Wednesday, April 9

Proportional Hazards and the Survival Function

Let $h_0(t)$ and $S_0(t)$ be the "baseline" hazard and survival functions (i.e., the function when all $x_j = 0$). If the proportional hazards assumption hold so that

$$h(t) = h_0(t)e^{\beta_1 x_1}e^{\beta_2 x_2} \cdots e^{\beta_k x_k},$$

then it can be shown that

$$S(t) = S_0(t)^{\eta} \quad \text{where} \quad \eta = e^{\beta_1 x_1}e^{\beta_2 x_2} \cdots e^{\beta_k x_k}.$$

Thus the effect of increasing x_j in a proportional hazards model can be summarized as follows.

1. If $\beta_j > 0$ then $S(t)$ will be *decreased* as x_j increases, as will $E(T)$.
2. If $\beta_j < 0$ then $S(t)$ will be *increased* as x_j increases, as will $E(T)$.

Note: The signs of the β_j parameters will be *opposite* of what they are in a equivalent accelerated failure time model.

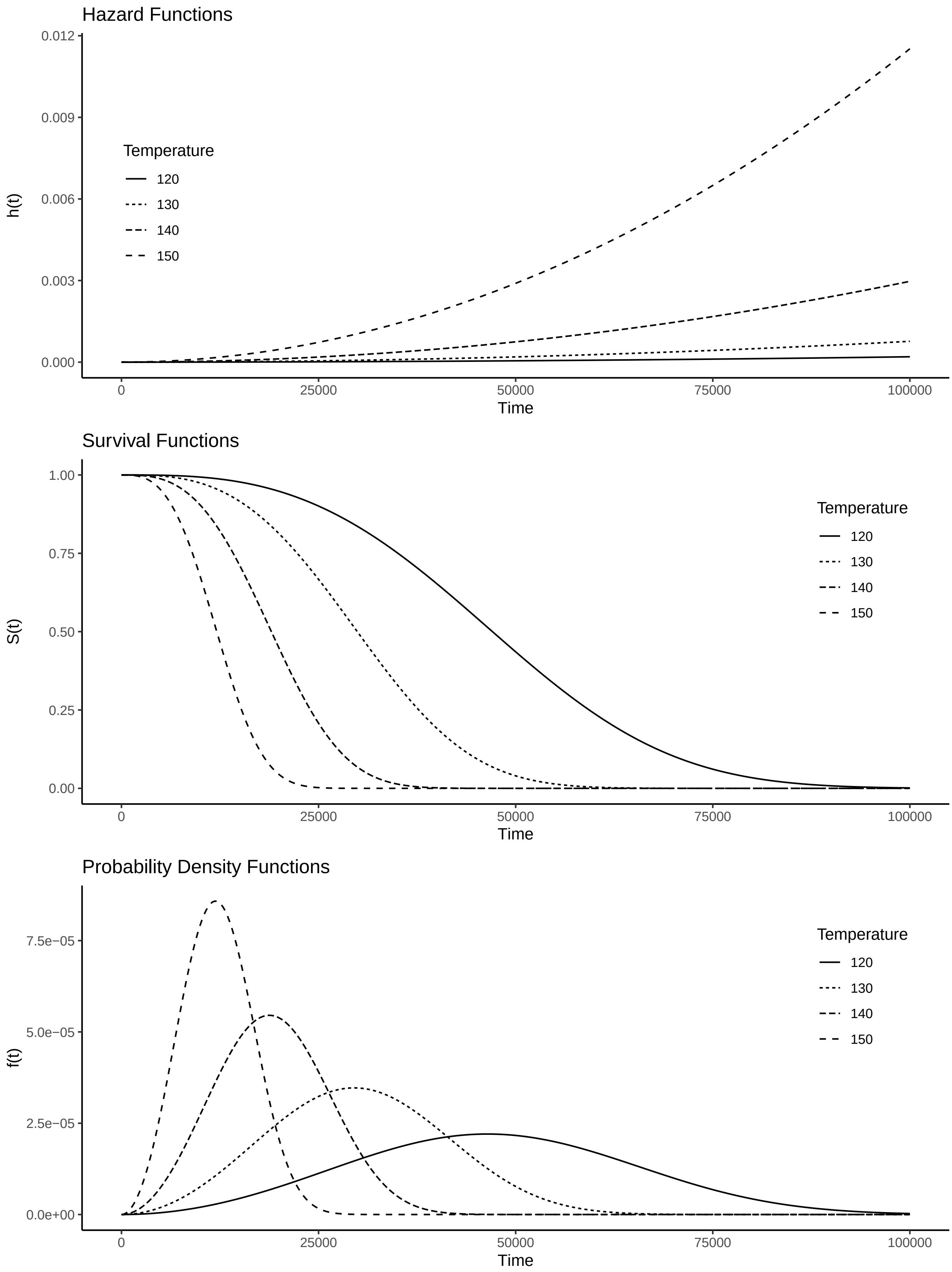
Example: Consider again a proportional hazards model for the `motors` data.

```r
library(flexsurv)
m <- flexsurvreg(Surv(time, cens) ~ temp, dist = "weibullPH", data = MASS::motors)
print(m)
```

```
Call:
flexsurvreg(formula = Surv(time, cens) ~ temp, data = MASS::motors, 
    dist = "weibullPH")

Estimates: 
       data mean  est        L95%       U95%       se         exp(est)  L95%       U95%     
shape        NA   2.99e+00   1.96e+00   4.56e+00   6.42e-01         NA         NA         NA
scale        NA   6.34e-22   1.46e-30   2.76e-13   6.43e-21         NA         NA         NA
temp   1.82e+02   1.36e-01   7.92e-02   1.92e-01   2.87e-02   1.15e+00   1.08e+00   1.21e+00

N = 40,  Events: 17,  Censored: 23
Total time at risk: 140654
Log-likelihood = -147, df = 3
AIC = 301
```


Hazard Functions
0.012
0.009
0.006
0.003
0.000
h(t)
Temperature
120
130
140
150
0
25000
50000
75000
100000
Time
Survival Functions
1.00
0.75
0.50
0.25
0.00
S(t)
Temperature
120
130
140
150
0
25000
50000
75000
100000
Time
Probability Density Functions
7.5e−05
5.0e−05
2.5e−05
0.0e+00
f(t)
Temperature
120
130
140
150
0
25000
50000
75000
100000
Time

Semi-Parametric (Cox) Proportional Hazards Model

A proportional hazards model assumes

$$h_i(t) = h_0(t)e^{\beta_1 x_{i1}}e^{\beta_2 x_{i2}} \cdots e^{\beta_k x_{ik}},$$

where again $h_0(t)$ is the "baseline" proportional hazards function. The functional form of $h_0(t)$ and thus $h_i(t)$ depends on the distribution of T_i.

1. A *parametric* proportional hazards model assumes a particular distribution and functional form of $h_0(t)$.
2. The *semi-parametric* proportional hazards model does not assume a particular distribution or functional form for $h_0(t)$.

The *marginal* or *partial* likelihood function permits maximum likelihood estimation of $\beta_1, \beta_2, \ldots, \beta_k$ *without* assuming a particular distribution. It is based only on the *rank order* of the times.

Comments about semi-parametric proportional hazards models.

1. Right-censoring can be easily handled with this model. But other types of censoring require additional assumptions.
2. Estimation of hazard and survival functions relies on a semi-parametric approach.
3. Stratification can be used when hazard functions are proportional within but not between strata.

The function `coxph` from the **survival** package will estimate a Cox proportional hazards model.

Example: Consider a Cox proportional hazards model for the `motors` data.

```r
library(survival) # for coxph function
m <- coxph(Surv(time, cens) ~ temp, data = MASS::motors)
summary(m)
```

```
Call:
coxph(formula = Surv(time, cens) ~ temp, data = MASS::motors)

  n= 40, number of events= 17

       coef exp(coef) se(coef)    z Pr(>|z|)
temp 0.0919    1.0962   0.0274 3.36  0.00079 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

     exp(coef) exp(-coef) lower .95 upper .95
temp       1.1      0.912      1.04      1.16

Concordance= 0.84  (se = 0.035 )
Likelihood ratio test= 25.6  on 1 df,   p=4e-07
Wald test            = 11.3  on 1 df,   p=8e-04
Score (logrank) test = 22.7  on 1 df,   p=2e-06
```

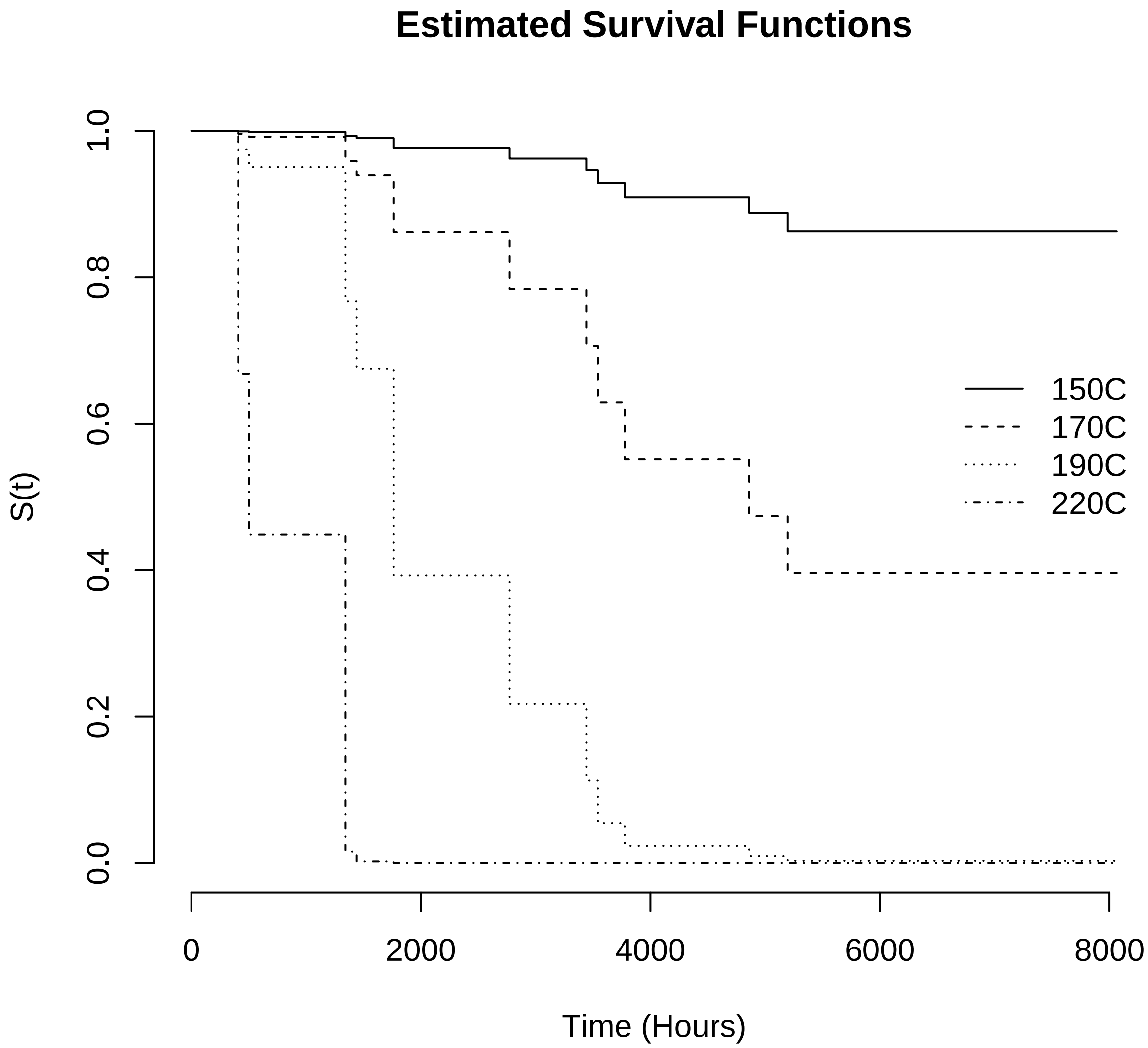
We can plot estimated survival functions from a `coxph` model object.

```r
d <- data.frame(temp = c(150,170,190,220))

# plot estimated survival functions
plot(survfit(m, newdata = d), bty = "n", lty = 1:4, xlab = "Time (Hours)", ylab = "S(t)")

# add a legend
legend(6500, 0.7, legend = c("150C", "170C", "190C", "220C"), lty = 1:4, bty = "n")
```

```r
# add a title
title("Estimated Survival Functions")
```



A common non-parametric estimator of a survival function is the Kaplan-Meier estimator, but it is largely limited to cases where you have a categorical explanatory variable with multiple times observed per category.

Discrete Survival Time Models

Discrete survival time models treat time-to-event as a discrete random variable rather than a continuous random variable. This is done for one of two reasons.

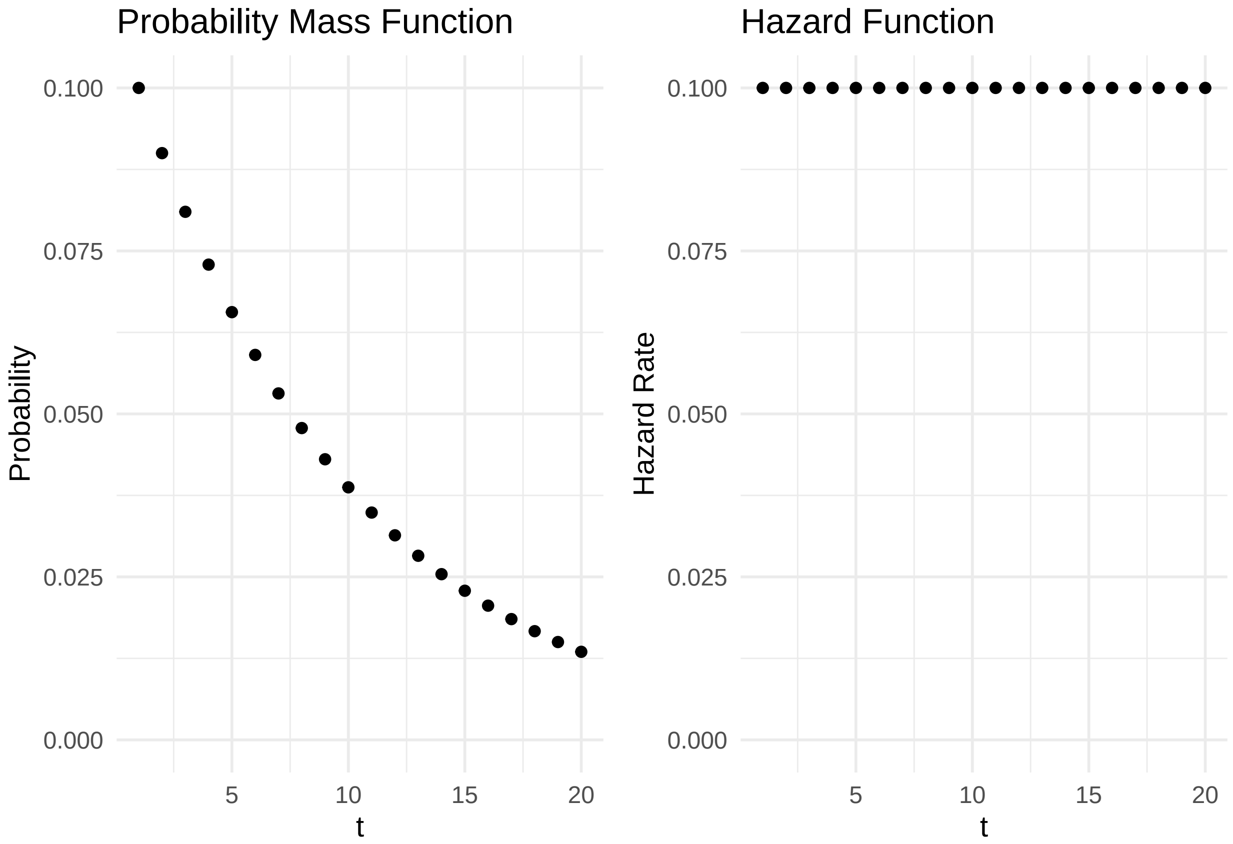
1. Time is actually continuous, but we treat it as discrete for convenience/simplicity, or because the observations are interval-censored (with common intervals, e.g., week, month, year).
2. The "time" is actually a count of "attempts" of an event (e.g., number of cycles until pregnancy, number of times to take a test until it is passed, number of times a machine is run until it fails).

For discrete time, the probability density, survival, and hazard functions are analogous to what they are for continuous time, but simpler because all of them give probabilities.

1. The *probability mass function* is $f(t) = P(T = t)$. This gives the probability that the event will happen at time t.
2. The *survival function* is, as before, $S(t) = P(T \geq t)$. This gives the probability that the event will happen at time t or later.
3. The *hazard function* is $h(t) = P(T = t|T \geq t)$. This gives the probability that the event will happen at time t *given* that it has not yet happened (i.e., the probability that it will happen at time t given that the unit has "survived" to that point).

It is important to not confused the probability mass function which gives the probability that the event will happen at time t, versus the hazard function which gives the probability that the event will happen at time t *given* that it has not yet happened.

Example: Suppose I have a fair ten-sided die. Let t be the number of rolls until I get a one. The figures below show the probability mass and hazard function.



Technical Details: Note that $f(t)$, $S(t)$, and $h(t)$ are related because $h(t) = f(t)/S(t)$. Also we can define $f(t)$ entirely in terms of $h(t)$. Consider that if a unit survives to time t, the probability that it *will not* survive past time t is

$$h(t) = P(T = t|T \geq t),$$

and the probability that it *will* survive past time t is

$$1 - h(t) = 1 - P(T = t|T \geq t) = P(T > t|T \geq t).$$

So we can write $f(t)$ in terms of $h(t)$ as follows.

1. For observations that *are not* right-censored at time t,

$$\begin{aligned}
f(1) &= h(1), \\
f(2) &= [1-h(1)]h(2), \\
f(3) &= [1-h(1)][1-h(2)]h(3), \\
f(4) &= [1-h(1)][1-h(2)][1-h(3)]h(4), \\
f(5) &= [1-h(1)][1-h(2)][1-h(3)][1-h(4)]h(5),
\end{aligned}$$

   and so on. In general for non-censored discrete times

$$f(t) = \begin{cases} h(t), & \text{if } t = 1, \\ h(t)\prod_{j=1}^{t-1}[1-h(j)], & \text{if } t > 1, \end{cases}$$

   Note that $1 - h(t) = 1 - P(T = t|T \geq t) = P(T > t|T \geq t)$.

2. For observations that *are* right-censored at time t,

$$\begin{aligned}
f(1) &= [1-h(1)], \\
f(2) &= [1-h(1)][1-h(2)], \\
f(3) &= [1-h(1)][1-h(2)][1-h(3)], \\
f(4) &= [1-h(1)][1-h(2)][1-h(3)][1-h(4)], \\
f(5) &= [1-h(1)][1-h(2)][1-h(3)][1-h(4)][1-h(5)],
\end{aligned}$$

   and so on. In general for right-censored discrete times

$$f(t) = \prod_{j=1}^{t}[1-h(j)].$$

   Note that $1 - h(t) = 1 - P(T = t|T \geq t) = P(T > t|T \geq t)$.

Discrete Survival Models as Binary Regression Models

Discrete survival time models can be expressed as *binary* regression models. We can model the probability that a unit will not survive past time t *given* that it survived to time t, or we can model the probability that it will survive past time t *given* that it survived to time t.

Suppose we code time-till-event with positive integers. For every T we define a set of binary responses such that if $T = t$ then we have t binary responses, $Y_1, Y_2, \ldots, Y_t$, such that

$$Y_t = \begin{cases} 1, & \text{if the event occurs at time } t \text{ (i.e., } T = t\text{)}, \\ 0, & \text{if the event occurs after time } t \text{ (i.e., } T > t\text{)}. \end{cases}$$

Note that if T is right-censored then we let $T = t$ where t is the last time we know the event had not failed, but $Y_t = 0$.

Example: The observed event times are $T = t$ where t = 1, 2, 3, 4, or 5. Then we define $Y_1, Y_2, \ldots, Y_5$ as follows.

Example: T is censored such that $T > t$ where t = 1, 2, 3, 4, or 5. Then we define $Y_1, Y_2, \ldots, Y_5$ as follows.

Not: If time is discrete due to interval-censoring the *maximum possible time* does not need a binary variable.

Technical Details: The distribution of T can be stated in terms of the Y_t. It follows that $h(t) = P(Y_t = 1)$ and $1 - h(t) = 1 - P(Y_t = 1) = P(Y_t = 0)$, so if T is not censored then

$$f(t) = \begin{cases} P(Y_1 = 1), & \text{if } t = 1, \\ P(Y_t = 1)\prod_{j=1}^{t-1}P(Y_t = 0), & \text{if } t > 1, \end{cases}$$

t	Y_1	Y_2	Y_3	Y_4	Y_5
1	1				
2	0	1			
3	0	0	1		
4	0	0	0	1	
5	0	0	0	0	1

t	Y_1	Y_2	Y_3	Y_4	Y_5
1	0				
2	0	0			
3	0	0	0		
4	0	0	0	0	
5	0	0	0	0	0

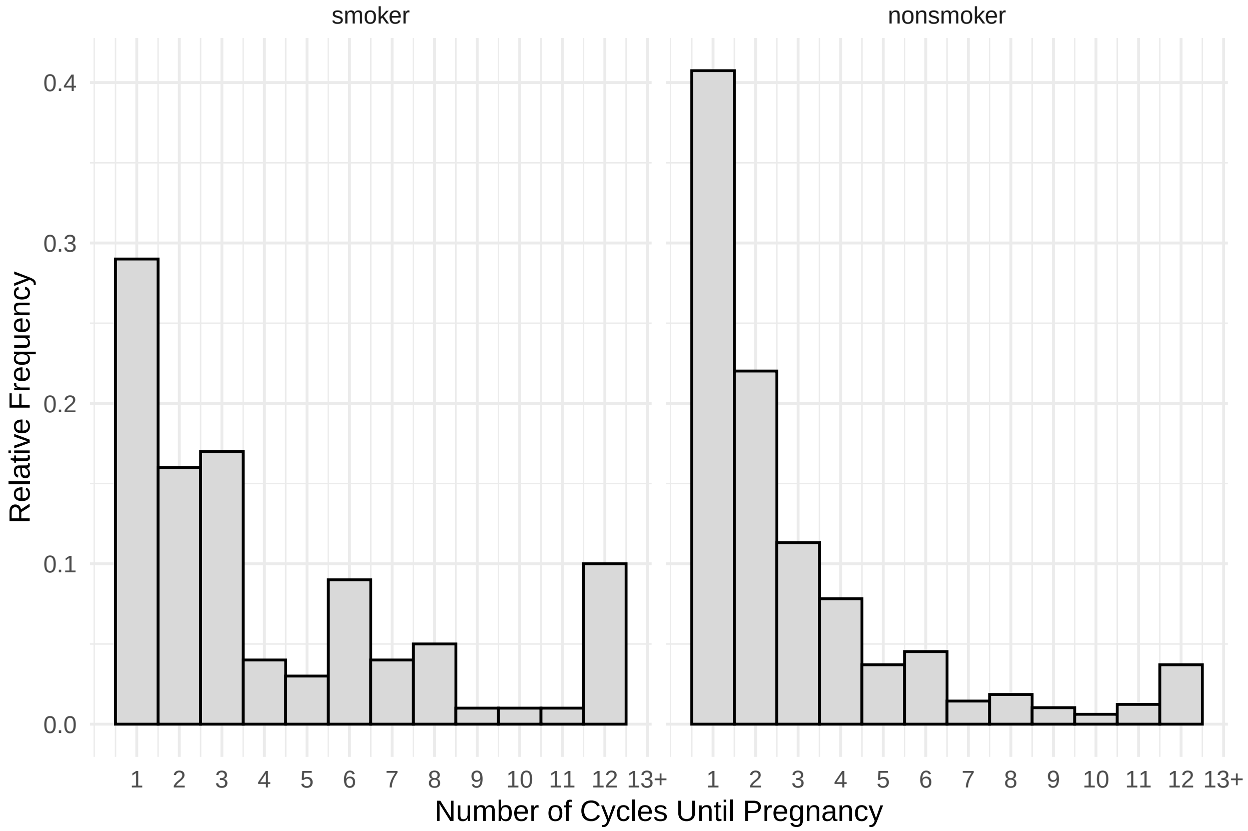
and if T is censored such that $T > t$ then

$$f(t) = \prod_{j=1}^{t} P(Y_t = 0).$$

TLDR: Many discrete time survival models can be estimated as binary regression models (e.g., logistic regression) where the response variable is an indicator variable for if the event happened at a given time.

Example: Consider the following data from a study comparing mothers who smoke to those who do not with respect to the number of menstrual cycles until pregnancy.

```r
library(trtools) # for cycles data
p <- ggplot(cycles, aes(x = cycles, y = after_stat(density))) +
  facet_wrap(~ mother) +
  geom_histogram(binwidth = 1, center = 1,
    color = "black", fill = grey(0.85)) +
  scale_x_continuous(breaks = 1:13, labels = c(1:12,"13+")) +
  labs(x = "Number of Cycles Until Pregnancy",
       y = "Relative Frequency") + theme_minimal()
plot(p)
```



Note that for a relative frequency histogram, use `y = after_stat(density*w)` where `w` is a number indicating the bin width (which is one in this case).

It is important to note that all reported values of 13 cycles are actually right-censored and so represent 13 *or more* cycles. The observed censoring times are between 1 and 12 cycles, with all recorded cycles of 13 representing right-censored observations only known to be *more than 12 cycles*. We need to create an indicator variable for *observed* times and to change values of 13 to 12 since that was the last observed time.

```
cycles$status <- ifelse(cycles$cycles == 13, 0, 1)
cycles$cycles <- ifelse(cycles$cycles == 13, 12, cycles$cycles)
```

Here are some mothers of observed (i.e., not censored) times.

```
    cycles    mother status
102      1 nonsmoker      1
216      1 nonsmoker      1
358      2 nonsmoker      1
437      3 nonsmoker      1
449      3 nonsmoker      1
```

Here are some mothers with censored times.

```
    cycles    mother status
576     12 nonsmoker      0
577     12 nonsmoker      0
581     12 nonsmoker      0
582     12 nonsmoker      0
584     12 nonsmoker      0
```

The function `dsurvbin` from the **trtools** package helps convert a data frame with a discrete time-till-event into a format with binary variables as discussed above (a similar function is available in the **discSurv** package).

```
cycles.bin <- dsurvbin(cycles, y = "cycles", event = "status")
```

So depending on the number of cycles up to twelve indicator variable are created *for each observational unit*. For example, here is a mother where pregnancy occurred after *three* cycles.

```
    cycles mother status unit t y
541      3 smoker      1   46 1 0
542      3 smoker      1   46 2 0
543      3 smoker      1   46 3 1
```

And here is a mother where pregnancy occurred after *five* cycles.

```
    cycles mother status unit t y
793      5 smoker      1   67 1 0
794      5 smoker      1   67 2 0
795      5 smoker      1   67 3 0
796      5 smoker      1   67 4 0
797      5 smoker      1   67 5 1
```

And here is a mother where pregnancy occurred after *twelve* cycles.

```
     cycles mother status unit  t y
1081     12 smoker      1   91  1 0
1082     12 smoker      1   91  2 0
1083     12 smoker      1   91  3 0
1084     12 smoker      1   91  4 0
1085     12 smoker      1   91  5 0
1086     12 smoker      1   91  6 0
1087     12 smoker      1   91  7 0
1088     12 smoker      1   91  8 0
1089     12 smoker      1   91  9 0
1090     12 smoker      1   91 10 0
1091     12 smoker      1   91 11 0
1092     12 smoker      1   91 12 1
```

But for comparison, here is a mother where pregnancy was right-censored and is only known to have occurred (if it occurred) *after twelve cycles*.

```
     cycles mother status unit  t y
1117     12 smoker      0   94  1 0
1118     12 smoker      0   94  2 0
1119     12 smoker      0   94  3 0
1120     12 smoker      0   94  4 0
1121     12 smoker      0   94  5 0
1122     12 smoker      0   94  6 0
1123     12 smoker      0   94  7 0
1124     12 smoker      0   94  8 0
1125     12 smoker      0   94  9 0
1126     12 smoker      0   94 10 0
1127     12 smoker      0   94 11 0
1128     12 smoker      0   94 12 0
```

Note: Here's a way to rearrange the data using tools from the `dplyr` package.

```
cycles.bin <- trtools::cycles %>%
  mutate(status = ifelse(cycles == 13, 0, 1)) %>%
  mutate(cycles = ifelse(cycles == 13, 12, cycles)) %>%
  mutate(unit = 1:n()) %>% uncount(cycles, .remove = FALSE) %>%
  arrange(unit) %>% group_by(unit) %>% mutate(t = 1:n()) %>%
  mutate(y = ifelse(t < cycles | status == 0, 0, 1))
```

Now consider a logistic regression model for the binary response variable `y`. This model effectively estimates the hazard rate (i.e., probability of pregnancy) under given circumstances (e.g., whether or not the mother is a smoker).

```
m <- glm(y ~ mother, family = binomial, data = cycles.bin)
cbind(summary(m)$coefficients, confint(m))
```

```
                Estimate Std. Error z value Pr(>|z|)  2.5 % 97.5 %
(Intercept)       -1.242      0.118  -10.55 5.08e-26 -1.478 -1.016
mothernonsmoker    0.541      0.130    4.15 3.31e-05  0.289  0.801
```

Odds ratio for smoking.

```
exp(cbind(coef(m), confint(m)))
```

```
                      2.5 % 97.5 %
(Intercept)     0.289 0.228  0.362
mothernonsmoker 1.718 1.336  2.228
```

```
trtools::contrast(m, tf = exp,
  a = list(mother = "nonsmoker"),
  b = list(mother = "smoker"))
```

```
 estimate lower upper
     1.72  1.33  2.22
```

```
trtools::contrast(m, tf = exp,
  a = list(mother = "smoker"),
  b = list(mother = "nonsmoker"))
```

```
 estimate lower upper
    0.582 0.451 0.751
```

Estimated probabilities of pregnancy on any given cycles.

```
trtools::contrast(m, a = list(mother = c("nonsmoker","smoker")),
  tf = plogis, cnames = c("nonsmoker","smoker"))
```

```
          estimate lower upper
nonsmoker    0.332 0.308 0.357
smoker       0.224 0.187 0.267
```

Note that with this model the hazard function is "flat" — i.e., the probability of pregnancy each cycle (given pregnancy has not yet happened) is the same.[1] This is reasonable here, but in other cases we might expect there to be time-varying effects (e.g., season or temperature in animals), which can be handled easily since we can let an explanatory variable vary over time (recorded as `t` in the data frame). Although over a longer time span we might consider a model where the hazard function decreases due to age.

Example: Consider the following data on the grade when adolescent males first experience sexual intercourse.

[1] In such cases we say that the number of trials until something happens has a *geometric* distribution.

```
firstsex <- read.table("https://stats.idre.ucla.edu/stat/examples/alda/firstsex.csv",
    sep = ",", header = TRUE)
head(firstsex)
```

```
  id time censor pt    pas
1  1    9      0  0  1.979
2  2   12      1  1 -0.545
3  3   12      1  0 -1.405
4  5   12      0  1  0.974
5  6   11      0  0 -0.636
6  7    9      0  1 -0.243
```

There is right-censoring (i.e., boys who did not experience sex by the 12th grade). We need a proper status variable for that.

```
firstsex$status <- ifelse(firstsex$censor == 1, 0, 1)
```

One key explanatory variable is whether or not a boy experienced a "parenting transition" prior to the 7th grade. The variable is `pt` but is a binary variable. We'll convert it to a factor with clear level labels.

```
firstsex$transition <- factor(firstsex$pt,
   levels = c(0,1), labels = c("no","yes"))
```

We can verify that these changes were done correctly.

```
head(firstsex)
```

```
  id time censor pt    pas status transition
1  1    9      0  0  1.979      1         no
2  2   12      1  1 -0.545      0        yes
3  3   12      1  0 -1.405      0         no
4  5   12      0  1  0.974      1        yes
5  6   11      0  0 -0.636      1         no
6  7    9      0  1 -0.243      1        yes
```

Now we need to transform the data to create indicator variables for whether or not a boy experienced sex for the first time in a given grade.

```
library(trtools)
firstsex <- dsurvbin(firstsex, "time", "status")
head(firstsex)
```

```
  id time censor pt    pas status transition unit t y
1  1    9      0  0  1.979      1         no    1 7 0
2  1    9      0  0  1.979      1         no    1 8 0
3  1    9      0  0  1.979      1         no    1 9 1
7  2   12      1  1 -0.545      0        yes    2 7 0
8  2   12      1  1 -0.545      0        yes    2 8 0
9  2   12      1  1 -0.545      0        yes    2 9 0
```

Here is a boy who first had sex in the 9th grade.

```
subset(firstsex, id == 1)
```

```
  id time censor pt  pas status transition unit t y
1  1    9      0  0 1.98      1         no    1 7 0
2  1    9      0  0 1.98      1         no    1 8 0
3  1    9      0  0 1.98      1         no    1 9 1
```

Here is a boy who first had sex in the 12th grade.

```r
subset(firstsex, id == 5)
```

```
   id time censor pt   pas status transition unit  t y
19  5   12      0  1 0.974      1        yes    4  7 0
20  5   12      0  1 0.974      1        yes    4  8 0
21  5   12      0  1 0.974      1        yes    4  9 0
22  5   12      0  1 0.974      1        yes    4 10 0
23  5   12      0  1 0.974      1        yes    4 11 0
24  5   12      0  1 0.974      1        yes    4 12 1
```

Here is a boy who did not first have sex by the 12th grade (but may have first had sex later — i.e., right-censored).

```r
subset(firstsex, id == 3)
```

```
   id time censor pt  pas status transition unit  t y
13  3   12      1  0 -1.4      0         no    3  7 0
14  3   12      1  0 -1.4      0         no    3  8 0
15  3   12      1  0 -1.4      0         no    3  9 0
16  3   12      1  0 -1.4      0         no    3 10 0
17  3   12      1  0 -1.4      0         no    3 11 0
18  3   12      1  0 -1.4      0         no    3 12 0
```

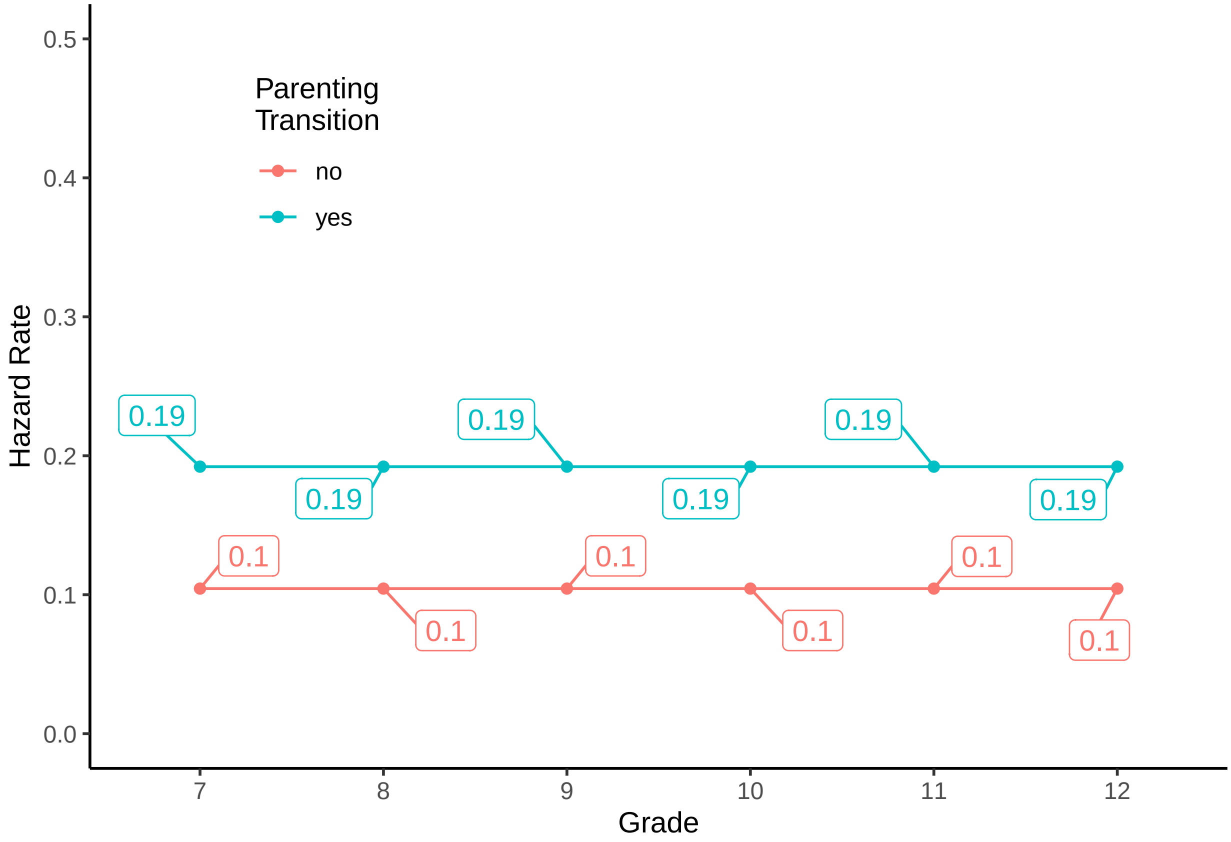
First consider a model for a flat/constant hazard function $h(t) = P(T = t | T \geq t)$, where here T is grade. However we will let the hazard rate depend on whether or not there was a parenting transition.

```r
m <- glm(y ~ transition, family = binomial, data = firstsex)
summary(m)$coefficients
```

```
              Estimate Std. Error z value Pr(>|z|)
(Intercept)     -2.149      0.171  -12.54 4.55e-36
transitionyes    0.713      0.208    3.42 6.23e-04
```

```r
d <- expand.grid(t = c("7","8","9","10","11","12"), transition = c("no","yes"))
d$yhat <- predict(m, newdata = d, type = "response")

library(ggrepel) # for geom_label_repel
p <- ggplot(d, aes(x = t, y = yhat, color = transition)) + theme_classic() +
   geom_point() + geom_line(aes(group = transition)) + ylim(0, 0.5) +
   geom_label_repel(aes(label = round(yhat,2)),
    box.padding = 0.75, show.legend = FALSE) +
   labs(x = "Grade", y = "Hazard Rate", color = "Parenting\nTransition") +
   theme(legend.position = "inside", legend.position.inside = c(0.2,0.8))
plot(p)
```



```
# odds ratio
contrast(m, tf = exp,
    a = list(transition = "yes", t = c("7","8","9","10","11","12")),
    b = list(transition = "no", t = c("7","8","9","10","11","12")),
    cnames = paste("Grade", 7:12))
```

```
         estimate lower upper
Grade 7      2.04  1.36  3.07
Grade 8      2.04  1.36  3.07
Grade 9      2.04  1.36  3.07
Grade 10     2.04  1.36  3.07
Grade 11     2.04  1.36  3.07
Grade 12     2.04  1.36  3.07
```

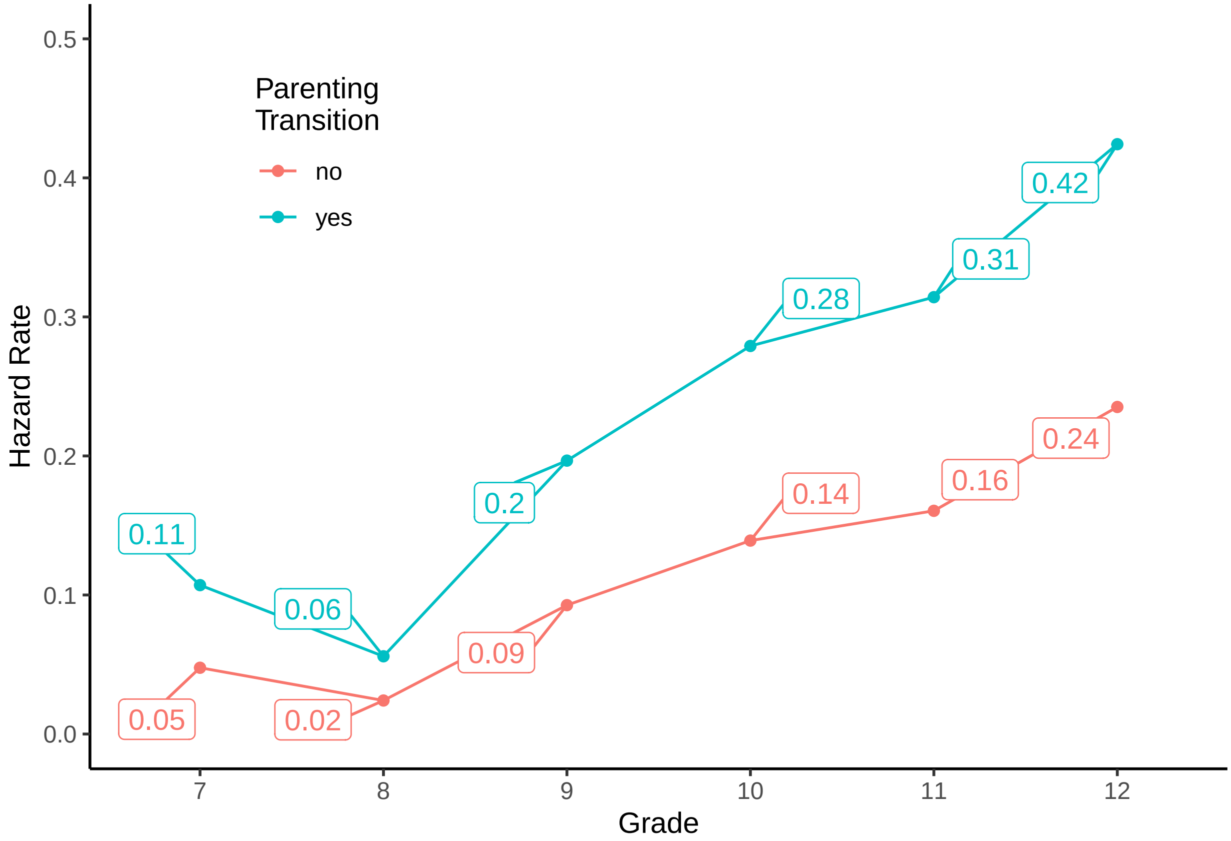
Now consider a model where the hazard rate is not necessarily constant over grades. This can be done by including an “effect” for time (i.e., grade).

```
m <- glm(y ~ transition + t, family = binomial, data = firstsex)
summary(m)$coefficients
```

```
              Estimate Std. Error z value Pr(>|z|)
(Intercept)     -2.994      0.318   -9.43 4.07e-21
transitionyes    0.874      0.217    4.02 5.86e-05
t8              -0.706      0.473   -1.49 1.36e-01
t9               0.713      0.352    2.03 4.27e-02
t10              1.172      0.345    3.39 6.89e-04
t11              1.340      0.359    3.74 1.88e-04
t12              1.815      0.367    4.94 7.78e-07
```

```r
d <- expand.grid(t = c("7","8","9","10","11","12"), transition = c("no","yes"))
d$yhat <- predict(m, newdata = d, type = "response")

p <- ggplot(d, aes(x = t, y = yhat, color = transition)) + theme_classic() +
   geom_point() + geom_line(aes(group = transition)) + ylim(0, 0.5) +
   geom_label_repel(aes(label = round(yhat,2)),
      box.padding = 0.75, show.legend = FALSE) +
   labs(x = "Grade", y = "Hazard Rate", color = "Parenting\nTransition") +
   theme(legend.position = "inside", legend.position.inside = c(0.2,0.8))
plot(p)
```



```r
# odds ratio
contrast(m, tf = exp,
    a = list(transition = "yes", t = c("7","8","9","10","11","12")),
    b = list(transition = "no", t = c("7","8","9","10","11","12")),
    cnames = paste("Grade", 7:12))
```

```
         estimate lower upper
Grade 7       2.4  1.56  3.67
Grade 8       2.4  1.56  3.67
Grade 9       2.4  1.56  3.67
Grade 10      2.4  1.56  3.67
Grade 11      2.4  1.56  3.67
Grade 12      2.4  1.56  3.67
```