DEM 7223 - Event History Analysis - Cox Proportional Hazards Model Part 2

Corey S. Sparks, PhD

29 September, 2022

Review of Cox Regression Assumptions

- Although the Cox PH model offers an attractive alternative to parametric models, especially when ties are present, the assumptions of the model need to be assessed
- The primary assumption we are concerned with are:
- The time-constant covariate effect, i.e. the effect does not vary with time
- Grambsch and Therneau (1994) derived a method for checking the proportionality assumption for the Cox model using the residuals from the Cox model fit
- First we need to look at the various kinds of residuals from the Cox model and their properties

Cox Model Residuals

The basic principle for the construction of residuals is:

- Observed value – Predicted value
- This lets you get an idea of how well you are modeling your data with your covariates
- In hazard models, this principle is a bit more difficult because of censoring in the data, but despite this, we can define several types of model residuals and use them to diagnose problems with the model.

Schoenfeld Residuals

- These are used to test the proportionality assumption of the model
- If there are p covariates and n observations, with observed duration, censoring indicators and covariates, then the Schoenfeld residual is defined as the observed - expected value of a covariate at a particular *failure time*
- Plotting these residuals versus time should show any time dependency in the covariate, which violates the proportionality assumption of the model.

Martingale residuals

- These residuals can be thought of as the difference between an expected event occurring and an actual event occurring
- These use the censoring indicator for each observation and the estimate of the cumulative hazard function
- These residuals are given by:

$$M_i(t) = \delta_i(t) – H_i(t)$$

- Where $H_i(t)$ is the cumulative hazard function
- This residual is derived from counting process theory and represents the difference between the observed count of failures at any time, minus those that are predicted by the cumulative hazard function.
- Martingale residuals are also useful for assessing the functional form of a covariate
- Meaning, is the effect linear or quadratic
- A plot of the martingale residuals against the values of the covariate will indicate if the covariate is being modeled correctly
- If a line fit to these residuals is a straight line, then the covariate has been modeled effectively, if it is curvilinear, you may need to enter the covariate as a quadratic
- This is usually not a problem for binary covariates

Testing non-proportional effects

- As mentioned before, Grambsch and Therneau (1994) derived a method for checking the proportionality assumption for the Cox model using the residuals from the Cox model fit
- Now that we know what these residuals are, we can see the test
- Their test is equivalent to regressing the Schoenfeld residual on time for each covariate
- If there is a significant trend (correlation) between the residual and time, then non-proportionality is likely.

Model stratification

- One common method for dealing with non proportionality of hazards is via model stratification.
- If one of the covariates exhibits non-proportionality we can re-specify the model so that each group will have its own baseline hazard rate
- The effect of the other covariates in the model is assumed to behave the same in both groups!
- This creates the model:

$$h_{is}(t) = h_{0s} exp(x'\beta)$$

- which allows for different baseline hazard rates for each of the s strata, which should control for their unequal hazards of experiencing the event.
- This procedure is slightly different than fitting separate models for each level of the stratification variable
- This method will allow the effects of the covariates to vary between strata
- Unfortunately, when we split the data and run separate models for each level, we lose the ability to discuss "between level" effects, since each analysis is run on a different sample

Data examples

This example will illustrate how to examine the fit of the Cox Proportional hazards model to a discrete-time (longitudinal) data set and examine various model diagnostics to evaluate the overall model fit. The data example uses data from the ECLS-K. Specifically, we will examine the transition into poverty between kindergarten and third grade.

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

```r
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]


# 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='hisp
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")
```

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.

NOTE I need to remove any children who are already in poverty age 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 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_6s
               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()%>%
  filter(complete.cases(age_enter, age_exit, povtran,
                        mlths, mgths, race_rec,w6c6p_6psu))
```

Construct survey design and fit basic Cox model

Now we fit the Cox model using full survey design. In the ECLS-K, I use the longitudinal weight for waves 1-7, 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.

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

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+race_rec, de
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 + race_rec, design = des2)

  n= 4084, number of events= 221

                   coef exp(coef) se(coef) robust se      z Pr(>|z|)
mlths            1.0034    2.7275   0.1715    0.1669  6.014 1.81e-09 ***
race_recnhasian -0.8670    0.4202   0.3735    0.3624 -2.393   0.0167 *
race_recnhblack -0.1382    0.8710   0.2200    0.2475 -0.558   0.5767
race_recnhwhite -1.3062    0.2709   0.1586    0.2075 -6.296 3.06e-10 ***
race_recother   -0.8036    0.4477   0.2761    0.1981 -4.056 5.00e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                exp(coef) exp(-coef) lower .95 upper .95
mlths              2.7275     0.3666    1.9667    3.7826
race_recnhasian    0.4202     2.3797    0.2065    0.8549
```

```
race_recnhblack    0.8710     1.1482    0.5362    1.4147
race_recnhwhite    0.2709     3.6921    0.1804    0.4068
race_recother      0.4477     2.2335    0.3036    0.6602

Concordance= 0.695  (se = 0.024 )
Likelihood ratio test= NA  on 5 df,   p=NA
Wald test            = 115.7  on 5 df,   p=<2e-16
Score (logrank) test = NA  on 5 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).
```

Model Residuals

There are several types of residuals for the Cox model, and they are used for different purposes.

First, we will extract the *Shoenfeld* residuals, which are useful for examining non-proportional hazards with respect to time. This means that the covariate effect could exhibit time-dependency.

First we extract the residuals from the model, then we fit a linear model to the residual and the observed (uncensored) failure times

WE DO NOT WANT TO SEE A SIGNIFICANT MODEL HERE!!!!!

that would indicate dependence between the residual and outcome, or *non-proportionality*, similar to doing a test for heteroskedasticity in OLS

```
schoenresid<-resid(fitl1, type="schoenfeld")

fit.sr<-lm(schoenresid~des2$variables$age_enter[des2$variables$povtran==1])

fit.sr%>%
  broom::tidy()%>%
  filter(term != "(Intercept)")%>%
  select(response, estimate, statistic, p.value)
```

```
# A tibble: 5 x 4
  response        estimate statistic p.value
  <chr>              <dbl>     <dbl>   <dbl>
```

```
1 mlths           -0.0114     -0.318   0.751
2 race_recnhasian -0.0140     -0.705   0.481
3 race_recnhblack -0.0317     -1.25    0.212
4 race_recnhwhite  0.00812     0.221   0.825
5 race_recother    0.0312      1.47    0.142
```

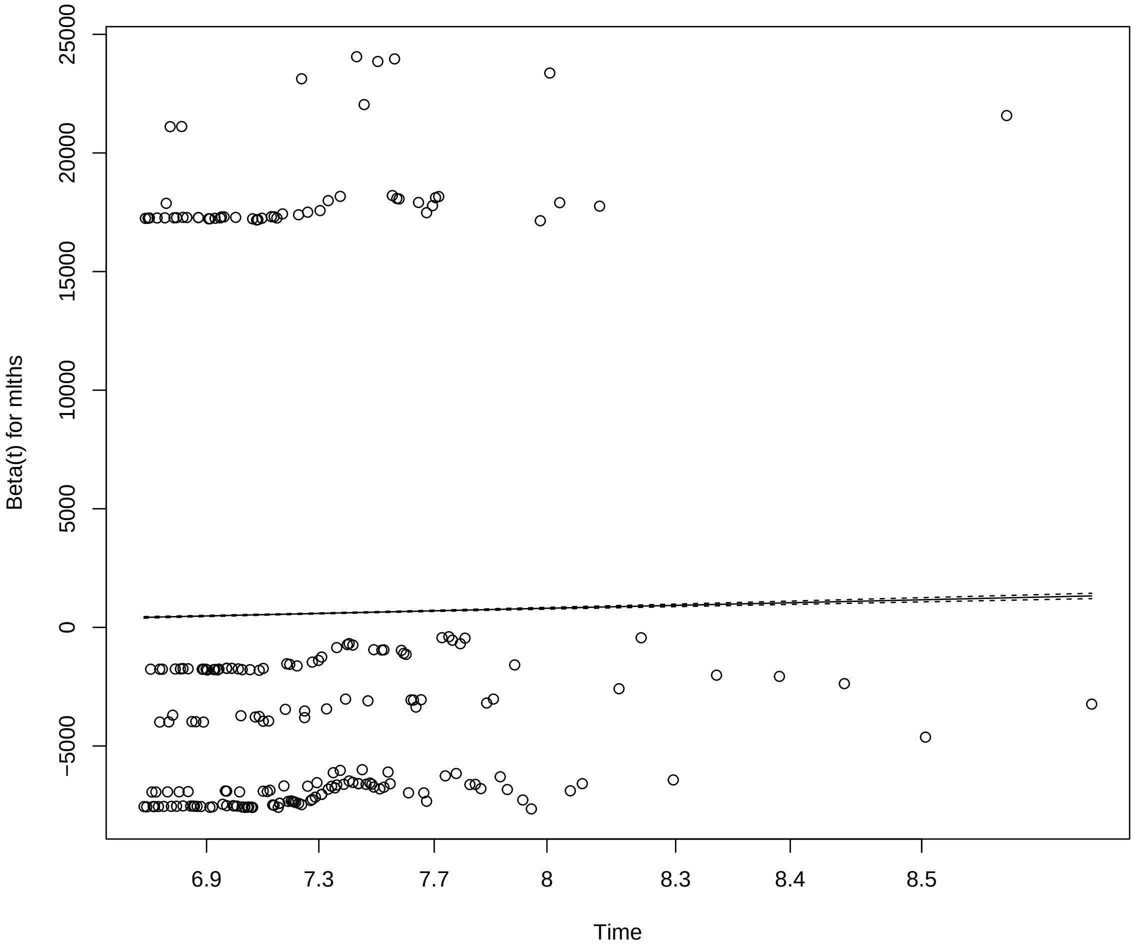
From these results, it appears that none of the variables are correlated with the timing of transition. This is what you want to see

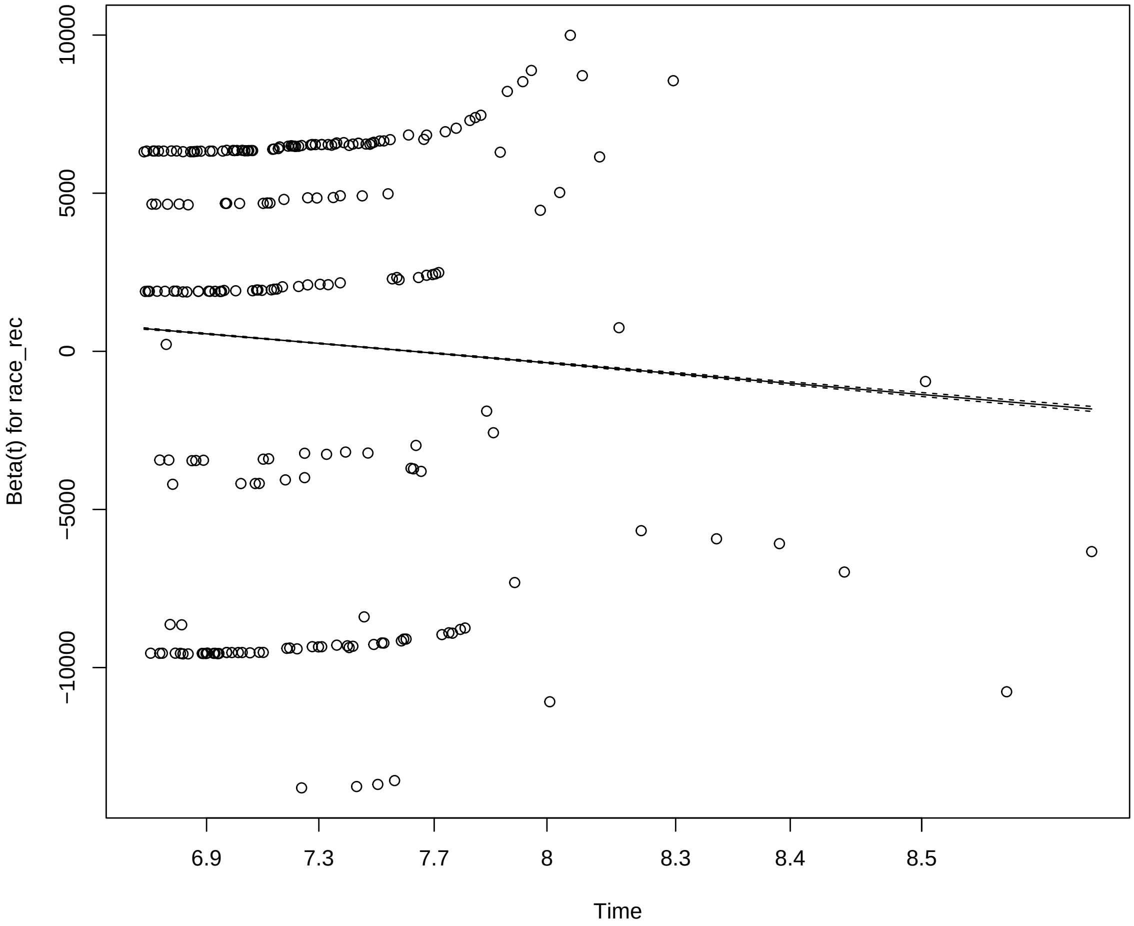
We can also get a formal test using weighted residuals in a nice pre-rolled form with a plot, a la Grambsch and Therneau (1994) :

```
fit.test<-cox.zph(fitl1)
fit.test
```

```
            chisq df    p
mlths    0.000479  1 0.98
race_rec 0.000499  4 1.00
GLOBAL   0.000897  5 1.00
```

```
plot(fit.test, df=2)
```

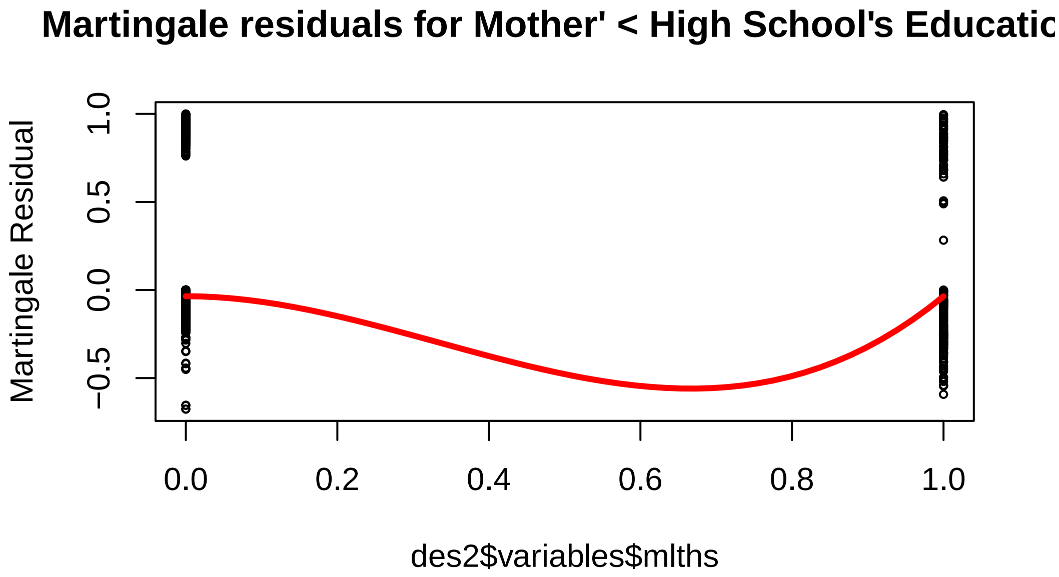

25000
20000
15000
10000
5000
0
−5000
Beta(t) for mlths
6.9
7.3
7.7
8
8.3
8.4
8.5
Time



Here, we see the same result, with no significant relationship detected by the formal test. **This is what you want to see**.

Next we examine Martingale residuals. Martingale residuals are also useful for assessing the functional form of a covariate. A plot of the martingale residuals against the values of the covariate will indicate if the covariate is being modeled correctly, i.e. linearly in the Cox model. If a line fit to these residuals is a straight line, then the covariate has been modeled effectively, if it is curvilinear, you may need to enter the covariate as a quadratic, although this is not commonly a problem for dummy variables.

```
#extract Martingale residuals
res.mar<-resid(fit11, type="martingale")

#plot vs maternal education
scatter.smooth(des2$variables$mlths, res.mar,degree = 2,
               span = 1, ylab="Martingale Residual",
               col=1,  cex=.5, lpars=list(col = "red", lwd = 3))
title(main="Martingale residuals for Mother' < High School's Education")
```



Which shows nothing in the way of non-linearity in this case.

Stratification

Above, we observed evidence of non-proportional effects by education. There are a few standard ways of dealing with this in practice. The first is *stratification* of the model by the offending predictor. If one of the covariates exhibits non-proportionality we can re-specify the model so that each group will have its own baseline hazard rate. This is direct enough to do by using the `strata()` function within a model. This is of best use when a covariate is categorical, and not of direct importance for our model (i.e. a control variable).

```
fitl2<-svycoxph(Surv(time = age_enter, time2 = age_exit, event = povtran)~race_rec+strata(
                design=des2)
summary(fitl2)
```

```
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) ~
    race_rec + strata(mlths), design = des2)

  n= 4084, number of events= 221

                   coef exp(coef) se(coef) robust se      z Pr(>|z|)
race_recnhasian -0.8614    0.4226   0.3737    0.3643 -2.364 0.018059 *
race_recnhblack -0.1583    0.8536   0.2207    0.2549 -0.621 0.534452
race_recnhwhite -1.3318    0.2640   0.1624    0.2204 -6.044 1.51e-09 ***
race_recother   -0.7997    0.4495   0.2762    0.2057 -3.888 0.000101 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                exp(coef) exp(-coef) lower .95 upper .95
race_recnhasian    0.4226      2.366    0.2069    0.8630
race_recnhblack    0.8536      1.172    0.5179    1.4066
race_recnhwhite    0.2640      3.788    0.1714    0.4066
race_recother      0.4495      2.225    0.3003    0.6726

Concordance= 0.659  (se = 0.026 )
Likelihood ratio test= NA  on 4 df,   p=NA
Wald test            = 50.52  on 4 df,   p=3e-10
Score (logrank) test = NA  on 4 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).
```

Non-proportional effects with time

We can also include a time by covariate interaction term to model directly any time-dependence in the covariate effect. Different people say to do different things, some advocate for simply interacting time with the covariate, others say use a nonlinear function of time, e.g. log(time) * the covariate, others say use time-1 * covariate, which is called the "heavy side function", according to Mills.

In this example, time is so limited that it doesn't make sense to do this.

ANOVA like tests for factors

You can use the `regTermTest()` function in the `survey()` package to do omnibus tests for variation across a factor variable.

```r
fit3<-svycoxph(Surv(time = age_enter, time2=age_exit, event = povtran)~mlths+race_rec,
           design=des2)
summary(fit3)
```

```
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 + race_rec, design = des2)

  n= 4084, number of events= 221

                   coef exp(coef) se(coef) robust se      z Pr(>|z|)
mlths            1.0034    2.7275   0.1715    0.1669  6.014 1.81e-09 ***
race_recnhasian -0.8670    0.4202   0.3735    0.3624 -2.393   0.0167 *
race_recnhblack -0.1382    0.8710   0.2200    0.2475 -0.558   0.5767
race_recnhwhite -1.3062    0.2709   0.1586    0.2075 -6.296 3.06e-10 ***
race_recother   -0.8036    0.4477   0.2761    0.1981 -4.056 5.00e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                exp(coef) exp(-coef) lower .95 upper .95
mlths              2.7275     0.3666    1.9667    3.7826
race_recnhasian    0.4202     2.3797    0.2065    0.8549
race_recnhblack    0.8710     1.1482    0.5362    1.4147
race_recnhwhite    0.2709     3.6921    0.1804    0.4068
race_recother      0.4477     2.2335    0.3036    0.6602
```

```
Concordance= 0.695  (se = 0.024 )
Likelihood ratio test= NA  on 5 df,   p=NA
Wald test            = 115.7  on 5 df,   p=<2e-16
Score (logrank) test = NA  on 5 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).
```

```
regTermTest(fit3, ~mlths, method="LRT")
```

```
Working (Rao-Scott+F) LRT for mlths
 in svycoxph(formula = Surv(time = age_enter, time2 = age_exit, event = povtran) ~
    mlths + race_rec, design = des2)
Working 2logLR =  22.79896 p= 1.0015e-05
df=1;  denominator df= 72
```

```
#regTermTest(fit3, ~race_rec, method="LRT")
```

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
  mutate(agefb = (age - (int.cmc - fbir.cmc)/12))%>%
  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)

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

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

sub<- sub%>%
```

```r
  filter(complete.cases(secbi, partnerhiedu, educ.high, agefb))

des<-svydesign(ids=~psu, strata=~strata,
               data=sub[sub$secbi>0,],
               weight=~weight )
```

Fit the model

```r
#use survey design

cox.s<-svycoxph(Surv(secbi,b2event)~educ.high+partnerhiedu+agefb+age2,
                design=des)
summary(cox.s)
```

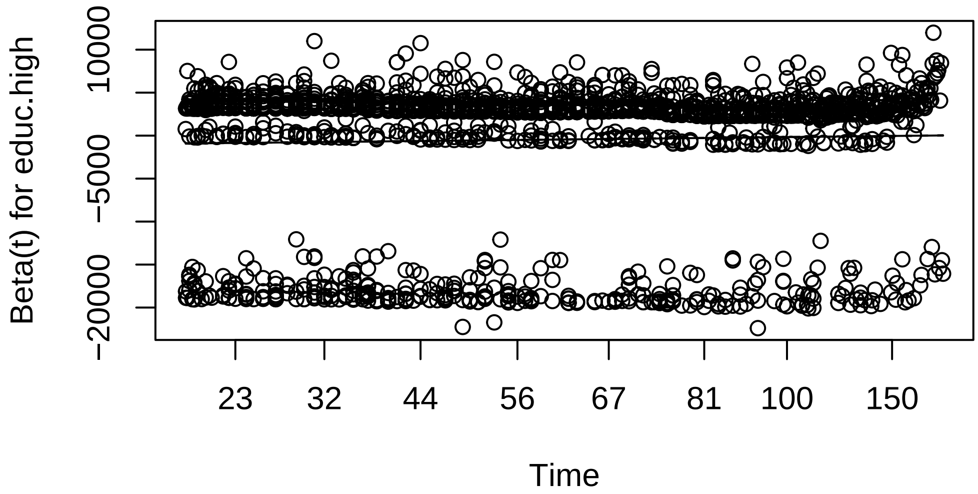
```
Stratified 1 - level Cluster Sampling design (with replacement)
With (670) clusters.
svydesign(ids = ~psu, strata = ~strata, data = sub[sub$secbi >
    0, ], weight = ~weight)


Call:
svycoxph(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
    agefb + age2, design = des)

  n= 2492, number of events= 1980

                   coef  exp(coef)   se(coef)  robust se       z Pr(>|z|)
educ.high    -0.2493914  0.7792749  0.0670396  0.0878240  -2.840  0.00452 **
partnerhiedu  0.0536853  1.0551525  0.0663283  0.0886441   0.606  0.54476
agefb         0.4995361  1.6479565  0.0348747  0.0423061  11.808  < 2e-16 ***
age2         -0.0079153  0.9921159  0.0005905  0.0007076 -11.187  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

             exp(coef) exp(-coef) lower .95 upper .95
educ.high       0.7793     1.2832    0.6560    0.9256
partnerhiedu    1.0552     0.9477    0.8869    1.2554
agefb           1.6480     0.6068    1.5168    1.7904
age2            0.9921     1.0079    0.9907    0.9935

Concordance= 0.55  (se = 0.01 )
```

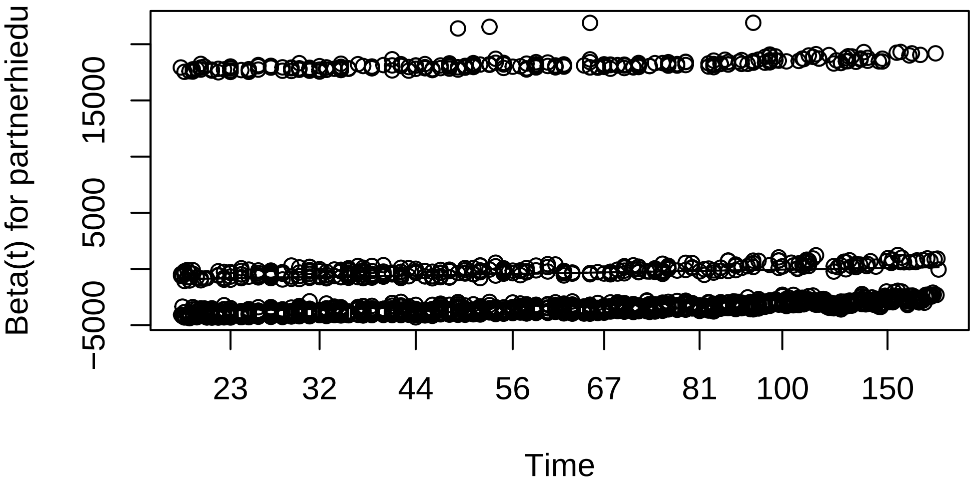
```
Likelihood ratio test= NA  on 4 df,   p=NA
Wald test            = 165.1  on 4 df,   p=<2e-16
Score (logrank) test = NA  on 4 df,   p=NA

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

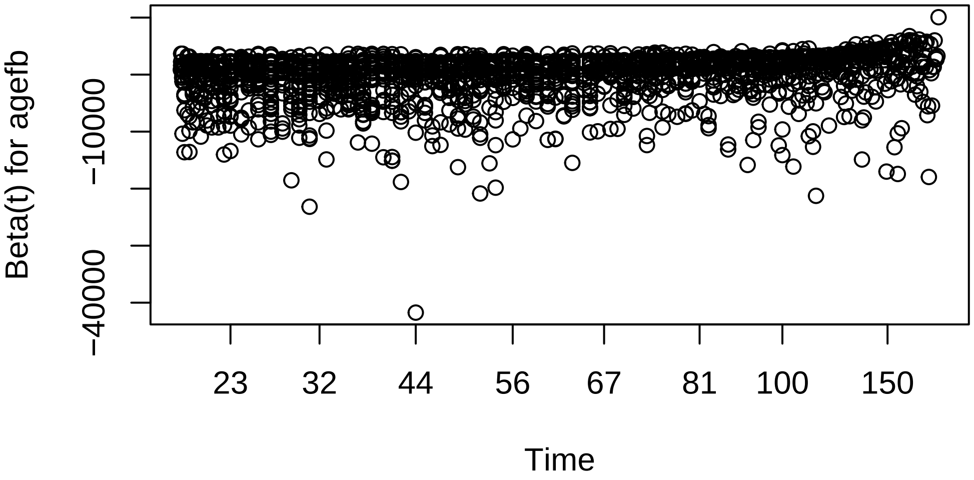
```
#Schoenfeld test
fit.test<-cox.zph(cox.s)
fit.test
```

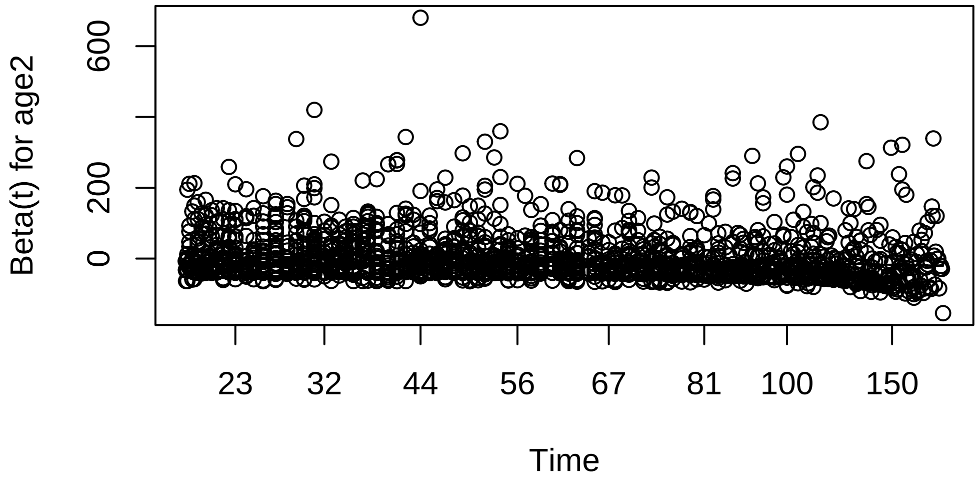
```
                chisq df    p
educ.high    0.001026  1 0.97
partnerhiedu 0.000798  1 0.98
agefb        0.060697  1 0.81
age2         0.052766  1 0.82
GLOBAL       0.077515  4 1.00
```

```
plot(fit.test, df=2)
```

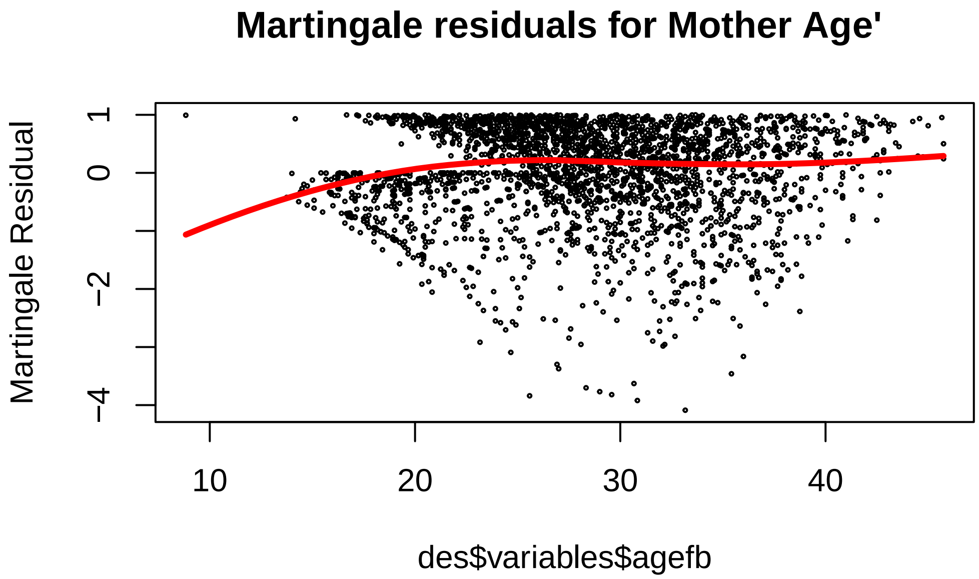



Beta(t) for partnerhiedu
−5000
5000
15000
23
32
44
56
67
81
100
150
Time


Beta(t) for agefb
−40000
−10000
23
32
44
56
67
81
100
150
Time



```
#martingale residuals
#extract Martingale residuals
res.mar<-resid(cox.s, type="martingale")

#plot vs maternal age
scatter.smooth(des$variables$agefb, res.mar,
               degree = 2,
               span = 1, ylab="Martingale Residual",
               col=1,  cex=.25, lpars=list(col = "red",
                                           lwd = 3))
title(main="Martingale residuals for Mother Age' ")
```



Non-proportional effects with time

We can also include a time by covariate interaction term to model directly any time-dependence in the covariate effect. Different people say to do different things, some advocate for simply interacting time with the covariate, others say use a nonlinear function of time, e.g. log(time) ∗ the covariate, others say use time-1 ∗ covariate, which is called the "heavy side function", according to Mills. Mills cites Allison, in saying that, to interpret the heavy side function, you go with the rule : "If β_2 is positive, then the effect of the covariate x increases over time, while if β_2 is negative, the effect of x decreases over time."

```
sub.split<-survSplit(Surv(secbi, b2event)~.,
                     data= sub[sub$secbi>0,], cut=36, episode = "timegroup")
sub.split<-sub.split[order(sub.split$CASEID, sub.split$timegroup),]

sub.split$hv1<-sub.split$agefb*(1-sub.split$timegroup)
sub.split$hv2<-sub.split$agefb*(sub.split$timegroup)

head(sub.split, n=20)
```

```
            CASEID int.cmc fbir.cmc sbir.cmc marr.cmc rural educ age    agec
1         1  13  2    1400     1204       NA     1333     1    1  41 (40,45]
2         1  13  2    1400     1204       NA     1333     1    1  41 (40,45]
3         1  36  2    1402     1351       NA     1372     1    2  27 (25,30]
4         1  36  2    1402     1351       NA     1372     1    2  27 (25,30]
5         2  29  2    1401     1088       NA     1240     2    2  45 (40,45]
6         2  29  2    1401     1088       NA     1240     2    2  45 (40,45]
7         2  29  3    1401     1286     1221     1383     2    1  36 (35,40]
8         2  29  3    1401     1286     1221     1383     2    1  36 (35,40]
9         2  34  4    1401     1371     1339     1391     2    2  26 (25,30]
10        2  35  2    1401     1146     1076     1145     2    1  47 (45,50]
11        2  35  2    1401     1146     1076     1145     2    1  47 (45,50]
12        2  59  2    1401     1168     1128     1132     2    1  47 (45,50]
13        2  59  2    1401     1168     1128     1132     2    1  47 (45,50]
14        2  65  2    1401     1275     1175     1164     2    1  39 (35,40]
15        2  65  2    1401     1275     1175     1164     2    1  39 (35,40]
16        2  79  1    1401     1301     1115     1252     2    3  48 (45,50]
17        2  79  1    1401     1301     1115     1252     2    3  48 (45,50]
18        2  96 11    1401     1401     1381     1345     2    2  23 (20,25]
19        3   2  2    1402     1386     1302     1132     1    2  39 (35,40]
20        3   2  2    1402     1386     1302     1132     1    2  39 (35,40]
   partneredu partnerage   weight psu strata    agefb educ.high      age2
1           1         44 1.396726   1     18 24.66667         0  608.4444
2           1         44 1.396726   1     18 24.66667         0  608.4444
3           2         31 1.396726   1     18 22.75000         1  517.5625
4           2         31 1.396726   1     18 22.75000         1  517.5625
5           0         48 1.206331   2     13 18.91667         1  357.8403
6           0         48 1.206331   2     13 18.91667         1  357.8403
7           2         52 1.206331   2     13 26.41667         0  697.8403
8           2         52 1.206331   2     13 26.41667         0  697.8403
9           2         33 1.206331   2     13 23.50000         1  552.2500
10          1         55 1.206331   2     13 25.75000         0  663.0625
11          1         55 1.206331   2     13 25.75000         0  663.0625
12          0         57 1.206331   2     13 27.58333         0  760.8403
13          0         57 1.206331   2     13 27.58333         0  760.8403
14          1         47 1.206331   2     13 28.50000         0  812.2500
15          1         47 1.206331   2     13 28.50000         0  812.2500
16          3         52 1.206331   2     13 39.66667         1 1573.4444
17          3         52 1.206331   2     13 39.66667         1 1573.4444
18          2         34 1.206331   2     13 23.00000         1  529.0000
19          1         42 0.560815   3      9 37.66667         1 1418.7778
20          1         42 0.560815   3      9 37.66667         1 1418.7778
   partnerhiedu tstart secbi b2event timegroup       hv1      hv2
1             0      0    36       0         1   0.00000 24.66667
2             0     36   196       0         2 -24.66667 49.33333
```

```
3             0      0     36       0         1   0.00000 22.75000
4             0     36     51       0         2 -22.75000 45.50000
5             0      0     36       0         1   0.00000 18.91667
6             0     36    313       0         2 -18.91667 37.83333
7             0      0     36       0         1   0.00000 26.41667
8             0     36     65       1         2 -26.41667 52.83333
9             0      0     32       1         1   0.00000 23.50000
10            0      0     36       0         1   0.00000 25.75000
11            0     36     70       1         2 -25.75000 51.50000
12            0      0     36       0         1   0.00000 27.58333
13            0     36     40       1         2 -27.58333 55.16667
14            0      0     36       0         1   0.00000 28.50000
15            0     36    100       1         2 -28.50000 57.00000
16            1      0     36       0         1   0.00000 39.66667
17            1     36    186       1         2 -39.66667 79.33333
18            0      0     20       1         1   0.00000 23.00000
19            0      0     36       0         1   0.00000 37.66667
20            0     36     84       1         2 -37.66667 75.33333
```

```
des3<-svydesign(ids=~psu, strata = ~strata ,
                weights=~weight, data=sub.split[is.na(sub.split$partnerhiedu)==F,])

cox.s2<-svycoxph(Surv(secbi,b2event)~educ.high+partnerhiedu+hv1+hv2,
                 design=des3)
summary(cox.s2)
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (670) clusters.
svydesign(ids = ~psu, strata = ~strata, weights = ~weight, data = sub.split[is.na(sub.split$
    F, ])


Call:
svycoxph(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
    hv1 + hv2, design = des3)

  n= 4320, number of events= 1980

                  coef exp(coef)  se(coef) robust se      z Pr(>|z|)
educ.high    -0.090737  0.913258  0.066556  0.091208 -0.995    0.320
partnerhiedu  0.066127  1.068362  0.066210  0.085337  0.775    0.438
hv1           0.150921  1.162905  0.007882  0.005713 26.415   <2e-16 ***
```

```
hv2           0.086805  1.090684  0.004594  0.004530 19.162   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

             exp(coef) exp(-coef) lower .95 upper .95
educ.high       0.9133     1.0950    0.7638     1.092
partnerhiedu    1.0684     0.9360    0.9038     1.263
hv1             1.1629     0.8599    1.1500     1.176
hv2             1.0907     0.9169    1.0810     1.100

Concordance= 0.651  (se = 0.008 )
Likelihood ratio test= NA  on 4 df,   p=NA
Wald test            = 864  on 4 df,   p=<2e-16
Score (logrank) test = NA  on 4 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).
```

So, for us β_2 in the heavyside function is positive, suggesting that the age effect increase over time

Aalen's additive regression model

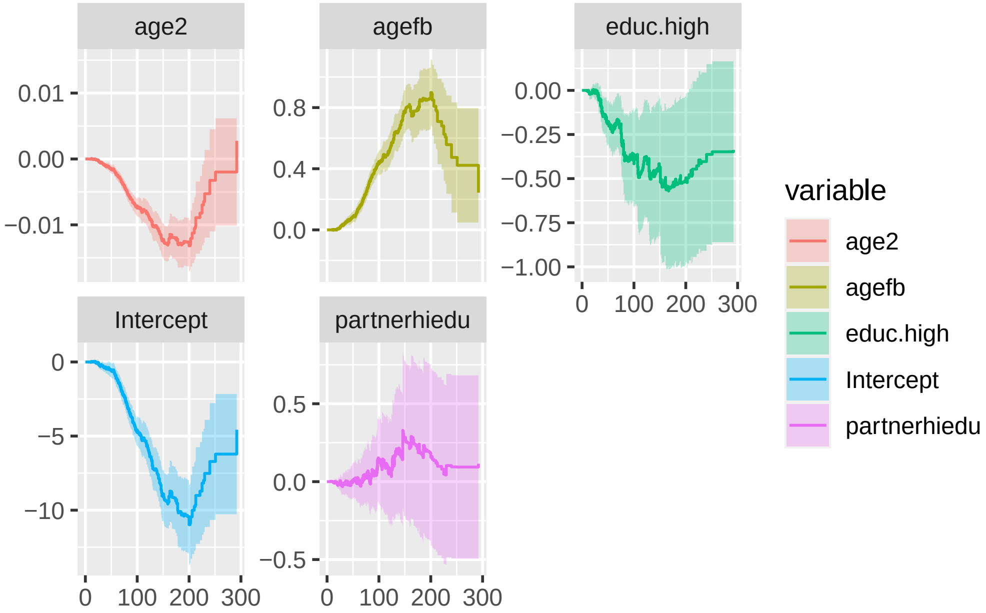
An alternative model proposed by Odd Aalen in 1989 and 1993 describe a model that is inherently nonparametric and models the changes in relationships in a hazard model.

```
fita<-aareg(Surv(secbi,b2event)~educ.high+partnerhiedu+agefb+age2+cluster(strata),
            sub, weights = weight)

summary(fita)
```

```
                 slope      coef se(coef) robust se      z        p
Intercept    -0.055100 -2.85e-03 3.00e-04  2.47e-04 -11.60 7.09e-31
educ.high    -0.007290 -2.12e-04 6.65e-05  7.33e-05  -2.89 3.81e-03
partnerhiedu  0.000603  3.90e-05 6.44e-05  4.48e-05   0.87 3.84e-01
agefb         0.005790  2.51e-04 2.27e-05  2.27e-05  11.10 2.19e-28
age2         -0.000098 -3.95e-06 4.06e-07  4.07e-07  -9.71 2.80e-22

Chisq=369.83 on 4 df, p=<2e-16; test weights=aalen
```

```r
library(ggfortify)
autoplot(fita)
```

```
Warning: `gather_()` was deprecated in tidyr 1.2.0.
Please use `gather()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```



What is seen in the plots are the time-varying coefficients of the hazard model. For example the effect of `educ.high` is globally negative, suggesting higher education decreases the hazard, as we saw in the Cox model above. In the plot, the regression function initially decreases sharply but then plateaus, suggesting the education effect is really only time varying until about 100 months after the first birth.