DEM 7223 - Event History Analysis - Parametric Hazard Models

true

September 21, 2020

Contents

Regression modeling of duration data

- Up until now we have not been concerned with the effects of individual characteristics on the risk of experiencing an event.
- We did see earlier that we can express the risk of experiencing an event conditional on individual risk factors, or covariates.
- We first discuss the use of parametric models for doing this

Parametric models

- When we consider a parametric model in hazards analysis, we are saying that we intend to explicitly define the fundamental shape of our *hazard function*, or that we are assuming a specific distribution for our durations.
- If we make a poor assumption on either of these points, our analysis is often incorrect, because we have effectively defined the wrong model.
- This is bad because our parameters that we think are telling us something, really are telling us nothing.
- We've seen this concept before when considering the *Generalized Linear Model* vs the *Linear Model*. i.e. Don't us the linear model for a binary outcome
- We can use regression models for duration data in two ways:

Proportional Hazards Model (PH)

$$h(t_i) = h_0 \; g(x_i)$$

usually letting $g(x_i) = \exp\,(x_i'\beta)$

Accelerated Failure Time Model (AFT)

$$log(t_i) = x_i'\beta + z_i$$

letting z_i have a parametric density

Which model form to use?

What is being modeled? Hazard or time?

- In a proportional hazard model if a $\beta > 0$ it says that the hazard increases, if a $\beta < 0$ it says that the hazard decreases.
- This is different than we are used to seeing for other regression models
- If the hazard is higher, then the risk is greater. This implies that subjects experience the event at a faster rate, and on average the durations are shorter.

In a accelerated failure time model if a $\beta > 0$ it says that the time, or duration increases, if a $\beta < 0$ it says that the time, or duration decreases.

This is similar to what we are used to seeing for other regression models.

Parameters and distributions

Parameters are unknown quantities that we estimate from data.

They define characteristics of mathematical functions, and variations in said functions.

Some examples of parametric models:

Linear regression, using the Normal distribution

$$y \sim Normal(b_0 + b_1 * x, \sigma_e^2)$$

has 2 parameters, the mean, here shown as the linear mean function, and the variance in the residuals

Logistic regression has mean function that is a transform of the mean

$$y \sim Binomial\left(\frac{1}{1 + exp^{(b_0 + b_1 * x)}}\right)$$

In both of these models, we estimate the parameters, b_0 and b_1 to describe how x affects y

In Parametric hazard models, we estimate regression parameters as well, but we also estimate parameters that describe the *shape of the distribution* of duration times

E.g. the normal distribution function is defined by 2 parameters: μ and σ, which define the characteristic bell-shaped curve

Common distributions in event history analysis

- Exponential This is a 1 parameter distribution, the hazard model for this is:

$$h_i(t, x_i) = h_0 \ exp(x'\beta)$$

The exponential is often a starting point that we don't use very much. The biggest reason we don't use it is because the hazard function is assumed to be a constant (h_0 isn't a function of time)

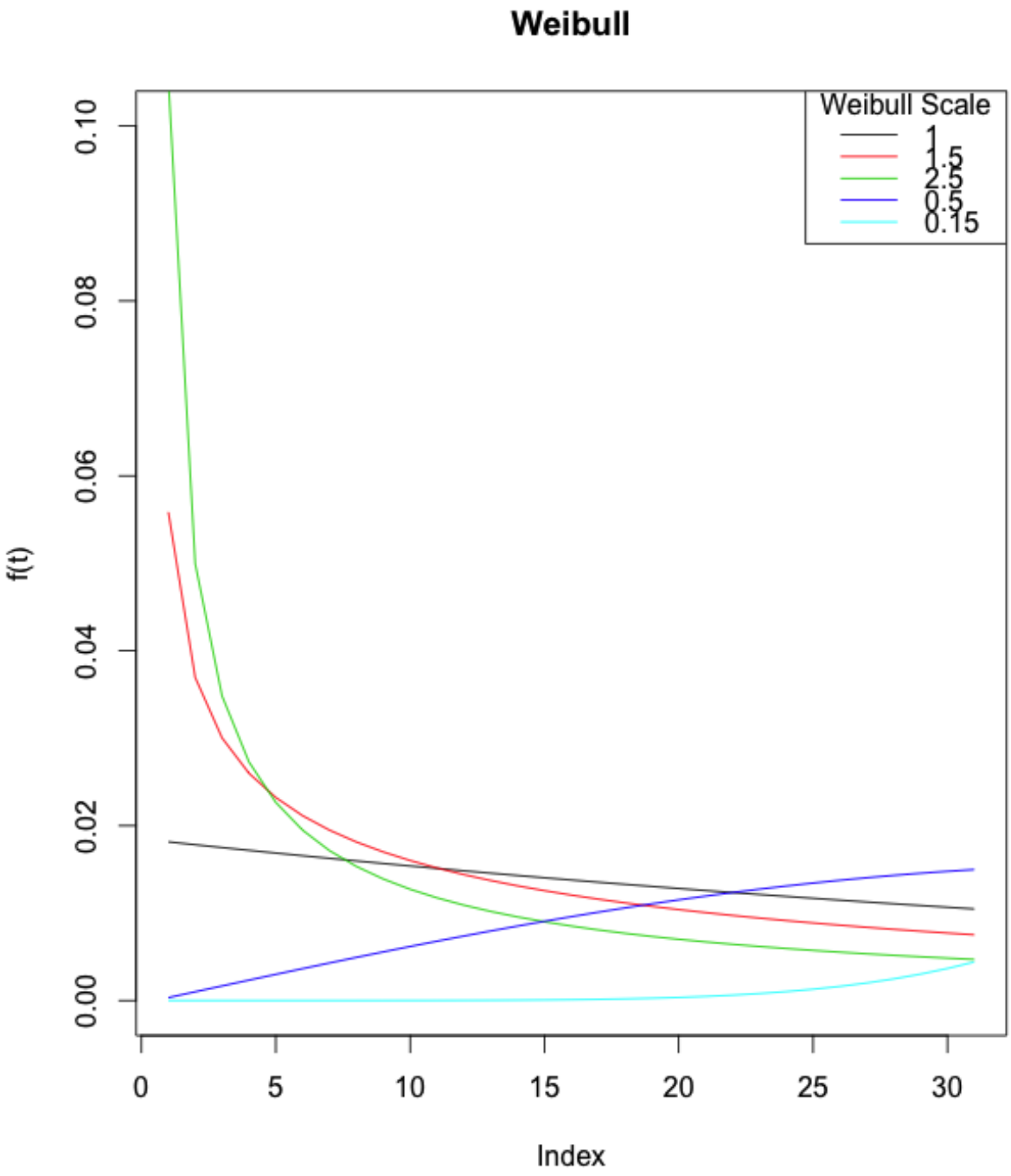
- Weibull The Weibull is a two parameter distribution, it's hazard function without covariates is:

$$h_i(t) = \theta\gamma t^{\gamma-1}$$

it's hazard function with covariates is:

$$h_i(t, x_i) = \gamma exp(x'\beta)t^{\gamma-1}$$

You notice that θ is replaced with the mean function in the second equation. The Weibull is a much more flexible distribution, and the shape of the hazard function change as γ changes.



- Log-normal - another 2 parameter distribution, yes, the log of the Normal distribution. Strictly positive. Hazard function is this monster:

$$h(t) = \frac{\frac{1}{t\sigma\sqrt{2\pi}} exp\left[\frac{-1}{2\sigma^2} ln(t) - \mu^2\right]}{1 - \Phi\left\{\frac{ln(t)-\mu}{\sigma}\right\}}$$

Yikes! This is a much more flexible model, because the hazard can actually increase and decrease, which the Weibull cannot do.

- Log-logistic - another 2 parameter distribution, very flexible

$$h(t) = \frac{\lambda \frac{1}{\gamma} t[\frac{1}{\gamma} - 1]}{\gamma[1 + (\lambda t)^{\frac{1}{\gamma}}]}$$

Yikes! If $\gamma <$ then the hazard rises, then falls, if $\gamma \geqslant 1$, the hazard is declining The parameter λ is the location, and can be parameterized as the linear mean function: $\lambda = e^{-X'\beta}$

- Gompertz - very famous demographic model for adult mortality, hazard function is:

$$h(t) = \lambda e^{\gamma t}$$

Where $\lambda = e^{X'\beta}$

If $\gamma < 1$, then the hazard is monotone decreasing over time, if $\gamma >1$, then the hazard is increasing over time, and if $\gamma = 1$ then the hazard is flat, and we have the exponential.

In general, more parameters allow for more flexibility to the shape of any distribution, and hence more flexibility when it comes to fitting the distribution to data.

But be aware that more complicated models are not always better than simple ones, and you should compare the fit of the model versus its complexity

Parsimony is the backbone of science!

More on the exponential model

AFT form

Since the exponential distribution is solely determined by the parameter, λ, and $\lambda > 0$, we need a model to accommodate this.

The exponential model can be specified two ways The accelerated failure time model is:

$$log(T_i) = x'\beta + z_i$$

Where the β's are regression parameters relating covariate values (the x's) to the duration time

PH form If we treat the hazard rate, λ, as a function of the covariates and the β's, we can write λ as

$$\lambda_i = exp^{(-x'\beta)}$$

So the hazard rate, is given by the covariates x

More on proportional hazards

An important aspect of the exponential model is called the *proportional hazards interpretation* If x is either 1 or 0, and the first term in the β's is a constant (the intercept term), we can write our hazard model as:

$$\frac{h_i(t|x=1)}{h_i(t|x=0)} = \frac{exp(-\beta_0 + \beta_1 * 1)}{exp(-\beta_0 + \beta_1 * 0)} = exp^{(\beta_1)}$$

So β is a constant, called the *baseline hazard*

Changes to this baseline hazard happen through the effect of β_1, or the covariate effects, we can consider the relative change in the hazard for someone with x =1, verses someone with x = 0

This is known as the *proportional hazards property*

Since the hazard rate in the Exponential model is invariant with respect to time, it represents a very simplistic model and one that often does not occur in the real world

Be careful with interpretations!

For Accelerated failure time model Y=log (duration), so if $exp(\beta_1) > 1$, you have an increase in time (implies a decrease in risk), if $exp(\beta_1) < 1$ you have a decrease in time (and an implied increase in risk)

For Proportional Hazards models

Y=hazard(time), so if $exp(\beta_1) > 1$, you have an increase in hazard (and a decrease in duration), if $exp(\beta_1) < 1$ you have a decrease in hazard (and an increase in duration)

Data example

This example will illustrate how to fit parametric hazard models to continuous duration data (i.e. person-level data). In this 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.

This is an important data file, because for each woman, it gives information on all of her births, arrayed in columns.

```
#Load required libraries
library(haven)
library(survival)
library(car)
```

```
## Loading required package: carData
```

```
library(survey)
```

```
## Loading required package: grid
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'survey'

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

```
library(muhaz)
library(eha)

#load the data
model.dat<-read_dta("https://github.com/coreysparks/data/blob/master/ZZIR62FL.DTA?raw=true")
model.dat<-zap_labels(model.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 6161 women who are at risk of a second birth.

```
table(is.na(model.dat$bidx_01))
```

```
##
## FALSE  TRUE
##  6161  2187
```

```
#now we extract those women
sub<-subset(model.dat, model.dat$bidx_01==1&model.dat$b0_01==0)

#Here I keep only a few of the variables for the dates, and some characteristics of the women, and deta
sub2<-data.frame(CASEID=sub$caseid,
                 int.cmc=sub$v008,
                 fbir.cmc=sub$b3_01,
                 sbir.cmc=sub$b3_02,
                 marr.cmc=sub$v509,
                 rural=sub$v025,
                 educ=sub$v106,
                 age=sub$v012,
                 partneredu=sub$v701,
                 partnerage=sub$v730,
                 weight=sub$v005/1000000,
```

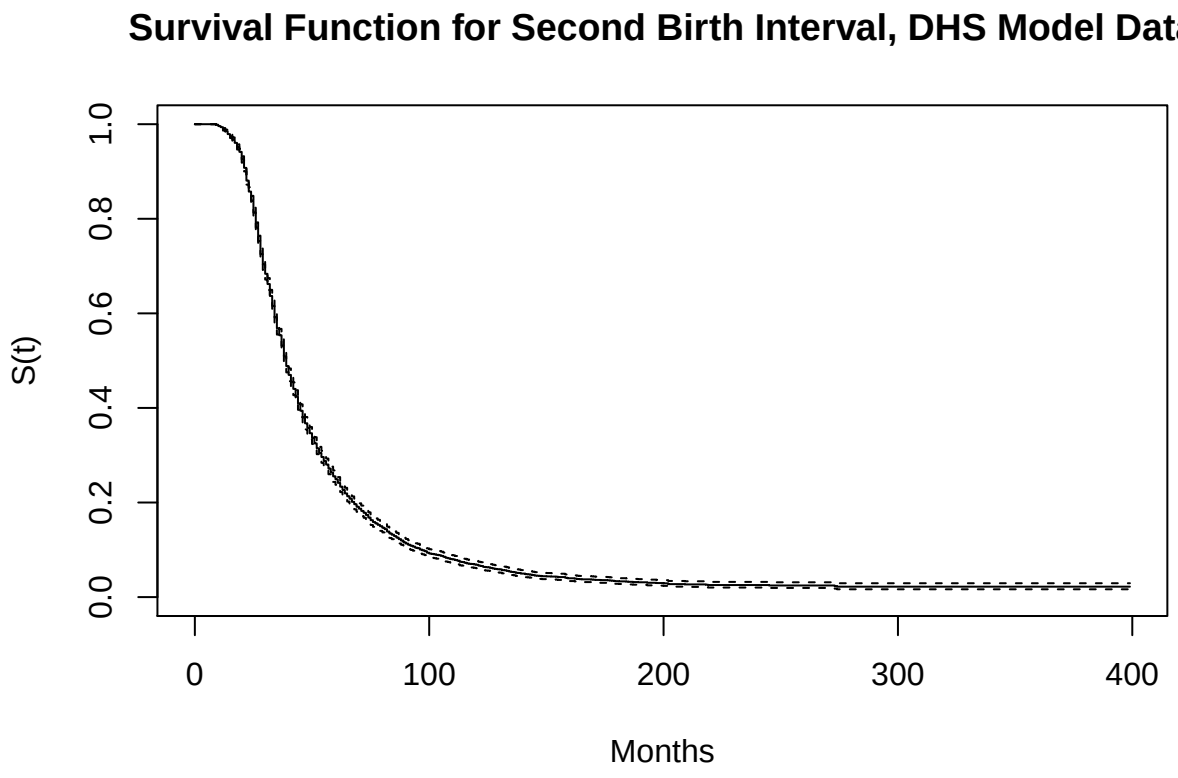
```
                psu=sub$v021, strata=sub$v022)

sub2$agefb = (sub2$age - (sub2$int.cmc - sub2$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?)

```
sub2$secbi<-ifelse(is.na(sub2$sbir.cmc)==T,
                   ((sub2$int.cmc))-((sub2$fbir.cmc)),
                   (sub2$fbir.cmc-sub2$sbir.cmc))

sub2$b2event<-ifelse(is.na(sub2$sbir.cmc)==T,0,1)
fit<-survfit(Surv(secbi, b2event)~1, sub2)
fit
```

```
## Call: survfit(formula = Surv(secbi, b2event) ~ 1, data = sub2)
##
##       n  events  median 0.95LCL 0.95UCL
##    6026    4789      39      38      40
```

```
plot(fit, conf.int=T, ylab="S(t)", xlab="Months")
title(main="Survival Function for Second Birth Interval, DHS Model Data")
```



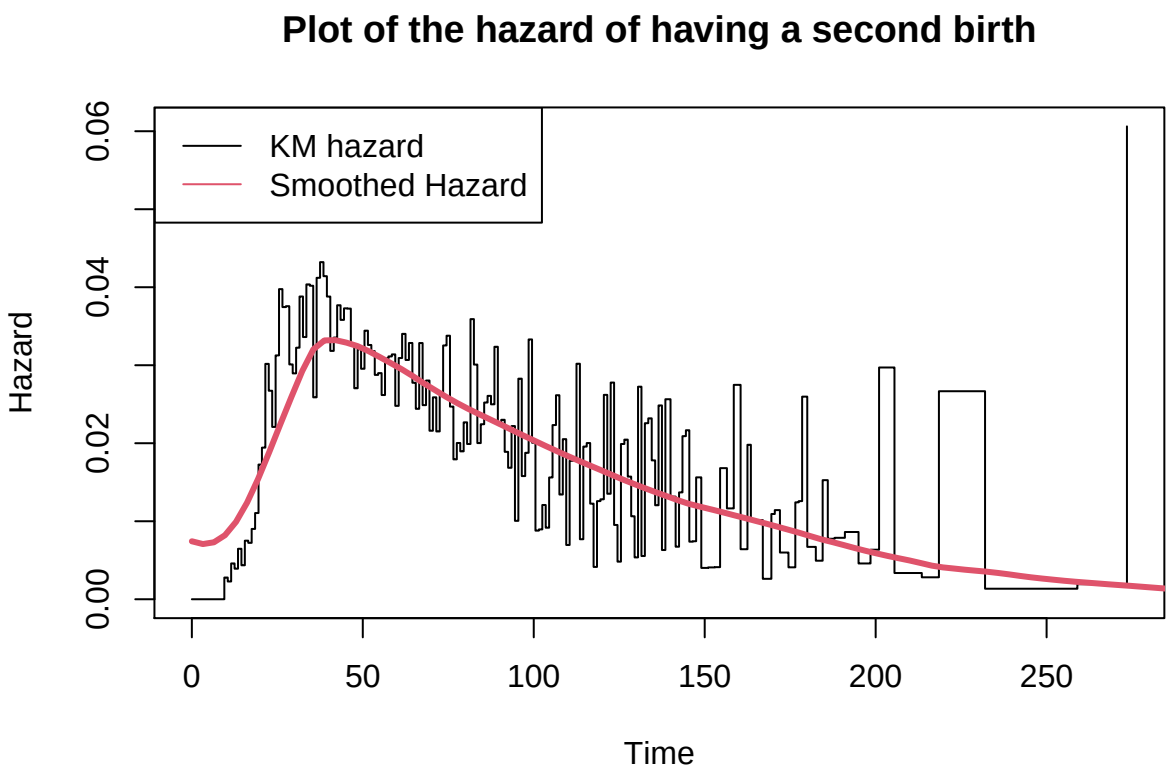
Estimating Parametric Hazard Models

While parametric models are not so common in demographic research, fundamental understanding of what they are and how they are constructed is of importance.

Some outcomes lend themselves very readily to the parametric approach, but as many demographic duration times are non-unique (tied), the parametric models are not statistically efficient for estimating the survival/hazard functions, as they assume the survival times are continuous random variables.

In this section, we first estimate the empirical hazard function and then fit a variety of parametric models to it (Exponential, Weibull, Log-normal and Piecewise exponential). Ideally, a parametric model's hazard function should approximate the observed empirical hazard function, *if the model fits the data.*

```
#since these functions don't work with durations of 0, we add a very small amount to the intervals
fit.haz.km<-kphaz.fit(sub2$secbi[sub2$secbi>0],
                      sub2$b2event[sub2$secbi>0] ,
                      method = "product-limit")

#this is a version of the hazard that is smoothed using a kernel-density method
fit.haz.sm<-muhaz(sub2$secbi[sub2$secbi>0], sub2$b2event[sub2$secbi>0] )

#Empirical hazard function (product-limit estimate) plot
kphaz.plot(fit.haz.km,main="Plot of the hazard of having a second birth")
#overlay the smoothed version
lines(fit.haz.sm, col=2, lwd=3)
legend("topleft", legend = c("KM hazard", "Smoothed Hazard"),
       col=c(1,2), lty=c(1,1))
```



So now we see what the empirical hazard function looks like, in both the observed and smoothed estimate of it.

Create covariates

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

```
sub2$educ.high<-ifelse(sub2$educ %in% c(2,3), 1, 0)
sub2$age2<-(sub2$agefb/5)^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 )

rep.des<-as.svrepdesign(des, type="bootstrap" )
```

Fit the models

Now we fit the models.

I use the `eha` package to do this, since it fits parametric proportional hazard models, not accelerated failure time models.

I prefer the interpretation of regression models on the hazard scale vs. the survival time scale. EHA is not the only package that will fit parametric survival models, be sure you *read the documentation for the procedure you use!!* Different functions fit different parameterizations of the distributions. For example, the `survreg()` function in the `survival` library fits accelerated failure time models only.

Exponential Model

Often the exponential model isn't directly available in packages, so we can fit a weibull model with a fixed shape parameter. This is 100% legal.

The exponential distribution has a constant hazard rate, $\lambda(t) = \lambda$. The survival function is $S(t) = \exp(-\lambda t)$

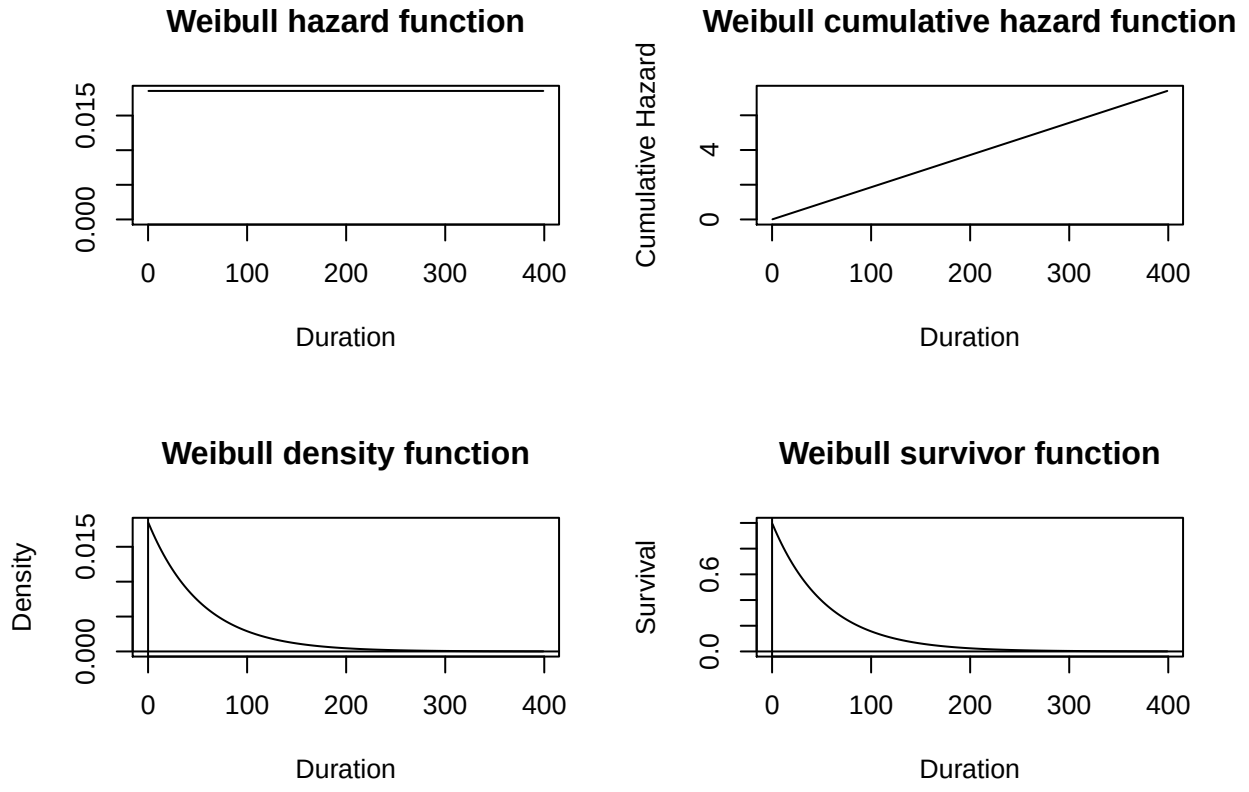
To specify the model in terms of covariates, you can write the hazard as a log-linear model : $\log\lambda = x`\beta$

```
#exponential distribution for hazard, here we hard code it to be
#a weibull dist with shape ==1
fit.1<-phreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(agefb/5)+age2,
             data=sub2[sub2$secbi>0,], dist="weibull", shape = 1)
summary(fit.1)
```

```
## Call:
## phreg(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
##     I(agefb/5) + age2, data = sub2[sub2$secbi > 0, ], dist = "weibull",
##     shape = 1)
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
```

```
## educ.high            0.167    -0.280     0.756     0.048     0.000
## partnerhiedu         0.071    -0.180     0.835     0.069     0.009
## I(agefb/5)           5.744     1.130     3.095     0.084     0.000
## age2                35.186    -0.090     0.914     0.007     0.000
##
## log(scale)                      7.254               0.244     0.000
##
##  Shape is fixed at  1
##
## Events                     4527
## Total time at risk         237141
## Max. log. likelihood       -22296
## LR test statistic          303.75
## Degrees of freedom         4
## Overall p-value            0
```

```
plot(fit.1)
```



Which shows us what the constant hazard model looks like, it assumed the hazard is constant with respect to time, which after seeing the plots above, we know is false. We see the effects of both woman's and partner's education are negative, which makes sense. Women with more education, and who have partners with more education lower risks of having a second birth. We also see the age effect is significant, meaning older women in this sample are more likely to have a second birth but the hazard doesn't go up forever, as the curvilinear term shows a negative slope.

###Interpreting the model coefficients To interpret the effects specifically, you can use the `Exp(Coef)` column. So, for example for women who have secondary or higher education, their hazard of having a

second child is 24.436 lower than a woman with less than a secondary education. To get that number I do : $100 * 1 - \exp(\beta_{\text{educ.high}})$

Likewise, for the effect of age, we can compare the hazards for a women who is age 35 to a woman who is age 20. To do this comparison for a continuous covariate, you have to form the ratio of the hazards at two different plausible values. For this comparison, we see that women who are age 35 are 1.512 times more likely to have a second birth than women who are 20. To get this, I find:

$$\text{Hazard Ratio} = \frac{\exp\left(\beta_{\text{I(age/5)}} * 7 + \beta_{\text{age2}} * 7\right)}{\exp\left(\beta_{\text{I(age/5)}} * 4 + \beta_{\text{age2}} * 4\right)}$$

I choose 7 because `7 * 5` $= 35$, and 4 because `4*5` $= 20$. Remember, I divided Age by 5 when I created my variables.

AFT model specification

If you wanted to do the AFT model, you can either `aftreg()` in the `eha` package or `survreg()` in the `survival` package. Generally AFT models are written as:

$\log T = -x`\beta + \sigma W$ Where W is an error (residual) term, which is assumed to follow some distribution.

```
fit.1.aft<-survreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+age2 + age2,
                   data=sub2[sub2$secbi>0,],dist = "exponential")

summary(fit.1.aft)
```

```
##
## Call:
## survreg(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
##     I(age/5) + age2 + age2, data = sub2[sub2$secbi > 0, ], dist = "exponential")
##                  Value Std. Error     z       p
## (Intercept)   3.45845    0.07168 48.25 < 2e-16
## educ.high     0.29630    0.04791  6.18 6.2e-10
## partnerhiedu  0.12346    0.06859  1.80   0.072
## I(age/5)      0.13581    0.01511  8.99 < 2e-16
## age2         -0.01326    0.00138 -9.62 < 2e-16
##
## Scale fixed at 1
##
## Exponential distribution
## Loglik(model)= -22354.2   Loglik(intercept only)= -22447.6
##  Chisq= 186.77 on 4 degrees of freedom, p= 2.6e-39
## Number of Newton-Raphson Iterations: 4
## n=5279 (728 observations deleted due to missingness)
```

Which shows, compared to the PH model, that the coefficients are all backwards. That's because if a predictor lowers the hazard, then, by default it extends survival.

Lower risk == longer survival times!

Weibull Model

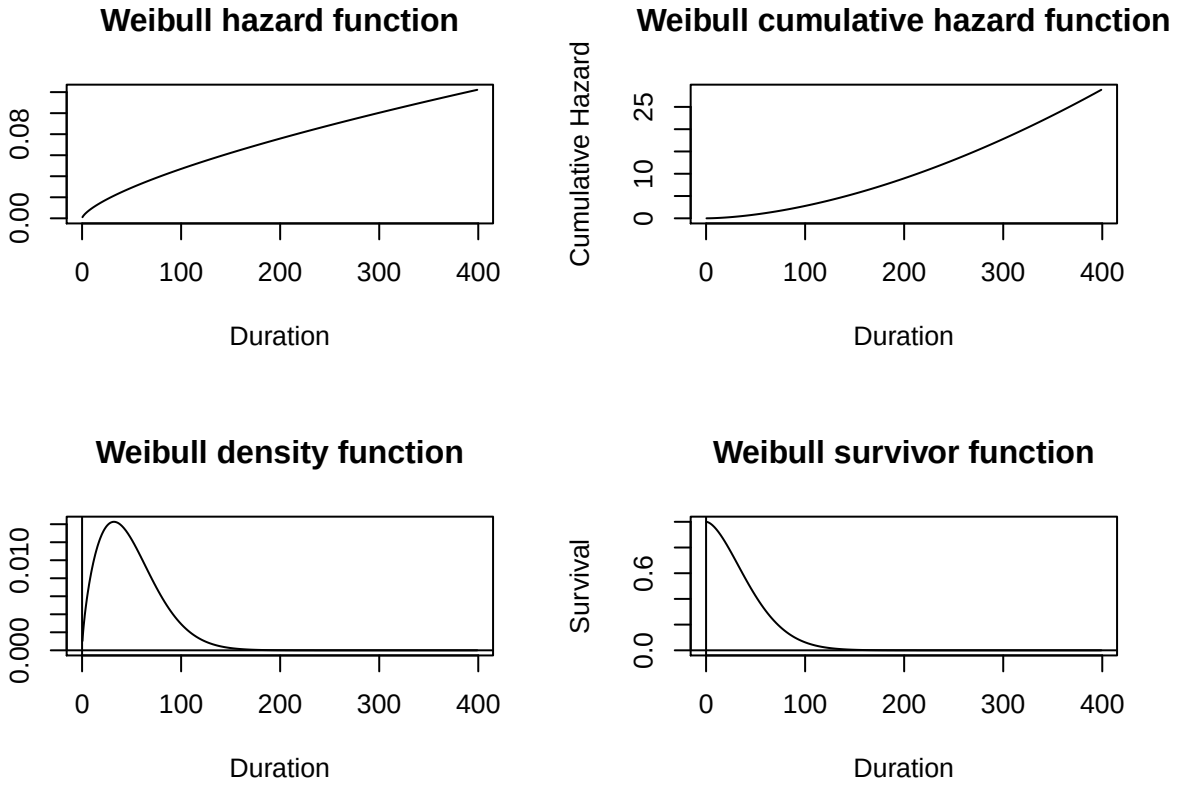
The Weibull model is more flexible than the Exponential, because it's distribution function has two parameters, scale and shape.

The Weibull distribution has hazard rate, $\lambda(t) = \lambda^p pt^{p-1}$. Where λ is the scale and p is the shape. The survival function is $S(t) = exp(-(\lambda t)^p)$

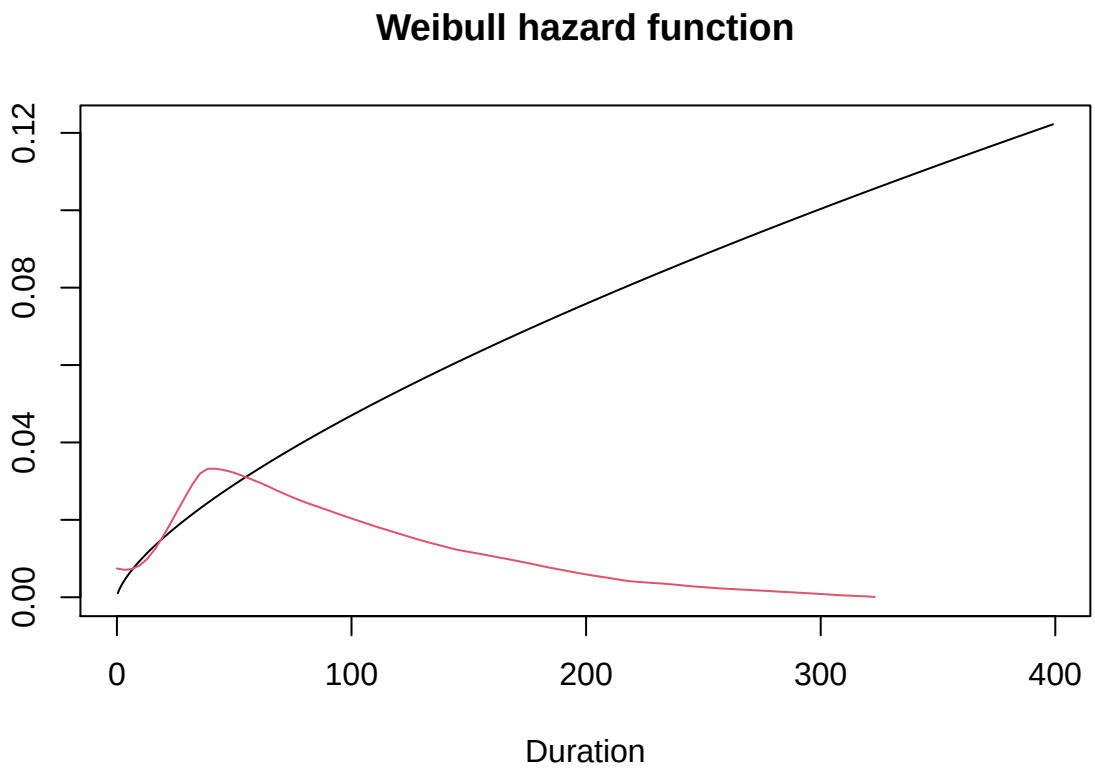
```
#weibull distribution for hazard
fit.2<-phreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+age2,
             data=sub2[sub2$secbi>0,], dist="weibull")
summary(fit.2)
```

```
## Call:
## phreg(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
##     I(age/5) + age2, data = sub2[sub2$secbi > 0, ], dist = "weibull")
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
## educ.high           0.167    -0.369     0.691     0.048     0.000
## partnerhiedu        0.071    -0.229     0.796     0.068     0.001
## I(age/5)            6.859    -0.340     0.712     0.016     0.000
## age2               35.186     0.024     1.024     0.001     0.000
##
## log(scale)                    3.133               0.043     0.000
## log(shape)                    0.525               0.011     0.000
##
## Events                     4527
## Total time at risk         237141
## Max. log. likelihood       -21461
## LR test statistic          684.43
## Degrees of freedom         4
## Overall p-value            0
```

```
plot(fit.2)
```




```
plot(fit.2, fn="haz")
lines(fit.haz.sm, col=2)
```



Here, we see a more realistic situation, where the hazard function changes over time (Weibull allows this), but compared to the empirical hazard, the model is a very poor fit, as empirically, the hazard goes up, but then goes down. The Weibull hazard just goes up, as the model does not allow the hazard to change direction, only rate of increase (i.e. it can increase at a slower or faster rate, but not change direction). We see the Age effects begin to go away, because the baseline hazard is accounting for the age effects on fertility.

##Note on exponential and Weibull models AFT vs PH parameterization and, as a nice trick for the exponential and weibull models, you can rescale the AFT beta's to PH model betas (see here)

```
#re-scaled beta's
(betaHat <- -coef(fit.1.aft) / fit.1.aft$scale)
```

```
##  (Intercept)    educ.high partnerhiedu     I(age/5)         age2
##  -3.45844846  -0.29630384  -0.12346198  -0.13580668   0.01326201
```

```
#beta's from the PH model
coef(fit.1)
```

```
##    educ.high partnerhiedu   I(agefb/5)         age2   log(scale)
##  -0.28019154  -0.17987145   1.12983459  -0.09019032   7.25416728
```

So for these two models, you can go back and forth.

Log-Normal Model

The Log-normal distribution is more flexible and allows the hazard to change direction.

The Log-normal distribution has hazard rate, $h(t) = \frac{\phi\left(\frac{logt}{\sigma}\right)}{\left[1-\Phi\left(\frac{logt}{\sigma}\right)\right]\sigma t}$

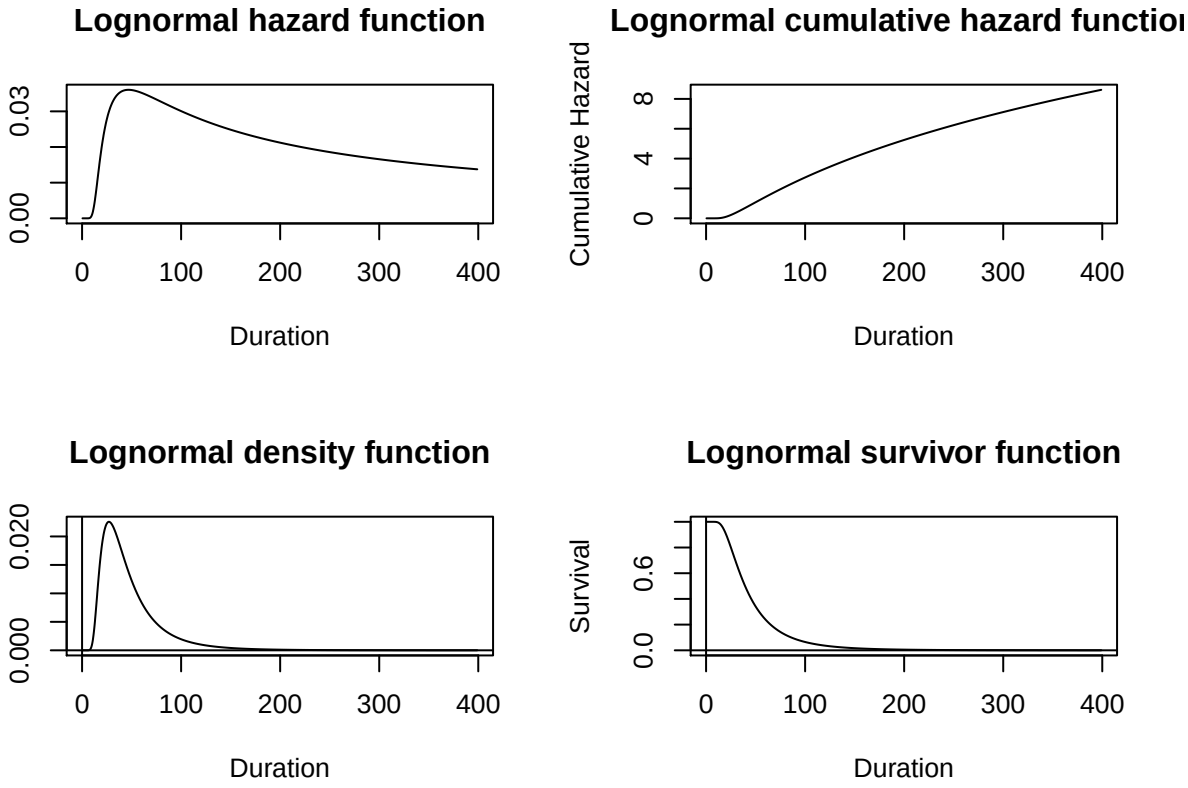
. Where σ is the shape.

The survival function is $S(t) = 1 - \Phi\left(\frac{logt-\mu}{\sigma}\right)$

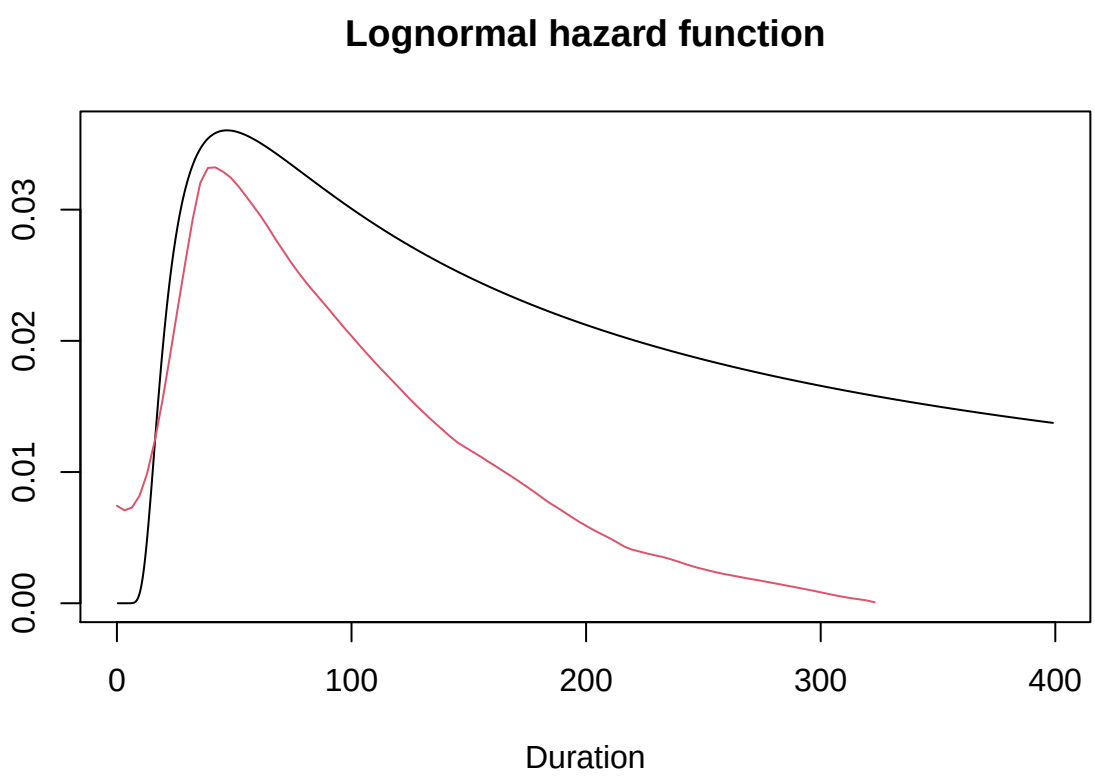
```
#log-normal distribution for hazard
fit.3<-phreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+age2,
             data=sub2[sub2$secbi>0,], dist="lognormal", center=T)
summary(fit.3)
```

```
## Call:
## phreg(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
##     I(age/5) + age2, data = sub2[sub2$secbi > 0, ], dist = "lognormal",
##     center = T)
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
## (Intercept)                  -1.002              0.170     0.000
## educ.high           0.167    -0.350     0.705     0.048     0.000
## partnerhiedu        0.071    -0.196     0.822     0.068     0.004
## I(age/5)            6.859    -0.208     0.813     0.016     0.000
## age2               35.186     0.014     1.014     0.001     0.000
##
## log(scale)                    2.997              0.037     0.000
## log(shape)                    1.267              0.055     0.000
##
## Events                    4527
## Total time at risk        237141
## Max. log. likelihood      -20596
## LR test statistic         303.40
## Degrees of freedom        4
## Overall p-value           0
```

```
plot(fit.3)
```




```
#plot the hazard from the log normal vs the empirical hazard
plot(fit.3, fn="haz")
lines(fit.haz.sm, col=2)
```



We now see the age effect completely gone from the model.

So, the log-normal model fits the empirical hazard pretty well up to ~150 months, where the empirical rate drops off faster. The `eha` package allows one other parametric distribution, the log-logistic, so we will consider that one too:

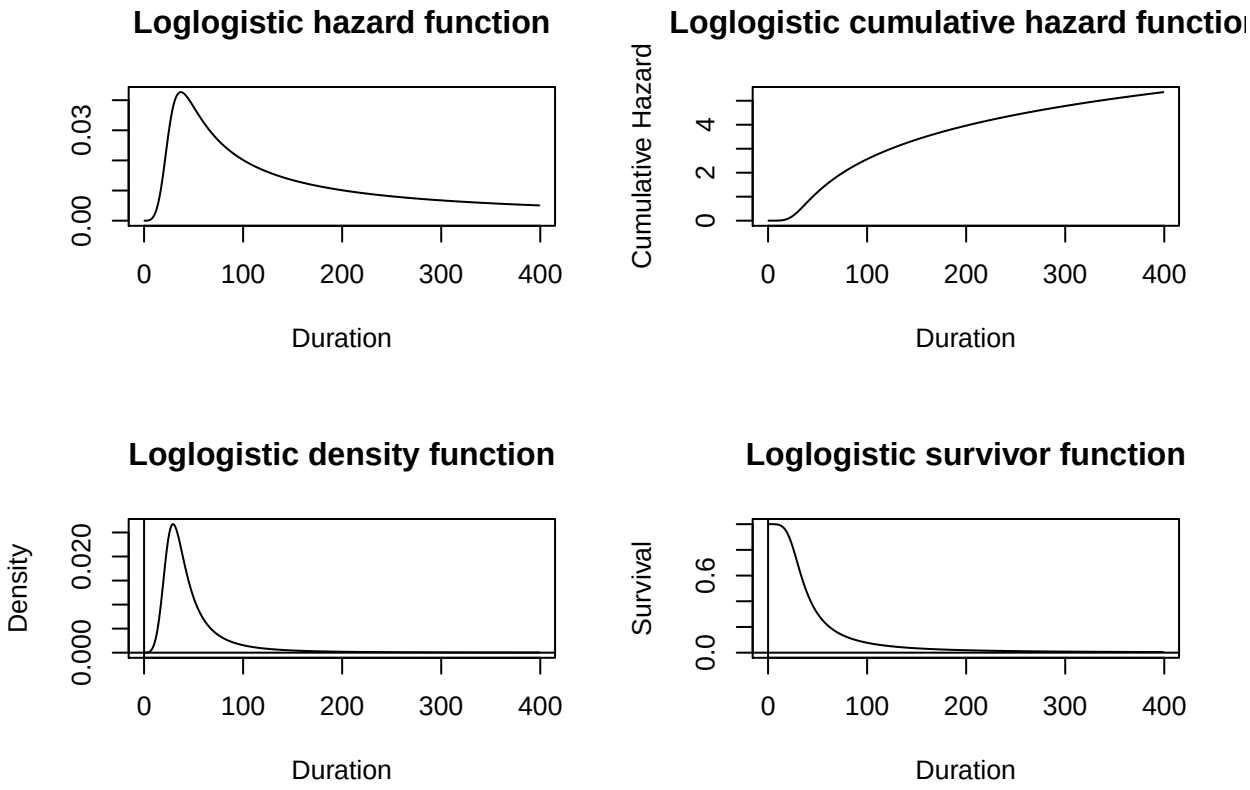
Log-logistic Model

```
#log-normal distribution for hazard
fit.4<-phreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+age2,
             data=sub2[sub2$secbi>0,], dist="loglogistic", center=T)
summary(fit.4)
```

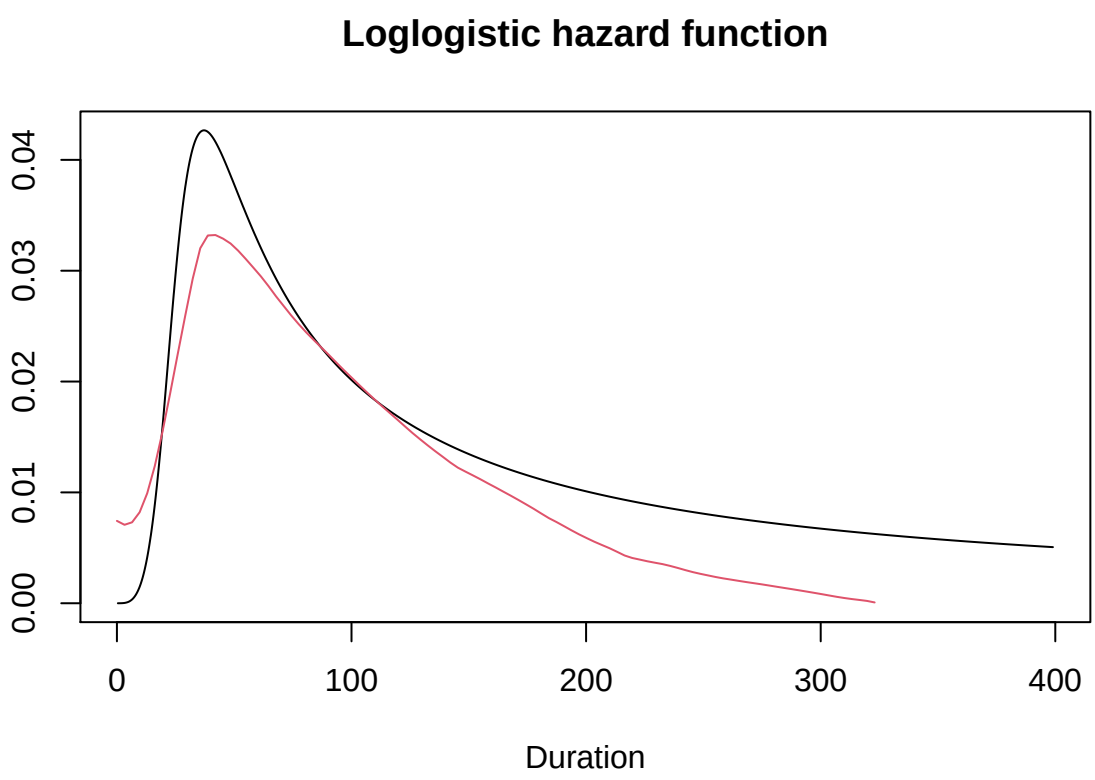
```
## Call:
## phreg(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
##     I(age/5) + age2, data = sub2[sub2$secbi > 0, ], dist = "loglogistic",
##     center = T)
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
## (Intercept)                  -0.121              0.095     0.202
## educ.high           0.167    -0.338     0.713    0.048     0.000
## partnerhiedu        0.071    -0.185     0.831    0.068     0.007
## I(age/5)            6.859    -0.154     0.857    0.016     0.000
## age2               35.186     0.011     1.011    0.001     0.000
##
## log(scale)                    3.336              0.018     0.000
## log(shape)                    1.534              0.028     0.000
```

```
##
## Events                    4527
## Total time at risk        237141
## Max. log. likelihood      -20534
## LR test statistic         199.20
## Degrees of freedom        4
## Overall p-value           0
```

```r
plot(fit.4)
```




```r
#plot the hazard from the log normal vs the empirical hazard
plot(fit.4, fn="haz")
lines(fit.haz.sm, col=2)
```



Whose hazard function drops off faster than the log-normal.

We may want to compare the models to one another based off AIC values. the `eha` package doesn't give this to you, so we must calculate it:

```
AIC1<--2*fit.1$loglik[2]+2*length(fit.1$coefficients); AIC1
```

```
## [1] 44601.37
```

```
AIC2<--2*fit.2$loglik[2]+2*length(fit.2$coefficients); AIC2
```

```
## [1] 42933.17
```

```
AIC3<--2*fit.3$loglik[2]+2*length(fit.3$coefficients); AIC3
```

```
## [1] 41206.94
```

```
AIC4<--2*fit.4$loglik[2]+2*length(fit.4$coefficients); AIC4
```

```
## [1] 41082
```

And we see the log-logistic model best fits the data, based on the minimum AIC criteria

Piecewise constant exponential model

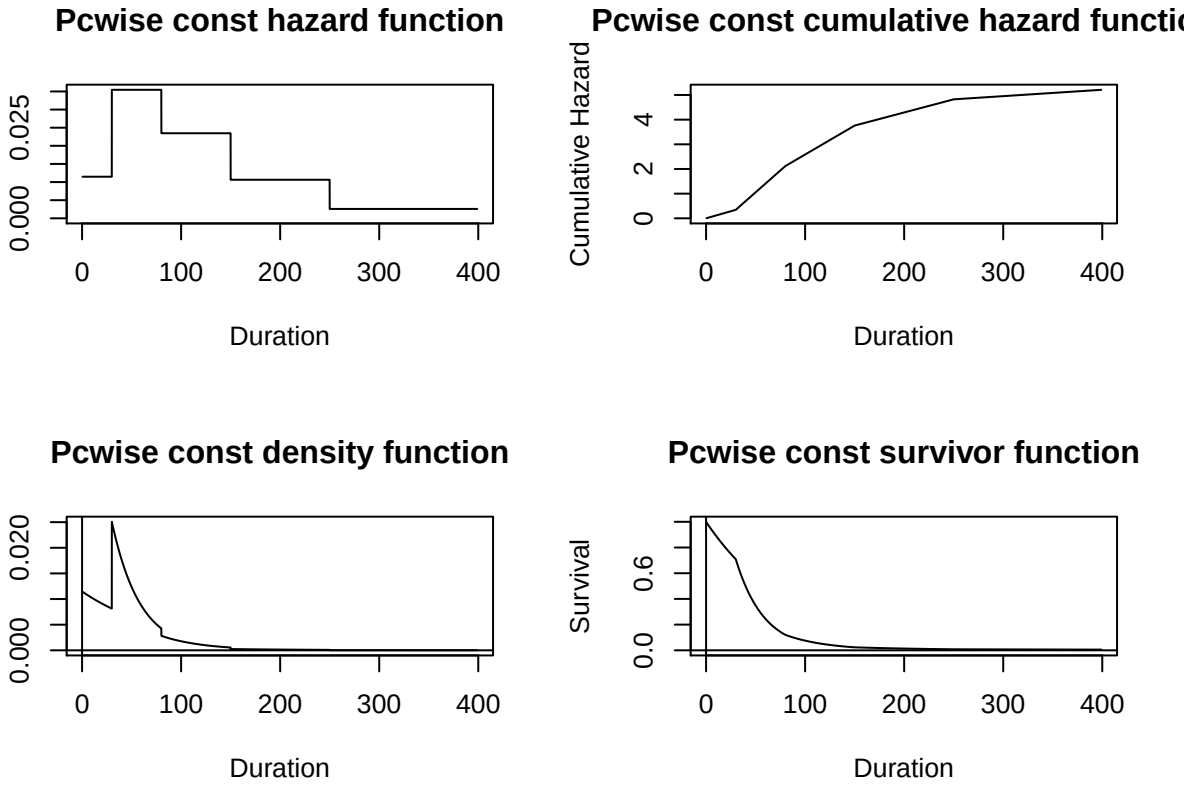
The final model we consider is the Piecewise constant exponential model. This model breaks the data into pieces, where we may fit constant hazards within these pieces.

For instance, given the observed hazard function above, we may break the data into an early piece, say < 30 months, a high piece,30-80 months and maybe two low pieces (80-150 and >150), so to mimic the form of the hazard function.

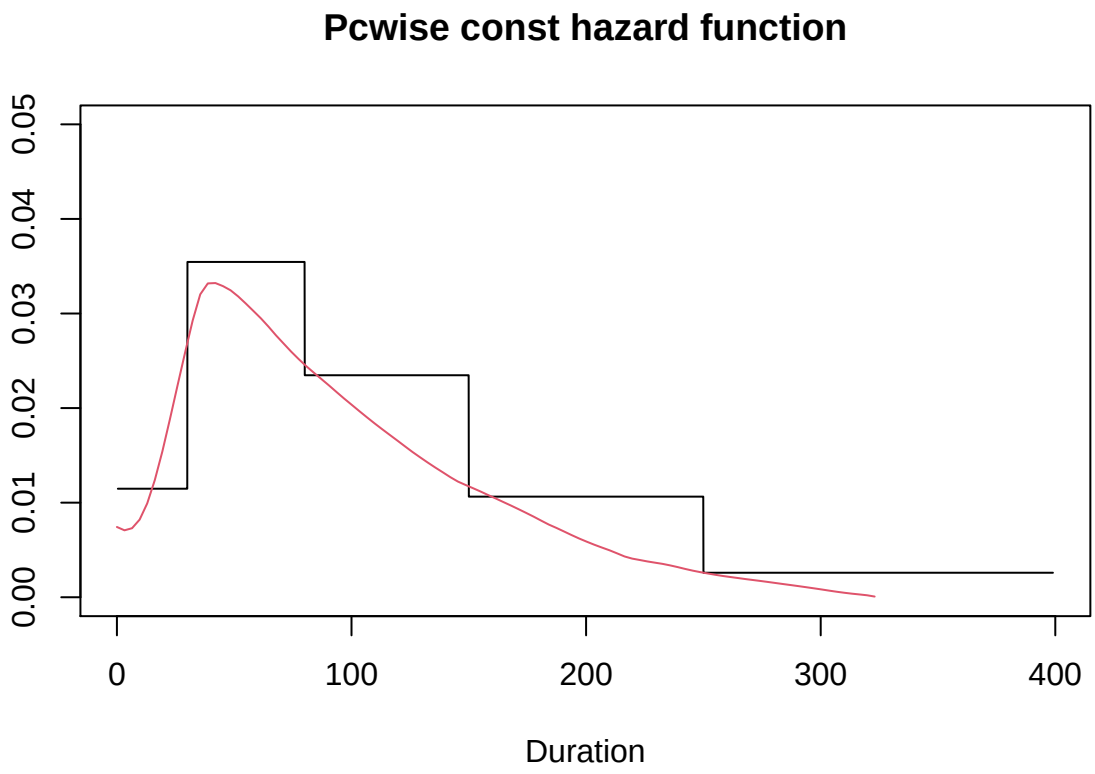
```
# here I must supply the times for the "pieces" where I expect the  hazard to be constant
fit.5<-phreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+age2,
             data=sub2[sub2$secbi>0,], dist="pch",
             cuts=c(30, 80, 150,250))
summary(fit.5)
```

```
## Call:
## phreg(formula = Surv(secbi, b2event) ~ educ.high + partnerhiedu +
##     I(age/5) + age2, data = sub2[sub2$secbi > 0, ], dist = "pch",
##     cuts = c(30, 80, 150, 250))
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
## educ.high           0.167    -0.345     0.708     0.048     0.000
## partnerhiedu        0.071    -0.175     0.839     0.068     0.011
## I(age/5)            6.859    -0.158     0.853     0.016     0.000
## age2               35.186     0.012     1.012     0.001     0.000
##
##
## Events                    4527
## Total time at risk        237141
## Max. log. likelihood      -21695
## LR test statistic         201.04
## Degrees of freedom        4
## Overall p-value           0
```

```
plot(fit.5)
```




```
plot(fit.5, fn="haz", ylim=c(0, .05))
lines(fit.haz.sm, col=2)
```



Which looks like it actually fits the data pretty good. The AIC's show the log-logistic model still fitting better.

```
AIC5<--2*fit.5$loglik[2]+2*length(fit.5$coefficients); AIC5
```

```
## [1] 43397.15
```

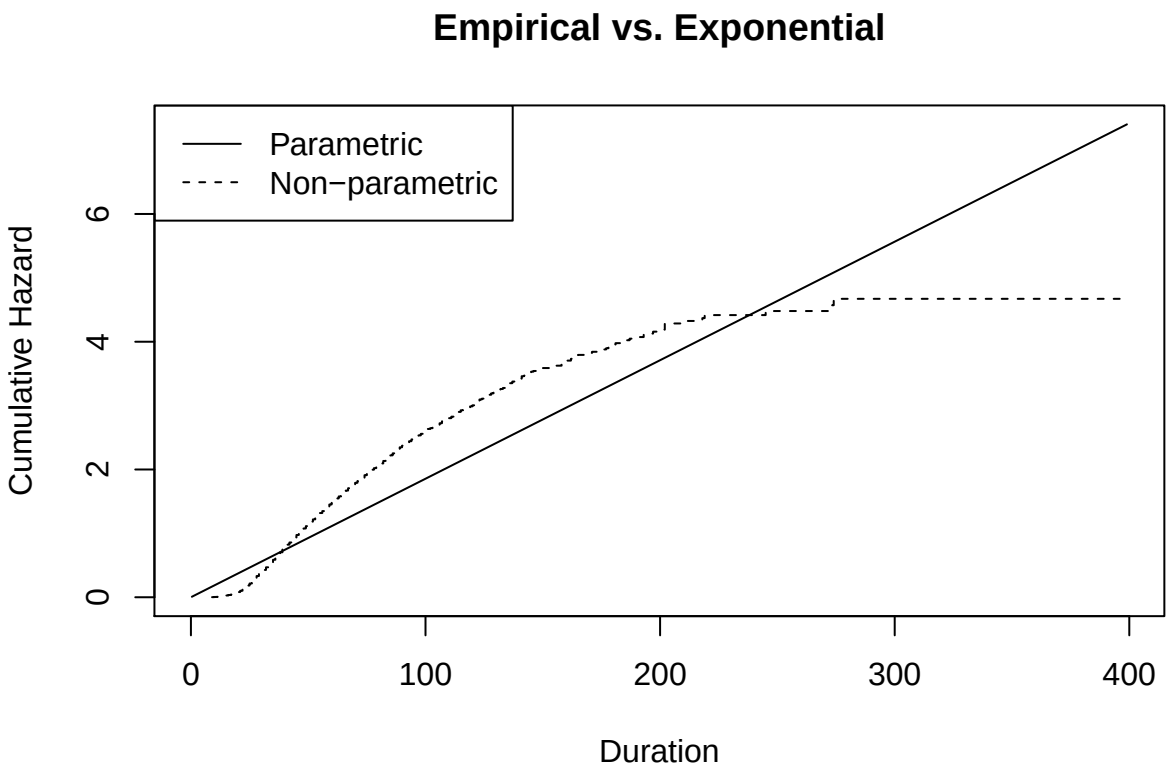
```
AIC4
```

```
## [1] 41082
```

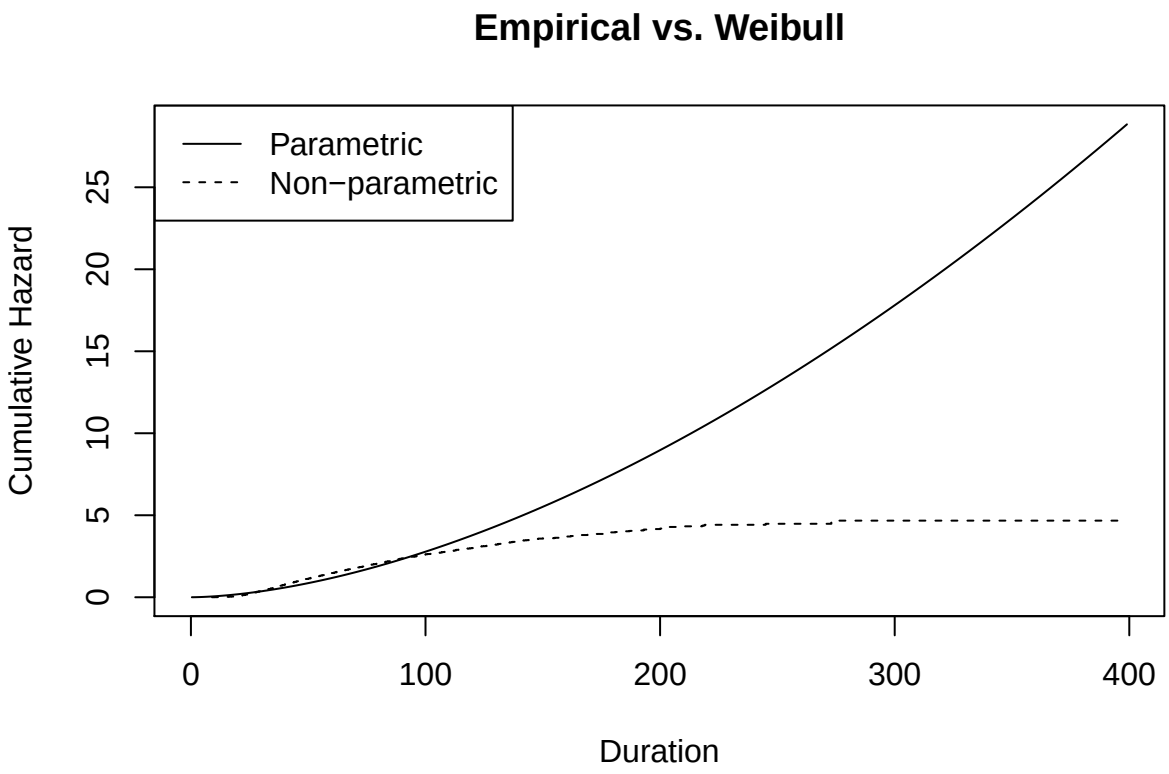
Graphical checks on the model fit

The `eha` package also provides a graphical method for the Cumulative hazard function, which allows us to visualize these models even better. It uses the empirical hazard, as fit in the Cox model (more on this next week), and compares the parametric models to the empirical pattern:

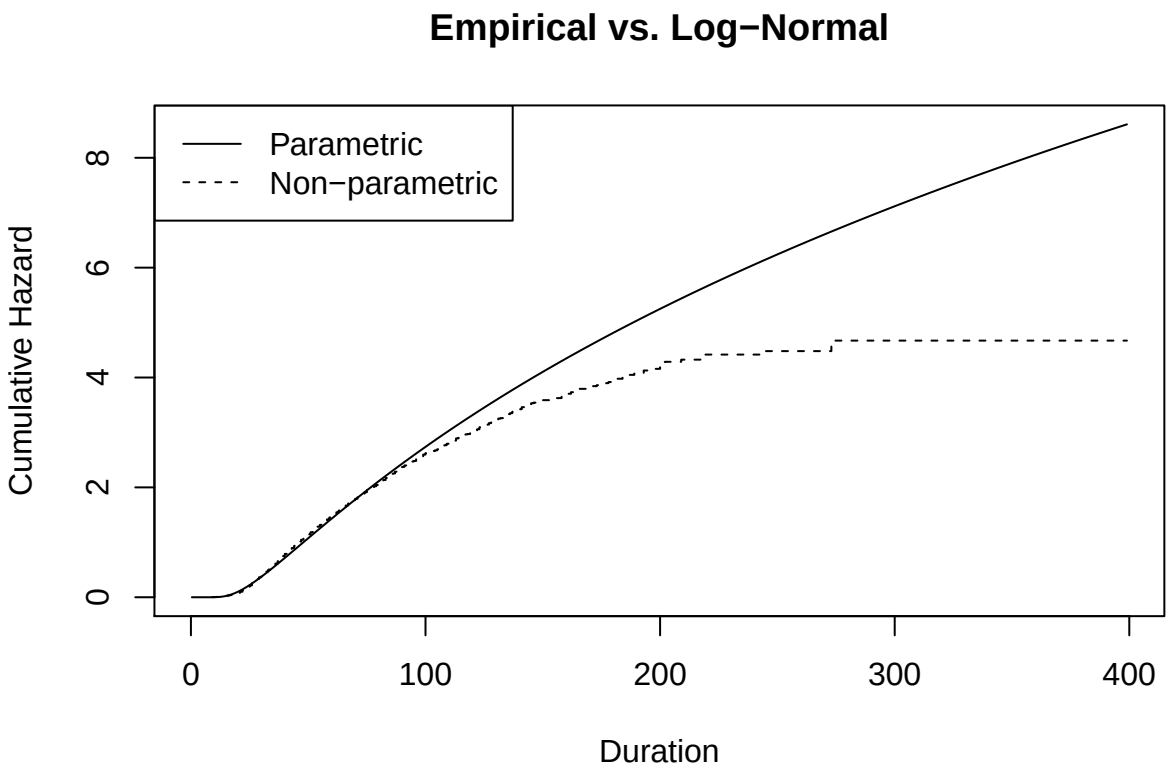
```
emp<-coxreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+age2,
            data=sub2[sub2$secbi>0,])

check.dist(sp=emp,pp=fit.1, main = "Empirical vs. Exponential")
```



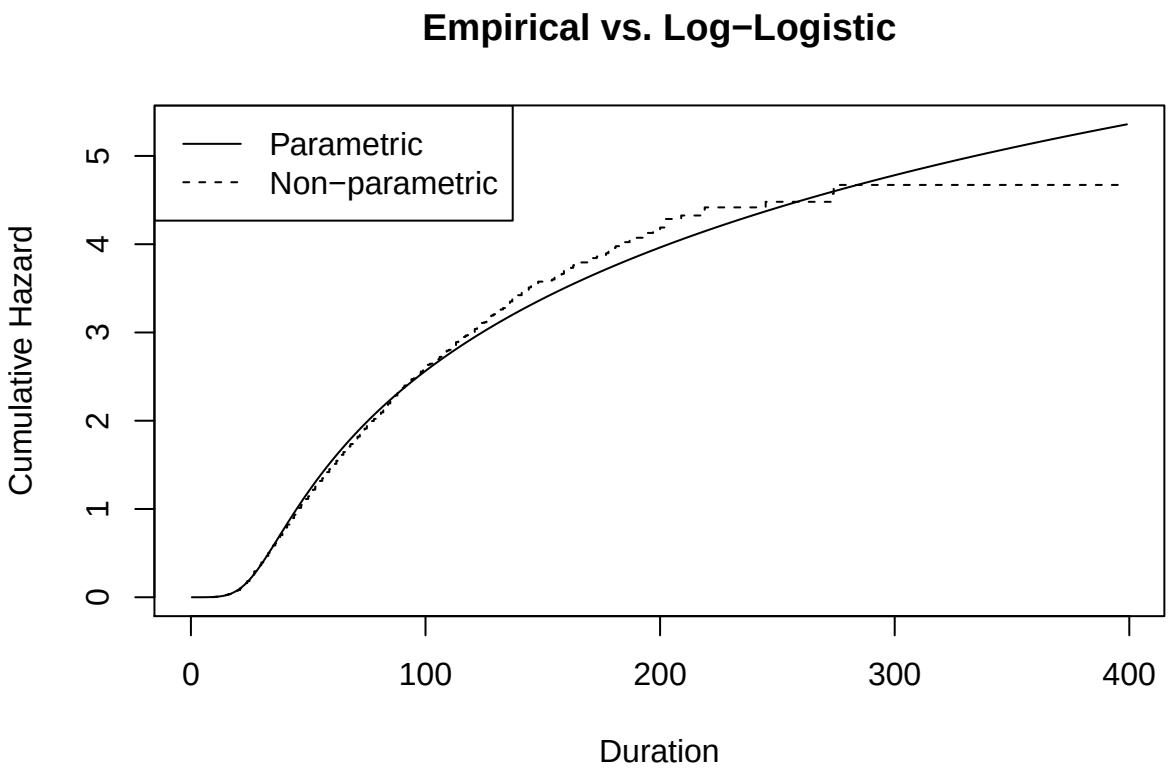

```
check.dist(sp=emp,pp=fit.2, main = "Empirical vs. Weibull")
```



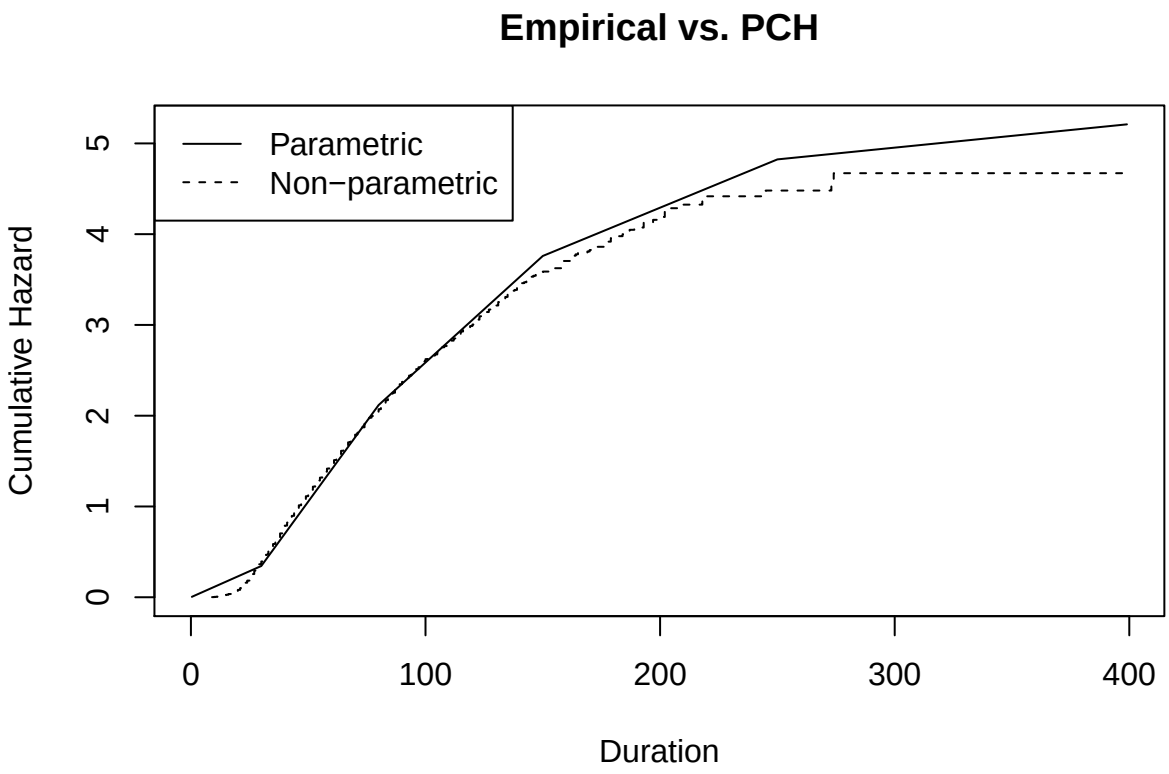

```
check.dist(sp=emp,pp=fit.3, main = "Empirical vs. Log-Normal")
```




```
check.dist(sp=emp,pp=fit.4, main = "Empirical vs. Log-Logistic")
```




```
check.dist(sp=emp,pp=fit.5, main = "Empirical vs. PCH")
```



We see that the PCH model and the log-logistic models both appear to fit the empirical hazard function better than the other parametric models.

Using Survey design

There are no survey analysis functions to fit parametric hazard models, so we must roll our own using advice from Thomas Lumely in his book Appendix E **You can get this on campus through the library.**

```
survey.fit <- withReplicates(rep.des,
                             quote(coef(survreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+ag
survey.est<-as.data.frame(survey.fit)
survey.test<-data.frame(beta = rownames(survey.est), estimate=survey.est$theta, se.est= survey.est$SE)
survey.test$t<-survey.test$estimate/survey.test$se.est
survey.test$pval<-2*pnorm(survey.test$t,lower.tail = F )
survey.test
```

```
##           beta     estimate      se.est         t         pval
## 1  (Intercept)  3.385308441 0.042932916 78.851118 0.000000e+00
## 2    educ.high  0.221824182 0.036261902  6.117279 9.518650e-10
## 3 partnerhiedu  0.130191973 0.067648529  1.924535 5.428754e-02
## 4     I(age/5)  0.067808360 0.010882979  6.230680 4.644146e-10
## 5         age2 -0.004319788 0.001189222 -3.632449 1.999719e+00
```

```
fit.2.aft<-survreg(Surv(secbi, b2event)~educ.high+partnerhiedu+I(age/5)+age2 + age2, data=sub2[sub2$sec

fit.2.aft.sum<-summary(fit.2.aft)

#Compare the se's of the parameters
survey.test$se.est/sqrt(diag(fit.2.aft.sum$var[-6, -6]))
```

```
## [1] 1.089144 1.414877 1.799860 1.254418 1.392521
```

```
#survey based errors are larger, as they should be.
```

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 parametric model to a longitudinally collected data set. Here I use data from the ECLS-K. Specifically, we will examine the transition into poverty between kindergarten and third grade.

First we load our data

```
eclskk5<-readRDS("C:/Users/ozd504/OneDrive - University of Texas at San Antonio/classes/dem7223/dem7223
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
eclskk5<-eclskk5[,myvars]
```

```
eclskk5$age1<-ifelse(eclskk5$x1kage_r==-9, NA, eclskk5$x1kage_r/12)
eclskk5$age2<-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 mo
eclskk5$age3<-ifelse(eclskk5$x5age==-9, NA, eclskk5$x5age/12)

eclskk5$pov1<-ifelse(eclskk5$x2povty==1,1,0)
eclskk5$pov2<-ifelse(eclskk5$x4povty_i==1,1,0)
eclskk5$pov3<-ifelse(eclskk5$x6povty_i==1,1,0)

#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='nhas
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<-subset(eclskk5, is.na(pov1)==F&is.na(pov2)==F&is.na(pov3)==F&is.na(age1)==F&is.na(age2)==F&is.
```

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.long<-reshape(data.frame(eclskk5), idvar="childid", varying=list(c("age1","age2"),
                                                    c("age2", "age3")),
                v.names=c("age_enter", "age_exit"),
                times=1:2, direction="long" )
e.long<-e.long[order(e.long$childid, e.long$time),]

e.long$povtran<-NA

e.long$povtran[e.long$pov1==0&e.long$pov2==1&e.long$time==1]<-1
e.long$povtran[e.long$pov2==0&e.long$pov3==1&e.long$time==2]<-1

e.long$povtran[e.long$pov1==0&e.long$pov2==0&e.long$time==1]<-0
e.long$povtran[e.long$pov2==0&e.long$pov3==0&e.long$time==2]<-0

#find which kids failed in earlier time periods and remove them from the second & third period risk set
failed1<-which(is.na(e.long$povtran)==T)
e.long<-e.long[-failed1,]


e.long$age1r<-round(e.long$age_enter, 0)
e.long$age2r<-round(e.long$age_exit, 0)
head(e.long, n=10)
```

```
##             childid x_chsex_r x_raceth_r x1kage_r x4age x5age  x6age  x7age
## 10000014.1 10000014         1          1    67.82 85.94 91.73  97.51 106.85
## 10000014.2 10000014         1          1    67.82 85.94 91.73  97.51 106.85
## 10000020.1 10000020         2          5    68.38 88.57 93.37 100.34 111.12
## 10000020.2 10000020         2          5    68.38 88.57 93.37 100.34 111.12
## 10000022.1 10000022         2          8    68.61 87.68 92.98  99.19 110.99
## 10000022.2 10000022         2          8    68.61 87.68 92.98  99.19 110.99
## 10000029.1 10000029         2          1    69.40 86.86 92.68  99.32 110.40
## 10000029.2 10000029         2          1    69.40 86.86 92.68  99.32 110.40
## 10000034.1 10000034         1          2    76.24 93.30 99.55 105.96 115.10
## 10000034.2 10000034         1          2    76.24 93.30 99.55 105.96 115.10
##            x2povty x4povty_i x6povty_i x8povty_i x12par1ed_i s2_id w6c6p_6psu
## 10000014.1       3         3         3         3           3  1433          2
## 10000014.2       3         3         3         3           3  1433          2
## 10000020.1       3         3         3         3           3  1365          2
## 10000020.2       3         3         3         3           3  1365          2
## 10000022.1       3         3         3         3           6  1405          1
## 10000022.2       3         3         3         3           6  1405          1
## 10000029.1       2         2         2         2           1  2042          2
## 10000029.2       2         2         2         2           1  2042          2
## 10000034.1       2         2         1        NA           3  2008          1
## 10000034.2       2         2         1        NA           3  2008          1
##            w6c6p_6str w6c6p_20 pov1 pov2 pov3 race_rec male mlths mgths time
## 10000014.1         39 328.0577    0    0    0  nhwhite    1     0     0    1
## 10000014.2         39 328.0577    0    0    0  nhwhite    1     0     0    2
## 10000020.1         53 136.5265    0    0    0  nhasian    0     0     0    1
```

```
## 10000020.2         53 136.5265    0    0    0  nhasian    0     0     0    2
## 10000022.1         35 163.1234    0    0    0    other    0     0     1    1
## 10000022.2         35 163.1234    0    0    0    other    0     0     1    2
## 10000029.1         60 341.5456    0    0    0  nhwhite    0     1     0    1
## 10000029.2         60 341.5456    0    0    0  nhwhite    0     1     0    2
## 10000034.1         50 289.4607    0    0    1  nhblack    1     0     0    1
## 10000034.2         50 289.4607    0    0    1  nhblack    1     0     0    2
##            age_enter age_exit povtran age1r age2r
## 10000014.1  5.651667 7.161667       0     6     7
## 10000014.2  7.161667 7.644167       0     7     8
## 10000020.1  5.698333 7.380833       0     6     7
## 10000020.2  7.380833 7.780833       0     7     8
## 10000022.1  5.717500 7.306667       0     6     7
## 10000022.2  7.306667 7.748333       0     7     8
## 10000029.1  5.783333 7.238333       0     6     7
## 10000029.2  7.238333 7.723333       0     7     8
## 10000034.1  6.353333 7.775000       0     6     8
## 10000034.2  7.775000 8.295833       1     8     8
```

So, this shows us the repeated measures nature of the longitudinal data set.

```
library(survminer)
```

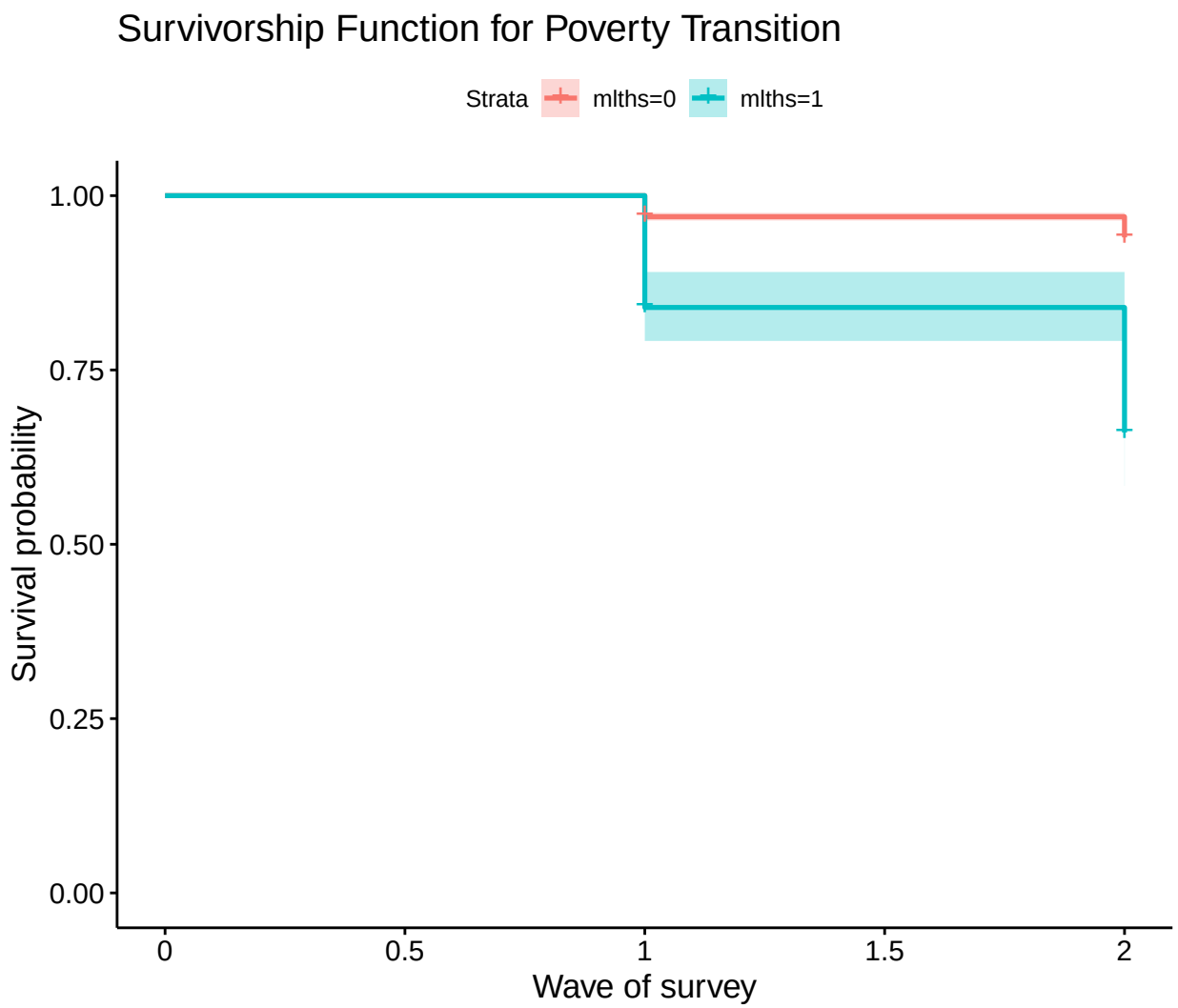
```
## Loading required package: ggplot2
```

```
## Loading required package: ggpubr
```

```
#poverty transition based on mother's education at time 1.
fit<-survfit(Surv(time = time, event = povtran)~mlths, e.long)
summary(fit)
```

```
## Call: survfit(formula = Surv(time = time, event = povtran) ~ mlths,
##     data = e.long)
##
## 9 observations deleted due to missingness
##                 mlths=0
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1   3774     114     0.97 0.00279        0.964        0.975
##     2   1830      56     0.94 0.00475        0.931        0.949
##
##                 mlths=1
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1    212      34     0.84  0.0252        0.792        0.891
##     2     89      19     0.66  0.0415        0.584        0.747
```

```
ggsurvplot(fit,conf.int = T, risk.table = F, title = "Survivorship Function for Poverty Transition", xl
```



Now we fit the models, I only show the Exponential, Weibull and PCH model fit here, but the others follow the example from above. I specify the age of the transition using an interval-censored notation to show when a child began and ended each risk period.

```
#Exponential
#interval censored
fitl1<-phreg(Surv(time = time, event = povtran)~mlths+mgths+race_rec, data=e.long, dist = "weibull", sha
summary(fitl1)
```

```
## Call:
## phreg(formula = Surv(time = time, event = povtran) ~ mlths +
##     mgths + race_rec, data = e.long, dist = "weibull", shape = 1)
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
## mlths               0.051     0.464     1.591     0.188     0.013
## mgths               0.797    -1.178     0.308     0.160     0.000
## race_rec
##         hispanic    0.213     0         1           (reference)
##          nhasian    0.085    -0.655     0.519     0.291     0.024
##          nhblack    0.063     0.004     1.004     0.242     0.986
```

```
##          nhwhite    0.568    -1.097     0.334     0.179     0.000
##            other    0.071    -0.338     0.713     0.286     0.236
##
## log(scale)                    2.072              0.136     0.000
##
##  Shape is fixed at  1
##
## Events                     223
## Total time at risk          5905
## Max. log. likelihood      -849.1
## LR test statistic         209.06
## Degrees of freedom        6
## Overall p-value           0
```

```
#Weibull
fitl2<-phreg(Surv(time = time, event = povtran)~mlths+mgths+race_rec, data=e.long, dist = "weibull")
summary(fitl2)
```

```
## Call:
## phreg(formula = Surv(time = time, event = povtran) ~ mlths +
##     mgths + race_rec, data = e.long, dist = "weibull")
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
## mlths               0.051     0.495     1.641     0.188     0.009
## mgths               0.797    -1.197     0.302     0.160     0.000
## race_rec
##         hispanic    0.213     0         1           (reference)
##          nhasian    0.085    -0.657     0.518     0.291     0.024
##          nhblack    0.063     0.007     1.007     0.243     0.978
##          nhwhite    0.568    -1.112     0.329     0.179     0.000
##            other    0.071    -0.350     0.705     0.286     0.221
##
## log(scale)                    1.077              0.056     0.000
## log(shape)                    1.024              0.055     0.000
##
## Events                     223
## Total time at risk          5905
## Max. log. likelihood      -735.96
## LR test statistic         217.30
## Degrees of freedom        6
## Overall p-value           0
```

```
#Piecewise constant
fitl3<-phreg(Surv(time = time, event = povtran)~mlths+mgths+race_rec,data=e.long, dist = "pch", cuts=c(
summary(fitl3)
```

```
## Call:
## phreg(formula = Surv(time = time, event = povtran) ~ mlths +
##     mgths + race_rec, data = e.long, dist = "pch", cuts = c(1))
##
## Covariate          W.mean      Coef Exp(Coef)  se(Coef)    Wald p
## mlths               0.051     0.466     1.594     0.188     0.013
## mgths               0.797    -1.179     0.308     0.160     0.000
```

```
## race_rec
##          hispanic    0.213     0         1           (reference)
##           nhasian    0.085    -0.655     0.519     0.291     0.024
##           nhblack    0.063     0.004     1.004     0.242     0.986
##           nhwhite    0.568    -1.097     0.334     0.179     0.000
##             other    0.071    -0.339     0.712     0.286     0.235
##
##
## Events                     223
## Total time at risk          5905
## Max. log. likelihood      -848.88
## LR test statistic         209.38
## Degrees of freedom        6
## Overall p-value           0
```

```
#AIC for exponential
-2*fitl1$loglik[2]+2*length(fitl1$coefficients)
```

```
## [1] 1712.204
```

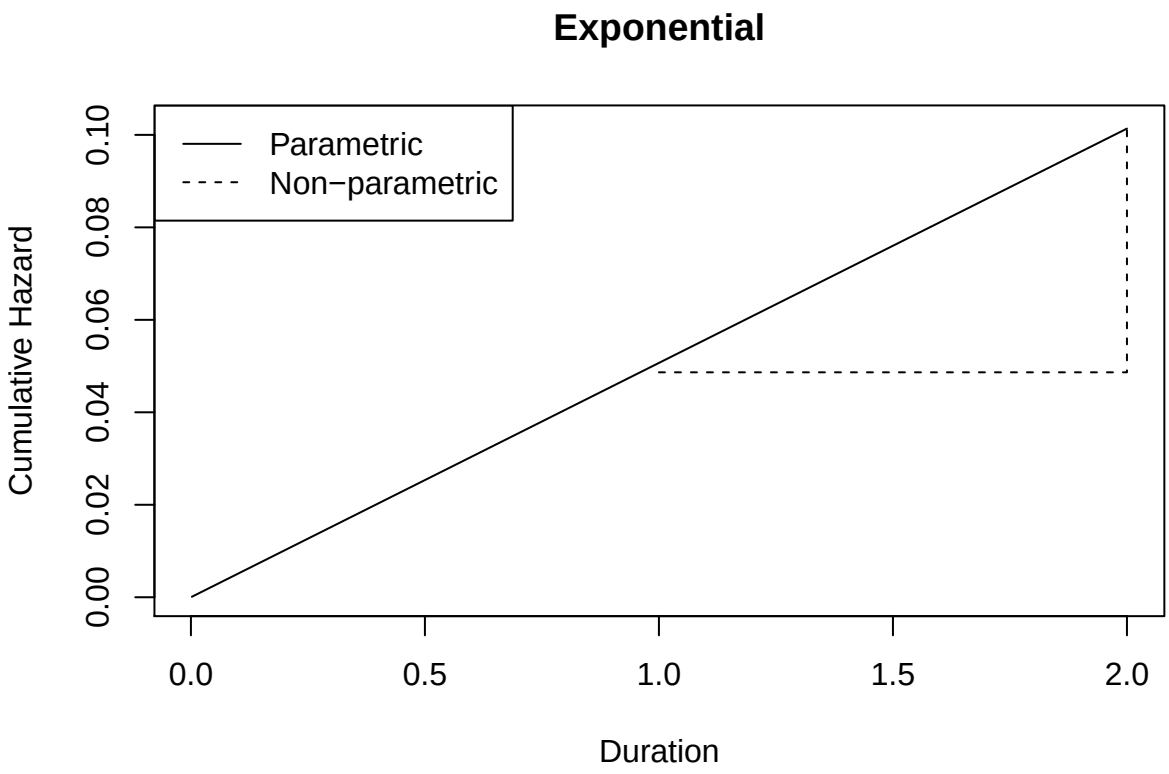
```
#AIC for weibull
-2*fitl2$loglik[2]+2*length(fitl2$coefficients)
```

```
## [1] 1487.925
```

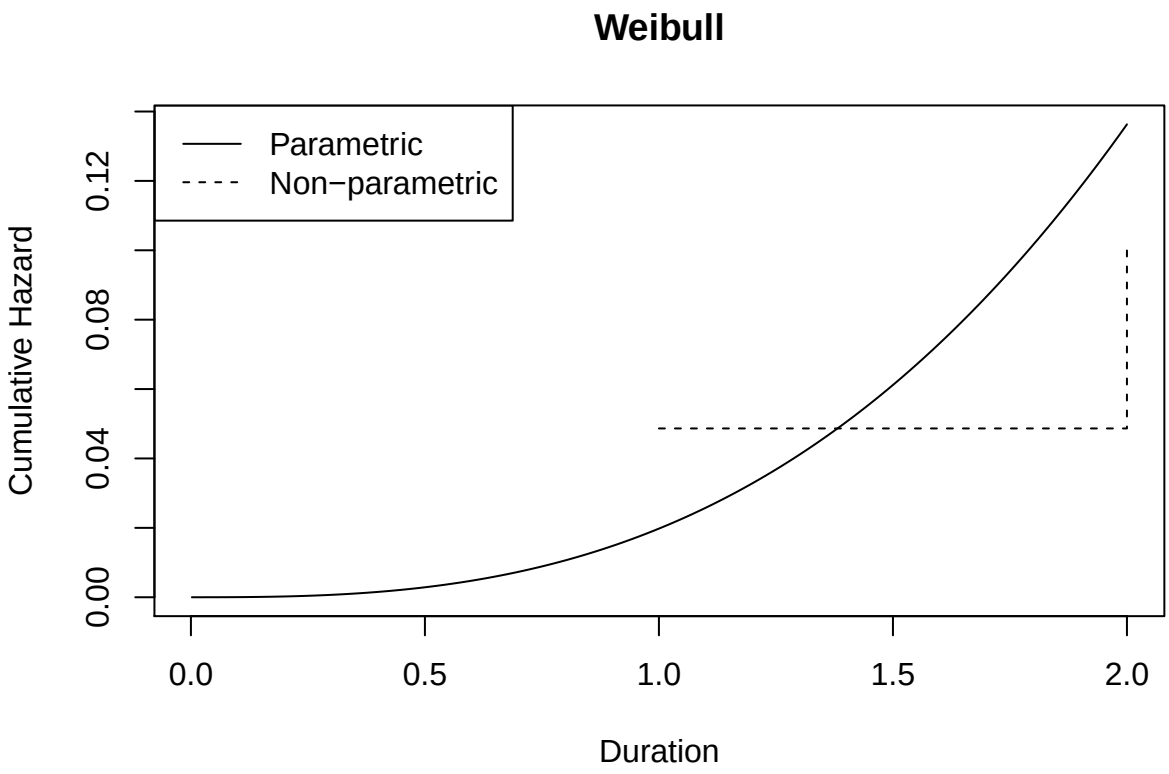
```
#AIC for weibull
-2*fitl3$loglik[2]+2*length(fitl3$coefficients)
```

```
## [1] 1709.755
```

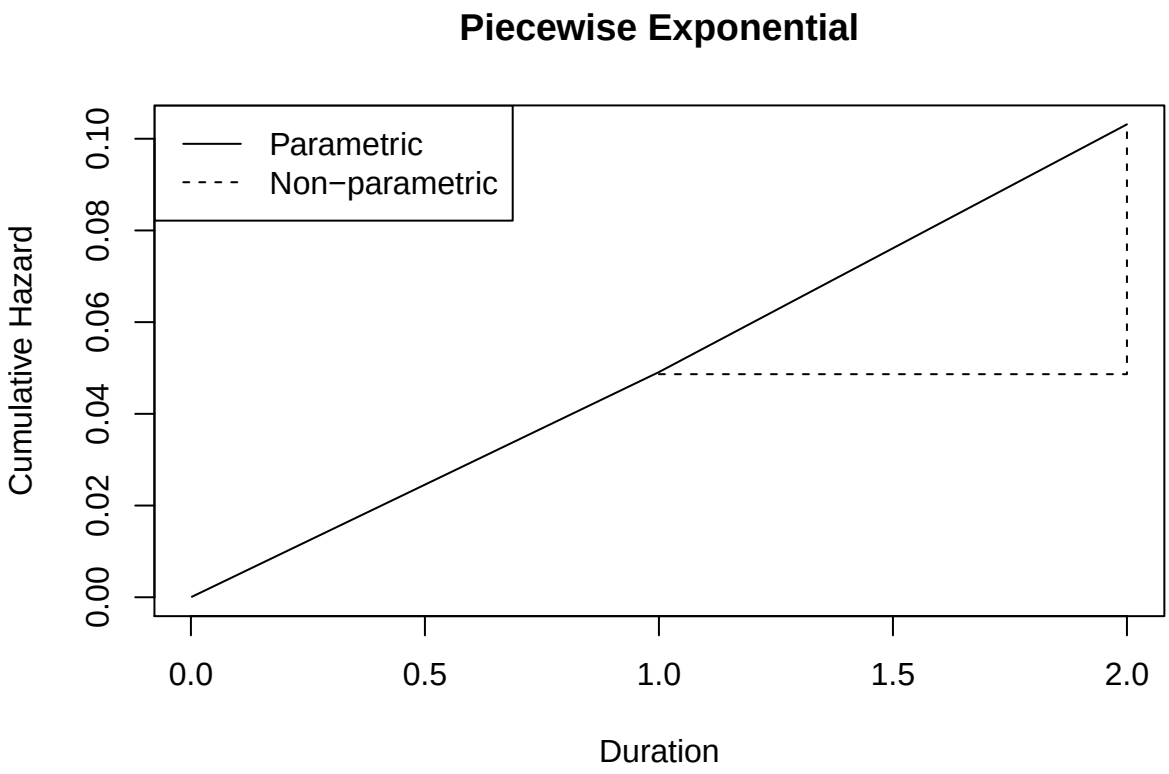
```
#Empirical (Cox)
fitle<-coxreg(Surv(time = time, event = povtran)~mlths+mgths+race_rec, data=e.long)

check.dist(fitle, fitl1, main = "Exponential")
```




```
check.dist(fitle, fitl2, main = "Weibull")
```




```
check.dist(fitle, fitl3, main = "Piecewise Exponential")
```



According to the AIC, the Weibull model is fitting better here.