1, col=c(1,"red", "green"), cex=.8)
```

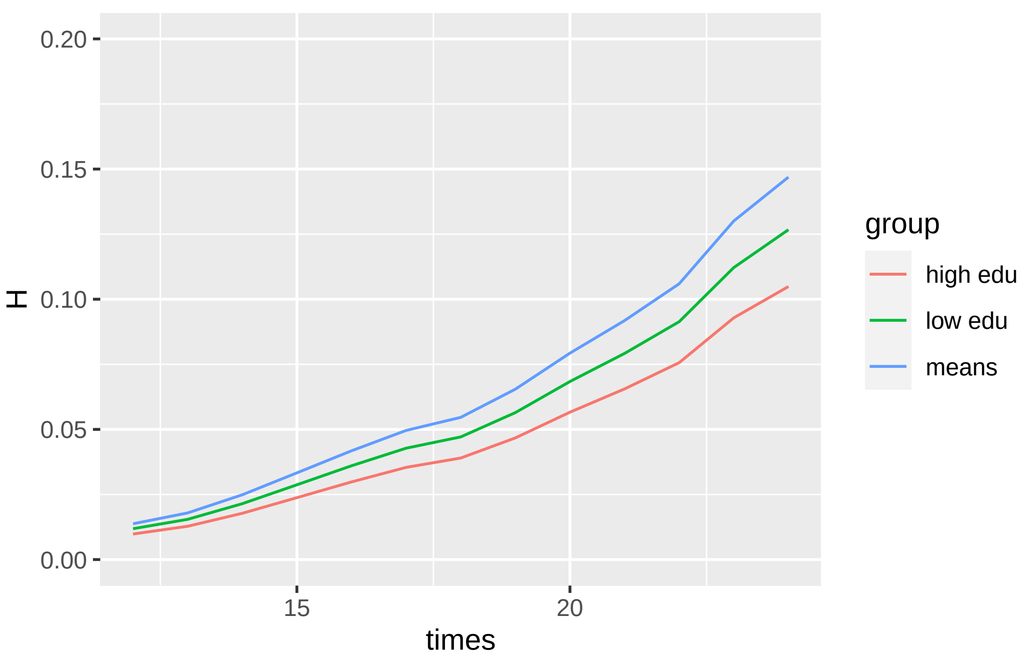


Now we look at some more plots we can examine from the models

```
#Now we look at the cumulative hazard functions
sf1<-survfit(fit.cox2)

sf2<-survfit(fit.cox2,
             newdata=data.frame(educ.high=0,
                                partnerhiedu=0,
                                agec="(25,30]"))

sf3<-survfit(fit.cox2,
             newdata=data.frame(educ.high=1,
                                partnerhiedu=0,
                                agec="(25,30]"))



H1<--log(sf1$surv)
H2<--log(sf2$surv)
H3<--log(sf3$surv)

test<-data.frame(H=c(H1, H2, H3),
```

```r
                   group=c(rep("means",length(H1)),
                           rep( "low edu", length(H1)),
                           rep("high edu",length(H1))),
                   times=sf1$time)



test%>%
   ggplot(aes(x=times, y=H))+
  geom_line(aes(group=group, color=group))+
  xlim(12,24)+
  ylim(0, .2)
```

```
Warning: Removed 738 row(s) containing missing values (geom_path).
```




```r
#and the hazard function
times <- sf1$time
hs1<-loess(diff(c(0,H1))~times, degree=1, span=.25)
hs2<-loess(diff(c(0,H2))~times, degree=1, span=.25)
hs3<-loess(diff(c(0,H3))~times, degree=1, span=.25)
```

```
plot(predict(hs1)~times,type="l", ylab="smoothed h(t)", xlab="Months",
     ylim=c(0, .04))
title(main="Smoothed hazard plots")
lines(predict(hs2)~times, type="l", col="red")
lines(predict(hs3)~times, type="l", col="green")
legend("topright",
       legend=c("Means of Covariates", "low edu, age 25-30", "high edu, age 25-30"),
       lty=1, col=c(1,"red", "green"), cex=.8)
```